McAfee Web Gateway 10.2.x Product Guide

Product overview

McAfee® Web Gateway protects your network against threats arising from the web.

Using different features in a complex process, it filters the traffic that goes out and comes in when users of your network access the web, allowing or blocking this traffic based on the rules of your web security policy.

You can configure this policy to suit your requirements by modifying and enlarging the default policy that is in place after the initial setup.

Overview

McAfee® Web Gateway is a web security product that protects your network against threats arising from the web.

Web Gateway is installed as a physical or virtual appliance, which serves as a gateway that connects your network to the web.

Following the implemented web security rules, Web Gateway filters the traffic that goes out and comes in. Malicious and inappropriate content is blocked, while useful matter is allowed to pass through.

Web Gateway is part of a solution known as McAfee® Web Protection.

- Within this solution, Web Gateway protects your network against threats that arise when on-premise users access the web from inside your network.
- McAfee® Web Gateway Cloud Service (McAfee® WGCS) is the part of the solution that protects web usage by cloud users, who access the web from outside your network, for example, while traveling or working at home.

As an integrated solution, Web Protection allows you to enforce the same security policy for web access by both on-premise and cloud users.

Key features

Filtering web traffic is a complex process. The key features of Web Gateway contribute to this process in different ways.

- **Interception of web traffic** — Intercepting web traffic is a prerequisite for any filtering. It is accomplished by the proxy functions of Web Gateway, which can be performed under different network protocols, such as HTTP, HTTPS, HTTP2, FTP, XMPP, and others.

  Depending on what you configure, Web Gateway can run in explicit proxy mode or in one of several transparent modes.
- **Authentication** — The authentication functions of Web Gateway check the authorization of users, relying on information from internal and external databases and using authentication methods such as NTLM, LDAP, RADIUS, Kerberos, and others.
- **Web filtering** — The anti-malware functions of Web Gateway scan and filter web traffic and block web objects if they are infected.

  Other functions filter URLs that users request access to, using information from the McAfee® Global Threat Intelligence™ (McAfee GTI) system, or perform media type and application filtering.

  The filtering functions are supported by functions that complete such jobs as counting user requests for web access or indicating the progress made in downloading web objects.
- **Monitoring** — The monitoring functions of Web Gateway provide a comprehensive and continuous overview of the filtering process.

  They include a dashboard, which displays information on alerts, web usage, filtering activities, and system behavior, as well as logging and tracing functions.

  Options to get external components involved in the monitoring process, for example, McAfee® ePolicy Orchestrator® (McAfee® ePO™) or an SNMP agent, are also provided.

How it works

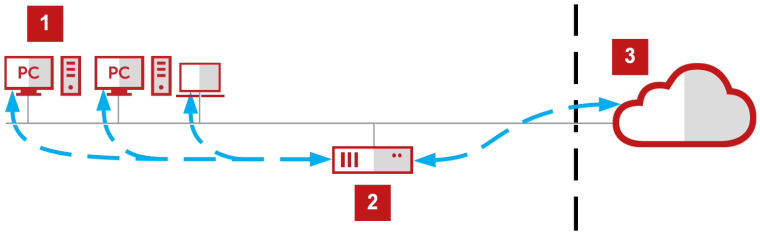
To protect your network against threats arising from the web, Web Gateway filters the traffic that goes out and comes in.

Following the implemented web security rules, Web Gateway filters the requests that users send to the web from within your network, and the responses that are sent back from the web. Embedded objects sent with requests or responses are also filtered.

As a result of the filtering process, requests, responses, and embedded objects, are blocked or allowed.

The workflow is as follows:

1. Requests are sent from your network to the web.
2. Web Gateway filters requests and responses.
3. Responses are sent from the web to your network.



To perform the filtering process, Web Gateway uses a rule engine and several filter engines or modules, for example, the Anti-Malware module or the URL Filter module. These modules complete particular jobs when the implemented web security rules are processed.

You can configure the web security rules and the behavior of the filter modules to adapt them to the requirements of your organization.

The filtering process also relies on the operating system of Web Gateway, which is MLOS 3 (McAfee Linux Operating System, version 3).

Working with Web Gateway

You work with Web Gateway as an administrator to ensure it protects your network against threats arising from the web.

To provide this protection, you install Web Gateway as an appliance within your network. The appliance filters web traffic that occurs when users access the web from within your network. The filtering process follows the rules of your web security policy.

After the initial setup, you continue on the user interface, dealing mainly with:

- **System settings** — Control the appliance system
- **Policy rules** — Ensure web security

At the initial setup, you are already prompted to configure some system settings. A default web security policy is also in place after the initial setup.

To be sure that your web security policy works as expected and to take measures if it does not, you are also dealing with:

- **Monitoring** — Shows performance parameters
- **Troubleshooting** — Offers measures to address issues

To carry out any administrator activities, access rights are required. So, you also need to manage:

- **Administrator accounts** — Control access to the different fields of administrator activities

An account for a Super Administrator is already configured after the initial setup.

Configuring system settings

System settings include initial settings and others that are configured after the initial setup.

Initial system settings

System settings configured when initially setting up Web Gateway include the primary network interface, host name, root password, and others.

For more information, see the relevant sections of the *McAfee Web Gateway Installation Guide.*

After the initial setup you can configure more system settings and also modify the initial settings.

System settings

System settings are configured to control the behavior of the appliance system. They include the settings that were configured at the initial setup and others, for example, settings for domain name services or port forwarding.

Among these settings, the following play an important part:

- **Network interfaces** — Network interfaces are configured to enable processing of web traffic on a Web Gateway appliance. When completing this configuration, you specify the host name, gateways, use of the IPv4 or IPv6 protocol, and other parameters.

  Note: Some of these settings are already configured during the initial setup.
- **Proxies** — Web Gateway can be configured to run as a proxy that intercepts web traffic and transmits it if this is allowed by the rules of your web security policy.

  Proxies can be configured in different ways regarding;

  - **Network mode** — The network mode can be an explicit (also known as direct) proxy mode or a transparent mode.
  - **Network protocol** — The network protocol can be, for example, HTTP, HTTPS, FTP, ICAP, or IFP, to enable the filtering of web traffic that is going on under any of these protocols.
- **Cluster nodes** — Instead of running a Web Gateway appliance in a standalone mode, you can run multiple appliances as nodes in a cluster.

  To configure a cluster with several appliances as nodes, the Central Management system settings are provided.

Configuring policy rules

Configuring policy rules is one of your main activities when working as a Web Gateway administrator.

Policy rules

To protect your network against threats arising from the web, Web Gateway enforces a web security policy, which is implemented during the initial setup. You can configure this policy later on to meet the requirements of your network.

A policy consists of rules, which are grouped in rule sets. Each rule set usually covers a particular field of web security, implementing filtering functions to protect your network.

After the initial setup, default rule sets provide these filtering functions:

- **Anti-malware filtering** — Protects your network against viruses and other malware.
- **URL filtering** — Protects your network against threats caused by accessing inappropriate URLs.
- **Media type filtering** — Protects your network against troubles arising from usage of complex media.

You can exempt web objects, such as hosts and clients, that you consider safe from filtering to make sure they are accessible, using the default rule set for Global Whitelisting.

Some functions that support web filtering, for example, web caching and file opening, are also provided by default rule sets.

More rule sets, which do not run by default on Web Gateway, can immediately be enabled or imported from the built-in or an online library. These rule sets cover other kinds of web filtering or support it. They include, for example:

- **HTTPS scanning** — Protects your network by scanning web traffic going on over SSL-secured connections.

  To enhance web security, private keys for certificates used in secure communication can be stored on a Hardware Security Module.
- **Application control** — Protects your network against threads arising from usage of various applications.

Policy rules can also improve web security not by filtering web objects, but in other ways, for example, by imposing restrictions on users. These include:

- **Authentication** — Protects your network by asking users to authenticate when they request web access.
- **Usage quotas** — Protects your network by imposing quotas for web usage.

Rule sets for these fields of web security and others can be imported from the libraries.

Lists and modules for rules

Rules consist of several elements, which you can configure, including:

- **Lists** — Support rules by listing objects that are relevant for web security, for example, URLs or media types.
- **Modules** — Support rules by handling filtering activities, for example, the Anti-Malware module, which calls engines that scan web objects for infections by viruses and other malware. These modules are also known as *engines* themselves.

  You can configure particular settings for each module.

Cloud use

The rules of your web security policy are applied to the traffic that is created when the users of your organization access and use the web.

Unless you configure it differently, however, these rules are only applied to the web usage of those users who access the web from inside your local network. This kind of usage is also known as *on-premise use*.

You can, however, enable rule sets for *cloud use*. This means that the rules in these rule sets are also enforced when users of our organization access the web from outside your local network, for example, when traveling or working from home.

Enabling cloud use for the rule sets on Web Gateway is also referred to as the *Hybrid Solution*. This solution requires that you also run McAfee Web Gateway Cloud Service.

Monitoring performance

You can monitor an appliance when it executes the filtering that ensures web security for your network.

Monitoring is performed in different ways. Default monitoring on an appliance includes:

- **Dashboard** — Displays key information on the appliance system and activities
- **Logging** — Writes information about important events on an appliance into log files
- **Error handling** — Takes measures when incidents and errors occur on an appliance

You can measure the performance of appliance functions and also use external devices for monitoring, such as a McAfee ePO server or an SNMP Agent.

Troubleshooting issues

Several methods and tools are available for troubleshooting problems on an appliance.

Managing administrator accounts

Administrator accounts are set up and managed on a Web Gateway appliance to control access to the different fields of administrator activities.

High-level steps for configuring Web Gateway

How you configure Web Gateway depends on where you have set it up and what your requirements regarding web security are. There is no fixed order of steps for completing this configuration.

The following high-level steps are suitable for administrators in various environments and with many different purposes.

Task

1. Complete the initial setup.

   During this setup, some basic system settings are already configured, for example, host name, root password, and primary network interface.

   Information about how to complete different types of the initial setup is provided in the *McAfee Web Gateway Installation Guide*.

2. Configure network interfaces.

   A primary network interface is already configured during the initial setup. On the user interface, which is available after this setup, you can configure more of them using the Network Interfaces settings.

   You can configure IP addresses, subnet masks, IP aliases, and other settings for these interfaces under IPv4 or IPv6.

3. Configure proxies.

   Proxies are set up on a Web Gateway appliance to have web traffic redirected to them. This traffic is going on between users' systems that are configured as clients and websites that users request access to. The traffic is filtered and forwarded to its original destinations if the filtering rules allow it.

   Proxies can be set up for the different network protocols that web traffic follows, for example, as HTTP or FTP proxies.

   They can also run in different network modes, for example, in an explicit mode, where the clients are aware that they are redirected, or in a transparent mode, where they are unaware.

   The Proxies settings are provided on the user interface for configuring proxies.

4. Configure a cluster.

   You can run multiple Web Gateway appliances as nodes in a cluster and administer them using the Central Management functions of Web Gateway.

For example, you can add an appliance as a node to a cluster, create node groups, or generate certificates for running web traffic on connections that are secured under the SSL or TLS protocol.

The Central Management settings are provided on the user interface for cluster administration.

5. Configure a web policy.

   A web policy consists of web security rules that are processed to filter traffic that is redirected to Web Gateway.

   Default rules grouped in rule sets that cover different fields of web security are implemented on Web Gateway during the initial setup. They include, for example, an anti-malware rule that blocks the download of malware infected files to a users's system within your network.

   You can modify or delete existing rules, import rules from libraries. and create your own rules. The Policy top-level menu of the user interface provides submenus with these functions

System configuration

The system of a Web Gateway appliance is configured to support the filtering functions that protect your network against threats arising from the web.

When performing this configuration, you will mainly be dealing with system settings and files.

Some system settings are already configured during the initial setup of Web Gateway, others can be configured later on the user interface.

Initial system settings

System configuration is in part performed during the initial setup of Web Gateway.

Settings that are configured during this setup include the primary network interface, host name, root password, and others.

System settings

After the initial setup you can configure more system settings and also modify the initial settings.

This includes configuring system settings for:

- **Network interfaces** — Network interfaces are configured to enable processing of web traffic on a Web Gateway appliance, specifying the host name, gateways, use of the IPv4 or IPv6 protocol, and other settings.

  Note: Some of these settings are already configured during the initial setup.
- **Proxies** — Web Gateway can be configured to run as a proxy that intercepts web traffic and transmits it if this is allowed by the rules of your web security policy.

  Proxies can be configured in different ways regarding;

  - **Network mode** — The network mode can be an explicit (also known as direct) proxy mode or a transparent mode.
  - **Network protocol** — The network protocol can be, for example, HTTP, HTTPS, FTP, ICAP, or IFP, to enable the filtering of web traffic that is going on under any of these protocols.
- **Cluster nodes** — Instead of running a Web Gateway appliance in a standalone mode, you can run multiple appliances as nodes in a cluster.

  To configure a cluster with several appliances as nodes, the Central Management system settings are provided.
- **Update schedules** — Updates are scheduled to ensure hat the latest available information is used by the filtering functions on Web Gateway.

System files

System files contain particular parameters of the appliance system. They can be modified using the File Editor.

Additional activities

System configuration can also include several other activities.

- **Network interface bonding** — Bonding two or more network interfaces enables them to act as one while increasing bandwidth and providing High Availability.
- **Cache volume resizing** — Logical volumes for web caching and for storing temporary and log files can be resized on an appliance using a wizard.
- **Closed networks** — Web Gateway appliances can be operated and updated in networks that have no internet connectivity for security or other reasons. These networks are also known as "closed" or "isolated" networks.

Update handling

Information retrieved from databases and lists for use in the filtering process must be updated from time to time.

Web objects are filtered on an appliance in a rule-based process. The filtering rules require information about these objects to know whether an action must be executed, such as blocking access to an object. They rely for this information on particular modules (engines).

For example, a malware filtering rule relies on the Anti-Malware module to find out whether an object is malware-infected. A URL filtering rule relies on the URL Filter module for category information.

The modules retrieve this information from particular sources, such as databases and lists. An example for these sources are the virus signatures used in anti-malware filtering, which are stored in DAT files and located in an external database.

Another example is the list of public domain name suffixes. When a URL is filtered, this list is used to find the domain suffix based on the host name within the URL.

Update methods are:

- **Manual engine update** — Information is manually updated for the modules of a particular appliance.
- **Automatic engine update** — Information is updated by automatic procedures running in regular intervals for the modules of a particular appliance.

  These updates can retrieve information:

  - **From the internet** — Information is downloaded from external databases.

    Note: Information is for the first time updated in this way immediately after the initial setup of an appliance.
  - **From other nodes in a cluster** — Information is downloaded from other nodes in a Central Management cluster. You can configure for each node whether uploading information from this node to others is allowed.

  You can already configure these updates when setting up a Central Management cluster.

Update database information manually

You can update database information for the modules of an appliance manually.

The update applies to the modules of the appliance you are logged on to and to those of other appliances that you have included as nodes in a Central Management configuration.

Task

1. Select Configuration → Appliances.
2. On the appliances toolbar, click Manual Engine Update.

   The update is performed.

Schedule automatic engine updates

You can schedule automatic updates of database information for the modules of an appliance.

When you are running multiple appliances as nodes in a Central Management configuration, you can schedule updates for the modules (also known as *engines*) on the nodes as part of configuring settings for this configuration.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to schedule automatic updates on and click Central Management.
3. Scroll down to Automatic Engine Updates and configure update settings as needed.
4. Click Save Changes.

System files

System files contain settings for functions of the appliance system. You can edit these settings using the File Editor.

The settings that are stored in system files include settings of parameters the appliance system uses for network communication, for example, IP addresses, the maximum message size, or the maximum number of messages in a queue.

Other settings are used to configure functions of the appliance system such as logging, access restrictions, and others.

An example for a system file is the */etc/hosts* file, which contains entries for IP addresses and host names, including the local IP address and host name of the appliance itself.

The File Editor allows you to edit the settings in these files. It is accessible on a tab of the user interface.

Caution: To edit system files, only use the File Editor. If you open these files outside the File Editor to edit them manually, your changes will be overwritten when an upgrade to a new version of Web Gateway is performed.

Network interface bonding

Bonding two or more network interfaces enables them to act as one while increasing bandwidth and providing High Availability.

The network interfaces on Web Gateway, for example, the eth2 and eth3 interfaces, can be bound together to form a single channel. A bonding kernel module is created this way and made accessible through a common network interface, which is referred to as the bonding interface.

The network interfaces that are bound together under the bonding interface are referred to as the bonded interfaces. These interfaces can be provided by different NICs.

The terms "master" and "subordinate" are also used to refer to a bonding and a bonded interface, respectively. In some system messages, you will also see the term "slave" used for a bonded interface.

Note: With regard to the components and processes that are involved, network interface bonding is also known as NIC bonding, ethernet bonding, or channel bonding.

You can configure network interface bonding on the user interface of Web Gateway. To verify that a bonding interface has successfully been configured, you can run some suitable commands from a system console.

A VLAN can be configured on a bonding interface in the same way as on an ordinary network interface, using the relevant configuration options of the user interface.

Note: When the transparent bridge or router mode are configured for a network, network interface bonding cannot be implemented.

Configure network interface bonding

To configure network interface bonding, create a bonding interface and configure parameters for this interface and the bonding configuration.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure network interface bonding on and click Network Interfaces.

   The Network Interfaces settings appear in the configuration pane.
3. Create a bonding interface.
   a. Under Enable these network interfaces, select a network interface that you want to run as a bonded interface, for example, eth2.
   b. Under IP settings, select Disable IPv4.
   c. On the Advanced tab, select Bond enabled and in the Name field type the name of the bonding interface that you want to create, for example, `bond1`.

   Repeat substeps a, b, and c for another network interface that you want run as a bonded interface under this bonding interface.

Note: You can also add further network interfaces as bonded interfaces and have more than two network interfaces in the bonding configuration.

   d. Click Save Changes.
   e. Log out and log on again.
      After the logon, the new bonding interface appears in the list under Enable these network interfaces.

4. Configure parameters for the bonding interface.
   a. Select the bonding interface and click the IPv4 or IPv6 tab, according to the protocol version that is used in your network.
   b. Select Configure manually and under IP address and subnet mask type an IP address and the values for a subnet mask.
      You can leave the default value under MTU, which specifies the maximum number of bytes in a single transmission unit, as it is.
5. Configure parameters for the bonding configuration.
   a. Select the bonding interface and click the Advanced tab.
   b. Under Mode, select one of the following bonding modes.
      - Active/Passive — In this mode, only one bonded interface in the bonding configuration is active at any time. A different bonded interface becomes active only if the active bonded interface fails.

        The MAC address of the bonding interface is only visible externally on one port, which avoids address confusion for a network switch.

        Note: This mode is referred to in some system messages as *mode 1*.

        The mode is selected by default.
      - 802.3ad/LACP — In this mode, all bonded interfaces in the bonding configuration are active.

        The bonded interface for outgoing traffic is selected according to the configured hash policy.

        Note: This mode is referred to in some system messages as *mode 4*.

        When this mode is selected, the LACP rate and Hash policy options become accessible.
   c. Under Miimon, configure monitoring for the bonding interface.

      The value that you configure here sets the time interval (in milliseconds) for sending the polling messages of the MII monitoring program.

      The default interval is 100 milliseconds.
   d. If you have selected 802.3ad/LACP as bonding mode, select options that are specific to this mode.

      Under LACP rate, select the transmission rate for the LACP-DU data packets that are exchanged between bonding and bonded network interfaces.
      - Slow — With this transmission rate, data packets are sent every 30 seconds.

        This transmission rate is selected by default.
      - Fast — With this transmission rate, data packets are sent every second.

      Under Hash policy, select one of the following options.
      - Layer2 — This policy uses a combination of layer 2 values to calculate the hash. The values that are included in this combination are hardware MAC addresses and packet type ID addresses.

        This hash policy is selected by default.
      - Layer2+3 — This policy uses a combination of layer 2 and layer 3 protocol information to calculate the hash.
6. Click Save Changes.

Checking the bonding configuration

You can verify that you have successfully configured a bonding network interface from a system console.

To verify that the bonding configuration runs with the parameters that you have configured, you can use a suitable network script. An additional command enables you to check the status of the bonding interface and the network interfaces that are bound to it.

Verifying the configuration parameters

The *ifcfg* network script allows you to verify that the network interfaces of the bonding configuration are running with the configured parameters, such as the bonding mode or the IP address of the bonding interface.

To view the parameters for the bonding interface, for example, *bond 1*, run the network script using the following command:

```
cat /etc/sysconfig/network-scripts/ifcfg-bond1
```

The command returns, for example, the following lines.

```
### BEGIN AUTOGENERATED CONFIG BONDING_OPTS:='mode=1 miimon=600' BOOTPROTO='none' DEVICE='bond1'
IPADDR='10.11.12.12' ...
```

To view the parameters for a bonded interface, for example, *eth2 1*, run the following command:

```
cat /etc/sysconfig/network-scripts/ifcfg-bond1
```

The command returns, for example, the following lines.

```
### BEGIN AUTOGENERATED CONFIG BOOTPROTO='none' MASTER='bond1' SLAVE:'yes' DEVICE='eth2' ...
```

Checking the network interface status

You can check whether the bonded network interfaces are running properly under the bonding interface and which of the bonded interfaces is currently in active (slave) status.

Run the following command, for example, if the bonding interface is *bond1*:

```
cat /proc/net/bonding/bond1
```

The command returns, for example, the following lines.

```
### Ethernet Channel Bonding Driver: v. 3.7.1 (April 27, 2015) Bonding Mode: fault-tolerance (active-backup)
Primary Slave: None MII Status: up MII Polling Interval (ms): 600 Up Delay (ms): 0 Down Delay (ms): 0 Slave
Interface: eth2 MII Status: up Speed: 1000 Mbps Duplex: full Link Failure Count: 0 Permanent HW Addr: 00:0c:
29:e0:a7:37 Slave Queue ID: 0 Slave Interface: eth3 MII Status: up ...
```

Source-based routing

When configuring routing for traffic in your network, you can let routing decisions be based on the source IP address. This routing method is known as source-based routing.

Using this method you can separate the management traffic that an administrator creates when accessing the user interface of a Web Gateway appliance from the traffic that the administrator or end users create when accessing the web. The two kinds of traffic can also be protected by a separate firewall for each of them.

To implement the method, you allow administrator access to the user interface only through a particular network interface on the appliance. This network interface is the management network interface, while a different network interface is configured for access to the web.

You can also configure that monitoring information, for example, SNMP messages, must access the appliance through the management network interface.

After passing through the management interface, traffic can be identified for further routing by its source IP address, which is the address of the management interface.

Configuring the routing for this traffic includes two main steps:

- Configuring a routing table
- Configuring a route within this table

The source IP address is specified in both steps to ensure that traffic with this address is routed according to a particular table and route.

Different routing tables can be configured and entered in a list on Web Gateway while different routes can be configured for each table.

You can configure routes for use under IPv4 or IPv6, depending on which version of this protocol is followed within your network.

Configure source-based routing for a management network interface

Configure source-based routing to separate other traffic from traffic that has a management network interface as its source.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure source-based routing on.
3. Configure use of the management network interface for administrator access to the user interface.
   a. Click User Interface.
   b. Under HTTP Connector, proceed as follows.
      - Make sure Enable local user interface over HTTP is selected.
      - In the HTTP connector field, type the IP address and listener port of the management network interface.
4. Configure use of the management network interface for SNMP messages.
   a. Click SNMP.
   b. Under SNMP Port Settings, click the Add icon on the toolbar of the Listener address list.
      The Add SNMP Listeners window opens.
   c. In the Listener address field, type the IP address and listener port of the management network interface.
   d. Click OK.
      The window closes and the listener address appears in the list.
5. Configure source-based routing for traffic that is sent and received through the management network interface.
   a. Click Static Routes.
   b. Under Source-based routing, select Source-based routing for IPv4 or Source-based routing for IPv6, depending on the IP version used in your network.
      Two lists for configuring source-based routing appear.
   c. On the toolbar of the Static source routing table number list, click the Add icon.
      The Add ApplianceSourceBasedRoutingTable window opens.
   d. Configure an entry for the routing table as follows.
      - In the Source information to look up routing table field, type the IP address of the management network interface.
      - In the Routing table number field, type the number of the routing table for the traffic that is sent and received through the management network interface.
   e. Click OK.
      The window closes and the routing table entry appears in the list.
   f. On the toolbar of the Source-based routing list for IPv4 (or the list for IPv6), click the Add icon.
      The Add ApplianceSourceBasedRoutingIPv4 window (or the window for IPv6) opens.
   g. Configure a routing entry as follows.
      - In the Destination field, type the IP address range in CIDR notation for the destinations of the traffic that is sent through the management network interface.
      - In the Routing table number field, type the number of the routing table for the traffic that is sent and received through the management network interface.
      - In the Gateway field, type the IP address of the gateway for the traffic that is sent and received through the management network interface.
      - In the Device field, type the name of the network interface that you want to configure as the management network interface.
      - In the Source IP field, type the IP address of the network interface that you want to configure as the management network interface.

h. Click OK.

   The window closes and the routing entry appears in the list.

6. Click Save Changes.

Cache volume resizing

Logical volumes for web caching and for storing temporary and log files can be resized on an appliance using a wizard.

After Web Gateway has been installed on an appliance, the logical volume for web caching is larger than that for storing temporary and log files. The appliance volume wizard is provided, which allows you to change this sizing and provide more disk space for storing temporary and log files.

Volume size is shown on the wizard pages in GiB. Before the resizing, sizes could be, for example, as follows:

- Web cache volume: 197 GiB
- Temporary and log files volume: 40 GiB

After the resizing the size relation is inverted:

- Web cache volume: 40 GiB
- Temporary and log files volume: 197 GiB

The wizard guides you through the resizing when you set up a Web Gateway appliance for the first time. After completing work with the configuration wizard that is provided for configuring initial system settings, the appliance restarts and the wizard appears.

If the wizard process is interrupted, you can restart it from the command line of a system console using the following command:

```
mwg-cache-wizard
```

When the `yum upgrade` command is used to set up an appliance, the wizard must also be started manually.

The path and file name for the main log that records the activities of the wizard are */var/log/resize-cache.log*.

If the resizing has already been performed on an appliance, the wizard displays a corresponding message.

If you still need to resize the appliance volumes, contact McAfee support.

Closed networks

Web Gateway appliances can be operated and updated in networks that have no internet connectivity for security or other reasons. These networks are known as "closed" or "isolated" networks, and sometimes also as "dark" networks.

When appliances that run in these networks need to be updated, they cannot connect to the usual McAfee update servers. An offline update procedure must be performed instead.

You can select and download an update package from a McAfee portal that is provided for this purpose, store it on portable media and use this media to apply the update package to one or more appliances in a closed network.

Update packages contain updated information for modules (engines) and malware patterns used in the filtering process on an appliance. Only full updates (as opposed to incremental updates) are made available on the portal.

After entering the portal, you need to submit the version number of Web Gateway on the appliance you want to update, and are provided with a list of features that updated information is currently available for.

According to your selection, an update package including all files required for the update is created in zipped format for downloading.

Update an appliance in a closed network

To update an appliance in a network with no internet connectivity, download an update package, store it on portable media, and use the media to perform the update.

Task

1. Download an update package.
    a. Use a browser to go to the update page of the Content & Cloud Security at: https://contentsecurity.mcafee.com/update
    b. On the update page, enter the version number for an appliance you want to update.
    A list of features that updated information is available for appears.
    c. Select the features you want to update.
    An update package is created according to your selection.
    d. Download the update package to your system.
2. Use portable media, for example, a USB drive, to transfer the update package from the system you used for the download to your administration system in the closed network.
3. For each appliance in the closed network that you want to update, perform the following steps:
    a. Select Configuration → Appliances.
    b. Click Update Engines, then select Upload Update File.
    The Engine Update by File Upload window opens.
    c. Click Browse, go to the location on the administration system where you stored the update package, and select the update package file.
    d. Click Update.
    The appliance is updated using the information from the update package.
    e. Click Close to close the window.

Proxies

A Web Gateway appliance uses its proxy functions to intercept web traffic sent by its clients and transmit it if this is allowed by the filtering rules.

Default settings are in place for these functions after the initial setup. You can configure and extend them to meet the requirements of your network.

Using these functions, one or more proxies can be run on Web Gateway to cover web traffic going on under different network protocols. With regard to the use of its proxy functions, Web Gateway is also itself referred to as proxy.

How Web Gateway uses its proxy functions depends on the way you have set it up within your network. Different modes can be configured to account for different setups.

So, the following are key settings when configuring proxies:

- **Network mode** — Is configured depending on how you have set up Web Gateway within your network.

  There is a network mode for running a proxy that clients must be explicitly made aware of if web traffic originating from them is to be redirected to it. For other network modes, this is handled transparently, which means the clients are not aware that their web traffic is redirected.
- **Network protocol** — Is configured depending on the type of web traffic that Web Gateway should intercept and transmit.

  Proxies can be set up for web traffic going on under HTTP, HTTPS, FTP, and other protocols.

There are several other proxy settings that you can configure, for example, timeouts or the maximum number of client connections. You can also implement special proxy solutions, for example, a reverse HTTPS proxy configuration.

Configure proxies

Configure the proxy functions of a Web Gateway appliance to meet the requirements of your network.

Task

1. Review the proxy settings.
   a. Select Configuration → Appliances
   b. On the appliances tree, select the appliance where you want to review the proxy settings. Then click Proxies

   These key settings are configured by default:

   ◦ Network mode: Explicit Proxy

     In this network mode, the clients must be explicitly configured to have their web traffic redirected to a proxy on Web Gateway.

     The mode is enabled when Proxy (optional WCCP) is selected under Network Setup.

     Use of WCCP is, however, not enabled by default. If you enable it or the L2 transparent redirection method, the network mode is no longer explicit, but becomes transparent.

   ◦ Network protocol: HTTP

     This protocol is enabled when Enable HTTP Proxy is selected under HTTP Proxy.
2. Modify these default settings as needed.

   You can select a transparent instead of the explicit proxy network mode. You can also enable other proxies while keeping or disabling the HTTP proxy.
3. Configure other proxy settings, for example, timeouts or the maximum number of client connections.
4. Save your changes.

Network modes

When setting up a proxy on Web Gateway, you set it up to run in a particular network mode.

These network modes include:

- Explicit proxy mode
- Transparent Proxy with WCCP mode
- Transparent Proxy with L2 transparent mode
- Proxy HA mode
- Transparent Router mode
- Transparent Bridge mode

Network Setup settings

Settings for implementing a network mode

When a network mode is selected, specific settings for this mode appear below these settings.

Network Setup

Option	Definition
Proxy (optional WCCP)	When selected, the explicit proxy mode is used and WCCP services can redirect web traffic to an appliance.
Proxy HA	When selected, the explicit proxy mode with High Availability functions is used for running proxies.
Transparent Router	When selected, the Transparent Router mode is used for running proxies.
Transparent Bridge	When selected, the Transparent Bridge mode is used for running proxies.

Explicit proxy mode

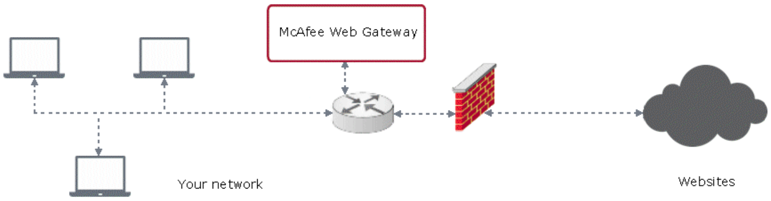
In explicit proxy mode, the clients that have their web traffic filtered on the appliance “know” they are connected to it. They must explicitly be configured to direct their web traffic to the appliance.

Explicit proxy mode configuration

If the clients of an appliance are explicitly configured to direct their web traffic to it, it is less important where the appliance is deployed within your network. Typically, it is placed behind a firewall and connected to its clients and the firewall by a router.

The following diagram shows a configuration in explicit proxy mode.

Explicit proxy mode



Configure the explicit proxy mode

You can configure the proxy functions of an appliance in explicit proxy mode, which is the default mode for these functions.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure the explicit proxy mode for and click Proxies (HTTP(S), FTP, ICAP, and IM).
3. Under Network Setup, select one of the two options for the explicit proxy mode.
    - Proxy — For the explicit proxy mode

      Note: This is the default proxy mode.

      When it is selected, specific settings for configuring transparent features of the explicit proxy mode appear below the Network Setup settings.
    - Proxy HA — For an explicit proxy mode with High Availability functions

      After selecting this option, specific Proxy HA settings appear below the Network Setup settings.
4. Configure specific and common settings for the selected option as needed.
5. Click Save Changes.

Transparent Proxy settings

The Transparent Proxy settings are used for configuring transparent features of the explicit proxy mode.

Transparent Proxy

Settings for configuring the explicit proxy mode with transparent features

Transparent Proxy

Option	Definition
Supported client redirection methods	Provides methods for intercepting web traffic and redirecting it to an appliance. • WCCP — When selected, HTTP client requests sent to web servers under IPv4 and IPv6 are intercepted by an additional network device and redirected to the appliance using the Web Cache Communication Protocol (WCCP).

Option	Definition
	The clients are not aware of the redirection, it remains transparent for them. In the same way as for client requests, responses from web servers are directed back to the appliance. When using the WCCP redirection method, you need to configure one or more WCCP services on the appliance to let them perform the redirection. You also need to configure the network device that intercepts the client requests and server responses. This device can be configured as a router or switch with routing functions. After selecting this option, the WCCP Services inline list appears for configuring and adding WCCP services. • L2 transparent — When selected, client requests sent to a web server under IPv4 and IPv6 are intercepted by an additional network device and directed to the appliance using the Layer 2 redirection method. Under this method, client requests are accepted on the appliance even if their destination IP addresses are not addresses of the appliance. The redirection is transparent to the clients. You need to enter the original ports for those client requests that are to be intercepted and redirected in a list on the appliance together with the ports that these requests are redirected to. The additional network device must be configured accordingly. When this option is selected, requests can not be transmitted using a connection in active FTP mode. Only the passive FTP mode is then available. After selecting this option, the Port Redirects inline list appears for entering ports.

The following two tables describe list entries in the lists of WCCP services and port redirects.

Advanced Outgoing Connection Settings

Settings specifying methods for handling information contained in client requests sent to web servers that are requirements for the network environment of the appliance

Advanced Outgoing Connection Settings

Option	Definition
IP spoofing (HTTP, HTTPS, FTP)	When selected, the appliance keeps the client IP address that is contained in a client request as the source address and uses it in communication with the requested web server under various protocols. When WCCP services are used for intercepting web traffic and directing it to the appliance, you need to configure two services for each port on the appliance that listens to client requests: one for the requests that come in from the clients, and one for responses to these requests that are sent by the web servers. When this option is not selected, the appliance chooses a source port and uses it in this communication.

Option	Definition
	• IP spoofing for explicit proxy connections — When selected, client addresses are kept in explicit proxy mode, in which web traffic is not intercepted by an additional device. • Use same source port as client for IP spoofing — When selected, client source ports are kept and used in addition to client source addresses for communication with web servers. When this option is not selected, the appliance chooses a random source port and uses it in this communication.
HTTP: Host header has priority over original destination address (transparent proxy)	When selected, the destination address that is provided in the HOST header part of a client request under HTTP is used for communication with the requested web server. In a transparent proxy configuration, communication with a web server could also use the destination address that is specified under TCP for the connection that serves to transmit a client request. This address is also known as the *original* destination address. Both methods of communication are available to a transparent proxy on an appliance that intercepts client requests or to a WCCP service that intercepts requests and redirects them to an appliance. Using the HOST header destination address is the preferred method, however, for some configurations it can be necessary to deselect this option and use the original destination address for communication with a web server. • If web traffic is processed on multiple appliances with transparent proxies running on them and client requests are routed to them according to destination addresses, it must be ensured that the proxies use the original destination addresses when connecting to web servers. • This applies also if a WCCP service intercepts client requests and redirects them to multiple appliances, using destination addresses for load distribution.

Sample WCCP service settings for IP spoofing

Sample settings for configuring WCCP services with IP spoofing

Note: Configure these settings only if you want to perform IP spoofing. It is usually not required that you configure two services for redirecting web traffic to the appliance under WCCP.

You can use IP spoofing in a configuration with WCCP services that intercept web traffic and direct it to the appliance. In this case, you need to configure two services for all ports on the appliance that listen.

One of these services is for the requests that come in from the clients and another one for the responses to these requests that are sent by the web servers.

The following table shows sample parameter values for these services.

Sample parameter values for two WCCP services configured with IP spoofing

Option	Service for client requests	Service for web server responses
Service ID	51	52
Service priority	0	0
WCCP router definition	10.150.107.254	10.150.107.254

Option	Service for client requests	Service for web server responses
Ports to be redirected	80, 443	80, 443
Ports to be redirected are source ports	false	true
Proxy listener IP address	10.150.107.251	10.150.107.251
Proxy listener port	9090	9090
MD5 authentication key	* * * * *	* * * * *
Input for load distribution	*This main item does not appear in the settings list, but is visible in the Add and Edit windows. The following four elements are related to it*	
Source IP	true	false
Destination IP	false	true
Source port	true	false
Destination port	false	true
Assignment method	*This main item does not appear in the settings list, but is visible in the Add and Edit windows. The following four elements are related to it*	
Assignment by mask	true	true
Assignment by hash	false	false
Assignment weight	100	100
Forwarding method	*This main item does not appear in the settings list, but is visible in the Add and Edit windows. The GRE-encapsulated and L2-rewrite to local NIC elements are related to it*	
GRE-encapsulated	false	false
L2-rewrite to local NIC	true	true
L2-redirect target	eth1	eth1
Magic (Mask assignment)	-1	-1
Comment		

Transparent Proxy with WCCP mode

When implementing a proxy on a Web Gateway appliance, you can configure it to run in Transparent Proxy mode and use WCCP (Web Cache Communication Protocol) to redirect web traffic from the clients to Web Gateway.

Use of this protocol considerably enhances the capabilities for load balancing and failover.

Alternatively, you can use of the L2 (Layer 2) method to redirect web traffic. You can also configure both methods for redirecting web traffic at the same time.

Configure the Transparent Proxy with WCCP mode

To configure the Transparent Proxy with WCCP protocol, configure a router and one or more Web Gateway appliances for handling web traffic according to this protocol.

Task

1. Configure a router for handling web traffic according to the WCCP protocol.
   Configuring the router mainly includes specifying the ID of the WCCP service. For more information, see the router documentation.
2. Configure a Web Gateway appliance for handling web traffic according to the WCCP protocol.

   Configuring the appliance mainly includes setting up a WCCP service on it.

   a. Select Configuration → Appliances.

   b. Select the appliance that you want to configure for use of the WCCP protocol and click Proxies (HTTPS, FTP, SOCKS, ICAP ...).

   c. Under Network Setup make sure that Proxy (optional WCCP) is selected. Under Transparent Proxy select WCCP.
      The WCCP services list appears.

   d. On the list toolbar, click the Add icon.
      The Add WCCP Service window opens.

   e. To add a service, provide values for the service parameters. When you are done, click OK.
      The new service appears in the WCCP services list.

   f. Click Save Changes.

   You can include more than one appliance in a WCCP configuration. Configure a WCCP service on every appliance that you want to include.

   Note:

   When configuring the explicit proxy mode with WCCP after using a different network mode before (Proxy HA mode, transparent router or bridge mode), you must restart the appliance.

   The restart unloads the network driver that handles the transparent interception and redirection of web traffic. Restarting is only required once. Later on you can enable and disable use of the WCCP protocol without restarting.

Transparent Proxy settings (for use with WCCP)

Settings for the Transparent Proxy mode when using WCCP services to redirect web traffic

Transparent Proxy

Option	Definition
Supported client redirection methods	Lets you select a method for redirecting web traffic. • WCCP — When selected, HTTP client requests sent to web servers under IPv4 and IPv6 are intercepted by an additional network device and redirected to the appliance using the Web Cache Communication Protocol (WCCP). The clients are not aware of the redirection, it remains transparent for them. In the same way as for client requests, responses from web servers are directed back to the appliance. When using the WCCP redirection method, you need to configure one or more WCCP services on the appliance to let them perform the redirection. You also need to configure the network device that intercepts the client requests and server responses. This device can be configured as a router or switch with routing functions.

Option	Definition
	After selecting this option, the WCCP Services inline list appears for configuring and adding WCCP services. **Note:** After selecting this option, the WCCP Services list appears below where you can configure and add WCCP services. • L2 transparent — When selected, client requests sent to a web server under IPv4 and IPv6 are intercepted by an additional network device and directed to the appliance using the Layer 2 redirection method. Under this method, client requests are accepted on the appliance even if their destination IP addresses are not addresses of the appliance. The redirection is transparent to the clients. You need to enter the original ports for those client requests that are to be intercepted and redirected in a list on the appliance together with the ports that these requests are redirected to. The additional network device must be configured accordingly. When this option is selected, requests can not be transmitted using a connection in active FTP mode. Only the passive FTP mode is then available. **Note:** After selecting this option, the Port Redirects list appears below where you can configure and add port for redirecting web traffic.

The following table describes the fields of an entry in the list of WCCP services used for redirecting web traffic.

WCCP Services – List entry

Option	Definition
Service ID	Identifies a service that redirects web traffic to an appliance under WCCP.
Service priority	Sets the priority for a WCCP service. When web traffic is redirected to Web Gateway under WCCP, a WCCP service handles the redirection. An incoming data packet is then assigned to the service with the highest priority. The priority value for a particular WCCP service is communicated by Web Gateway along with other information in the heartbeat messages that it sends to the WCCP router or switch in short intervals. Priority range: 0 - 255 Default: 0
WCCP router definition	Specifies the Multicast IP address and DNS name of a router (or switch with routing functions) that uses a WCCP service to direct web traffic to an appliance. You can configure multiple routers here, separating entries by commas.
IP protocol version preference for name resolution	Selects the IP version that is preferred when resolving a host name to an IP address.

Option	Definition
	The host name is the name of the host system that a request with data packets was sent from. The data packets are redirected using a WCCP router that Web Gateway registers with under the IP address that the host name is resolved to. • IPv4 — When selected, IPv4 is the preferred protocol version. • IPv6 — When selected, IPv6 is the preferred protocol version. • Use other protocol version as fallback — When selected, the host name is resolved using the other protocol version if the preferred version is not available. For example, if a router supports only IPv4 addresses, the host name is resolved to an IPv4 address even if you selected IPv6 as the preferred version.
Ports to be redirected	Lists the ports, for example, on web servers, that data packets must have in their address information to be redirected. You can specify up to eight port numbers here, separated by commas.
Ports to be redirected are source ports	Specifies whether the ports that are to be redirected are source ports. When configuring a WCCP service, you need to select this option if the service is used to redirect responses from web servers back to the appliance.
Proxy listener IP address	Specifies the IP address of an appliance when serving client requests.
Proxy listener port	Specifies a port for listening to client requests. The default port number is 9090.
MD5 authentication key	Sets a password used under the MD5 algorithm for signing and verifying control data packets. The Set button opens a window for setting the password. The password can have up to eight characters.
Assignment method	*This main item does not appear in the list, but is visible in the Add and Edit windows. The following two elements are related to it, specifying the assignment method.* • Assignment by mask — When selected, masking of the source or destination IP addresses is used for load distribution. • Assignment by hash — When selected, a hash algorithm is used for load distribution.
Input for load distribution	*This main item does not appear in the list, but is visible in the Add and Edit windows. The following elements are related to it, specifying what is used in a data packet as the criteria for load distribution* *Different elements are provided, depending on whether you have selected assignment by mask or hash.* When running multiple appliances, load distribution can be configured for the proxies on them. Data packets can be

Option	Definition
	distributed to these proxies based on their source or destination IP addresses and port numbers. When source or destination IP addresses are used for load distribution, they can be masked or a hash algorithm can be applied to them, see the options under *Assignment method*. When source or destination ports are used, only the hash algorithm method can be selected. *Load distribution elements for assignment by mask:* • Source IP mask — Specifies the mask for a source IP address. The default mask value is *0x15*. • Destination IP mask — Specifies the mask for a destination IP address. The default mask value is *0x15*. The maximum mask length is 4 digits, for example, *0xa000*. For both masks together, 6 bit can be set as a maximum. If a mask is set to *0x0*, it does not influence load distribution. So, if you want to use, for example, only source IP addresses for load distribution, you need to set the mask for destination IP addresses to this value. *Load distribution elements for assignment by hash:* • Source IP — When selected, load distribution is based on source IP addresses. • Destination IP — When selected, load distribution is based on destination IP addresses. • Source port — When selected, load distribution is based on source port numbers. • Destination port — When selected, load distribution is based on destination port numbers. When configuring one WCCP service for handling client requests and another for handling web server responses, you need to select Source IP and Destination IP in a "crosswise" corresponding manner. This means that if you select Source IP for the client requests service, you *must* select Destination IP for the web server responses service. If you select Source IP for the web server responses service, you *must* select Destination IP for the client requests service, and so on. The same applies when selecting Source port and Destination port.
Assignment weight	Sets a value to determine how much load is assigned to a proxy. Use this value to assign more load to a proxy on an appliance that has more CPU capacity. 0 means no load is distributed to a proxy.
Forwarding method	*This main item does not appear in the list, but is visible in the Add and Edit windows. The following two elements are related to it, specifying the forwarding method.* • GRE-encapsulated — When selected, data packets are encapsulated by the router before being redirected. When this element is selected, the following two elements are also shown:

Option	Definition
	◦ GRE-redirect target — Lets you select a network interface on an appliance. All data packets that were received on this network interface are redirected to the listener port that is configured for this WCCP service. ◦ Redirect for all interfaces — When selected, data packets are redirected to the same listener port regardless of the network interface that they were received on. • L2-rewrite to local NIC — When selected, data packets are redirected to the appliance by replacing the MAC address of the next device on the route to the web server with that of the appliance. When this element is selected, the following element is also shown: ◦ L2-redirect target — Lets you select a network interface on an appliance. All data packets that were received on this network interface are redirected to the listener port that is configured for this WCCP service.
Magic (Mask assignment)	Lets you set an unknown field in the mask that an appliance sends to the router. This setting is needed for ensuring compatibility with different versions of the vendor's operating system, which is used for the router. **Note:** This element is only provided if Assignment by mask was selected as the assignment method.
Comment	Provides a plain-text comment on a WCCP service.

Redirecting web traffic under WCCP

When implementing the explicit proxy mode on a Web Gateway appliance, you can configure the redirection of web traffic to Web Gateway under WCCP (Web Cache Communication Protocol). Use of this protocol considerably enhances the capabilities for load balancing and failover.

To enable redirection under WCCP, a suitable router must be placed between the client systems of the users in your network and the web. The router redirects requests for web access from the clients that are directed to particular ports to the Web Gateway appliance.

The router is also referred to as the WCCP device. Instead of a router, you can also use a switch as WCCP device.

On the appliance, you must configure a WCCP service. When configuring this service, you specify a service ID, the IP address of the router, the ports that requests are redirected from, and other information.

Multiple appliances can connect to the same router under WCCP for load balancing and failover. The appliances must be configured as nodes in a Central Management configuration and a WCCP service must be configured on each of them.

The redirection happens transparently, which means users are not aware that their requests are redirected. When the response to a request is received from a web server, Web Gateway forwards it to the client, using (spoofing) the IP address of the web server.

To start working with the router, Web Gateway subscribes to it. The router is not aware of Web Gateway until the subscription happens. No settings must be configured on the router to inform it about Web Gateway.

Communication between Web Gateway and the router

Under WCCP, data packets are exchanged to subscribe, negotiate settings, and as health checks. Web Gateway sends a "Here I Am" packet to the router and forwards the configured settings. These settings include the ports for redirection, the ID of the WCCP service, the IP address that traffic should be redirected to, and other information.

The router acknowledges with an "I See You" packet that the subscription has been successful and includes the router ID, which is the highest interface IP address on the router.

If a router does not receive a "Here I Am" packet over more than 25 seconds, it sends a removal query, requesting that Web Gateway respond immediately. If no response is received within another 5 seconds, Web Gateway is considered offline and removed from the pool of WCCP partners.

Load balancing and failover

In a WCCP configuration with multiple Web Gateway appliances, the first appliance that connects to the router distributes workload to the other appliance. Portions of workload that are distributed are also known as "buckets" in WCCP terminology.

When an appliances goes offline or returns, buckets are immediately reassigned. If the appliance that is currently assigning buckets goes offline, another appliance takes over its role.

We do not recommend using WCCP when the router, client systems, or the Web Gateway appliances are separated by a device that uses the method known as source NATing to handle client traffic. This method impacts the performance of load balancing under WCCP. It also prevents you from configuring rules for user authentication based on time or client IP addresses.

Fail-open and fail-closed strategies

If use of the WCCP protocol is configured on the router and no Web Gateway appliance is available, the router lets requests for web access pass through without redirection. This behavior follows a strategy known as *fail-open strategy*.

If you have a firewall in your network, you must configure it to allow requests for web access with any source IP addresses to enable this strategy. Requests can then go out to the web directly.

Under a *fail-close strategy*, requests are blocked if no Web Gateway appliance is available to redirect them to. For this strategy to work, you must configure the firewall to allow only requests with source IP addresses belonging to Web Gateway.

Using WCCP only or as fallback solution

You can use the explicit proxy mode with WCCP as your only network mode solution, which means all web traffic is handled in this mode. You can also use it as a fallback solution for special use cases in an explicit proxy configuration, for example, to deal with applications that do not recognize proxy settings. Another use case would be handling web traffic in a Wi-Fi network segment where users can bring their own devices.

As best practice, we recommend using two different proxy ports. Configure one for handling web traffic in explicit proxy mode with WCCP, and one for handling it without WCCP. Following this practice allows you to use the property for proxy ports in the criteria of web security rules.

Best practice: Tips for configuring WCCP service settings

Configuring the settings for a WCCP service includes setting service parameters.

For these parameters, consider the best-practice information that is provided here.

Service ID

The service ID identifies the WCCP service. The service is also included in the router configuration, where its ID must be the same.

Service IDs 0–50 are static under WCCP and reserved for *well known services* with standard configurations. Service IDs 51–255 are dynamic and involve negotiation between the partners in the WCCP configuration. When configuring your WCCP service, we recommend a value from 51 to 98.

Service priority

The service priority identifies the WCCP service. The service is also included in the router configuration, where its ID must be the same.

Service IDs 0–50 are static under WCCP and reserved for *well known services* with standard configurations. Service IDs 51–255 are dynamic and involve negotiation between the partners in the WCCP configuration. When configuring your WCCP service, we recommend a value from 51 to 98.

WCCP router definition

The IP address of the router for the WCCP service is configured in the router definition. Alternatively, you can specify the name to which a domain server will resolve this IP address.

You can configure multiple routers by specifying an IP address or DNS name for each of them or by using a multicast IP address. When an IP address is to serve as a multicast IP address, this is indicated by use of the keywords `group-address` and `group-listen`.

Ports to be redirected

The ports from which web traffic is redirected to the Web Gateway proxy port are listed here.

The redirection works for traffic under HTTP and HTTPS. Redirection of FTP traffic or traffic under any other protocol is not supported. This means that all ports that you configure here must be ports for HTTP and HTTPS traffic. Port 80 for HTTP traffic and port 443 for HTTPS traffic are by default included in the list.

If you add more ports for HTTPS traffic, you must also add them as ports to be treated as SSL.

Note: If version 1 of WCCP is used, only traffic for port 80 is redirected. You cannot add any other ports for redirection.

Proxy listener address

The proxy listener address is the physical IP address of the network interface card on a Web Gateway appliance that web traffic is redirected to.

Proxy listener port

The proxy listener port is the port on Web Gateway that listens to redirected requests.

For the redirection to work, you must bind this port to IP address 0.0.0.0. For example, when using default port 9090, bind it by specifying 0.0.0.0:9090.

You must not bind the port to the IP address of the appliance where you are working, by specifying `localhost`, nor bind it to any other IP address. Otherwise the redirection will not work and traffic will not be processed.

Assignment method

The assignment method is the method for assigning buckets (processing jobs) under WCCP to different Web Gateway appliances when a configuration consists of more than one appliance. The method can be assignment by mask or hash. Some routers only support the mask assignment method. For more information, see the router documentation.

Input for load distribution

Load distribution can based on the source or destination IP address or the source or destination port of a request. We recommend configuring load distribution based on the source IP address. This ensures that the same appliance will always receive the requests that a user sends from a particular client system. Breaking sessions is avoided this way.

Assignment weight

The assignment weight assigns traffic load to different Web Gateway appliances in a WCCP configuration. If 1000 is configured as default on all appliances, the load is distributed equally.

If an appliance performs better than the others, you can configure a higher value on this appliance and lower values on the others. If all appliances perform equally well, we recommend leaving the default on each of them.

GRE-encapsulated

When the Generic Routing Encapsulation (GRE) method is used for sending data packets, an original data packet is encapsulated inside a new packet with additional headers. The new packet is sent from the router to Web Gateway over a connection that is known as a GRE tunnel. This method requires more overhead, but has the advantage of working across subnets.

L2-rewrite to local NIC

When the L2-rewrite (Layer 2 rewrite) method is used for sending data packets, the destination MAC address is rewritten to the MAC address of the proxy. The packets are redirected to a network interface on an appliance. This method works only if the router and the appliance are on the same subnet.

L2-redirect target

The target for redirecting data packets under the L2-rewrite method is the network interface of a NIC on the appliance where you are working, for example, eth0.

Troubleshooting issues with WCCP services

When issues arise with WCCP services while configuring them for the Transparent Proxy mode, there are several ways to gather relevant information.

- You can review information about WCCP services on the appliance dashboard
- You can retrieve information about WCCP services by running suitable commands on the command line of a system console that is connected to the appliance.

Review information about WCCP services on the dashboard

Review information about WCCP services on the appliance dashboard to see whether troubleshooting activities are required.

Task

1. Select Dashboard → Charts and Tables.
2. On the navigation pane, click System Summary and scroll down to the WCCP Service Current Status Report table.

Results

The table shows values for several WCCP parameters, such as the ID of the WCCP service that the appliance has subscribed to, the IP address of the router, forwarding and return methods, and assigned buckets.

It also shows the time stamps of the latest "Here I Am" and "I See You" data packets, which allows you to verify that the health check is working.

Retrieving information about WCCP services over the command line

You can run several commands on the command line of a system console to retrieve information about WCCP services.

Enter the following command to see if web traffic is redirected to the configured port on a Web Gateway appliance.

```
iptables -t mangle -L
```

You will see, for example, an entry for the `chain WCCP0` with a line containing `redirect 10.10.73.72:9090`.

10.10.73.72 is the IP address of the network interface of the NIC on the Web Gateway appliance that you configured as destination of the redirected traffic. 9090 is the configured port.

You can check whether the appliance sends "Here I Am" and "I See You" data packets. Enter the following command:

```
tcpdump -npi eth0 port 2048
```

Within the data packets that are displayed, verify that the following applies:

- The IP address shown for the web cache is the IP address of the Web Gateway appliance.
- The bucket assignment method is the method that is also configured for Web Gateway.
- The redirect method is the method that is also configured for Web Gateway.

You can check whether the GRE-encapsulated or L2-rewritten data packets are received on the Web Gateway appliance.

- For GRE-encapsulation, enter the following command:

  ```
  tcpdump -npi eth0 ip proto 47
  ```

  Verify that the source IP address of the data packets is the IP address that is configured for the router on Web Gateway.
- For L2-rewriting, enter the following command:

  ```
  tcpdump -npi eth0 not host <IP address of the Web Gateway appliance
  ```

  Verify that the source IP address of the data packets is the IP address of the client that sent the request.

Note: To check that redirected data packets are received on Web Gateway, you can also enter the *ifconfig* command.

Transparent Proxy with L2 transparent mode

When setting up a proxy on a Web Gateway appliance, you can configure it to run in Transparent Proxy mode and use the L2 (Layer 2) transparent method to redirect web traffic from the clients to Web Gateway.

Alternatively, you can redirect web traffic using WCCP (Web Cache Communication Protocol). You can also configure and use both methods at the same time.

Configure the Transparent Proxy with L2 transparent mode

Configure the Transparent Proxy mode for proxies that run on Web Gateway and use the L2 (Layer 2) transparent method to redirect web traffic sent by clients.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance where you want to configure the Transparent Proxy with L2 transparent mode. Then click Proxies.
3. Under Network Setup, select Proxy (optional WCCP).
   The Transparent Proxy settings appear.
4. Select L2 transparent.
   The Port redirects list appears.
5. Add entries for port redirects to this list as needed.
   a. Click the Add icon.
      The Add Proxy Port Redirects window opens.
   b. Use the options in this window to create and add list entries.
      Each entry specifies particular ports that web traffic was originally directed to and the ports on the proxy that this traffic is redirected to.
6. Click Save Changes.

Results

Web traffic coming in from the clients is now redirected under the L2 transparent method to the ports that you configured.

Transparent Proxy settings (for use with L2 transparent)

Settings for the Transparent Proxy mode when using the L2 transparent method to redirect web traffic

Transparent Proxy

Option	Definition
Supported client redirection methods	Lets you select a method for redirecting web traffic. • WCCP — When selected, HTTP client requests sent to web servers under IPv4 and IPv6 are intercepted by an additional network device and redirected to the appliance using the Web Cache Communication Protocol (WCCP). The clients are not aware of the redirection, it remains transparent for them. In the same way as for client requests, responses from web servers are directed back to the appliance. When using the WCCP redirection method, you need to configure one or more WCCP services on the appliance to let them perform the redirection. You also need to configure the network device that intercepts the client requests and server responses. This device can be configured as a router or switch with routing functions. After selecting this option, the WCCP Services inline list appears for configuring and adding WCCP services. **Note:** After selecting this option, the WCCP Services list appears below where you can configure and add WCCP services. • L2 transparent — When selected, client requests sent to a web server under IPv4 and IPv6 are intercepted by an additional network device and directed to the appliance using the Layer 2 redirection method. Under this method, client requests are accepted on the appliance even if their destination IP addresses are not addresses of the appliance. The redirection is transparent to the clients. You need to enter the original ports for those client requests that are to be intercepted and redirected in a list on the appliance together with the ports that these requests are redirected to. The additional network device must be configured accordingly. When this option is selected, requests can not be transmitted using a connection in active FTP mode. Only the passive FTP mode is then available. **Note:** After selecting this option, the Port Redirects list appears below where you can configure and add port for redirecting web traffic.

The following table describes the fields of an entry in the list of port redirects for the L2 transparent redirection method.

Port redirects – List entry

Option	Definition
Original destination port	Specifies the port or ports that web traffic coming in from a client was originally directed to.
Destination proxy port	Specifies the port on the proxy that this traffic is redirected to.

Option	Definition
Comment	Provides a plain-text comment on the redirection

Proxy HA mode

The Proxy HA network mode that you can configure on Web Gateway is an explicit proxy mode with High Availability functions. It allows you to perform failover and load balancing without using external load balancers.

Director node and scanning nodes

When multiple appliances run in a Proxy HA configuration, this configuration is also called a High Availability cluster. One of the appliances in this cluster is configured as the director node, while the other appliances are configured as scanning nodes.

The director node performs load balancing within the cluster by distributing load to the scanning nodes. Usually, the director node also acts as a scanning node. You also configure at least one scanning node as backup node that replaces a failed director node,

Appliances take their roles according to a priority value that you configure for each of them. The director node has the highest value configured, the values for the backup nodes are below this, but greater than 0, whereas scanning nodes that have 0 as their value will not perform backup functions.

The node that has the director role at a given point in time is known as the *active director*. The active director uses a virtual IP address (VIP) as an alias IP address on its interface for communication with the clients that have their web traffic redirected to Web Gateway for filtering.

We recommend that you also configure the appliances that you want to include in a Proxy HA configuration as members of a Central Management cluster.

These configuration types do not depend on each other for running successfully. But if the appliances are not controlled and synchronized by Central Management, each appliance might follow different web security rules after some time.

Load balancing

Load balancing in a Proxy HA configuration takes into account resource usage and active number of connections. So, if one scanning node is overloaded, others get more traffic to compensate.

When load balancing is performed, requests from the same client usually go to the same scanning node.

Failover

If the director node fails, the backup node with the highest priority value takes over the director role. When the original director node returns to active status, it takes over the director role again.

To verify that nodes are available, VRRP (Virtual Router Redundancy Protocol) is used for health checks. You must configure the following for VRRP on each appliance to enable the health checks: A VRRP interface and a virtual router ID that is the same for all members of the High Availability cluster.

Each node sends a multicast packet per second to IP address 224.0.0.18. If no multicast packet from the active director is seen for 3–4 seconds, a failover is performed. The failover lets the backup node with the highest priority become the director node. This node takes on ownership of the virtual IP address of the High Availability cluster and informs the other nodes about its new director role.

Gratuitous ARP (Address Resolution Protocol) messages are used to update the ARP tables of participating clients and routers. Each time the common virtual IP address changes ownership (a failover occurs), the new director node sends a gratuitous ARP message. Subsequent TCP/IP packets can thus reach this node.

IP addresses

On the appliances that are to run as director and backup nodes, IP addresses must be configured as follows:

- The IP addresses of all scanning nodes must be entered in the scanner table that is filled in on the director and backup nodes. If you also configure the director node to take the role of a scanning node, you must enter its address in this table as well.
- The virtual IP address that the director and backup nodes use on their interfaces as network IP address to communicate with the Web Gateway clients must also be known.

You must add this network interface IP address to the settings that are configured on these nodes for the HTTP and FTP proxies with ports that listen to requests coming in from the clients.

Configure the Proxy HA mode

Configure the Proxy High Availability (Proxy HA) mode for a group of Web Gateway appliances to perform load balancing and failover without using external load balancers.

This group of appliances is also referred to as Proxy HA configuration or High Availability cluster.

Configure the appliances in this cluster one after another. Set up one of them as the director node, which provides the load balancing functions, and the others as backup nodes or as scanning nodes only.

Select different values for the various settings options depending on the particular role of a node.

If no information regarding the different roles is provided for a configuration step in the following, complete it in the same way for each appliance.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select an appliance that you want to include in the cluster, then click Proxies.
3. Under Network Setup, select Proxy HA.

   The Proxy HA settings appear immediately below the Network Setup settings.
4. Configure these settings as follows.
   a. Begin with Director priority, which is located below the scanner table — Move the slider on the scale that is provided here to a numerical value for the appliance to configure its role in the cluster.

      - For a director node — Highest priority, for example, *99*

        A load balancer runs on this node to provide load balancing functions in the cluster.
      - For a backup node — Lower priority, but higher than *0*, for example, *89*

        A backup node, also known as *peer* node, performs a failover to replace the director node when this node fails and no other node with a higher priority is active. Otherwise the backup node works as scanning node.
      - For a node that runs as a scanning node only — *0*

      Moving the slider also makes the remaining Proxy HA settings accessible. Continue with configuring these settings.

   b. Scanners table — In this table, fill in IP addresses and roles of the nodes in the cluster that participate in the scanning. Roles are referred to as *types* in this table.

      Entries are filled in this table when setting up a director or a backup node. When setting up a scanning-only node, you need not deal with this table.

      When filling in an entry for the node that you are currently working on, always specify Scanner as role, regardless of the fact that the node might be a director or backup node.

      To add an entry, click the Add icon, and proceed as follows:

      - For a director node — If this node is to participate in the scanning, fill in its own IP address and select Scanner as type. Otherwise do not fill in an entry for this node.

        Add the IP addresses of all other nodes that run as scanning nodes. Select Scanner as role *(type)* if a node is a scanning-only node, and Peer/Director if it's a backup node.

        For example, you have a cluster with a director node (appliance 1) and a backup node (appliance 2) that both participate in the scanning along with two nodes that only run as scanning nodes (appliances 3 and 4).

        Then the table should include these entries:

        - IP address of the director node (appliance 1) — Type: Scanner
        - IP address of the backup node (appliance 2) — Type: Peer/Director
        - IP address of one scanning only node (appliance 3) — Type: Scanner

- IP address of the other scanning only node (appliance 4) — Type: Scanner

- For a backup node — Fill in its own IP address and select Scanner as role *(type)*.

  Add the IP addresses of all other nodes that run as scanning nodes. Select Scanner as role if a node is a scanning-only node, and Peer/Director otherwise.

  For example, you have the same cluster as above. Then the table should include these entries:

  - IP address of the director node (appliance 1) — Type: Peer/Director
  - IP address of the backup node (appliance 2) — Type: Scanner
  - IP address of one scanning only node (appliance 3) — Type: Scanner
  - IP address of the other scanning only node (appliance 4) — Type: Scanner

c. Relay port — For a director or backup node, configure a TCP port as relay port. This is a port that the scanning nodes in the cluster will use when forwarding web traffic to external destinations.

d. Probe interval — For a director or backup node, set this interval as the time (in milliseconds) to elapse before the director node sends the next probe packet. Probe packets are sent to scanning nodes to verify they are still alive.

   If you specify 0, no probe packets are sent.

e. Inactivity timeout — Set a timeout (in seconds) for inactivity on the connections between the clients and the internal load balancer.

f. Load balancing algorithm — Select a load balancing algorithm for the load balancer.

   Select one of the following:

   - Round robin — Traffic is forwarded to the scanning nodes one after another.
   - Leastconn (Least connections) — Traffic is forwarded to the scanning node with the lowest number of currently active connections.

g. Stickiness — Enable sticky sessions between the clients and the scanning nodes using the client IP addresses as sources.

   If you want to run an FTP proxy under the Proxy HA network mode, this option must be enabled.

h. Virtual IPs — For a director or backup node, specify a Virtual IP address (VIP address) that is to serve as the cluster address.

   Select a network interface, for example, eth0, to assign this VIP address to it.

   You can assign more than one VIP address to a network interface. You can also select more than one network interface.

   The cluster address is used by the node that is currently the active director. Using this address, the director node connects to the scanning nodes as well as to the clients that have their requests for web access redirected to Web Gateway.

   Any network interface that you select or leave here as selected by default is one of those that you have configured under the Network Interfaces settings, which are part of the system settings on a Web Gateway appliance.

   Note: We recommend selecting network interfaces here that you have configured with a /32 subnet mask.

i. Configure the settings for health checks under the Virtual Router Redundancy Protocol (VRRP).

   - Virtual router ID — ID used for the health checks

     This ID must be the same on all nodes.

     Default: 51

     You can leave the default ID unless you are already using VRRP elsewhere in your network with ID 51. Then change it here to make it unique for the High Availability cluster.

   - VRRP interface — Interface used for the health checks

     Default: eth0

     You can leave this default unless you are not using the eth0 interface on your appliances.

     The network interface that you select or leave here as selected by default is one of those that you have configured under the Network Interfaces settings, which are part of the system settings on a Web Gateway appliance.

     The VIP address of the network interface that is selected here is used when this node connects as active director to a scanning node in passive FTP mode. If more than one VIP address is configured for this interface, the address that was configured last is used.

     Note: If no network interface is selected as the VRRP interface, no connections can be run under FTP in Proxy HA mode.

j. List of egress IPs for load distribution — Configure egress IP addresses in this list to be able to use more connections when forwarding incoming web traffic to the scanning nodes.

   Configuring egress IP addresses is optional. Configure them if more than 50,000 active connections are needed on one scanning node at the same time.

   As egress IP addresses, enter addresses that you added as IP aliases for network interfaces when you configured them under the Network Interfaces settings, which are part of the system settings on a Web Gateway appliance.

   The following must also apply to the IP aliases that you add as egress IP addresses. You must also have configured these IP aliases as IP addresses for the network interface that you selected as the VRRP interface in *substep i*.

   The load balancer on the active director node distributes incoming web traffic among the scanning nodes. The number of ports that can be used by this load balancer when connecting to these nodes is limited. By configuring egress IP addresses you can overcome this limit and increase performance.

5. If you want to run the appliances in the cluster nodes as proxies under HTTP, configure the HTTP Proxy settings.
   - For a director or backup node — Under Listener Address in the HTTP port definition list, fill in the IP address of this node.

     Use the default entry that is provided in first position on the list, and replace the 0.0.0.0 with this IP address, keeping port 9090, for example, `192.168.2.100:9090`.

     Leave the values in the remaining fields as they are unless you have a particular reason for changing them.
   - For a node that runs as scanning node only — Configure an unbound listener under Listener Address in the HTTP port definition list, for example, `0.0.0.0:9090`.

     To configure an unbound listener, you can leave the default entry that is provided in first position on the list.
6. If you want to run the appliances in the cluster as proxies under FTP, configure the FTP proxy settings.
   - For a director or backup node — Under Listener Address in the FTP port definition list, fill in the IP address of this node.

     Use the default entry that is provided in first position on the list, and replace the 0.0.0.0 with this IP address, keeping port 2121, for example, 192.168.2.100:2121.

     Leave the values in the remaining fields as they are unless you have a particular reason for changing them.
   - For a node that runs as scanning node only — Configure an unbound listener under Listener Address in the FTP port definition list, for example, `0.0.0.0:9090`.

     To configure an unbound listener, you can leave the default entry that is provided in first position on the list.
7. If you want to run the appliances in the cluster together with one or more ICAP servers, configure ports for them in the ICAP port definition list of the ICAP Server settings.

   Configure entries for director and backup nodes as well as for scanning only nodes in the same way as shown for HTTP in step 5.
8. Click Save Changes.

Configure the Proxy HA mode for SOCKS traffic

You can configure the Proxy High Availability (Proxy HA) mode also for web traffic that is going on under the SOCKS protocol.

This network mode provides high availability and load balancing functions to the filtering process Web Gateway uses to protect your network against threats arising from the web.

To implement this mode, set up a cluster of Web Gateway appliances as nodes in a Proxy HA cluster with one of them acting as the director node.

The director node distributes the work load that the filtering process requires to the other nodes in the cluster, which are referred to as scanning nodes. A director node can participate in the filtering process and take the role of a scanning node in parallel.

You can also configure one of the other nodes or more of them as backup nodes that take over the director role from the director node if this node becomes unavailable.

To extend the filtering process to web traffic coming in under the SOCKS protocol, you set up a SOCKS proxy on each node in the cluster.

On each of the nodes that act as scanning nodes, you also implement a rule set that provides the filtering functions for SOCKS traffic.

Task

1. Set up a director node for the Proxy HA cluster.
   a. Select Configuration → Appliances.
   b. On the appliances tree, navigate to the appliance you want to set up as director node. Then select Proxies and under Network Setup, select Proxy HA.
   c. Begin with Director priority, which is located below the Scanners table, and move the slider on the slider scale to a high value, for example, 97.
   d. In the Scanners table, enter an IP address for each appliance in the cluster and the role it takes.

      The role of the appliance that you have currently selected for configuration is always specified as `Scanner`, even if it is a director or backup node.

      Otherwise the role for director and backup nodes is `Peer/Director`. For a scanning-only node, the role is always `Scanner`.

      For example, enter the following.

      - For the director node: `192.169.10.10 — Scanner`
      - For a backup node: `192.169.10.11 — Peer/Director`
      - For a scanning-only node: `192.169.10.12 — Scanner`

   e. Under Virtual IPs, specify a network interface on this appliance and a virtual IP address (VIP address) for it. This VIP address serves as the cluster address.

      You can specify more than one network interface here and more than one VIP address for each interface.

      For example, you have configured `eth2` and `eth2.10` as network interfaces on this appliance and specified IP addresses for them under both IPv4 and IPv6.

      Then you can enter the following.

      - `192.169.10.250/16 — eth2`
      - `fd02:169::250/64 — eth2`
      - `192.170.10.250/16 — eth2.10`
      - `fd02:170::250/64 — eth2.10`

   f. Under VRRP interface, specify any of the network interfaces you have configured on this appliance.

   For the other options under Proxy HA,, you can leave the default values.
2. Set up a SOCKS proxy on the director node.
   a. Scroll down to the SOCKS Proxy section.
   b. Select Enable SOCKS proxy.
   c. Under Listener address in the table that is provided here, enter the IP address and the port on the director node that listens to requests for web access coming in from the clients as traffic under the SOCKS protocol.

      You can enter addresses under IPv4 and IPv6 here.

      For example, enter the following.

      - `192.169.10.10:1080`
      - `fd02:169::10:1080`

        For the second entry, you can omit the address and only enter the port.
3. Set up other appliances as nodes in the Proxy HA cluster.

   Set up at least one appliance as a scanning-only node. We recommend that you also set up an appliance as a backup node.

Repeat these substeps for every appliance you want to include in the cluster.

a. Navigate to another appliance on the appliance tree. Then select Proxies and under Network Setup, select Proxy HA.
b. Begin with Director priority and move the slider on the slider scale to a value as follows.
   - For a backup node: Lower than what you configured for the director node, but greater than 0, for example, 56
   - For a scanning-only node: 0
c. Fill entries in the Scanners table only when setting up a backup node. When setting up a scanning-only node, no entries are required in this table.

   For example, enter the following when setting up a backup node.

   - For the director node: `192.169.10.10 — Peer/Director`
   - For the backup node: `192.169.10.11 — Scanner`
   - For a scanning-only node: `192.169.10.12 — Scanner`
d. Configure all other options under Proxy HA for a backup or a scanning-only node as in step 1.
e. Set up a SOCKS proxy on this appliance in the same way and with the same values as in step 2.

4. Implement the SOCKS Proxy rule set on all scanning nodes in the cluster.

   If the director node also works as a scanning node, implement the rule set there as well.

   a. Select Policy → Rule Sets.
   b. Import the SOCKS Proxy rule set from the Common Rules group of the library.
   c. In the key elements view of this rule set, enter the protocol versions for the SOCKS traffic that is to be filtered to a whitelist
5. In the key elements view of this rule set, you can configure settings and add the protocol versions for the SOCKS traffic that is filtered to a whitelist.

   SOCKS traffic coming in under different protocol versions is blocked.

Results

You have now set up a Proxy HA cluster with Web Gateway appliances as nodes that filter SOCKS traffic coming in from the clients.

Resolving issues with a Proxy HA configuration

Several measures can be taken when trying to resolve issues with a Proxy HA configuration, also known as High Availability cluster.

Look up VRRP health check messages

Messages about the VRRP health checks are logged on an appliance system under:

```
/var/log/messages
```

These messages also inform you about whether an appliance is in director or backup node status.

Find out which node blocked a request

To find out which of the nodes in a High Availability cluster blocked a request, edit the user message template for Block actions. Insert the System.HostName property.

Test a node

To test the behavior of a particular node, enter only its IP address in the table of scanning nodes, leaving out all other addresses, before operating the High Availability cluster.

Identify the active director

To identify the active director node that owns the virtual IP address of the High Availability cluster, set up an SSH session with each node. Then run the `ip addr show` command on each of them.

Turn a director node into a scanning node

When an issue occurred with a director node, you can change its role and turn it into a scanning node that performs no other functions besides scanning.

First set the director priority for this node to 0. Be sure to save what you configured here.

Then change the settings that you configured on this node for the HTTP and FTP proxies with ports that listen to requests coming in from the clients. These settings include the network interface IP address. Set this address to 0.0.0.0.

Inspect failure to distribute web traffic

If all web traffic is processed on the director node or another single node instead of being distributed to other nodes, it could have these reasons:

- The director node does not know about any other nodes because no IP addresses of other scanning nodes have been entered in the scanner table.
- All traffic is coming from the same source IP address because there is a downstream proxy or a NAT device in place. Then the usual behavior for load balancing is to direct this traffic to the same node again and again.

Best practices - High Availability configuration size limits

When configuring the Proxy HA (High Availability) network mode, you need to consider the number of Web Gateway appliances to include in the configuration.

In most cases, multiple appliances are run in a network and configured as nodes that are administered using Central Management functions.

Usually, one of these nodes is configured as the *director* node that directs incoming web traffic to the other nodes, which are termed *scanning* nodes since their job is to scan this traffic.

On a particular appliance, network interfaces are usually configured in a two-leg solution, which uses separate interfaces for incoming and outgoing web traffic, or in a three-leg solution, which uses an additional interface for Central Management communication.

When working with a network that is configured in this way, the following should be taken into account:

- The added maximum throughput of the scanning nodes must not exhaust or exceed the maximum throughput that can be achieved by the director node.
- Under the Proxy HA network mode, only a scaling of up to 1 gigabit per second is possible due to internal restrictions.
- We recommend that you leave a clear safety margin regarding the number of scanning nodes that could theoretically be configured under these conditions.
    - For example, with a throughput of 100 megabits per second for a scanning node and a director node that uses a 1 GbE network interface, ten nodes would be possible, but we recommend five.
    - With a throughput of 300 megabits per second on a scanning nodes and the same director node, three nodes would be possible, but we recommend two.

The maximum throughput on a scanning node varies with the appliance model that is used as a node and how a node is configured, for example, whether anti-malware filtering or the web cache are enabled or not. To find this value for a node, you can use a sizing calculator.

Calculations look different when a director node uses a 10GbE network interface, rather than a 1GbE network interface, or when IP spoofing is enabled in a configuration. This is explained in the following.

10GbE network interfaces

When a 10GbE network interface is installed on the director node, the maximum throughput for this node increases accordingly. However, the internal scaling limit for a Proxy HA configuration must still not be exhausted or exceeded.

- For example, with a throughput of 100 megabits per second for a scanning node, more than five nodes are possible, but we still recommend to keep the number of nodes below ten.
- With a throughput of 300 megabits per second, three nodes are possible, and we recommend not to use more.

IP spoofing

When IP spoofing is configured, data packets pass through the director node twice, once when the director node directs them to the scanning nodes and a second time when they are returned from the scanning nodes to the director node, as this node forwards the data packets to their original IP addresses.

This means the maximum throughput is only 500 megabits per second on the director node if a 1GbE network interface is used while the internal scaling limit for a Proxy HA configuration remains the same.

The number of scanning nodes must be adapted accordingly.

- For example, with a throughput of 100 megabits per second for a scanning node and a director node that uses a 1GbE network interface, the number of scanning nodes must be less than five.

  If a 10GbE network interface is used, the number of scanning nodes can be higher, but we still recommend five.
- With a throughput of 300 megabits per second for a scanning node and a director node that uses a 1GbE network interface, there should be only one scanning node.

  If a 10GbE network interface is used, we recommend not to configure more than three scanning nodes.

Transparent router mode

The transparent router mode is a mode you can configure for the proxy functions of a Web Gateway appliance if you do not want to use an explicit mode.

In transparent router mode, the clients are unaware of the appliance and need not be configured to direct their web traffic to it.

The appliance is placed as a router immediately behind a firewall. A switch can be used for connecting the appliance to its clients. A routing table is used to direct the traffic.

Director and scanning nodes

If you are running several appliances as nodes within a complex configuration, for example, in a Central Management cluster, one node is usually configured as director, while the other nodes are configured as scanning nodes.

The director node receives web traffic from the clients and distributes it to the scanning nodes, which perform filtering activities on the traffic according to the rules that are implemented. Further handling of the traffic by the director or scanning nodes differs depending on what is configured. The director node can also perform filtering activities.

If you are only running one Web Gateway appliance in your network and want to configure it in transparent router mode, you must still configure the director role for it to let it receive, filter, and forward web traffic.

Tip: Best practice: Configure at least two director nodes to avoid problems in case one of them goes offline.

Configure the transparent router mode

To configure the proxy functions of an appliance in transparent router mode, complete the following high-level steps.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure the transparent router mode for and click Proxies (HTTP(S), FTP, ICAP, and IM).
3. Under Network Setup, select Transparent Router.

   After selecting this mode, specific Transparent Router settings appear below the Network Setup settings.

   Common settings follow the specific settings, including settings for configuring the HTTP, FTP, and other network protocols.
4. Configure specific and common settings as needed.
5. Click Save Changes.

Results

When running several appliances as nodes in a Central Management configuration, you can configure the transparent router mode on each of them.

Configure nodes in transparent router mode

You can configure the transparent router mode for two or more appliances that are nodes in a Central Management configuration. One of the nodes takes the director role, which means it directs data packets, while the scanning nodes filter them.

Node configuration includes configuring network and proxy settings.

Configure network settings for a director node in transparent router mode

To configure a director node in transparent router mode, configure network interfaces for inbound and outbound web traffic.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure as a director node and click Network interfaces.
3. Configure network interfaces as is suitable for your network.
   You need at least one interface for inbound and one for outbound web traffic.
4. Click Save Changes.
   You are logged off and logged on to the appliance again.

Configure proxy settings for a director node in Transparent Router mode

To configure proxy settings for a director node in Transparent Router mode, configure the director role for this node as well as port redirects and proxy ports.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance that runs as director node, then click Proxies.
3. Under Network Setup, select Transparent Router.
   Specific Transparent Router settings appear below the Network Setup settings.
4. Configure one or more port redirects that let requests sent from Web Gateway clients be redirected to a particular port.
   1. Under Port redirects, click Add.
      The Add Port Redirects window opens.
   2. Configure the following for a new port redirect that applies to connections under HTTP or HTTPS:
      - Protocol name — http
        http covers connections under both HTTP and HTTPS.
      - Original destination ports — 80, 443
        These are the default destination ports. They cover connections under both HTTP and HTTPS.

If you also want to filter HTTPS traffic, enable the SSL Scanner rule set, which is by default provided on the rule sets tree, but not enabled.

- Destination proxy port — 9090

  9090 is the default proxy port on an appliance.

  If you need to use other ports due to the requirements of your network, change these settings as needed.

  To configure a port direct for connections under FTP, select this protocol. Default ports are then preconfigured, which you can change as needed.

5. Continue with Director priority, which is located below the Scanners table.

   Move the slider on the scale that is provided here to a high value, for example, *99.*

   Moving the slider also makes the remaining Transparent Router settings accessible. Continue with configuring these settings.
6. Scanners table — In this table, fill in the IP addresses of the outbound network interfaces for the nodes in a cluster that run as scanning nodes as well as their roles. Roles are referred to as *types* in this table.

   Click the Add icon and proceed as follows:

   a. Fill in an entry for the director node itself if this node participates in the scanning. Otherwise do not fill in an entry for this node.
   If the director node participates in the scanning, select Scanner as role, regardless of the fact that it is a director node.

   b. Fill in entries for all other scanning nodes in the cluster, including scanning-only nodes, as well as backup nodes that participate in the scanning.
   Select Scanner as role if a node is a scanning-only node and Peer/Director if it is a backup node.

   For example, you have a cluster with a director node (appliance 1) and a backup node (appliance 2) that both participate in the scanning as well as two nodes that only run as scanning nodes (appliances 3 and 4).

   Then four entries are required in this table, one for the director node and three more for the other appliances:

   - Outbound IP address of the director node (appliance 1) — Type: Scanner
   - Outbound IP address of the backup node (appliance 2) — Type: Peer/Director
   - Outbound IP address of one scanning only node (appliance 3) — Type: Scanner
   - Outbound IP address of the other scanning only node (appliance 4) — Type: Scanner
7. Relay port — Configure a TCP port as relay port. This is a port that the scanning nodes in the cluster will use to forward web traffic to external destinations.
8. Probe interval — Set this interval as the time (in milliseconds) to elapse before the director node sends the next probe packet to the scanning nodes. Probe packets are sent to verify that the scanning nodes are still alive.
   If you specify 0, no probe packets are sent.
9. Inactivity timeout — Set a timeout (in seconds) for inactivity on the connections between the clients and the internal load balancer.
10. Load balancing algorithm — Select a load balancing algorithm for the load balancer.

    Select one of the following:

    - Round robin — Traffic is forwarded to the scanning nodes one after another.
    - Leastconn (Least connection) — Traffic is forwarded to the scanning node with the lowest number of currently active connections.
11. Stickiness — Enable sticky sessions between clients and the scanning nodes using the client IP addresses as sources.
    If you want to run an FTP proxy under the Transparent Router network mode, this option must be enabled.
12. Virtual IPs — Specify a virtual IP address (VIP address) that is to serve as the cluster address when multiple Web Gateway appliances are running in a cluster.

    Select a network interface, for example, eth0, to assign this VIP address to it.

    You can assign more than one VIP address to a network interface. You can also select more than one network interface.

    Any network interface that you select or leave here as selected by default is one of those that you have configured under the Network Interfaces settings, which are part of the system settings on a Web Gateway appliance.

    The cluster address is used by the node that is currently the active director. Using this address, the director node connects to the scanning nodes as well as to the clients that have their requests for web access redirected to Web Gateway.
13. Configure the settings for health checks under the Virtual Router Redundancy Protocol (VRRP).

    ◦

Virtual router ID — ID used for the health checks

This ID must be the same on all cluster nodes.

Default: 51

You can leave the default ID unless you are already using VRRP elsewhere in your network with ID 51. Then change it here to make it unique for a cluster.

◦

VRRP interface — Interface used for the health checks

Default: eth0

You can leave this default unless you are not using the eth0 interface on your appliances.

The network interface that you select or leave here as selected by default is one of those that you have configured under the Network Interfaces settings, which are part of the system settings on a Web Gateway appliance.

The VIP address of the network interface that is selected here is used when this node connects as active director to a scanning node in passive FTP mode. If more than one VIP address is configured for this interface, the address that was configured last is used.

Note: If no network interface is selected as the VRRP interface, no connections can be run under FTP in Transparent Router mode.

14. List of egress IPs for load distribution — Configure egress IP addresses in this list to be able to use more connections when forwarding incoming web traffic to the scanning nodes.

    Configuring egress IP addresses is optional. Configure them if more than 50,000 active connections are needed on one scanning node at the same time.

    As egress IP addresses, enter addresses that you added as IP aliases for network interfaces when you configured them under the Network Interfaces settings, which are part of the system settings on a Web Gateway appliance.

    The following must also apply to the IP aliases that you add as egress IP addresses. You must also have configured these IP aliases as IP addresses for the network interface that you selected as the VRRP interface in *step 13*.

    The load balancer on the active director node distributes incoming web traffic among the scanning nodes. The number of ports that can be used by this load balancer when connecting to these nodes is limited. By configuring egress IP addresses you can overcome this limit and increase performance.
15. Configure IP spoofing as needed.
16. If you want to run this director node as a proxy under HTTP, configure the **HTTP Proxy** settings as follows.
    a. Under HTTP proxy port, make sure Enable HTTP proxy is selected.

       This setting is selected by default. An entry for port 9090 is also configured by default.

    b. Fill in the IP address of the outbound network interface for the director node under **Listener Address** in the **HTTP port definition list**.

       Use the default entry that is provided in first position on the list, and replace the 0.0.0.0 with the outbound IP address, keeping port 9090.

       Leave the values in the remaining fields as they are unless you have a particular reason for changing them.

    Clicking Add opens the Add HTTP Proxy Port window, which allows you to add more HTTP proxy ports. Configure the outbound IP address of the director node for each of them.
17. If you want to run this director node as a proxy under FTP, configure a listener address for it in the **FTP Proxy** settings.
    a. Under FTP proxy port, select Enable FTP proxy.

       An entry with FTP control port 2121 and FTP data port 2020 is configured by default.

    b. Fill in the IP address of the outbound network interface for the director node under **Listener Address** in the **FTP port definition list**.

       Use the default entry that is provided in first position on the list, and replace the 0.0.0.0 with the outbound IP address, keeping port 2121.

       Leave the values in the remaining fields as they are unless you have a particular reason for changing them.

    Clicking Add opens the Add FTP Proxy Port window, which allows you to add more FTP proxy ports.
18. Click Save Changes.

Configure a scanning node in transparent router mode

To configure a scanning node in transparent router mode, configure at least one network interface for outbound traffic. Configure the proxy settings in the same way as for a director node in this mode, except for the scanning role.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure as a scanning node and click Network Iinterfaces.
3. Configure network interfaces as is suitable for your network.
   You need at least one interface for outbound web traffic.
4. Click Save Changes.
5. You are logged off and logged on to the appliance again.
6. On the appliances tree, select the appliance you want to configure as a scanning node and click Proxies.
7. Under Network Setup, select Transparent Router.
   Specific Transparent Router settings appear below the Network Setup settings.
8. Configure the same port redirects as for the director node.
9. Set Director priority to 0.
10. Configure IP spoofing in the same way as for the director node.
11. If you have configured HTTP and FTP proxies for the director node, configure them also for this scanning node, using the options provided under HTTP proxy and FTP proxy.

    Proceed in the same way as for the director node, but under **Listener Address** in the **HTTP port definition list** and the **FTP port definition list**, do not specify an outbound IP address like the one you specified for the director node.

    Leave 0.0.0.0 instead, which is configured in the entries that are provided by default, or specify it when configuring more entries.
12. Click Save Changes.

Results

To run more than one scanning node in transparent router mode, configure additional appliances in the same way.

Under the transparent router mode, you can also turn a director or backup node into a node for scanning only. Select an appliance that runs as director or backup node and configure it using the settings for a scanning node as shown here.

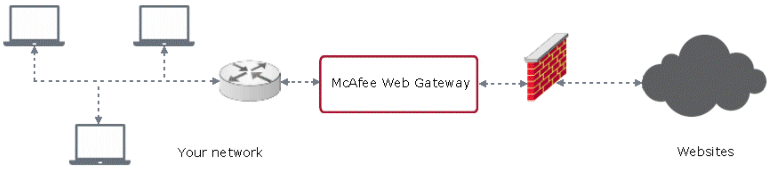
Transparent bridge mode

The transparent bridge mode is one of the transparent modes you can configure for the proxy functions of the appliance if you do not want to use an explicit mode.

In this mode, the clients are unaware of the appliance and need not be configured to direct their web traffic to it. The appliance is usually placed between a firewall and a router, where it serves as an invisible bridge.

The following diagram shows a configuration in transparent bridge mode.

Transparent bridge mode



Configure the transparent bridge mode

To configure the proxy functions of an appliance in transparent bridge mode, complete the following high-level steps.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure the transparent bridge mode for and click Proxies (HTTP(S), FTP, ICAP, and IM).
3. Under Network Setup, select Transparent Bridge.

   After selecting this mode, specific Transparent Bridge settings appear below the Network Setup settings.

   Common settings follow the specific settings, including settings for configuring the HTTP, FTP, and other network protocols.
4. Configure specific and common settings as needed.
5. Restart the appliance.

   The restart includes the reloading of network drivers, which ensures that the appropriate drivers for this network mode are applied.

   Tip: Best practice: Restart the appliance also after switching from the transparent bridge mode to another network mode.
6. Click Save Changes.

Results

When running several appliances as nodes in a Central Management configuration, you can configure the transparent bridge mode on each of them.

Configure nodes in transparent bridge mode

You can configure the transparent bridge mode for two or more appliances that are nodes in a Central Management configuration. One of the nodes takes the director role, which means it directs data packets, while the scanning nodes filter them.

Node configuration includes configuring network, Central Management, and proxy settings.

Configure network and Central Management settings for a director node in transparent bridge mode

To configure a director node in transparent bridge mode, configure a network interface for the transparent bridge functions and let its IP address be used for Central Management communication.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure as a director node and click Network interfaces.
3. Prepare the network interface for the transparent bridge functions.
   a. Select a still unused network interface of the appliance, but do not enable it yet.
   b. On the Advanced tab, select Bridge enabled.
   c. In the Name field, type `ibr0` as the name of the interface.
   d. On the IPv4 tab, under IP Settings, select Disable IPv4.
   e. Click Save Changes.

      You are logged off and logged on to the appliance again.
4. Configure the network interface for the transparent bridge functions.

   a. Select Configuration → Appliances. Then select the appliance again, and click Network interfaces.
      An additional network interface named ibr0 is now available.
   b. Select the ibr0 interface.
   c. On the IPv4 tab, configure an IP address, a subnet mask, and a default route for this interface.
   d. Select the checkbox next to the interface to enable it.
5. Configure the network interface that is currently used to access the appliance as the network interface for the transparent bridge functions.
   a. Select the network interface that is currently used to access the appliance.
   b. On the Advanced tab, select Bridge enabled.
   c. In the Name field, type `ibr0` as the name of the interface.
   d. On the IPv4 tab, under IP Settings, select Disable IPv4.
6. Enable the ibr0 network interface that you selected in step 3 from the until now unused network interfaces.
7. Configure Central Management settings.
   a. Select Central Management.
   b. Under Central Management Settings, add the IP address you configured for the ibr0 network interface to the list that is provided.
8. Click Save Changes.

Results

If you want to use more than one network interface for the transparent bridge mode, configure more unused network interfaces of an appliance in the same way.

Configure proxy settings for a director node in transparent bridge mode

To configure proxy settings for a director node in transparent bridge mode, configure the director role for it, as well as port redirects and proxy ports.

The director role is configured by giving the node a priority value > 0.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure as a director node and click Proxies (HTTP(S), FTP, ICAP, and IM).
3. Under Network Setup, select Transparent Bridge.
   Specific Transparent Bridge settings appear below the Network Setup settings.
4. Configure one or more port redirects that let requests sent from clients of Web Gateway be redirected to a particular port.
   a. Under Port redirects, click Add.
   b. Configure the following for a new port redirect that applies to connections under HTTP or HTTPS:
      - Protocol name — http

        http covers connections under both HTTP and HTTPS.
      - Original destination ports — 80. 443

        These are the default destination ports. They cover connections under both HTTP and HTTPS.

        If you want to filter also HTTPS traffic, you need to enable the SSL Scanner rule set, which is by default provided on the rule sets tree, but not enabled.
      - Destination proxy port — 9090

        9090 is the default proxy port on an appliance.

        If you need to use other ports due to the requirements of your network, change these settings as needed.

        To configure a port direct for connections under FTP, select this protocol. Default ports are then preconfigured, which you can change as needed.

5. Set Director priority to a value > 0.
6. In the Management IP field, type the IP address you specified for ibr0 when configuring the network settings.
7. Configure IP spoofing as needed.
8. Under HTTP proxy port, make sure Enable HTTP proxy is selected.

   The setting is selected by default. An entry for port 9090 is also configured by default on the HTTP Port Definition List.

   - You can change this port as needed. Clicking Add opens the Add HTTP Proxy Port window, which allows you to add more proxy ports.
   - To configure one or more FTP proxies, select Enable FTP Proxy under FTP Proxy. An entry for FTP control port 2121 and FTP data port 2020 is then preconfigured on the FTP Port Definition List

9. Click Save Changes.

Configure a scanning node in transparent bridge mode

To configure a scanning node in transparent bridge mode, configure the same settings as for a director node in this mode, except for the scanning role.

The scanning role is configured by giving the node 0 as the priority value.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure as a scanning node and click Proxies (HTTP(S), FTP, SOCKS, ICAP ...).
3. Under Network Setup, select Transparent Bridge.

   Specific Transparent Bridge settings appear below the Network Setup settings.
4. Configure the same port redirects as for the director node.
5. Set Director priority to 0.
6. Configure IP spoofing in the same way as for the director node.
7. Configure the same HTTP and FTP proxy ports as for the director node.
8. Click Save Changes.

Results

To run more than one scanning node in transparent bridge mode, configure additional appliances in the same way.

Transparent Bridge settings

The Transparent Bridge settings are specific settings for configuring the proxy functions of an appliance in transparent bridge mode.

Transparent Bridge

Settings for configuring the transparent bridge mode

Transparent Bridge

Option	Definition
Port redirects	Provides a list for entering the ports that requests for web access sent by users of your network are redirected to.
Director priority	Sets the priority (ranging from 0 to 99) that an appliance takes in directing the data packets that are sent in a request.

Option	Definition
	The highest value prevails. 0 means an appliance is what is known as a scanning node, which never directs data packets, but only filters them. **Note:** You can use this option only to configure a node as a scanning node (priority = 0) or a director node (priority > 0). Differences in node priorities greater than 0 are not evaluated. After configuring node priorities greater than 0 for multiple appliances in transparent bridge mode, you need to watch their behavior to find out which one has actually become the director node that directs data packets.
Management IP	Specifies the source IP address of an appliance that directs data packets when sending heartbeat messages to other appliances.
IP spoofing (HTTP, HTTPS)	When selected, the appliance keeps the client IP address that is sent with a request as the source address and uses it in communication with the requested web server under various protocols. The appliance does not verify whether this address matches the host name of the request.
IP spoofing (FTP)	When selected, the appliance communicates with a file server under the FTP protocol in the same way as under the HTTP or HTTPS protocol to perform IP spoofing For active FTP, this option must be enabled.

The following table describes an entry in the list of port redirects.

Port redirects – List entry

Option	Definition
Protocol name	Specifies the name of the protocol used for sending and receiving requests.
Original destination ports	Specifies the ports that redirected requests must originally be sent to if they are to be redirected.
Destination proxy port	Specifies the port that requests are redirected to.
Source IP based exceptions	Excludes requests that have been received from clients with the specified IP addresses from redirecting. • For each IP address, a net mask must also be specified. • When a request is excluded from redirecting, it is not processed by any of the filtering rules that are implemented. • You can configure redirection exceptions in this way to let requests received from trusted sources skip further processing on Web Gateway or for troubleshooting connection problems.

Option	Definition
Destination IP based exceptions	Excludes requests that are sent to a destination with the specified IP address from redirecting. • For each IP address, a net mask must also be specified. • When a request is excluded from redirecting, it is not processed by any of the filtering rules that are implemented. • You can configure redirection exceptions in this way to let requests sent to trusted destinations skip further processing
Optional 802.1Q VLANs	Lists the IDs of the network interfaces for VLAN traffic that are configured.
Comment	Provides a plain-text comment on a port redirect.

Best practice: Fine-tuning the Transparent Bridge mode

When configuring Web Gateway in transparent bridge mode, you can complete several activities in addition to the basic steps to improve the configuration.

These activities include the following:

- Configuring port redirects
- Setting up more than one appliance
- Appropriate handling of the STP protocol

Configuring port redirects

Web Gateway is by default configured to scan and filter requests for web access arriving on ports 80 and 443. All requests arriving on other ports are passed on to the web unfiltered unless you specify additional ports.

You can configure port redirects as exceptions for requests coming from a particular client IP address or going to a particular destination IP address. These exceptions are also passed on to the web unfiltered.

Setting up more than one appliance

When configuring Web Gateway in transparent bridge mode, we recommend setting up more than one appliance.

When a Web Gateway appliance is configured in this mode, it is implemented in an in-line position within your network. This means that all traffic is physically passing through the appliance, even if no ports are configured to receive the traffic and enable its filtering. Setting up only one appliance would therefore make it a single point of failure.

If you set up at least one other appliance, it can serve as a failover device. Another appliance will, however, not only perform failover functions, but also load balancing, receiving, and processing web traffic.

Avoiding a port shutdown under STP

When the Web Gateway appliances in your network are directly connected to switches that use the Spanning Tree Protocol (STP), ports needed for load balancing communication might be shut down under this protocol.

On most network switches, STP is used to avoid loops and ensure a single path of communication, shutting down redundant ports that cause such loops.

The protocol is also used, however, when two or more Web Gateway appliances are configured in transparent bridge mode. One of the appliances then takes the director role, in which it directs the web traffic that occurs to the other appliance or appliances for processing.

STP is used to communicate this role and the load balancing measures between the appliances.

If network switches with STP are directly connected to the Web Gateway appliances, it is highly likely that ports needed for this load balancing communication are shut down.

You can proceed in one of the following ways to avoid a shutdown:

- Disable STP on every switch that is directly connected to a Web Gateway appliance.

  Note: Do not use this method if other components of your network rely on these switches and STP.
- Install a second switch without STP between every Web Gateway appliance and every switch with STP that the appliance would be connected to.

  Setting up your network in this way ensures that load balancing on the Web Gateway appliances and other network components that rely on switches with STP are not impacted.

Configure port redirects for the transparent bridge mode

Configure port redirects for the transparent bridge mode to pass on particular requests to the web unfiltered.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure port redirects on and click Proxies (HTTP(S), FTP, SOCKS, ICAP ...).
3. Under Network Setup, select Transparent Bridge.

   The Transparent Bridge settings appear below the Network Setup settings.
4. In the list under Port redirects, specify an IP address and subnet mask for every port redirect that you want to configure.
5. Click Save Changes.

Network protocols

Network protocols determine the various ways in which data can be transmitted over a network. To intercept this data and make it available for filtering, proxies are set up on Web Gateway and configured for the different protocols.

You can set up and configure proxies for the following network protocols:

- HTTP (Hypertext Transfer Protocol and HTTPS (Hypertext Transfer Protocol - Secure)
- FTP (File Transfer Protocol)
- SOCKS (SOCKetS) protocol
- TCP (Transmission Control Protocol)
- IFP (Internet Fax Protocol)

The proxy functions on Web Gateway can also be configured for the network protocol known as ICAP (Internet Content Adaption Protocol) to set up an:

- ICAP server

HTTP proxy

This type of proxy is configured to cover web traffic going on under HTTP. It is configured by default on a Web Gateway appliance.

HTTP is a protocol that is widely used for transferring web pages and other data. Connections for data transfer can also be SSL-secured.

Configure an HTTP proxy

Configure an HTTP proxy on Web Gateway for web traffic going on under this protocol.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance where you want to configure an HTTP proxy. Then click Proxies.
3. Under HTTP Proxy, make sure Enable HTTP proxy is selected.
4. Add entries for listener ports to the HTTP port definition list as needed.

   After the initial setup, one entry is already inserted in this list by default. It has `0.0.0.0:9090` as its listener address and `443` as SSL port.

   a. Click the Add icon.

      The Add HTTP Proxy Port window opens.

   b. Use the options in this window to create and add list entries.

      Each entry specifies a listener address, as well as several other port parameters.
5. Configure the remaining settings as needed, for example, to adjust the Content type header when requests come in under HTTP that are directed to archives.
6. Click Save Changes.

Results

An HTTP proxy is now running on Web Gateway with the settings that you configured.

HTTP Proxy settings

Settings for running a proxy on Web Gateway under HTTP.

HTTP Proxy

Option	Definition
Enable HTTP proxy	When selected, a proxy is run on an appliance under HTTP.
HTTP port definition	Provides a list for entering the ports on an appliance that listen to client requests.
Anonymous login for FTP over HTTP	Specifies the user name for logging on as an anonymous user when requests are transmitted to an FTP server by an HTTP proxy on an appliance.
Password for anonymous login for FTP over HTTP	Sets a password for a user name.
Add Via HTTP header	When selected, a Via HTTP header is added to a request that is processed on an appliance. This option is selected by default.
Adjust content-type header for requests to archives (depending on the content encoding)	When selected, a content-type header in a request for access to an archive file is adjusted if this header does not match the content encoding that was detected for the archive.
Host header has priority over original destination address (transparent proxy)	When selected, requests that are sent to the proxy on an appliance in transparent proxy mode are recognized as traffic in explicit proxy mode and processed accordingly. Requests can, for example, be received on an appliance in transparent mode when they have been forwarded by a load balancer. If the proxy does not recognize the requests as traffic in explicit proxy mode, they will be forwarded to the web without filtering.

Option	Definition
	This option is only available if the explicit proxy mode is not already configured on an appliance. If the option is available, it is selected by default.

The following table describes the fields of an entry in the list of ports for web traffic going on under HTTP.

HTTP port definition – **List entry**

Option	Definition
Listener address	Specifies the IP address and port number for a port that listens to HTTP requests. If the port is configured for a director or backup node in a cluster where web traffic is processed in Proxy HA or Transparent Router mode, the network interface address of the cluster is specified here.
Serve transparent SSL connections	When selected, SSL-encrypted data can also be transferred using this proxy.
Ports treated as SSL	Provides a list of ports that handle incoming data as SSL-encrypted. Entries in this list are separated by commas. The list includes port 443 by default.
Transparent common name handling for proxy requests	When selected, common names sent within a request to the proxy are handled transparently.
McAfee Web Gateway uses passive FTP over HTTP connections	When selected, data can be transferred in FTP passive mode using HTTP connections.
Accept Proxy Protocol header	When selected, a Proxy Protocol header sent by a proxy forwarding web server data downstream is processed on Web Gateway. Sending of this header is optional, not required for the downstream proxy. The header information is extracted and different parts of it are stored as values of the Connection.IP, Connection.Port, and Connection.OriginalDestination.IP properties.
Comment	Provides a plain-text comment on a port that listens to HTTP requests.

FTP proxy

A proxy of this type is configured to process web traffic going on under FTP.

This protocol is well suited for transferring files, using separate connections for control functions and data transfer.

FTP Proxy settings

Settings for running a proxy on an appliance under FTP.

When a file is uploaded to the web from an FTP client and processed on Web Gateway, you can send progress indicators to the client by inserting the FTP Upload Progress Indication event into a suitable rule.

This will prevent a timeout on the client when processing takes more time, for example, due to scanning the file for infections by viruses and other malware.

FTP Proxy

Option	Definition
Enable FTP proxy	When selected, a proxy is run on an appliance under FTP.
FTP port definition list	Provides a list for entering the ports on an appliance that listen to client requests.
Allow character @ in FTP server user name (Authentication using USER ftpserveruser@ftpserver)	When selected, this character is allowed in a user name.
Enable authentication using USER proxyuser@ftpserveruser@ftpserver	When selected, this syntax is allowed for a user name.
Enable authentication using USER ftpserveruser@proxyuser@ftpserver	When selected, this syntax is allowed for a user name.
Enable customized welcome message	When selected, you can edit the welcome message that is shown to a user who sends a request for web access under the FTP. Type the welcome message into the Customized welcome message text field, using the appropriate values for the variables that are contained in the message. `Welcome to $MWG-ProductName$ $MWG-Version$ - build $MWG.BuildNumber$ Running on $System.HostName$ - $System.UUID$ $Proxy.IP$:$Proxy.Port$`
Select the command to be used for next-hop proxy login	Allows you to select the command that Web Gateway sends for logon when connecting to a next-hop proxy under FTP. The following commands can be selected: • SITE • OPEN • USER@Host

The following table describes an entry in the FTP port definition list.

FTP port definition list – List entry

Option	Definition
Listener address	Specifies the IP address and port number for a port that listens to requests from clients coming in on a Web Gateway appliance under FTP.
Port range for client listener	Configures a range of numbers for ports that listen to FTP requests received from clients. The range is configured by specifying port numbers for its beginning and end.
Port range for server listener	Configures a range of numbers for ports that listen to FTP responses received from web servers that requests were forwarded to.

Option	Definition
Allow clients to use passive FTP connections	When selected, requests can be sent from clients using passive connections under the FTP protocol.
McAfee Web Gateway uses same connections (active/passive) as clients does	When selected, Web Gateway uses the same type for forwarding web traffic as a client that sent a request to Web Gateway.
McAfee Web Gateway uses passive FTP connections	When selected, Web Gateway can forward web traffic using passive connections under FTP. **Note:** When the FTP-over-HTTP mode is configured, Web Gateway always uses active connections to reach out to the FTP server even if this checkbox is selected.
Comment	Provides a plain-text comment on a port that listens to FTP requests.

Best practices - Configuring FTP over HTTP

Working with FTP over HTTP, users can retrieve files from an FTP server without setting up and configuring an FTP client.

FTP over HTTP is a type of HTTP traffic going on between a web browser and a proxy, such as the proxy provided by Web Gateway. In most respects, sending an FTP-over-HTTP request is like sending any other HTTP request. The difference is that the requested resource resides on an FTP server rather than an HTTP server. FTP over HTTP traffic is configured in explicit proxy mode.

An FTP-over-HTTP request contains a URL prefixed with *ftp://* instead of *http://*. The HTTP host header value includes port 21 instead of no port number (in which case port 80 is assumed).

The following is an example of an FTP-over-HTTP request that a client sends to Web Gateway at the beginning of the communication.

```
GET ftp://10.10.80.200/ HTTP/1.1 User-Agent: Mozilla/4.0 (compatible; MSIE 8.0) Host: 10.10.80.200:21 Proxy-
Connection: Keep-Alive
```

When Web Gateway receives this request, it recognizes that the requested resource resides on an FTP server, due to the presence of *ftp://*" and port 21.

Web Gateway then uses native FTP to retrieve the resource from the FTP server. Any FTP response traffic is "translated" on Web Gateway into HTTP terms before passing it on to the client. Sometimes this requires Web Gateway to create an HTML page, so the client can display the results, for example, a listing of FTP directories, in the web browser.

Use of the HTTP proxy port

FTP over HTTP is handled on Web Gateway much like any other HTTP communication with a client. Requests should therefore be sent from the client to the HTTP proxy port on Web Gateway, which is 9090 by default.

There is no need to configure the FTP proxy port, which is 2121 by default, as use of this port is only required if a client sends requests under the native FTP protocol.

Advantages and disadvantages

FTP over HTTP has advantages and disadvantages. Using this method to retrieve files from an FTP server allows users to work with a web browser, which saves them the effort of setting up and configuring an FTP client, such as the open-source Filezilla client.

The major disadvantage of FTP over HTTP is that it does not allow users to upload files. File upload requires the use of native FTP, an FTP client program on the client system, and use of the FTP proxy port on Web Gateway if users choose to send traffic this way.

Another disadvantage of FTP over HTTP is that most web browsers have issues with it.

Configure your own FTP credentials for anonymous logon

When a client sends an FTP-over-HTTP request that does not contain user credentials to Web Gateway, preconfigured credentials are used by default to perform anonymous logon to the FTP server. You can configure FTP credentials of your own to replace the default credentials.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance that you want to configure FTP credentials on, then click Proxies.
3. Scroll down to HTTP Proxy and proceed as follows:
   a. Under Anonymous login for FTP over HTTP, type a user name.
   b. Under Password for anonymous login for FTP over HTTP, type a password.
4. Click Save Changes.

Troubleshooting browser issues arising for FTP over HTTP

Testing has shown that Mozilla Firefox is the only browser that does not require special attention from the user or administrator when doing FTP over HTTP.

Most web browsers have issues when requests are sent using FTP over HTTP. For some of these issues, you can implement workarounds.

Anonymous and non-anonymous logon

Some browsers can only handle FTP over HTTP when anonymous logon is allowed on an FTP server, as these browsers cannot send credentials as part of the URL in a request.

Other browsers prompt users for credentials when anonymous logon is not allowed on an FTP server.

When working with browsers that can handle non-anonymous logon, but do not prompt users, credentials can be submitted in one of the following ways:

- Credentials can be entered on the authentication page that is sent to the browser by Web Gateway.
- Credentials can be inserted in the URL that is sent from the browser to the FTP server. The URL format must be as follows:
  `ftp://`<user name>`:`<password>`@`<name of the FTP server>

Special characters within credentials

Some browsers do not encode FTP user names and passwords containing special characters correctly, rendering them useless and causing logon to fail.

There is no workaround for this issue, other than avoiding special characters in credentials. These browsers can be used, but are not recommended for FTP over HTTP when credentials for non-anonymous logon are required.

Proxy authentication

When doing FTP over HTTP using a proxy, for example, the proxy provided by Web Gateway, the proxy has to authenticate to the FTP server.

Some browsers cannot handle this authentication process. When Web Gateway sends a message that proxy authentication is required, these browsers do not send the user credentials back.

As a workaround, you can exempt these browsers from proxy authentication. For this exemption, a rule must be inserted in the rule set that you are using to control authentication. The rule recognizes the browser through the information provided by the user-agent information in the header of the request that is submitted.

This rule might look as follows if the browser is, for example, Google Chrome:

Name		
Exempt FTP over HTTP with Chrome from proxy authentication		
Criteria	Action	Event
Header.Get("User-Agent" matches "Chrome" AND URL.Protocol equals "ftp" –>	Stop Rule Set	

Proxy settings

Some browsers ignore the proxy settings when the protocol information in a URL is *ftp://*. Instead of sending an FTP over HTTP request to the proxy, they send a native FTP request directly to the FTP server.

There is no workaround for this issue. These browsers cannot be used for FTP over HTTP traffic.

Issues with commonly used web browsers

The following table shows issues that arise more often when doing FTP over HTTP in relation to some of the most commonly used web browsers.

Issues with web browsers when doing FTP over HTTP

Web browser	Issues	Solution
Mozilla Firefox	No issues known.	Can be used without taking additional measures for FTP over HTTP.
Microsoft Internet Explorer	Does not prompt users for credentials when anonymous logon is not allowed for FTP over HTTP on an FTP server. Encodes special characters within credentials incorrectly.	Can be used for FTP over HTTP when anonymous logon is allowed on an FTP server. When user for non-anonymous logon, credentials must be submitted in one of the following ways: • Entering credentials on the authentication page that is displayed by Web Gateway. • Inserting credentials in the URL that is sent to the FTP server The credentials must not contain special characters.
Google Chrome	Can only handle FTP over HTTP if an FTP server allows anonymous logon. Cannot handle proxy authentication.	Can be used if an FTP server allows anonymous logon, but requires a rule for skipping proxy authentication.
Opera	Cannot handle proxy authentication.	Can be used for FTP over HTTP, but requires a rule for skipping proxy authentication.
Safari	Ignores proxy settings when the protocol information in a URL is *ftp://*. Sends native FTP requests from a client directly to the FTP server instead, bypassing the configured proxy, for	No workaround: Cannot be used for FTP over HTTP.

Web browser	Issues	Solution
	example, the proxy provided by Web Gateway.	

Using WCCP to redirect FTP traffic

Requests that clients of Web Gateway send to servers under the FTP protocol can be redirected to Web Gateway using the WCCP (Web Cache Control Protocol) redirection method.

To send a request to a server under the FTP protocol, a client of Web Gateway opens the initial FTP connection. The client uses the IP address of the server for this connection. To let Web Gateway act as a proxy, the request is redirected to the IP address of the appliance that Web Gateway runs on.

Under the default settings, the client considers this redirection as a security risk and does therefore not continue with opening the FTP data connection. When redirection is performed using the WCCP protocol, you can solve this problem by modifying the settings as follows:

- Using the active FTP mode for the connection from the client to the proxy

  Clients are by default allowed to use the passive FTP mode. You can enforce the active FTP mode by disabling an option of the proxy settings on the user interface of Web Gateway.
- Configuring a port for redirection to the proxy

  This port must be entered in the list of ports that are redirected under WCCP.
- Letting the proxy use the IP address of the FTP server instead of its own IP address

  Setting a particular parameter ensures that the proxy uses this address.

After modifying the settings in this way, a client uses the active FTP mode. It sends the proxy an IP address and a port number to connect to. The proxy returns a synchronization message. In this message, the IP address of the FTP server is used as the source IP address of the proxy. The port number is 21 or 2020.

The client responds with the IP address of the FTP server as its destination IP address and the same port number. Requests from the client to the FTP server are then redirected to the proxy, using WCCP as the redirection method.

Note: The WCCP redirection method cannot be used for FTP traffic in transparent bridge or router mode.

Configure the use of WCCP for redirecting FTP traffic

To enable the use of the WCCP redirection method for requests that clients send to servers under the FTP protocol, configure the proxy settings as follows.

Task

1. Enforce use of the active FTP mode by clients.
   a. Select Configuration → Appliances.
   b. On the appliances tree, select the appliance that you want to enable use of the WCCP redirection method for, then click Proxies (HTTP(S), FTP, SOCKS, ICAP ...).
   c. Scroll down to FTP Proxy and make sure that Enable FTP proxy is selected.
   d. Select an entry in the FTP port definition list, click Edit, and under FTP Proxy Port, deselect Allow clients to use passive connections.
      Repeat this substep for all entries in the list.
2. Add ports 21 and 2020 to the ports that are used for redirection under WCCP.
   a. Within the Proxies settings, scroll to Transparent Proxy, and under Supported redirection methods, make sure that WCCP is selected.
   b. Select an entry in the WCCP services list, click Edit, and under Ports to be redirected type `21,2020`.
      Repeat this substep for all entries in the list.
3. Click Save Changes.
4. Within the relevanr settings, set the *ftp.match.client.data* parameter to *yes*.

This setting ensures that Web Gateway uses the IP address that it received from the client as its source IP address when responding to the client.

This address is the IP address of the FTP server in question, not the IP address of the Web Gateway appliance. The client does therefore not suspect a security risk.

Results

Requests sent from a client to a server under the FTP protocol are now redirected to Web Gateway, using the WCCP redirection method, and processed without problems.

Using the Raptor syntax for FTP logon

When Web Gateway is configured to run as an FTP proxy, the Raptor syntax can be used for logging on to an FTP server with Web Gateway as a proxy.

To perform this logon, the user who wants to access the FTP server can run the USER, PASS, and ACCEPT commands from a suitable FTP client. Using these commands, the FTP server is specified together with user names and passwords for both the FTP server and the Web Gateway proxy.

The command syntax is as follows:

`USER` <ftpuser>`@`<ftpserver> <proxyuser>

`PASS` <ftpuserpass>

`ACCT` <proxyuserpass>

The following table describes the meanings of the command parameters.

Command parameter for FTP logon

Option	Definition
ftpserver	FTP server that access is requested to
ftpuser	User name on the FTP server
ftpuserpass	Password for the FTP server
proxyuser	User name on the Web Gateway proxy
proxyuserpass	Password for the Web Gateway proxy

ICAP server

You can let Web Gateway appliances take the roles of servers and clients with web traffic going on between them under ICAP.

Under this protocol, an ICAP server modifies requests and responses when communicating with ICAP clients. Traffic can, therefore, go on in what is known as the REQMOD and the RESPMOD mode.

Secure ICAP

ICAP traffic can also be SSL-secured. It is then referred to as Secure ICAP or ICAPS traffic.

To secure ICAP traffic, you need to import a server certificate for each port on the appliance that takes the server role, listening to SSL-secured requests from its clients.

Requests that ICAP clients send to the server must include ICAPS as specification in the server address to enable SSL-secured communication.

ICAP clients are not required to submit certificates to the server.

Configure servers for ICAP communication

Configure servers for communication with clients under ICAP in each of the two ICAP modes.

When a Web Gateway appliance connects as an ICAP client to an appliance that takes the role of an ICAP server, it selects this server from a list that you must configure.

You must specify IP addresses or fully qualified domain names for them and complete this for both the REQMOD and the RESPMOD mode.

Note: A Web Gateway appliance that takes the ICAP server role must be configured in explicit proxy or Proxy HA (High Availability) mode. The Transparent Router and Bridge modes are not supported here.

When the Proxy HA (High Availability) mode is configured for an ICAP server, the virtual IP address that is specified here must also be configured as IP address for this server on the ICAP client, both within the REQMOD and RESPMOD parts.

The server port must be the one that you have also configured for this server in the ICAP Server settings.

Task

1. On a Web Gateway appliance that you want to act as an ICAP client, implement the ICAP Client rule set.
   a. Import the ICAP Client rule set from the library.
   b. Select Policy → Rule Sets, then click ICAP Client.
2. Configure servers that communicate with the clients in REQMOD mode.
   a. Click Edit next to ReqMod server.

      The Edit List (ICAP Server) window opens.

   b. Under List Content, click Add.

      The Add ICAP Server window opens.

   c. In the URI field under ICAP Server, specify a server.

      Type the IP address or the fully qualified domain name of a server, followed by the ICAP mode. Optionally, you can add a port. If you add no port, the default port 1344 is configured.

      The syntax for specifying this information is displayed above the field. For example, you can type one of the following strings:

      - `icap://10.213.246.89/reqmod`
      - `icap://10.213.246.89:1346/reqmod`
      - `icap://test-icap.micmwg.com/reqmod`
      - `icap://test-icap.micmwg.com:1346/reqmod`

      You can add more than one server to the list. Web Gateway tries the listed servers in round-robin mode until a connection is established.

   d. Click OK in each of the two open windows.
3. Configure servers that communicate with the clients in RESPMOD mode.
   a. Click Edit next to RespMod server
   b. Proceed in the same way as for the RESPMOD mode while specifying `respmod` for the ICAP mode.
4. Click Save Changes.

ICAP Server settings

Settings for running a server on an appliance that modifies requests and responses in communication with clients under ICAP.

ICAP Server

Option	Definition
Enable ICAP server	When selected, an ICAP server is run on an appliance.

Option	Definition
ICAP Port Definition list	Provides a list for entering the ports on an appliance that listen to requests from ICAP clients. When multiple ICAP servers are configured on different appliances within your network, requests coming in from ICAP clients are distributed among these servers in round-robin mode.
Select TLS version	Allows you to select a TLS or SSL protocol version for the ICAP traffic originating from requests and responses sent and received between an ICAP server and its clients. If one of these protocols is in use, the ICAP traffic is going on as *secured* traffic, which is also referred to as secure ICAP or ICAPS traffic. You can select one or more of the following protocol versions. • TLS 1.3 • TLS 1.2 • TLS 1.1 • SSL 3.0 — Use this protocol version only if it is required to ensure compatibility with an existing configuration.

The following table describes an entry in the ICAP port definition list.

ICAP port definition list – List entry

Option	Definition
Listener address	Specifies the IP address and port number for a port on the ICAP server that listens for requests from ICAP clients.
Send early 204 responses	When selected, these responses are sent.
Include Realm into authentication attributes	When selected, the realm is included in the attributes that are evaluated during the authentication process that is performed in ICAP communication.
Wait for complete ICAP request	When selected, an ICAP request is only processed after it has been completely received on the ICAP server, depending, however, on what you select from the following. • Never — Processing never waits until a request has been completely received. • Only for REQMOD requests — Processing only waits if a request was sent in REQMOD mode. • Only for FTP requests — Processing only waits if an FTP request was sent. • Always — Processing always waits until a request has been completely received.
Maximum concurrent REQMOD connections	Limits the number of connections that can run in REQMOD mode at the same time. The default maximum number is 100.
Maximum concurrent RESPMOD connections	Limits the number of connections that can run in RESPMOD mode at the same time. The default maximum number is 400.

Option	Definition
Preview size	Sets the preview size.
ICAPS	When selected, the connections used for the ICAP communication are SSL-secured. When this option is selected, the options explained in the following are activated. These options are related to the certificate that the ICAP server submits when connecting to ICAP clients over SSL-secured connections.
Subject, Issuer, Validity, Extensions, Fingerprint, Key	These fields display information about the server certificate that is currently in use.
Server certificate	Provides options for handling a server certificate. • Generate New — Opens a window for generating a new server certificate. • Import — Opens a window for importing a server certificate. • Export — Lets you browse to a location within your file system that a server certificate can be exported to. • Export key — Lets you browse to a location within your file system that the key file for a server certificate can be exported to.
Comment	Provides a plain-text comment on a port that listens to requests from ICAP clients.

SOCKS proxy

You can configure Web Gateway to run as a proxy that forwards web traffic under the SOCKS (Sockets) protocol.

When web traffic goes on under the SOCKS protocol, it also follows an embedded protocol, which can be, for example, HTTP or HTTPS.

The embedded protocol can be detected on Web Gateway, and if filtering is supported for web traffic under this protocol, the configured filtering rules can be processed for this traffic. If filtering is not supported, the traffic can be blocked by a suitable rule.

There are some restrictions to using the SOCKS protocol for the proxy functions on Web Gateway:

- The SOCKS protocol version must be 5, 4, or 4a.
- The BIND method is not supported for setting up connections under the SOCKS protocol.

Web traffic that is forwarded by a next-hop proxy under the SOCKS protocol can be protected using level 1 or 2 of the Kerberos authentication method.

In this case, encryption that would also make this traffic SSL-secured cannot by applied, so SSL scanning is not required. The default SSL Scanner rule set therefore includes a criteria part that lets this traffic skip SSL scanning.

Configuring a SOCKS proxy

To configure Web Gateway as a SOCKS proxy, you need to complete several activities.

- Enable the SOCKS proxy.
- Specify one or more proxy ports that listen to the SOCKS proxy clients when they send requests to Web Gateway.
  These ports are specified as part of the common proxy settings on Web Gateway.

- Create rules that control the behavior of the SOCKS proxy.

These settings are configured as part of the common proxy settings on Web Gateway.

Using properties and an event in rules for a SOCKS proxy

Two properties and an event are available to create rules for controlling the behavior of Web Gateway when it runs as a SOCKS proxy.

Note: There is no preconfigured SOCKS proxy rule set available in the default rule set system or the rule set library. If you want to use such rules, you need to create them and insert them in an existing rule set or create a rule set for them.

- *ProtocolDetector.DetectedProtocol* — This property can be used to detect the embedded protocol that is followed in web traffic under the SOCKS protocol, for example, HTTP or HTTPS.

  Its value is the protocol name in string format. When the embedded protocol cannot be detected, the string is empty.
- *ProtocolDetector.ProtocolFilterable* — This property can be used to find out whether filtering is supported for web traffic following the embedded protocol that has been detected.

  Its value is *true* if this traffic is filterable and *false* otherwise.

  If this property is processed in a rule, the *ProtocolDetector.DetectedProtocol* property is also filled with a value. If this value is an empty string for the latter property, which means no the embedded protocol could not be detected, the value of the *ProtocolDetector.ProtocolFilterable* property is, consequently, set to *false*.
- *ProtocolDetector.ApplyFiltering* — This event can be used to enable processing of other rules that are configured on Web Gateway for filtering web traffic under the protocol that has been detected.

Accordingly, the following rule enables processing of other rules for filtering web traffic under the SOCKS protocol if an embedded protocol has been detected that is filterable.

Name		
Enable filtering for SOCKS traffic following an embedded protocol that is filterable		
Criteria	Action	Event
ProtocolDetector.ProtocolFilterable is true	StopCycle	ProtocolDetector.ApplyFiltering

The following rule blocks SOCKS traffic if no embedded protocol is detected.

Name	
Block SOCKS traffic if no embedded protocol can be detected	
Criteria	Action
ProtocolDetector.DetectedProtocol equals " "	Block

If no rule is configured that would enable the filtering of SOCKS traffic or block it if no embedded protocol is detected, this traffic is allowed.

This means that if a request for web access is received from a SOCKS client on Web Gateway, it is forwarded to the requested web server without any further processing.

Configure SOCKS proxy settings

You can configure settings for a SOCKS proxy as part of the common proxy settings on Web Gateway.

Task

1. Select Configuration → Appliances .
2. On the appliances tree, select the appliance you want to configure as a SOCKS proxy, then click Proxies (HTTP(S), FTP, ICAP, and IM).
   The settings for configuring proxy functions appear in the configuration pane.
3. Scroll down to the SOCKS Proxy settings.
4. Configure these settings as needed.
5. Click Save Changes.

SOCKS Proxy

Settings for running a proxy on an appliance under the SOCKS (sockets) protocol

SOCKS Proxy

Option	Definition
Enable SOCKS proxy	When selected, a proxy is run on an appliance under the SOCKS protocol.
SOCKS port definition list	Provides a list for entering the ports on an appliance that listen to client requests for the SOCKS proxy.

The following table describes an entry in the SOCKS port definition list.

SOCKS port definition list – List entry

Option	Definition
Listener address	Specifies the IP address and port number of a port that listens for SOCKS requests.
Port range for UDP	Sets the range of ports used for listening to requests sent under the UDP protocol when a SOCKS proxy is configured.
Comment	Provides a plain-text comment on a port that listens to SOCKS requests.

SOCKS Proxy rule set

The SOCKS Proxy rule set is a library rule set for filteirng traffic that is going on under the SOCKS protocol.

Library rule set – SOCKS Proxy
Criteria – *Always*
Cycles – Requests (and IM) and responses

The rule set contains the following rules.

Filter traffic under the SOCKS protocol with filterable embedded protocol
ProtocolDetector.ProtocolFilterable <Protocol Detector Settings> equals true –> Stop Cycle — ProtocolDetector.ApplyFiltering
The rule uses the *ProtocolDetector.ProtocolFilterable* property to check whether the protocol that is embedded in the SOCKS traffic is filterable on Web Gateway. Filterable protocols are HTTP and HTTPS.
If either of these two protocols is detected, filtering is enabled by the rule event. If no embedded protocol is detected, the rule does not apply and processing continues with the second rule.

Block traffic under the SOCKS protocol if no embedded protocol is detected
ProtocolDetector.ProtocolFilterable <Protocol Detector Settings> equals " " –> Block <Default>
The rule blocks requests if no embedded protocol is detected.

Block traffic under the SOCKS protocol if detected protocol is not on whitelist
ProtocolDetector.DetectedProtocol <Protocol Detector Settings> is not in list Protocol Whitelist –> Block <Default>
The rule blocks requests if an embedded protocol is detected, but is not on a particular whitelist.
The rule is not enabled by default.

Using UDP under SOCKS

You can configure UDP (User Datagram Protocol) when Web Gateway is running as a proxy under the SOCKS protocol.

When traffic going on under the SOCKS protocol is processed by the proxy functions on Web Gateway, traffic that follows UDP can also be detected and forwarded. This traffic is not filtered, but forwarded as it is.

To allow the handling of UDP traffic in this way, you must complete the following configuration activities.

- Set the range of ports that listen to UDP traffic.
- Set a timeout on connections for UDP traffic.

You need not explicitly enable the handling of UDP traffic in addition to configuring these settings, as it is basically enabled by default.

When a client of Web Gateway sends a request for setting up a connection that follows UDP under SOCKS, the command name sent with the request is stored as the value of a property.

The name of the property is Command.Name and its value is *SOCKSUDPASSOCIATE* then. You can use this property in a rule for monitoring or other purposes.

Note: You can also use this property in a rule to disable processing of UDP traffic on Web Gateway.

Use of UDP is also monitored and shown on the dashboard under SOCKS Traffic Summary.

Configure settings for UDP under SOCKS

Configure settings for UDP to enable filtering of traffic that is going on under this protocol when Web Gateway is running as a proxy under the SOCKS protocol.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure UDP settings on and click Proxies (HTTP(S), FTP, SOCKS, ICAP ...).
3. In the configuration pane, scroll down to SOCKS Proxy. Under Port range for UDP, set the range of ports that listen to UDP traffic.
4. Scroll further down to Timeouts for HTTP(S), FTP, ICAP, SOCKS, and UDP. Under UDP timeout, set the timeout on connections for UDP.
5. Click Save Changes.

TCP proxy

TCP Proxy

Settings for running a proxy on a Web Gateway appliance under TCP.

When an appliance is configured to run as a TCP proxy, further administrator activities relating to TCP traffic can be performed, for example, configuring port forwarding for this traffic.

TCP Proxy

Option	Definition
Enable TCP proxy	When selected, a proxy is run on an appliance under TCP.
TCP port definition list	Provides a list for entering the ports on an appliance that listen to client requests for the TCP proxy.

The following table describes an entry in the TCP port definition list.

TCP port definition list **– List entry**

Option	Definition
Listener address	Specifies the IP address and port number for a port that listens for TCP requests.
Comment	Provides a plain-text comment on a port that listens to TCP requests.

Configuring TCP window scaling

You can configure TCP window scaling to improve network throughput. Using a particular configuration method, you can configure it differently for each TCP connection.

The size of the window for receiving TCP data packets, determines the amount of data that Web Gateway can receive from a web server or client on a given connection before an acknowledge (ACK) packet must be sent.

This size can vary up to a particular maximum value in a process known as *window scaling*. A larger maximum window size improves network throughput, especially on high-latency connections.

The risk with having a larger maximum window size is that devices such as routers and firewalls might not accept it. This can lead to a breach of window scaling with slow or no throughput.

To configure TCP window scaling, you can use the system settings, a rule event, or a system file.

If you use more than one of these methods, be sure not to configure conflicting values.

System settings

You can configure TCP window scaling as part of specifying the settings of the appliance system.

The Advanced Settings section of the Proxies settings provides options for enabling TCP window scaling and for setting the maximum window size.

Rule event

You can configure TCP window scaling using an event in a rule.

The Enable.ProxyControl event is provided for this purpose. It is executed with the settings for the Proxy Control module. These settings include options for enabling TCP window scaling and for setting the maximum window size.

Using this method, you can configure TCP window scaling differently for each connection.

There is no default or library rule set for TCP window scaling. So, if you want to use this method, you must create your own rule set with a suitable rule.

System file

You can configure TCP window scaling by entering parameter names and values in a system file.

The file name is sysctl.con. You can edit this file using the editor that is provided on the Web Gateway interface.

Using this method, you can configure the maximum and minimum sizes for the TCP window.

Precedence of the configured values

If more than one method is used for configuring TCP window scaling, the configured values are effective according to particular rules. In general:

- System file entries take precedence over system settings and event settings.
- System settings take precedence over event settings.

This means, for example:

- The maximum window size in the system file cannot be exceeded by configuring a larger size using any of the other methods.
- If the minimum window size in the system file is greater than a particular value, TCP window scaling cannot be disabled using any of the other methods.
- If TCP window scaling is disabled in the system settings, it cannot be enabled by the event settings.

Configure the TCP window size in a system file

You can configure the TCP window size using the sysctl.conf system file.

You can edit the system file on the Web Gateway interface.

Task

1. Select Configuration → File Editor.
2. On the files tree, navigate to the branch for the appliance where you want to configure the TCP window size, then select sysctl.conf.
3. In the editing area on the right, type parameter names and values for the TCP window size.

   Type these entries after this line:

   ```
   ### END AUTOGENERATED CONFIG
   ```

   Otherwise the parameter values will be overridden by those configured under Configuration → Appliances as part of the proxy system settings.

   ◦

     `net.core.rmem_max = <maximum size in KB>` — Sets the maximum size of the TCP window, specified as a number (without `KB`).

     This size is the absolute limit. If a larger size is configured using a different method, it is only effective up to the number of bytes configured here.

     Alternatively, you can set the maximum size using this parameter: `net.ipv4.tcp.rmem_max = <maximum size in KB>`. This size cannot be exceeded either by using any other configuration method.

If both parameters are entered in the system file, `net.core.rmem_max` takes precedence.

◦

`net.ipv4.tcp.rmem_min = <minimum size in KB>` — Sets the minimum size of the TCP window, specified as a number (without `KB`).

If you configure a value greater than 65535 here, window scaling cannot be disabled by any other setting on Web Gateway.

4. Click Save Changes.

IFP Proxy

Settings for running a proxy on an appliance under the IFP protocol

This protocol is used for transferring web pages.

IFP Proxy

Option	Definition
Enable IFP proxy	When selected, a proxy is run on an appliance under the IFP protocol.
IFP port definition list	Provides a list for entering the ports on an appliance that listen to client requests for the IFP proxy.
Maximum number of concurrent IFP requests allowed	Limits the number of IFP requests that are processed at the same time to the specified value. You can use this setting to prevent an overloading of the IFP proxy.

The following table describes an entry in the IFP port definition list.

IFP port definition list – List entry

Option	Definition
Listener address	Specifies the IP address and port number for a port that listens for IFP requests.
Send error message as redirect	When set to true, a user who sent a request is informed, for example, about a blocking of the request, by redirecting the request to an error message page. Otherwise the relevant information is sent as a normal message under the IFP protocol.
Comment	Provides a plain-text comment on a port that listens to IFP requests.

Instant messaging

Instant messaging proxies can be set up on an appliance to filter instant messaging (IM) chat and file transfer.

When users of your network participate in instant messaging communication, they send, for example, chat messages to an instant messaging server, receive responses to their messages, or send and receive files. An instant messaging proxy on an appliance can intercept and filter this traffic according to the implemented filtering rules. For this purpose, instant messaging traffic is redirected to the appliance.

The following network components are involved in the filtering process:

- **Instant messaging proxies** — Proxies can be set up on an appliance to filter instant messaging under different protocols, for example, a Yahoo proxy, a Windows Live Messenger proxy, and others.
- **Instant messaging clients** — These clients run on the systems of the users within your network to enable communication with instant messaging servers.
- **Instant messaging servers** — These are the destinations that are addressed by client from within your network.
- **Other components of your network** — Other components involved in instant messaging filtering can be, for example, a firewall or a local DNS server that redirects instant messaging traffic to an appliance.

When configuring instant messaging filtering, you need to complete configuration activities for the instant messaging proxy or proxies to ensure they intercept and filter the instant messaging traffic.

You also need to ensure that the instant messaging traffic is redirected to the instant messaging proxies. However, configuration activities for this are not performed on the clients, but on other components of your network. For example, DNS redirects or firewall rules are configured in a suitable manner.

An instant messaging proxy on an appliance is mainly intended to be used together with vendor IM client software that is provided, for example, by Yahoo, Microsoft, ICQ, or Google. But this client software can still change its behavior, for example, use a new logon server, without advance warning after a hidden update.

When using third-party client software, you should generally be aware that logon servers, protocol versions, or authentication methods could have been modified in comparison to those of the original client software, which can prevent an instant messaging proxy on an appliance from intercepting and filtering instant messaging traffic.

Configuring an instant messaging proxy

To configure an instant messaging proxy on an appliance, you need to configure the relevant parts of the Proxies settings of the Configuration top-level menu.

These are mainly settings for:

- Enabling an instant messaging proxy
- IP address and ports for listening to requests sent by instant messaging clients
- Settings for instant messaging servers
- Timeouts for instant messaging communication

Default values are preconfigured for all these settings after the initial setup of an appliance.

Instant messaging going on under the following protocols can be filtered:

- Yahoo
- ICQ
- Windows Live Messenger
- XMPP, which is the protocol used for Google Talk, Facebook Chat, Jabber, and other instant messaging services

The rules that are processed on an appliance for filtering instant messaging traffic are those that have *Requests (and IM)* configured as the processing cycle in the settings of their rule sets.

However, the *Responses* cycle can also be involved when instant messaging under the Yahoo protocol is filtered. Under this protocol, a requested file is transferred to a client in a response of the same kind as a response used for transferring files in normal web traffic. The file is stored on a server and retrieved by the client under HTTP, for example, using a suitable URL.

When problems arise in the communication between instant messaging client and proxy under a particular protocol, the client can also switch to using a different protocol and bypass the proxy this way. The client can even use a protocol for normal web traffic. On the dashboard of an appliance, this would result in a decrease of the IM traffic and an increase of the web traffic that is displayed.

Session initialization

During initialization of an instant messaging session between client and server, client requests can only be received on an appliance, but no responses can be sent back. As long as this is the case, the *IM.Message.CanSendBack* property will have *false* as its value when used in a rule.

We recommend that you do not implement any blocking rules with regard to session initialization, unless you want to block instant messaging traffic completely. You should also allow required helper connections, which are typically DNS requests or HTTP transfers.

Restrictions that you implement, for example, allowing only authenticated users, should rather apply to traffic going on during the session itself, such as chat messages and file transfers.

Configuring other network components for instant messaging filtering

The purpose of configuring other network components for instant messaging filtering is to redirect the instant messaging traffic that is going on between clients and servers to an appliance that has one or more instant messaging proxies running.

For example, under the ICQ protocol, clients send their requests to a server with the host name *api.icq.net*. For instant messaging filtering, you need to create a DNS redirecting rule that lets this host name be resolved not to the IP address of the ICQ server, but to that of the appliance.

In a similar way, firewall rules can be created to direct instant messaging traffic to an appliance rather than to an instant messaging server.

Filtering instant messaging traffic under Windows Live Messenger

When configuring the filtering of instant messaging traffic that is going on under the Windows Live Messenger protocol, the following is useful to know.

The host name of the instant messaging server is *messenger.hotmail.com*. This is the host name that must be resolved in a redirecting rule by the IP address of an appliance with an instant messaging proxy.

Sometimes a client connects to the server without requesting the host name to be resolved in a DNS lookup. In this case, it can help to find and remove the following registry entry within the client settings:

```
geohostingserver_messenger.hotmail.com:1863, REG_SZ
```

For a successful logon to a server, the following URL must be accessible to a client without authentication:

http://login.live.com

For this reason, you need to insert this URL in the whitelists that are used by the implemented web filtering rules on an appliance.

Filtering Instant messaging traffic under ICQ

When configuring the filtering of instant messaging traffic that is going on under the ICQ protocol, the following is useful to know.

The host names of the instant messaging servers are as follows:

- *api.icq.net* (Service request server: new since parting from AOL)
- *ars.icq.com* (File transfer proxy: new since parting from AOL)
- *api.oscar.aol.com* (Old service request server)
- *ars.oscar.aol.com* (Old file transfer proxy)
- *login.icq.com* (For new logon procedure)
- *login.oscar.aol.com* (For old logon procedure)

ICQ clients log on to a server in an encrypted process that cannot be intercepted by the instant messaging proxy on an appliance.

But after this, an ICQ client asks the service request server for information about the session server, using the magic token received after the logon. Here the instant messaging proxy intercepts. The filtering process then uses another logon procedure after the client name has been announced in the communication with the session server.

In contrast to the vendor Yahoo client, the vendor ICQ client ignores any Internet Explorer connection settings.

Filtering instant messaging traffic under Yahoo

When configuring the filtering of instant messaging traffic that is going on under the Yahoo protocol, the following is useful to know.

The list of instant messaging servers that requests are sent to can be very long. The following is a list of the host names of servers that are or have been in use. New servers can have appeared by now that would have to be added to the list.

- *vcs.msg.yahoo.com*
- *vcs1.msg.yahoo.com*
- *vcs2.msg.yahoo.com*
- *scs.yahoo.com*

- *cs.yahoo.com*
- *relay.msg.yahoo.com*
- *relay1.msg.dcn.yahoo.com*
- *relay2.msg.dcn.yahoo.com*
- *relay3.msg.dcn.yahoo.com*
- *mcs.msg.yahoo.com*
- *scs.msg.yahoo.com*
- *scsa.msg.yahoo.com*
- *scsb.msg.yahoo.com*
- *scs.msg.yahoo.com*
- *scs-fooa.msg.yahoo.com*
- *scs-foob.msg.yahoo.com*
- *scs-fooc.msg.yahoo.com*
- *scs-food.msg.yahoo.com*
- *scs-fooe.msg.yahoo.com*
- *scs-foof.msg.yahoo.com*
- *scsd.msg.yahoo.com*
- *scse.msg.yahoo.com*
- *scsf.msg.yahoo.com*
- *scsg.msg.yahoo.com*
- *scsh.msg.yahoo.com*

For a successful logon to a server, the following URLs must be accessible to a client without authentication:

- *http://vcs1.msg.yahoo.com/capacity*
- *http://vcs2.msg.yahoo.com/capacity*

For this reason, you need to insert these URLs in the whitelists that are used by the implemented web filtering rules on an appliance.

Even if the option Connect directly to the Internet has been enabled within the settings on a Yahoo client, it might still use Internet Explorer connection settings. This can cause the logon to fail in a later stage of the process. Therefore, we recommend that you also insert the URL **login.yahoo.com** in a whitelist.

Issues with instant messaging filtering

Issues with instant messaging filtering can involve, for example, the connection between client and server or the application of the implemented filtering rules.

Keep-alive data packets are sent in regular intervals as part of the instant messaging traffic to indicate the communication partners are still connected and responsive. Intervals vary between 20 and 80 seconds, depending on the IM protocol and client software. These data packets are not processed by the filtering rules that are implemented on an appliance.

If you detect such data packets in a troubleshooting situation, you can use rule engine tracing to see which rules are still executed.

When a client sends a request for logon to the server, it is redirected to the appliance if you have configured the appropriate settings. However, a client can at the same time try to log on to another server that requires SSL-secured authentication. If this fails, the client can also drop the connection to the appliance.

Some clients also provide options for performing basic troubleshooting tests after a failure to log on to the server.

XMPP proxy

When filtering instant messaging communication on an appliance, one of the methods you can use is to set up a proxy under the XMPP (Extensible Messaging and Presence Protocol).

This protocol is also known under the name of Jabber. It is used, for example, to participate in Facebook chats or Google talk going on between an XMPP client and server.

You can configure settings for the XMPP proxy on the user interface under Configuration → Proxies.

When the SSL Scanner rule set is not enabled on an appliance, traffic going on between an XMPP client and this appliance is not encrypted, but filtered by all rules that are enabled on the appliance. If the client does not accept unencrypted traffic, the connection is closed.

When the SSL Scanner rule set is enabled, traffic is encrypted and inspected using SSL scanning to make it available for filtering by other rules on the appliance.

Enable ICMP redirects

You can enable redirects to Web Gateway under ICMP for requests to access the web sent from clients.

ICMP redirects are not allowed by default on Web Gateway because they might create a security issue.

If you run Web Gateway in an environment where ICMP redirects are required, you can let them be accepted by editing the sysctl.conf system file.

Note: To edit sysctl.conf system file, use the File Editor that is provided for this purpose on the user interface.

Task

1. Select Configuration → File Editor.
2. Under Files in the navigation pane, expand the entry for the Web Gateway appliance where you want to allow ICMP redirects. Then select sysctl.conf.
3. After the last line of the file content that shows up in the configuration pane, append these lines by typing or pasting them:
   ```
   net.ipv4.conf.all.accept_redirects = 1 net.ipv4.conf.all.secure_redirects = 1
   net.ipv4.conf.default.accept_redirects = 1 net.ipv4.conf.default.secure_redirects = 1
   ```
4. Click Save Changes.
5. Restart the appliance to let the changed system file content become effective.

Results

ICMP redirects are now accepted on the Web Gateway appliance that you configured this acceptance for.

Controlling outbound source IP addresses

Using different source IP addresses for outbound connections from Web Gateway to web servers or next-hop proxies can lead to connection problems. To avoid these problems, you can replace these addresses with a single address.

Different source IP addresses might be used, for example, when load balancing is configured for multiple Web Gateway appliances. Load balancing can lead to connection problems on the side of the involved web servers or next-hop proxies. Problems can, for example, arise when source IP addresses change during a session period.

To avoid these problems, you can configure a rule that replaces changing source IP addresses with a single address.

This single address does not have to be fixed. You can set up a list of IP addresses and let the rule select an address in a particular position on the list. The address that replaces other addresses then varies according to what you have entered in that position.

Network setups for controlling outbound source IP addresses

Controlling outbound source IP addresses is supported for network setups with:

- IPv4 or IPv6
- HTTP, HTTPS, FTP, or SOCKS proxy

  Note: Instant messaging is not supported.
- Proxy (with optional WCCP) mode

  The transparent router mode is supported if the source IP addresses that are used for replacing other addresses are configured as aliases.

  The Proxy HA and transparent bridge modes are not supported.

Periodic rule engine triggering is also possible when control of outbound source IP addresses is implemented.

Sample rule for controlling outbound source IP addresses

A rule that replaces outbound source IP addresses by a single address, for example, when connections to next-hop proxies are set up, could look as follows:

Name		
Use proxy depending on the destination		
Criteria	Action	Events
URL.Destination.IP is in range list Next Hop Proxy IP Range List –> OR *URL.Destination.IP is in list Next Hop Proxy IP List*	Continue	Enable Next Hop Proxy<Internal Proxy> Enable Outbound Source IP Override(Proxy.OutboundIP(2))

The criteria of the rule specifies when a next-hop proxy is used. The first of the two events sets up a connection to a next-hop proxy.

The second event, *Enable Outbound Source IP Override*, is for controlling outbound source IP addresses. It replaces ("overrides") any source IP address that is submitted with a request by an IP address that is taken from a list.

An event parameter, which is itself a property, specifies the IP address. The name of the property is *Proxy.OutboundIP*. Its value is the IP address in the list position determined by the property parameter.

List of IP addresses for controlling outbound source IP addresses

The list of IP addresses that you can use to replace outbound source IP addresses is part of the *Proxies* settings. You can find it there under *Advanced Outgoing Connection Settings*. Its name is *Outbound Source IP list*.

The following applies regarding the position of an IP address in the list:

- The list index starts from 0. If you specify, for example, 2, as the parameter of the *Proxy.OutboundIP* property to determine a position, the *third* IP address on the list is selected.
- If you specify a parameter value that is higher than the number of list entries, the position is determined by calculating *<parameter-value> modulo <number-of-list-entries>*.

  For example, if you specify 5 for a list that has only three list entries, the result of the modulo calculation is 2. The third IP address on the list is then selected.

Network routing and IP address spoofing

The IP addresses that are inserted into data packets by the *Enable Outbound Source IP Override* event are non-local source IP addresses. You must therefore configure network routing in a suitable way.

Data packets that are sent back from a web server to a client must be routed to the proxy on Web Gateway. You can, for example, use static routes to route the data packets.

When the *Enable Outbound Source IP Override* event is triggered and you have IP address spoofing enabled, the event also overrides this setting.

Logging the use of outbound IP source addresses

Several properties are available for logging data about outbound connections, including the source IP address and port that Web Gateway uses when connecting to web servers or next-hop proxies.

These properties are set to particular values, regardless of whether you have configured a single source IP address, using the *Enable Outbound Source IP Override* event. But you can also use them in this case.

- *Proxy.Outbound.IP* — Stores the source IP address that Web Gateway uses when connecting to web servers and next-hop proxies.

Note: Do not confuse this property with *Proxy.OutboundIP*, which has no dot before *IP* and is used together with the *Enable Outbound Source IP Override* event to select a single source IP address from a list.

- *Proxy.Outbound.Port* — Stores the source port that is used by Web Gateway when connecting to web servers or next-hop proxies.
- *Proxy.Outbound.IPList* — Stores the list of source IP addresses that Web Gateway can select an address from when connecting to web servers and next-hop proxies.

  The list is configured as part of the *Proxies* settings under *Advanced Outgoing Connection Settings*. Its name is *Outbound Source IP list*. When a single source IP address for outbound connections is configured, it is taken from this list.

Configure control of outbound source IP addresses

Replace different outbound source IP addresses with a single address to avoid connection problems.

Task

1. Select Configuration → Appliances.
2. Select an appliance for configuring the replacement of IP addresses, then select Proxies (HTTP(S), FTP, SOCKS, ICAP ...) and scroll down to Advanced Outgoing Connection Settings.
3. Under Outbound source IP list, add one or more IP addresses to the list of source IP addresses for outbound requests.
4. Add the following event to an existing rule for connections to web servers or next-hop proxies: *Enable Outbound Source IP Override* with *Proxy.OutboundIP* property as a parameter.

Results

The rule now uses the list that you have configured to select an IP address for replacing different outbound source IP addresses.

Node communication protocols

When Web Gateway appliances run as director and scanning nodes in a Central Management configuration, communication between the nodes uses the Virtual Router Redundancy Protocol (VRRP) and MWG Management Protocol.

Use of the protocols depends on the proxy settings that you have configured on the appliances that run as nodes. The protocols differ with regard to the activities of director and scanning nodes that are covered by them.

Virtual Router Redundancy Protocol

The Virtual Router Redundancy Protocol is used when you have configured Web Gateway as a proxy in transparent router mode or High Availability proxy mode.

Under this protocol, virtual IP addresses are assigned to active director nodes and backup director nodes. The protocol also determines which director node takes the active director role.

MWG Management Protocol

The MWG Management Protocol is used in Transparent Router and High Availability proxy mode. Under this protocol, scanning nodes are identified that are available for processing web traffic.

The node that takes the active director role sends out broadcast messages to the scanning nodes, using the IP address that you have configured as its source IP address under the *Management IP* option of the respective proxy settings.

The protocol lets scanning nodes that are available within the same network segment respond in regular intervals to the discovery messages of the director node.

Security considerations

The security features of the Virtual Router Redundancy Protocol and MWG Management Protocol are similar to that of the Address Resolution Protocol (ARP).

The Virtual Router Redundancy Protocol uses multicast with an IP address that is not routed beyond the local broadcast domain. MWG Management Protocol uses broadcast messages.

A malicious node on the same network segment might send VRRP messages and hence impersonate itself as the active director node holding the respective virtual IP address. If that node decides to drop all data packets it receives for the virtual IP address, network connectivity stops for the clients that are connected to Web Gateway.

Tip: Best practice: Use IP addresses from a protected network segment when configuring proxy settings according to the Virtual Router Redundancy Protocol and the MWG Management Protocol. This will prevent malicious nodes from impacting Web Gateway activities.

Packet size handling

When communication between Web Gateway on an appliance and its clients requires that the size of data packets is handled in a flexible manner, only the explicit proxy mode can be configured as usual.

Other modes require an additional configuration effort in this case.

The size of data packets is measured by the MTU (Maximum Transmission Unit) parameter, which limits the number of bytes that can be sent in one packet.

The method of negotiating the value for this parameter between communication partners is known as *Path MTU Discovery*. It is not available for the Transparent Bridge mode.

For example, when Web Gateway sends a data packet to a client that it connects to through a VPN (Virtual Private Network) tunnel, the MTU that the VPN tunnel can handle might be 1412, whereas the MTU of the data packets is 1500.

The VPN gateway then sends a message under the ICMP protocol to inform its partner about the required size, but this message cannot be processed unless the configured network mode is the explicit proxy mode.

To solve this problem for the other modes, reduce the MTU parameter value for the network interface on Web Gateway that is used for the communication, in this case, for communication with clients behind a VPN tunnel. Set the parameter to the value that is required, for example, to 1412.

The MTU parameter is configured on the user interface as part of the Network Interfaces settings for the IPv4 or IP6 protocol, which can be accessed under Configuration → Appliances.

Advanced and extended proxy settings

In addition to configuring network modes and protocols for proxies on Web Gateway, you can use advanced settings to configure functions like IP spoofing or the maximum number of client connections. You can also extend proxy configuration to include DNS and DLX services.

The common web cache on Web Gateway is also enabled as part of the proxy settings.

Advanced settings are available for configuring various proxy functions and parameters. Additionally, there are advanced settings that apply only to the network mode that has been selected. Settings for timeouts are also provided.

- **Advanced settings** — Include settings for configuring advanced proxy functions and parameters, for example, the maximum number of connections that are available for communication between clients and a proxy or the maximum amount of data that is transmitted on a connection.
- **Advanced Outgoing Connection settings** — Include settings for configuring the use of IP spoofing with any of the network modes that can be selected for running proxies.
- **Timeouts for HTTP(S), FTP, ICAP, IFP, SOCKS and UDP** — Include settings for timeouts, for example, to limit the time interval that elapses between accepting a client connection and receiving the first request over this connection.

You can extend proxy configuration to configure settings for services that provide additional data to Web Gateway.

- **DNS settings** — Include settings for sending queries to a DNS (Domain Name System) server and for caching responses.
- **Data Exchange Layer settings** — Include settings for a timeout on a connection to a DXL (Data Exchange Layer) server and the topics that the DXL service is to send information about.

Advanced Settings (for proxies)

Settings for advanced proxy functions

Advanced Settings

Option	Definition
Maximum number of client connections	Limits the number of connections between a proxy on an appliance and its clients. Specifying 0 means that no limit is configured. Default: 50000 connections
Handle responses from server (content-encoding)	Provides options for handling the content in the body of a response from a web server that is forwarded to a client by Web Gateway. The content can be handled differently depending on whether it is compressed, for example, when GZIP encoding has been applied, or not. Compressed content can be extracted to allow access, inspection, and other treatment according to the rules that are configured on Web Gateway. Forwarding to the client is only performed if and to the extent that the rules allow it. • Extract but do not compress — Compressed content is extracted and forwarded uncompressed to the client. Uncompressed content is forwarded as it is. • Extract and compress if server response is compressed — Compressed content is extracted and compressed again before forwarding it to the client. Uncompressed content is forwarded as it is. • Extract and compress if client supports compression — Compressed content is extracted and compressed again before forwarding to the client if the client supports compression. Otherwise it is forwarded uncompressed. Uncompressed content is compressed and then forwarded if the client supports compression. Otherwise it is forwarded uncompressed. • Do not extract and not compress — Compressed content is not extracted and forwarded to the client compressed. Uncompressed content is forwarded uncompressed. Not extracting compressed content reduces load in content forwarding. This option is therefore useful when content inspection or other treatment is not required. For example, if you only want to apply URL filtering to web traffic, content extraction is unnecessary. Compressed content is, however, extracted under this option if the Dynamic Content Classifier (DCC) is called in case a URL could not be rated using Trusted Source information. To call the DCC, the following setting within the URL settings must be selected: Enable the Dynamic Content Classifier if GTI web categorization yields no result. The extracted content is forwarded uncompressed to the client.

Option	Definition
Handle compressed requests from client	Provides options for handling requests that were received in compressed format from a client of Web Gateway. • Ignore — The compressed content is not extracted and filtered, and the request is forwarded to the web server in compressed format. • Extract — The compressed content is extracted, so it can be filtered, but not compressed again before it is eventually forwarded to the web server. • Extract and compress again — The compressed content is extracted, so it can be filtered, and compressed again before it is eventually forwarded to the web server.
Number of working threads	Specifies the number of threads used for filtering and transmitting web objects when a proxy is run on an appliance.
Number of threads for AV scanning	Specifies the number of threads used to scan web objects for infections by viruses and other malware when a proxy is run on an appliance.
Use TCP no delay	When selected, delays on a proxy connection are avoided by not using the Nagle algorithm to assemble data packets. This algorithm enforces that packets are not sent before a certain amount of data has been collected.
Maximum TTL for DNS cache in seconds	Limits the time (in seconds) that host name information is stored in the DNS cache.
Timeout for errors for long running connections	Sets the time (in hours) that a long-running connection to another network component is allowed to remain inactive before Web Gateway closes the connection. The default time is 24 hours. This setting prevents the performance of a Web Gateway appliance from being impacted by long-running connections that run extremely long. Time is measured as follows for the different connection protocols to determine whether the timeout has been reached. • HTTP, HTTPS (with content inspection), ICAP, and similar protocols: Time is measured for every request that is sent on a connection. • SOCKS (when the embedded protocol is not followed), tunneled HTTP, HTTPS (without content inspection), and similar protocols: Time is measured for a connection as a whole. • FTP: Time is measured for the control connection. When the connection is closed, an error is generated, which can be handled by the rules in an Error Handler rule set.
Check interval for long running connections	Sets the time (in minutes) that elapses between check messages sent over a long-running connection.
Maximum amount of data per connection or request	Sets the amount of data (in MB) that can be sent on a long-running connection to another network component before Web Gateway closes the connection.

Option	Definition
	The default amount is 10,240 MB. This setting prevents the performance of a Web Gateway appliance from being impacted by long-running connections that carry a very high data load. Data load is measured as follows for the different connection protocols to determine whether the maximum amount has been reached. • HTTP, HTTPS (with content inspection), ICAP, and similar protocols: Data load is measured for every request that is sent on a connection. • SOCKS (when the underlying protocol is not followed), tunneled HTTP, HTTPS (without content inspection), and similar protocols: Data load is measured for a connection as a whole. • FTP: Data load is measured for the data connection. When the connection is closed, an error is generated, which can be handled by the rules in an Error Handler rule set. The following properties are then set to the value of the measured data to be available for the error handling rules: Bytes.ToClient, Bytes.ToServer, Bytes.FromClient, Bytes.FromServer.
Volume interval for connections	Sets the volume interval for long-running connections.
Internal path ID	Identifies the path an appliance follows to forward internal requests (not requests received from clients), for example, requests for style sheets used to display error messages.
Bypass RESPmod for responses that must not contain a body	When selected, responses sent in communication under the ICAP protocol are not modified according to the RESPMOD mode if they do not include a body.
Call log handler for progress page updates and objects embedded in error templates	When selected, the rules in the log handler rule set that is implemented on the appliance are processed to deal with the specified updates and objects.
Allow connections to use local ports using proxy	When selected, local ports can be used for requests on an appliance that a proxy is run on.
Use virtual IP as the Proxy.IP property value	When selected, the value for the Proxy.IP property in High Availability mode is a virtual IP address for all nodes in a configuration. It is the virtual IP address that is used by clients to connect to the proxy. When the director node redirects a request sent from a client to a scanning node, this address is the value of the Proxy.IP property also on the scanning node (not the physical address of the scanning node).
HTTP(S): Remove all hop-by-hop headers	When selected, hop-by-hop headers are removed from requests received on an appliance that an HTTP or HTTPs proxy is run on.

Option	Definition
HTTP(S): Inspect via headers to detect proxy loops	When selected, via headers in requests received on the appliance that an HTTP or HTTPS proxy is run on are inspected to detect loops.
HTTP(S): Host from absolute URL has priority over host header	When selected, the host names corresponding to absolute URLs in requests received on an appliance that an HTTP or HTTPS proxy is run on are preferred to the host names contained in the request headers.
Encode own IP address in progress page ID to enable non-sticky load balancers	When selected the own IP address is encoded in the progress page ID.
HTTP(S): Maximum size of a header	Sets a limit to the size (in MB) for the header of a request or response sent in HTTP(S) traffic. Default: 10 MB
Listen backlog	Specifies a value for the listen backlog. Default: 128
Limit for working threads doing IO in web cache	Sets a limit to the number of working threads for the web cache. Default: 25
Progress page limit	Sets a limit to the size (in KB) of the progress page. Default: 40,000 KB
Enable TCP window scaling	When selected, the initial size of the window for receiving TCP data packets can be increased up to a maximum value that depends on a scaling factor. This factor is configured under TCP window scale. With a larger window size, Web Gateway can receive more data from a web server or client on a given connection before an acknowledge (ACK) packet must be sent. **Benefit:** Improved network throughput, especially on high-latency connections **Risk:** If routers or firewalls do not accept a larger window size, window scaling might break up, leading to slow or no throughput. **Recommended:** Reduce the window to a size that results in an acceptable performance. Default: Enabled **Caution:** When this option is disabled, no window scaling is performed. Disable the option with caution.
TCP window scale	Sets the scaling factor that determines the maximum size of the window for receiving TCP data packets. If window scaling is enabled, the initial window size can be increased using this scaling factor, which is calculated by taking base 2 to the power of the value that you specify here. For example, if you specify 1, the scaling factor is 2^1 = 2, so the maximum window size is doubled.

Option	Definition
	If you specify 0 for a scaling factor 1, the initial window size is kept for Web Gateway. Window scaling can still be used then for the receive window of the communication partner. Range of values: 0–4 Default: 7 **Note:** With this default, the receive window can be increased to a maximum size of 8192 KB.

Proxy-Generated Error Messages

Settings for messages to the user generated on the proxy

Proxy-Generated Error Messages

Option	Definition
Language	Provides settings for selecting the language of a user message. • Auto (Browser) — When selected, the message is in the language of the browser that a request was received from.. • Force to — When selected, the message is in the language chosen from the list that is provided here.
Collection	Provides a list for selecting a collection of templates for user messages. • Add — Opens the Add Template Collection window for adding a template collection. • Edit — Opens the Template Editor for editing a template collection.

Periodic Rule Engine Trigger

Settings for connecting to web servers, calling the rule engine, and downloading data

Periodic Rule Engine Trigger

Option	Definition
Enable Periodic Rule Engine Trigger	When selected, connections to the web servers specified in list called URL definition list are set up in regular intervals. The interval for each web server connection is also specified on the list. When the interval has elapsed, the rule processing module (rule engine) on an appliance is called, a connection to the web server is set up, and data is downloaded from the web server and passed on to the rule engine for processing. Data is only downloaded under the HTTP and HTTPS protocols.

Option	Definition
	Web servers that connections are set up to in this way include next-hop proxy servers and other servers used for providing particular services in the web.
URL definition list	Provides a list of web servers that a connection can be set up to.

The following table describes a list entry in the URL definition list.

URL definition list – **List entry**

Option	Definition
Host	Specifies the IP address and port number or the URL of a web server that a connection can be set up to.
Trigger interval	Specifies the interval (in seconds) that elapses before the next attempt to set up a connection to a web server.
Comment	Provides a plain-text comment on a web server connection.

Timeouts for HTTP(S), FTP, and ICAP

Settings for timeouts on connections for communication under the HTTP, HTTPS, FTP, and ICAP protocols

Timeouts for HTTP(S), FTP, and ICAP

Option	Definition
Initial connection timeout	Limits the time (in seconds) that elapses before a newly opened connection is closed if no request is received to the specified value.
Connection timeout	Limits the time (in seconds) that elapses before a connection is closed if a client or server remains inactive during an uncompleted request communication to the specified value.
Client connection timeout	Limits the time (in seconds) that elapses before a connection from the proxy on an appliance to a client is closed between one request and the next to the specified value.
Maximum idle time for unused HTTP server connections	Limits the time (in seconds) that elapses before a connection from the proxy on an appliance to a server is closed between one request and the next to the specified value.

Domain Name System

Queries are sent to Domain Name System (DNS) servers that Web ´Gateway connects for retrieving the IP addresses that match the host names submitted in user requests.

Using DNS servers according to domains

The use of DNS (Domain Name System) servers to resolve domain information provided in URLs into IP addresses when requests for web access are processed on Web Gateway can be configured according to the domains of the requested destinations.

This use of DNS servers is also known as conditional DNS forwarding.

Domains, for example, testnet.webwasher.com, are entered into a list together with the IP address of the DNS server that is used to resolve the URL information. More than one DNS server can be specified this way for a domain.

When a request to a particular destination on the web is sent to Web Gateway, it is forwarded to a DNS server according to this list.

The use of a particular DNS server can be configured dynamically with DHCP (Dynamic Host Configuration Protocol. This is also the default setting after the initial setup of a Web Gateway appliance.

If both configuration with DHCP and conditional DNS forwarding are configured, DHCP takes precedence and conditional DNS forwarding is bypassed.

Note: If a BIND server is configured as a DNS server, the DNS server settings that are stored in a configuration file on Web Gateway will be overwritten. To keep these settings for domain name resolving, you need to enter them manually again.

Configure the use of DNS servers according to domains

To enable the use of DNS servers according to the domains of destinations in the web, configure the Domain Name Service settings in a suitable manner.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure the use of DNS servers for and click Domain Name Service.
3. Configure the settings in the Conditional DNS Forwarder Configuration section as needed.
4. Click Save Changes.

DNS proxy settings

Settings for handling queries to a domain name system server (DNS server).

DNS Settings

Option	Definition
IP protocol version preference	Lets you select the protocol version that is preferred when retrieving IP addresses from a DNS server. • Same as incoming connection — When selected, the protocol version is used that is already in use on the incoming connection. • IP4 — When selected, this protocol version is used. • IP6 — When selected, this protocol version is used. • Use other protocol version as fallback — When selected, the other protocol version is used if using the preferred version resulted in a failure. When this option is selected, you can also configure the following. ◦

Option	Definition
	Enable simultaneous DNS queries for IPv4 and IPv6 — When selected, DNS queries for IPv4 and IPv6 addresses are sent at the same time. When this option is selected, you can also configure the following. ◦ Time to wait for results with a preferred IP version (IPv4/IPv6) after initiating simultaneous DNS queries — Limits the time (in milliseconds) that elapses until a connection that uses the other protocol version is accepted when no connection could be set up using the preferred version. ◦ Count of IP addresses of the preferred version (IPv4/IPv6) to be used from the DNS query results — Limits the number of IP addresses that are tried under the preferred protocol version for setting up a connection before IP addresses are tried under the other version. The number of retries that can be configured ranges from 1 to 4. A query for retrieving IP addresses from a DNS server can result in multiple IPv4 of IPv6 addresses, Whether an IPv4 or an IPv6 address is used for setting up a connection, depends on what you have configured above. When multiple IP addresses are available within the same address family (IPv4 or IPv6), addresses are sorted according to several parameters. Connection attempts are then made using these addresses in the order in which they are sorted. The parameters for sorting IP addresses are listed in the following. They are applied in the order they are listed. • Precedence of an IP address The precedence of an IP address is calculated based on its address prefix. An IP address with a higher precedence value is tried for connecting before an address with a lower value. • Scope of an IP address An IP address can have different scopes as follows: ◦ Link local ◦ Site or uniquely local ◦ Global The scopes are used for sorting in the order they are listed here. • Connection time (round trip time) Connection history is recorded. So when less time was required for setting up a connection using a particular IP address on a previous occasion, this address is preferred over another IP address that required more time. • Least recently used IP address Connection history is also used to determine when IP addresses were used for the last time. An IP address that

Option	Definition
	was used less recently than another IP address is preferred of this address.
Minimal TTL for DNS cache	Sets a minimum time (in seconds) that must have elapsed before data stored in the DNS cache is deleted.
Maximal TTL for DNS cache	Set a maximum time (in seconds) that must have elapsed before data stored in the DNS cache is deleted.
Flush DNS cache	Flushes the DNS cache.

Using DNS servers according to domains

The use of DNS (Domain Name System) servers to resolve domain information provided in URLs into IP addresses when requests for web access are processed on Web Gateway can be configured according to the domains of the requested destinations.

This use of DNS servers is also known as conditional DNS forwarding.

Domains, for example, testnet.webwasher.com, are entered into a list together with the IP address of the DNS server that is used to resolve the URL information. More than one DNS server can be specified this way for a domain.

When a request to a particular destination on the web is sent to Web Gateway, it is forwarded to a DNS server according to this list.

The use of a particular DNS server can be configured dynamically with DHCP (Dynamic Host Configuration Protocol. This is also the default setting after the initial setup of a Web Gateway appliance.

If both configuration with DHCP and conditional DNS forwarding are configured, DHCP takes precedence and conditional DNS forwarding is bypassed.

Note: If a BIND server is configured as a DNS server, the DNS server settings that are stored in a configuration file on Web Gateway will be overwritten. To keep these settings for domain name resolving, you need to enter them manually again.

Configure the use of DNS servers according to domains

To enable the use of DNS servers according to the domains of destinations in the web, configure the Domain Name Service settings in a suitable manner.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure the use of DNS servers for and click Domain Name Service.
3. Configure the settings in the Conditional DNS Forwarder Configuration section as needed.
4. Click Save Changes.

DNS proxy settings

Settings for handling queries to a domain name system server (DNS server).

DNS Settings

Option	Definition
IP protocol version preference	Lets you select the protocol version that is preferred when retrieving IP addresses from a DNS server. • Same as incoming connection — When selected, the protocol version is used that is already in use on the incoming connection. • IP4 — When selected, this protocol version is used. • IP6 — When selected, this protocol version is used. • Use other protocol version as fallback — When selected, the other protocol version is used if using the preferred version resulted in a failure. When this option is selected, you can also configure the following. ◦ Enable simultaneous DNS queries for IPv4 and IPv6 — When selected, DNS queries for IPv4 and IPv6 addresses are sent at the same time. When this option is selected, you can also configure the following. ◦ Time to wait for results with a preferred IP version (IPv4/IPv6) after initiating simultaneous DNS queries — Limits the time (in milliseconds) that elapses until a connection that uses the other protocol version is accepted when no connection could be set up using the preferred version. ◦ Count of IP addresses of the preferred version (IPv4/IPv6) to be used from the DNS query results — Limits the number of IP addresses that are tried under the preferred protocol version for setting up a connection before IP addresses are tried under the other version. The number of retries that can be configured ranges from 1 to 4. A query for retrieving IP addresses from a DNS server can result in multiple IPv4 of IPv6 addresses, Whether an IPv4 or an IPv6 address is used for setting up a connection, depends on what you have configured above. When multiple IP addresses are available within the same address family (IPv4 or IPv6), addresses are sorted according to several parameters. Connection attempts are then made using these addresses in the order in which they are sorted. The parameters for sorting IP addresses are listed in the following. They are applied in the order they are listed. • Precedence of an IP address The precedence of an IP address is calculated based on its address prefix. An IP address with a higher precedence value is tried for connecting before an address with a lower value.

Option	Definition
	• Scope of an IP address An IP address can have different scopes as follows: ◦ Link local ◦ Site or uniquely local ◦ Global The scopes are used for sorting in the order they are listed here. • Connection time (round trip time) Connection history is recorded. So when less time was required for setting up a connection using a particular IP address on a previous occasion, this address is preferred over another IP address that required more time. • Least recently used IP address Connection history is also used to determine when IP addresses were used for the last time. An IP address that was used less recently than another IP address is preferred of this address.
Minimal TTL for DNS cache	Sets a minimum time (in seconds) that must have elapsed before data stored in the DNS cache is deleted.
Maximal TTL for DNS cache	Set a maximum time (in seconds) that must have elapsed before data stored in the DNS cache is deleted.
Flush DNS cache	Flushes the DNS cache.

Data Exchange Layer

You can use the DXL technology to send and receive information to and from web security products that are connected to Web Gateway in a common security architecture.

Data Exchange Layer settings

Settings for using the DXL (Data Exchange Layer) technology to exchange information between different web security products

Note:

You can implement a library rule set that uses DXL messages to exchange file reputation information between Web Gateway and a TIE server.

Implementing this rule set is currently the only way to use DXL messages on Web Gateway. The rule set works without any additional configuration of the Data Exchange Layer settings.

Data Exchange Layer

Option	Definition
Time to wait for replies to DXL service requests	Sets the time (in seconds) that Web Gateway waits for a response after sending a request to DXL service. The default waiting time is 60 seconds.
Subscription Topics	Provides a list of topics that a security product can subscribe to for receiving messages about these topics.

Option	Definition
Services	Provides a list of services that send messages about topics to security products.

The following tables describe entries in the Subscription Topics and Services lists.

Subscription Topics – List entry

Option	Definition
String	Specifies the name of a topic.
Comment	Provides a plain-text comment on a topic.

Services – List entry

Option	Definition
Service	Specifies the name of a service that sends messages about topics.
Comment	Provides a plain-text comment on a service.

Using DXL messages to exchange web security information

You can use the DXL technology to send and receive information to and from web security products that are connected to Web Gateway in a common security architecture.

McAfee® Data Exchange Layer (DXL) is a messaging technology for real-time information exchange. The technology is used to exchange security-related information, for example, file reputation scores between Web Gateway and other web security products that are connected to it.

This kind of information exchange is part of a security architecture that is provided by McAfee and is also known as *Security Connected*.

Scenarios for exchanging web security information

You can exchange information under DXL in two main scenarios: One is publishing a message about a security topic in an event and receiving this message after subscribing for the topic. The other is sending a query for information about a security topic to a service and receiving a response from this service.

The web security products that are connected to each other, including Web Gateway, take the various roles that belong to these scenarios. Products can be publishers and subscribers, they can send queries and also act as services that queries are sent to.

When a publisher sends DXL messages to the subscribers, they send no responses. When a DXL message is sent as a query for security-related information to a service, the service sends a response, providing information about the topic that was specified in the query.

Web Gateway supports the sending of DXL messages in events and as queries to a service. It can also receive DXL messages and act as a service that provides information about a web security topic

Note:

You can implement the Gateway Anti-Malware with TIE library rule set that uses DXL messages to exchange file reputation information between Web Gateway and a TIE server. This is the only way to use DXL messages on Web Gateway.

Configuring settings for the exchange of web security information

When information about web security topics is exchanged on Web Gateway, several settings are involved. These settings include credentials for a McAfee ePO server, as parts of the DXL architecture are managed by this administration product.

Topics and services for information exchange are part of the settings for the proxy functions of Web Gateway.

DXL messages can also be traced for troubleshooting after enabling the relevant option of the Troubleshooting settings.

Best practices - Working with the user-agent header

The user-agent header is a header in a request for web access sent under the HTTP protocol. This header identifies the software program that was used to send the request. You can work with this header to create a rule that performs a particular action on a request that contains this header.

The software used on a client for sending a request can be a browser, a media player, or a similar program. If you find, for example, that requests sent with a particular browser cause issues with user authentication on Web Gateway, you can create a rule that skips authentication for these requests or blocks them.

The rule contains the value of the user-agent header in the criteria for the action that is performed. When a request is processed on Web Gateway, this value is retrieved from the request to see whether it is the one for the software program that causes issues.

If not only one program causes issues or you expect that more will be found, you can also work with a list of user-agents. The value of the user-agent header within a request is then compared to the list entries to see whether it matches any of them.

Finding the user-agent

To create a rule with an action for a request that caused issues due to its user-agent, you must know which user-agent it is. There are several ways to find this out.

- **Access log** — You can inspect the access log that is maintained on Web Gateway. The data that this log records includes the user-agent header of a request by default.
- **Online resources** — You can find information about browsers, media players, and similar programs that run as user-agents on client systems using online resources, for example, performing an online search.

  Websites ae available that support your search for information, for example, by listing and describing the most common user-agents or by identifying the browser that is currently in use on a client.
- **TCP dump** — You can create a TCP dump of the request processing that Web Gateway performs, using the troubleshooting functions on the user interface. For more information about these functions, see the *Troubleshooting* chapter.

  When a TCP dump has been created, you can work with a packet tracing tool, for example, Wireshark, to follow the TCP stream. You can select a GET request sent for web access and inspect the data packets of this request with its headers.

  If you already have some information about the user-agent that causes issues, you can filter the output in Wireshark accordingly. Entering, for example, the following line returns all data packets that contain the text string "Mozilla".

  ```
  http.user_agent matches "Mozilla"
  ```

Note: Most user-agent headers for browsers begin with the text string "Mozilla". This does not necessarily mean that the user-agent is the Mozilla Firefox browser. It could be Firefox or a different browser.

Common user-agent headers

The following list provides information about some user-agent headers for software programs that are often found when TCP dumps created on Web Gateway are inspected.

Codes lines from the Wireshark packet tracing tool showing the relevant information are added for each user-agent header.

- **Firefox** — A user-agent header for a Mozilla Firefox browser contains the text string "Firefox/" followed by the version number.

  ```
  Mozilla/6.0 (Windows NT 6.2; WOW64; rv:16.0.1) Gecko/20121011 Firefox/16.0.1
  ```
- **Internet Explorer** — A user-agent header for a Microsoft Internet Explorer browser contains the text string "MSIE" followed by the version number.

  ```
  Mozilla/5.0 (compatible; MSIE 10.0; Windows NT 6.1; WOW64; Trident/6.0)
  ```
- **Chrome** — A user-agent header for a Google Chrome browser contains the text string "AppleWebKit".

  ```
  Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.22 (KHTML, like Gecko) Chrome/25.0.1364.172 Safari/537.22
  ```

  Do not confuse a header like this with a user-agent header for the Apple iPhone smartphone.
- **Windows Media Player** — A user-agent header for Windows Media Player contains the two text strings shown in this sample code block.

  ```
  Windows-Media-Player/10.0.0.xxxx NSPlayer/10.0.0.xxxx WMFSDK/10.0
  ```

- **iTunes** — A user-agent header for an Apple iTunes media player contains the text string "iTunes/" followed by the version number.
  `Mozilla/6.0 (Windows NT 6.2; WOW64; rv:16.0.1) Gecko/20121011 Firefox/16.0.1`
- **Safari on iPhone** — A user-agent header for an app that runs on an iPhone, for example, the Apple Safari browser, contains the text string "iPhone".
  `Mozilla/6.0 (Windows NT 6.2; WOW64; rv:16.0.1) Gecko/20121011 Firefox/16.0.1`

Sample rule for working with the user-agent header

In a rule that performs an action on a request with a user-agent header for a particular software program, the user-agent is included in the rule criteria. If the rule is to apply for more than one user-agent, you can work with a list of user-agents.

Note: We recommend using a list, even if you are presently interested in a particular user-agent only. Using a list makes it easier to modify the rule when more user-agents must be addressed in the future.

The rule criteria contains a property that is set to the value for the user-agent in the user-agent header when the rule is processed. The rule applies if this value matches one of the entries in a list of user-agents or a single user-agent if you have configured the rule this way.

The list might, for example, contain the entry "MSIE 10" for version 10 of the Microsoft Internet Explorer. If a request includes a user-agent header for this browser, the rule criteria matches, as the string that you entered in the list is also contained in the user-agent header.

The property that is used to retrieve the value for the user-agent from the user-agent header in a request is Header.Request.Get. To use the property for retrieving this value, you configure the string "User-Agent" as a parameter of the property.

The purpose of the sample rule is to let a request skip SSL scanning, It looks as follows.

Name	
Skip SSL Scanner for user-agents in list	
Criteria	Action
Header.Request.Get("User-Agent") matches in list –> User Agent Whitelist	Stop Rule Set

Note: We recommend including still another criteria part in a rule like this. As it is the client that provides the information about the user-agent, the client or a malware program might spoof a trusted user-agent to bypass filtering.

A sample rule that has its criteria extended by another part to protect the rule against user-agent spoofing looks as follows.

Name	
Skip SSL Scanner for user-agents in list	
Criteria	Action
Header.Request.Get("User-Agent") matches in list –> User Agent Whitelist AND URL.Host matches *samplesite.com	Stop Rule Set

In the sample rules, Stop Rule Set is configured as action. To address issues that a user-agent causes with regard to a function of Web Gateway, you insert the rule in the rule set for that function.

For example, if a user-agent causes issues with SSL scanning, insert it at the beginning of the SSL Scanner rule set. If the rule applies, processing of this rule set is stopped, which means that the relevant request skips SSL scanning. The rule can be used in a similar way to skip, for example, user authentication.

If you do not want to let requests skip anything due to issues with user-agents, you can replace the Stop Rule Set action with Block. You can then create a rule set for globally blocking requests (if it does not yet exist in your rule set system) and insert the rule.

Create a rule for working with the user-agent header

Create a rule that performs an action on requests depending on their user-agent headers to address issues caused by the user-agents.

The following procedure assumes that an issue with SSL scanning is caused by a particular user-agent. The rule that is created lets requests with user-agent headers containing this user-agent skip SSL scanning to avoid the issues.

Task

1. Select the rule set for the function that is skipped for requests with the user-agent that causes issues.
   a. Select Policy → Rule Sets.
   b. On the rules tree, select the SSL Scanner rule set.
   c. Click Unlock View on the configuration pane and confirm with Yes.
      The nesting SSL Scanner rule set is accessible for inserting rules.
2. Configure the name of the rule that lets requests skip the rules in the rule set.
   a. Click Add Rule.
      The Add Rule window opens with the Name step selected.
   b. In the Name field, type a name for the rule, for example, `Skip SSL Scanner for user-agents on list.`
3. Configure the property that is used to retrieve the user-agent.
   a. Click Rule Criteria and then Add.
   b. From the drop-down menu, select Advanced Criteria.
      The Add Criteria window opens.
   c. Click Filter, then select Engine → Header and from the filtered list of properties select Header.Request.Get.
   d. Click Parameters at the property.
   e. In the window that opens, make sure that Parameter value is selected and type User-Agent, then click OK to close the window.
4. Configure the operator and the list to compare the property value with.
   a. Leave the Matches in list operator that is suggested.
   b. From the lists under Compare with, select User Agent Whitelist.
      Note: The list is initially empty and you must insert an entry for the user-agent that causes issues.
   c. Click OK.
      The Add Criteria window closes and the complete criteria appears in the Add Rule window.
5. Configure the rule action.
   a. Click Action.
   b. From the Action list select Stop Rule Set
6. Complete the configuration.
   a. Click Finish.
      The Add Rule window closes and the rule appears in the SSL Scanner rule set.
      Note: The SSL Scanner rule set is empty by default, as the rules for the scanning functions are contained in nesting rule sets. If you find that the nesting rule set contains rules that were inserted after the initial setup, move the new rule into first position.
   b. Click Save Changes.

Bypassing for Office 365 and other Microsoft services

Requests sent to Office 365 and other Microsoft services, and responses received from these services, can be configured to bypass filtering to avoid a load increase on Web Gateway.

Bypassing is handled for these requests and responses by rules. A rule set with suitable rules is provided in the default rule set system and in the rule set library.

Office 365 and other Microsoft services

Microsoft offers several cloud-based applications that belong to the Office 365 application suite. These applications rely heavily on HTML5 features to provide an enriched user experience.

Consequently, some of these applications can set up a high number of connections and also several "endless" connections, which might considerably increase the load on a Web Gateway appliance. The increased load can have an impact on the proxy functions of Web Gateway, leading to slow or delayed web access, timeouts, and other issues.

To avoid such issues, you might want to let requests and responses in traffic to and from Office 365 and other Microsoft services bypass filtering on Web Gateway. Many of these requests and responses also use undocumented formats or protocols that are proprietary to Microsoft and cannot be scanned and filtered on Web Gateway.

Rule set for Microsoft services bypassing

The Bypass Microsoft (Office 365) Services rule set contains rules that enable bypassing for requests and responses in traffic to and from Office 365 and other Microsoft services.

IP address and URL lists published by Microsoft are used to recognize the requests that are submitted for accessing these services.

The rule set is placed at the top of the default rule set system.

Using a Domain Name System

The bypassing rule set requires Web Gateway to access a Domain Name System (DNS). In some configurations, for example when next-hop proxies are used, Web Gateway does not normally require DNS access, so this access might not be configured or even be blocked by a rule.

Most of the rules in the rule set, however, rely on evaluating the URL.Destination.IP property to recognize relevant requests. The DNS is then used to resolve the destination IP address of the request that is currently processed.

So, if a DNS is not correctly configured or not configured at all, you might encounter timeouts or slow performance when working with the rule set.

Reverse HTTPS proxy

A reverse HTTPS proxy configuration can prevent clients from uploading unwanted data, such as malware or particular media types, to web servers under the HTTPS protocol.

In this configuration, HTTPS traffic is redirected to an appliance that a proxy is run on. It is inspected and eventually forwarded or blocked, according to the rules implemented on the appliance.

You can configure this in the following ways:

- Set up a Transparent Router.
- Set up a DNS configuration that points directly to the appliance when access to a particular web server is requested.

Redirection to an appliance can also be achieved by configuring proxy-aware connections that rely on the use of CONNECT headers.

However, this method would require an additional network device to assemble these headers for incoming requests. It is therefore not recommended.

In addition to configuring your network, you need to configure the handling of SSL certificates.

Optionally, you can configure additional settings that are not SSL-related to ensure a smooth operation of the reverse HTTPS proxy.

Redirect HTTPS traffic in transparent router mode

In transparent router mode, you can use a port redirect rule (also known as port forwarding rule) to direct HTTPS traffic to the proxy port on an appliance.

You also need to ensure that the redirected requests are treated as SSL-secured communication.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to redirect traffic to and click Proxies .
3. In the Network Setup section, select Transparent router).
   The section with the specific router settings appears.
4. Under Port redirects, click Add.
   The Add Port Redirects window opens.
5. Configure the following settings for a new port redirect rule:
   - Protocol name — HTTP

     Note: This setting covers connections under both the HTTP and HTTPS protocols.
   - Original destination ports — 443

     If the web servers that are the destinations for requests can be reached under the HTTP protocol as well, you can add port 80 here (separated by a comma). This type of traffic is then also directed to the appliance.
   - Destination proxy port — 9090

     This is the default proxy port on an appliance.
6. Click OK.
   The window closes and the new rule appears on the list.
7. Under HTTP proxy port, make sure Enable HTTP proxy is selected and click Add.
   The Add HTTP Proxy Port window opens.
8. Make sure the following is configured:
   - Serve transparent SSL connections — Selected
   - Ports treated as SSL — 443
9. Leave the other settings at their default values and click OK.

   The window closes and the new HTTP proxy port appears on the list.
10. Click Save Changes.

Let the appliance listen to requests redirected by DNS entries

When requests under the HTTPS protocol are redirected to an appliance according to DNS entries, you can configure the proxy on the appliance to listen directly on the appropriate port. You also need to ensure that only SSL-secured connections are served.

Before you begin

If you want to configure the proxy in this way, make sure of the following:

- The host names of the requested web servers are not resolved to the appliance when the appliance does a DNS lookup.

  You can achieve this by entering the IP addresses of the web servers into the /etc/hosts file on the appliance or by using an appropriately configured internal DNS server.
- A rule set that handles content inspection is implemented on the appliance and enabled.

  A suitable rule set is provided in the default rule set system as nested rule set of the SSL Scanner rule set.

When using DNS entries, a port redirect rule cannot be applied because the purpose of such a rule is forwarding requests for other destinations to the appliance. However, due to the DNS entries, the appliance is already the destination.

You also need to ensure that only SSL-secured connections are served.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select an appliance for listening to requests and click Proxies (HTTP(S), FTP, ICAP, and IM).
3. Under HTTP proxy port, make sure Enable HTTP proxy is selected and click Add.

   The Add HTTP Proxy Port window opens.
4. Configure the following settings for a new HTTP proxy port:
   - Listener address — 0.0.0.0:443

     This setting lets the appliance listen to requests for any web servers, regardless of their IP addresses. You can also specify a particular IP address here and restrict the appliance to listening for requests to the server in question.

     If you are running several network interface cards on your appliance, you can specify IP addresses (separated by commas) for as many web servers as there are network interface cards.
   - Serve transparent SSL connections — Selected
   - Ports treated as SSL — *
5. Leave the other settings at their default values and click OK.

   The window closes and the new proxy port appears on the list.

   If a web server should also be accessible under the HTTPS protocol, you need to add another HTTP proxy port with listener address 0.0.0.0:80 or the address of a particular web server.
6. Click Save Changes.

SSL certificates in a reverse HTTPS proxy configuration

A reverse HTTPS proxy configuration is usually set up to protect a limited number of web servers against the upload of unwanted data by clients. You need to import SSL certificates for these servers and add them to the appliance configuration.

In a reverse HTTPS proxy configuration, the appliance communicates in SSL-secured mode with its clients. The SSL certificates that the appliance sends to the clients during the SSL handshake cannot be issued, however, by its SSL Scanner module. Therefore, the appliance uses the original certificates of the web servers that the clients request access to.

You can import these certificates when configuring the settings for the SSL Client Context without CA module.

The appliance uses several methods to find the appropriate certificates for sending to its clients.

Choosing certificates for sending to the clients

To find out which certificate should be sent to a client in a given situation, the appliance scans the list of imported certificates. On this list, certificates are mapped to the host names of the web servers they belong to. The appliance then sends the certificate that is mapped to the name of the host that a client requested access to.

In an explicit proxy setup, the host name would be transmitted and made known to the appliance in the header of the CONNECT request.

In a transparent setup, the appliance uses the following methods to detect the host names:

- If a client sends an SNI extension, the host name can be found in a way that is similar to detecting it in an explicit proxy configuration.
- If client requests are redirected to the appliance according to DNS entries, the host name is known by the IP address that you specified when configuring redirection.

  In this case, you also need to create a rule set with rules that set the URL.Host property to the appropriate value for every IP address the appliance has been configured to listen to. This is to let the appliance know where to forward a request to when it has been filtered and allowed.

- If the transparent setup does not use redirection by DNS entries, the appliance will send a handshake message to the web server that a client requested, extract the common name from the certificate it receives from the web server, and use this common name to detect the appropriate host name.

  This method requires that the appliance and the web server communicate in SSL-secured mode, too. You can configure a setting on the appliance to ensure this mode is used.

Create settings for SSL certificates in a reverse HTTPS proxy configuration

You can create settings for the SSL certificates that are used for web servers in a reverse HTTPS proxy configuration and import the certificates when configuring these settings.

Task

1. Select Policies → Settings.
2. On the settings tree, select Enable SSL Client Context without CA.
3. Click Add above the settings tree.

   The Add Settings window opens.
4. In the Name field, enter a name for the settings you want to add, for example, `Imported web server certificates`.
5. [Optional] In the Comments field, type a plain-text comment on the settings.
6. [Optional] Select the Permissions tab and configure who is allowed to access the settings.
7. In the Define SSL Client Context (Without Certificate Authority) section, configure the settings parameters.
   a. On the toolbar of the inline list Select server certificate by host or IP, click Add.

      The Add Host to Certificate Mapping window opens.

   b. Click Import and use the options of the Import Server Certificate window that opens to import an SSL certificate for a web server.

   c. Configure the other parameters in the Add Host to Certificate Mapping window as needed.

   d. Click OK.

      The window closes and a new entry for mapping an SSL certificate to the host name of a web server appears in the inline list.

   e. Repeat substeps a to d if you want to add more mapping entries to the inline list.

   f. Select or deselect SSL-Scanner functionality applies only to client connection, according to whether the connection to the web server should be SSL-secured or not.

      If you choose to let this connection be unsecured, you need to create a rule that changes the network protocol from HTTPS to HTTP.

   g. Configure the other settings parameters for the SSL client context as needed.

   h. Click OK.

      The Add Settings window closes and the new settings appear on the settings tree.
8. Click OK.

   The window closes and the new settings appear on the settings tree.
9. Click Save Changes.

Results

You can use these settings in the rule for setting the client context that is provided in the SSL Scanner rule set of the default rule set system.

Set the URL.Host property in a reverse HTTPS proxy configuration

When client requests are redirected to the appliance by DNS entries in a reverse HTTPS proxy configuration, you need to set the IP address of a web server as values of the URL.Host property to let the appliance know where to forward requests to.

After filtering a request has led to the result that it is allowed, the appliance uses the URL.Host property that was submitted with the request to forward it to the requested web server.

When requests are redirected according to DNS entries, web servers are known to the appliance by their IP addresses. If the URL.Host property has the IP address of a web server as its value, the appliance forwards the request to the appropriate destination.

Setting the value of a URL.Host property to an IP address can be done by a rule. You need to create such a rule for each web server that the appliance should forward requests to.

These rules can be contained in a rule set of their own.

Create a rule set for setting the URL.Host property

You can create a rule set with rules that set the IP address of a web server as the value of the URL.Host property.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, navigate to the position where you want to insert the rule set.
3. Above the tree, click Add and select Rule Set.
   The Add New Rule Set window opens.
4. Under Name, enter a suitable name for the new rule set, for example, Set value of URL.Host to IP address.
5. Make sure Enable is selected.
6. Under Applies to select Requests and IM.
7. Under Apply this rule set, select Always.
8. [Optional] Under Comment, type a plain-text comment on the rule set.
9. [Optional] Click the Permissions tab and configure who is allowed to access the rule set.
10. Click OK.
    The window closes and the new rule set appears on the rule sets tree.

Create rules for setting the URL.Host property

You can create rules that set the IP address of a web server as the value of the URL.Host property.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select the rule set you have created for the new rules, for example, Set value of URL.Host to IP address.
3. Click Add Rule.
   The Add Rule window opens with the Name step selected.
4. In the Name field, type a name for a new rule, for example, `Set value of URL.Host to 10.141.101.51`.
5. Select Rule Criteria, then If the following criteria is matched, and click Add.
   The Add Criteria window opens.
6. Configure the rule criteria as follows:
   a. From the list of properties in the left column, select URL.Destination.IP.
   b. From the list of operators in the middle column, select equals.
   c. In the operand field under Compare with in the right column, type an IP address.
7. Click OK.
   The window closes and the new criteria appears under Rule Criteria.
8. Click Action, select Continue, and leave the default settings for this action.
9. Click Events, then Add, and from the drop-down menu that appears, select Set Property Value.
   The Add Set Property window opens.
10. Set a property as follows:

    a. Under Set this property, select URL.Host.
    b. Under To concatenation of these strings, click Add.
       The Please Enter a String window opens.
    c. In the Parameter value field, type the host name of the web server that has the IP address you are using in this rule.
    d. Click OK.
       The window closes and the host name appears in the Add Set Property window.

11. Click OK.
    The window closes and the event for setting the *URL.Host* property appears under Events.
12. Click Finish.
    The Add Rule window closes and the new rule appears within the rule set that you have created for the value-setting rules.
13. Click Save Changes.

Complete optional activities for a reverse HTTPS proxy configuration

In addition to configuring the network setup and the SSL certificate handling, you can complete several other activities, which are optional, to ensure a smooth operation of the reverse HTTPS proxy.

- Deactivate proxy loop detection
- Restrict access to appliance ports
- Restrict access to web servers
- Address multiple web servers

Deactivate proxy loop detection

An appliance can detect proxy loops by evaluating the Via header of a client request. We recommend that you deactivate this detection process in a reverse HTTPS proxy configuration.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to deactivate proxy loop detection for and click Proxies (HTTP(S), FTP, ICAP, and IM).
3. In the Advanced Settings section, deselect HTTP(S): Inspect Via header to detect proxy loops.
4. Click Save Changes.

Restrict access to appliance ports

In a reverse HTTPS proxy configuration, access should be restricted to the proxy ports of an appliance. You need to configure the user interface and file server settings accordingly.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to restrict port access for and click User Interface.
3. Under HTTP Connector Port, enter the appliance proxy port (default: 9090).
4. Select File Server.
5. Under HTTP Connector Port, enter the appliance proxy port (default: 9090).
6. Click Save Changes.

Restrict access to web servers

A reverse HTTPS proxy configuration is usually implemented to protect a limited number of web servers against unwanted data uploads from clients. In this configuration, you should allow access to these servers only and block it for others.

After access to others servers has been requested and blocked, we also recommend that you let the appliance close these connections.

To restrict access:

- Create a list of the web servers you want to protect.
- Create a rule set for a blocking rule.
- Create a rule that blocks access to other web servers and closes connections to clients after blocking their requests.

Create a list of protected web servers

You can create a list the web servers that you want to protect in a reverse HTTPS proxy configuration.

Task

1. Select Policy → Lists.
2. Above the lists tree, click Add.

   The Add List window opens.
3. Configure the following settings for the list:
   - Name — List name, for example, `Protected web servers`
   - [Optional] Comment — A plain-text comment on the new list
   - Type — Wildcard Expression
4. [Optional] Click the **Permissions** tab and configure who is allowed to access the list.
5. Click OK.

   The window closes and the new list appears on the lists tree under Custom Lists → WildcardExpression.
6. To fill the list with entries, click Add above the settings pane.

   The Add Wildcard Expression window opens.

   To add multiple entries at once, click Add Multiple.
7. Enter one or more wildcard expressions matching the URLs for the web servers you want to protect. Separate multiple entries by commas.
8. Click OK.

   The window closes and the new entries appear on the list.
9. Click Save Changes.

Create a rule set for a blocking rule

You can create a rule set for the rule that blocks access to web servers in a reverse HTTPS proxy configuration.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, navigate to the position where you want to insert the rule set.
3. Above the tree, click Add and select Rule Set.

   The Add New Rule Set window opens.
4. Under Name, enter a name for the new rule set, for example, `Block web servers in a reverse HTTPS proxy configuration`.
5. Make sure Enable is selected.

6. Under Applies to, select Requests and IM.
7. Under Apply this rule set, select If the following criteria is matched. Then click Add.

   The Add Criteria window opens.
8. Configure the rule set criteria as follows:
   a. From the Property list, select URL.Protocol.
   b. From the Operator list, select equals.
   c. Under Operand, type `https`.
   d. [Optional] Under Comment, type a plain-text comment on the new rule set.
9. [Optional] Click the Permissions tab and configure who is allowed to access the rule set.
10. Click OK.

    The window closes and the new rule set appears on the rule sets tree.

Create a rule to block access to web servers

You can create a rule for blocking access to web servers when these are not on the list of protected servers in a reverse HTTPS proxy configuration.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select the rule set you have created for the blocking rule, for example, Block web servers in a reverse HTTPS proxy configuration.
3. Click Add Rule.

   The Add Rule window opens with the Name step selected.
4. In the Name field, type a name for the rule, for example, `Allow access only to protected web servers`.
5. Select Rule Criteria, then If the following criteria is matched and click Add.

   The Add Criteria window opens.
6. Configure the rule criteria as follows:
   a. From the list of properties in the left column, select URL.Host.
   b. From the list of operators in the middle column, select matches in list.
   c. From the list of operands in the right column, select the web server list you configured, for example, Protected web servers.
7. Click OK.

   The window closes and the new criteria appears under Rule Criteria.
8. Click Action, select Block and leave the default settings for this action.
9. Click Events, then Add and from the drop-down list that appears, select Event.

   The Add Event window opens.
10. Configure an event as follows:
    a. From the Event list, select Enable Proxy Control.
    b. From the Settings list, select Do not keep connection to client persistent.
11. Click OK.

    The window closes and the new event appears under Events.
12. Click Finish.

    The Add Rule window closes and the rule appears within the new rule set that you have created.
13. Click Save Changes.

Address multiple web servers

You can let an appliance forward consecutive requests to different web servers to achieve load balancing and ensure redundancy.

To implement this, you need to:

- Import the Next Hop Proxy rule set from the rule set library
- Create a list of next-hop proxies
- Create next-hop proxy settings
- Create a rule that uses the list and the settings to trigger the Enable Next Hop proxy event when a web server from the list of protected servers is requested.

  The list also uses a list of protected servers. For this list, you can use the one that you created to restrict access to these servers.

Create a list of next-hop proxies

You can create a list of the web servers that are addressed as next-hop proxies when a suitable rule triggers the Enable Next Hop Proxy event.

Task

1. Select Policy → Lists.
2. Above the lists tree, click Add.
   The Add List window opens.
3. Configure the following settings for the list:
   - Name — List name, for example, `Protected web servers as next-hop proxies`
   - [Optional] Comment — Plain-text comment on the new list
   - Type — NextHopProxy
4. [Optional] Click the Permissions tab and configure who is allowed to access the list.
5. Click OK.
   The window closes and the new list appears on the lists tree under Custom Lists → NextHopProxy.
6. To fill the list with entries, click Add above the settings pane.

   The Add Wildcard Expression window opens.

   To add multiple entries at once, click Add Multiple.
7. Enter one or more wildcard expressions matching URLs for the web servers you want to address. Separate multiple entries by commas.
8. Click OK.
   The window closes and the new entries appear on the list.
9. Click Save Changes.

Create next-hop proxy settings

You can create next-hop proxy settings for the rule that triggers the Enable Next Hop Proxy event when a server from the list of protected web servers is requested.

Task

1. Select Policy → Settings.
2. On the settings tree, select Enable Next Hop Proxy and click Add.
   The Add Settings window opens.
3. Configure the following settings parameters:
   - Name — Settings name, for example, `Protected web servers`
   - [Optional] Comment — A plain-text comment on the new settings
4. Under Next Hop Proxy Servers configure the following:

    a. From the List of next hop proxy servers, select the next hop proxy list you created, for example, `Protected web servers as next hop proxies`.
    b. Make sure Round Robin is selected.
    c. Deselect Proxy style requests.
5. Click OK.
   The window closes and the new settings appear on the settings tree.
6. Click Save Changes.

Create a rule for the Enable Next Hop proxy event

You can create a rule that triggers the Enable Next Hop Proxy event when a server from the list of protected web servers is requested.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select the Next Hop Proxy rule set.
   The rules of this rule set appear on the settings pane.
3. Click Add Rule.
   The Add Rule window opens with the Name step selected.
4. In the Name field, type a name for the rule, for example, `Address protected web servers as next-hop proxies`.
5. Select Rule Criteria, then If the following criteria is matched, and click Add.
   The Add Criteria window opens.
6. Configure the rule criteria as follows:
    a. From the list of properties in the left column, select URL.Host.
    b. From the list of operators in the middle column, select does not match in list.
    c. From the list of operands in the right column, select the web server list you configured to restrict access to these servers, for example, Protected web servers.
7. Click OK.
   The window closes and the new criteria appears under Rule Criteria.
8. Click Action, and leave the default Continue.
9. Click Events, then Add and from the drop-down list that appears, select Event.
   The Add Event window opens.
10. Configure an event as follows:
    a. From the Event list, select Enable Next Hop Proxy.
    b. From the Settings list, select the settings you configured for this rule, for example, Protected web servers.
11. Click OK.
    The window closes and the new event appears under Events.
12. Click Finish.
    The Add Rule window closes and the new rule appears within the Next Hop Proxy rule set.
13. Click Save Changes.

Proxy auto-configuration

One or more proxy auto-configuration (PAC) files can be made available on an appliance for web browsers on clients. The browsers can use them to find proxies for accessing particular web pages.

A proxy auto-configuration file usually has *.pac* as its file name extension. There can be several of them on an appliance, for example, a *proxy.pac* and a *webgateway.pac*.

Under the WPAD (Web Proxy Auto-Discovery) protocol, a proxy auto-configuration file must have *wpad.dat* as its file name. Therefore, it can exist on an appliance only once.

Make a .pac file available

You can make a .pac file available for proxy auto-configuration to a web browser on a client.

Task

1. Store the .pac file in the /opt/mwg/files folder on the appliance.
2. Start the browser and navigate to the network configuration settings.
3. In the Connection section, click Settings.
4. Select Automatic proxy configuration URL, then enter the path and file name for the .pac file.

   For example, enter:

   ```
   http://mwgappl.webwasher.com:4711/files/proxy.pac
   ```

   If you want the clients to use a dedicated port for downloading the file, you must first configure this port.

   If no dedicated port is used, clients are directed to the HTTP port for the user interface (the default port number is 4711).
5. Click OK.

Create a rule for downloading a wpad.dat file

To enable the download of a wpad.dat file by a web browser on a client, you need to configure a rule that forwards the download request to the appropriate port on an appliance.

Task

1. On the user interface of the appliance, select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to make the wpad.dat file available on and click Port Forwarding.
3. Under Port Forwarding Rules, click Add.
   The Add AppliancePortForwarding window opens.
4. Configure settings for a port forwarding rule as follows:
   - Source Host — 0.0.0.0
   - Target Port — 80
   - Destination Host — 127.0.0.1
   - Destination Port — <File download port>

     As <File download port>, enter the HTTP port for the user interface of the appliance (default: 4711) or a dedicated port that you have configured.
5. Click OK.
   The window closes and the rule appears in the list.

Configure auto-detection of a wpad host

You can let a web browser use auto-detection to find the appliance as the host where a wpad.dat file is stored.

Task

1. Start the web browser and go to the network configuration settings.
2. In the Connection section, click Settings.
3. Select Auto-detect proxy settings for this network.
4. Click OK.

Central Management

Central Management allows you to administer multiple Web Gateway appliances in your network as nodes in a common configuration.

A configuration of multiple appliances administered through Central Management is also referred to as a *cluster*.

When administering a Central Management cluster, you are dealing mainly with:

- **Nodes** — Appliances run as nodes that are connected to each other sending and receiving data to perform updates, backups, downloads, and other jobs.
- **Node groups** — Nodes are assigned to different types of node groups that allow different ways of transferring data.
- **Scheduled jobs** — Data can be transferred according to time schedules that you configure.

Note: A Central Management cluster is not necessarily a High Availability (HA) cluster with fail-over functions. To provide these functions, you must also configure the Proxy HA mode for the proxy functions of the appliances that are involved.

Nodes in a Central Management cluster

In a Central Management cluster, multiple appliances run as nodes and can be administered from any of these nodes.

The nodes in a Central Management cluster are connected within your network as follows:

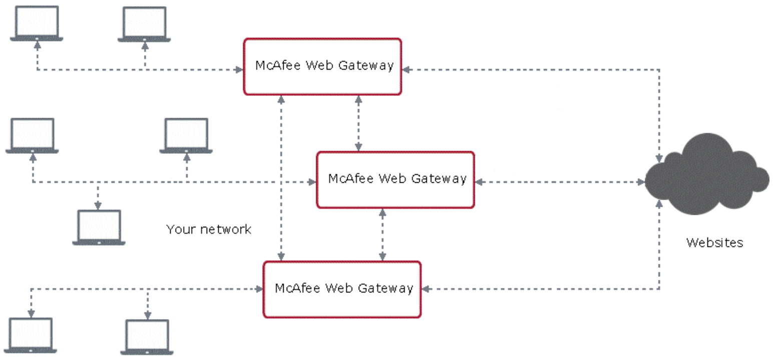
- Each node is connected to client systems of your network that direct their web traffic to it for filtering purposes.
- Nodes are assigned to node groups.
  - Node groups allow common administration activities for the group members, for example, transferring data for updates from one node to another node or several other nodes.

    When configuring appliances as nodes, make sure that they can "see" (connect to) each other. The default port on an appliance that listens to messages from other appliances is 12346.

    Using the *ping* command is a method to verify that appliances can connect. This method is, however, not applicable to all networks.
  - There are different types of node groups that allow different kinds of data transfer between the group members.

The following diagram shows how several appliances run as nodes in a Central Management cluster

Central Management cluster



Types of node groups

The nodes of a Central Management cluster can be assigned to node groups.

Node groups have names and differ with regard to their types. There are the following types of node groups:

- **Runtime group** — A node that is a member of a runtime group can share runtime data with all other nodes in the group.

  Runtime data is data that is created at runtime on an appliance. For example, the amount of time that is left for a user at a given point in time when a quota restriction has been imposed on web usage is runtime data.

  A node can only be a member of one particular runtime group.
- **Update group** — A node that is a member of an update group can share updates with all other nodes in the group.

  A node can only be a member of one particular update group.
- **Network group** — A node that is a member of a network group can immediately connect to all other node in the group.

  A node can be a member of different network groups at the same time.

  When a node is a member of different node groups, for example, of groups A and B, it is possible to transfer data through that node from other nodes in group A that are not members of group B to nodes in group B that are not members of group A.

Secure cluster communication

Communication between the nodes in a Central Management cluster is usually SSL-secured. Certificates and certificate authorities (CAs) with private keys are implemented to enable this kind of communication.

When setting up a cluster you can begin by generating the required security items on one appliance and import them to every other appliance that you include in the cluster later on.

Including appliances as nodes in a cluster

Once you have begun to set up a Central Management cluster by using a particular appliance as your first node, you can include more appliances.

To include an appliance, you can work on the interface of another appliance that is already a cluster node or on the interface of the appliance you want to include.

The following terms are used to denote the two methods.

- **Add** — Adding an appliance to a cluster means that you work on the interface of another appliance that is already a cluster node.
- **Join** — Joining an appliance or letting an appliance join a cluster means that you work on its own interface.

Both methods require that the necessary items for performing secure cluster communication are already implemented on the appliance that is to be included in the cluster.

Web security policy in a cluster

The web security policy that is implemented in a Central Management cluster is always the same for all cluster nodes.

When you set up a cluster, you begin with the web security policy that is implemented on the appliance that is your first node in the cluster. Any other appliance that you include as a node adopts the policy that already exists in the cluster.

If you make changes to this policy on one node, the changes are distributed to all other nodes.

In contrast to the common web security policy on the nodes in a cluster, the system settings for an individual appliance are retained when this appliance is included as a node.

If you change the system settings for an individual appliance, the other appliances in the cluster are not affected.

Scheduled jobs

You can schedule jobs on an appliance that is a node in a Central Management cluster, such as creating a configuration backup or downloading files, for execution at a particular time and date or in regular intervals.

You can also configure the schedule in the interface of the node where you are currently working and execute the job on another node in the cluster.

Overview of the cluster configuration procedure

You can configure the Central Management functions of Web Gateway to run and administer multiple appliances as nodes in a cluster.

Note: By default, appliances are not running as nodes in any Central Management cluster on Web Gateway, so all activities for setting up a cluster must be completed by the administrator.

1. Choose an appliance in your network that serves as your first node.
2. Implement the items required for SSL-secured communication between cluster nodes, such as a certificate and a certificate authority (CA) with private keys, on this appliance.
3. Include at least one more appliance as another node.
    - Implement the items for SSL-secured cluster communication on this appliance.
    - Working on the interface of this or the first appliance, configure at least:
        - Host name or IP address
        - Membership in a network node group

      You can also configure:
        - IP addresses and ports for communication between nodes
        - Membership in runtime and update node groups
        - Scheduled jobs
        - Updates
4. Complete more configuration activities as needed.

    For example:
    - Review the settings for Central Management on any node and adapt them to your requirements.

      Note: If you change the default Advanced Management Settings, make sure that Use unencrypted communication is set in the same way for all nodes.
    - Include more nodes in the cluster.
5. Save your changes.

Add an appliance to a Central Management cluster

You can add a Web Gateway appliance as a node to a Central Management cluster and assign it to a network group.

Before you begin

Make sure you have imported the items required for SSL-secured cluster communication to the appliance that you want to add.

Task

1. On the interface of an appliance that is already a node in the cluster, select Configuration → Appliances.
2. On the appliances toolbar, click Add/Join.
3. Type the host name or the IP address of the appliance that you want to add.
4. From the Network group list, select a network group for the new appliance.
5. Select Add appliance.
6. Click OK.

    The window closes and the added appliance appears on the appliances tree.

Results

The added appliance is now a node in the cluster that already included the appliance you have been working on.

Join an appliance to a Central Management cluster

You can join a Web Gateway appliance as a node to a Central Management cluster and assign it to a network group.

Before you begin

Make sure you have imported the items required for SSL-secured cluster communication to the appliance that you want to join.

Task

1. On the interface of the appliance that you want to join, select Configuration → Appliances.
2. On the appliances toolbar, click Add/Join.
3. Type the host name or the IP address of an appliance that is already a node in the cluster.
4. From the Network group list, select a network group for the appliance that you want to join.
5. Select Join cluster.
6. Click OK.

   The window closes and the appliance appears on the appliances tree.

Results

The appliance is now a node in the cluster.

Generate a cluster CA and its private key

Generate a cluster CA and its private key for use in generating certificates and private keys to ensure secure communication between Web Gateway appliances that are nodes in a Central Management cluster.

A cluster CA and its private key are first generated on a single appliance when you begin to create a cluster. More appliances can then be added to the cluster, after importing this cluster CA and its private key to each of them.

If you have already created a cluster and want to replace the cluster CA and the private key that are in use within this cluster, you can generate these items on any of the appliances that are its nodes.

Importing the cluster CA and its private key to other appliances is not required in this case, as certificates and private keys for all appliances in the cluster are generated when a cluster CA and its private key are generated on any of them.

Task

1. On the user interface of an appliance, select Configuration → Appliances.
2. At the top of the configuration pane, click Cluster CA.
3. In the Cluster CA window that opens, click Generate CA.
4. Use the Generate Cluster CA Certificate window that opens to generate a cluster CA and its private key.
   a. Under Common Name, type a common name for the cluster CA.

      If a cluster CA already exists on the appliance, its name is displayed here, together with the hash value of the name.

      Note:

      The remaining fields in the window are grayed out, as filling them out is not required.

      The validity period for the cluster CA is set to 15 years. The RSA key size is set to 3072.

   b. Click Apply and Export.

      A cluster CA and its private key are generated. The private key enables use of the cluster CA, which then signs the certificate that is generated on the appliance that you are currently working on.

      Together with this certificate, another private key is generated for enabling its use in cluster communication, and both items are stored on the appliance.

      If you are generating the cluster CA and its private key on an appliance that is a node in an already existing cluster, certificates and private keys for all nodes in this cluster are also generated and stored.

      The Generate Cluster CA Certificate window closes and the Save CA Certificate and Private Key window opens.
5. Use the Save CA Certificate and Private Key window to store the cluster CA and its private key.

Note: It is mandatory that you complete this step here, as no opportunity will be provided to store these items later on.

a. Next to Exported CA certificate location, click Browse and browse a location to store the cluster CA there.
b. Next to Exported private key location, click Browse and browse to a location to store the private key there.
c. Under Encryption password, type a password for the private key and hit the Enter key on your keyboard.
   The window closes and the cluster CA is stored with its private key.
d. When a message informs you that both have been stored, click OK to close the message window.

Results

A cluster CA and its private key have been generated and are stored in the places that you selected. The cluster CA is also stored on the appliance, but not its private key.

This private key was required, however, to enable use of the cluster CA when it signed the certificate that was generated on the appliance.

Note: Be aware that this private key will again be required to enable use of the cluster CA when it is imported with the private key to sign a certificate for another appliance that is added as a node to the cluster.

Import a cluster CA and its private key

Import a cluster CA and its private key to a Web Gateway appliance for signing the certificate that is generated to ensure secure communication between this appliance and other appliances that are nodes in a Central Management cluster.

Task

1. On the user interface of an appliance, select Configuration → Appliances.
2. At the top of the configuration pane, click Cluster CA.
3. In the Cluster CA window that opens, click Change CA.
4. Use the Import Certificate Authority for Cluster to import the cluster CA and its private key.
   a. Browse to the location where you stored the cluster CA after generating it, then browse to the location where you stored its private key.
      Note: This cluster CA must be used for signing the certificates of all appliances that are added as nodes to the cluster.
   b. Enter a password for the private key.
   c. Click Import.

   The cluster CA is imported with its private key. The private key enables use of the cluster CA, which then signs the certificate that is created on the appliance.

   Together with this certificate, another private key is generated for enabling its use in secure cluster communication, and both items are stored on the appliance.

   Note: The private key of the Cluster CA is required for enabling its use in completing these activities, but it is not stored on the appliance.
5. Click Save Changes.

Assign a node to network groups

You can assign a node to one or more network groups by entering the group name or names into the appropriate list.

Task

1. On the user interface of an appliance, select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to assign as a node to one or more network groups and click Central Management.
3. To assign the node to a network group other than the default `all` group, click the Add icon on the toolbar of the Group network inline list.

The default group is provided to give you the option of not using different network groups, but having only one network group for *all* nodes.

If you want to have more than one network group, you should delete the `all` group or rename it.

The Add String window opens.

4. Configure a new network group.
   a. In the Name field, type a name for the network group.
   b. [Optional] In the Comment field, type a plain-text comment on the network group.
   c. Click OK.

      The window closes and the new network group appears in the Group network inline list.

      The node is now a member of this network group.

   You can also add multiple network groups at once by clicking the Add multiple icon and working with the Add Strings window that opens.

   In the window, you can enter multiple group names, using a new line for each of them.

   The window provides also options for adding the same comment to all groups or add different comments to individual groups.
5. To include another node in the same network group or groups, select this node on the appliances tree, click Central Management again, and enter the same group name or names in the Group network inline list.
   Repeat this procedure for every node you want to include in the same network group or groups.
6. Click Save Changes.

Assign a node to a runtime group

You can assign a node to a runtime group by typing the group name in the appropriate input field.

Task

1. On the user interface of an appliance, select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to assign as a node to a runtime group and click Central Management.
3. In the Group runtime field of the section This Node Is a Member of the Following Groups, type the name of the runtime group you want to assign the node to.

   When typing the name, be sure to overwrite `all`, which appears in the field as the default name for a runtime group.

   This default name is provided to give you the option of not using different runtime groups, but having only one runtime group for *all* nodes.

   Note: If you delete the default `all` and do not enter a name, you assign the node to a group anyway, one that has an empty string as its name.
4. To include another node in the same runtime group, select this node on the appliances tree, click Central Management again, and type the same name in the Group runtime field.
   Repeat this procedure for every node you want to include in the same runtime group.
5. Click Save Changes.

Assign a node to an update group

You can assign a node to an update group by typing the group name in the appropriate input field.

Task

1. On the user interface of an appliance, select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to assign as a node to an update group and click Central Management.

3. In the the Group update field of the section This Node Is a Member of the Following Groups type the name of the update group you want to assign the node to.

   The procedure is the same as the one for assigning a node to a runtime group.

   Also to include other nodes in the group, proceed in the same way as for a runtime group.
4. Click Save Changes.

Best practice: Configuring Central Management node groups

In a Central Management cluster, nodes are assigned to node groups to enable different methods of communication between them. Node groups can include nodes running in different physical locations.

To ensure the efficient use of node groups in a cluster, the following must apply:

- Appropriate routes are configured in your network to allow communication between nodes.

  If nodes in different locations are protected by firewalls, they must allow use of the port that is configured on each node for communication with other nodes (default port: 12346).
- Time is synchronized. Node communication depends on this when it is determined which node has the most up-to-date configuration.

  We highly recommend that you configure the use of an NTP server on each node for automatic synchronization. This is done as part of configuring the Date and Time settings of the Configuration top-level menu.

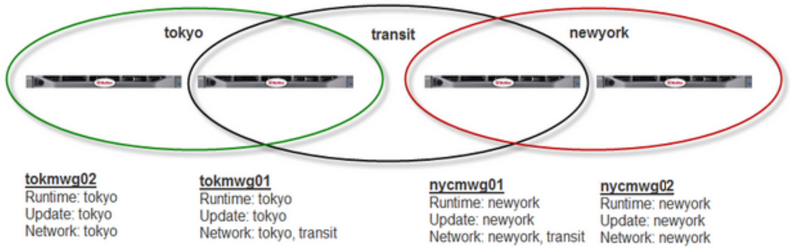
  Note: If you are not using an NTP server for your network, you can configure the default server that is provided by McAfee at *ntp.webwasher.com*.
- The same version and build of Web Gateway is running on all appliances that are configured as nodes.

Small sample configuration

In this sample configuration, there are two different locations (Tokyo and New York) with two nodes each. In both locations, the nodes are assigned to their own runtime, update, and network groups. The group names are *tokyo* and *newyork*, respectively, for all types of groups.

One node in each location is also assigned to the *transit* network group, which is the same for both locations.

The following diagram shows this configuration.



This way, the following is achieved:

- Policy changes that an administrator configures on any node are distributed to all other nodes, due to the existence of a transit group node in each location. This ensures the web security policy remains the same on all nodes.

  The changes are transferred, for example, from the non-transit node in New York to the transit node because both are in one network group. They are then transferred from this transit node to the node in Tokyo, again, because both are in one network group, the *transit* group.

  Finally, the changes are transferred from the Tokyo transit node to the other node in this location.
- Updates of anti-malware and URL filtering information for the respective modules (engines) of Web Gateway are only distributed between nodes in Tokyo and between nodes in New York.

  This allows you to account for differences in the network structure of locations, which is advisable regarding the download of potentially large update files.

  Nodes in one location with, for example, fast connections and LAN links can share these updates, while they are not distributed between these nodes and those in other locations with, for example, slower connections and WAN links.

Note: We generally recommend that you include only nodes of one location in the same update group.

- Runtime data, for example, the quota time consumed by users, is only distributed between nodes in Tokyo and between nodes in New York.

  This makes sense, as probably users in one location will only be directed to the local nodes when requesting web access. So it would not be required for a node in New York to be informed about, for example, the remaining quota time of a user in Tokyo.

  Note:

  If the nodes in one location are assigned to different user groups with regard to their web access, you can also configure these nodes in different runtime groups to avoid an information overhead on any node.

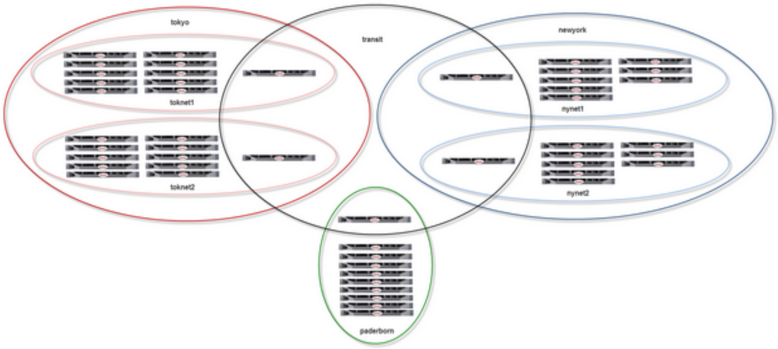
Larger sample configuration

Not more than 10 nodes should be configured for a network group together with a transit node. This means that in larger locations, you need to configure more than one node for the transit network group.

In the following sample configuration, there are 22 nodes in one location (Tokyo), which are split into two network groups (*toknet1* and *toknet2*), both of which include one node that is also a member of the *transit* group.

The 18 nodes in the second location (New York) are configured in the same way, whereas the 9 nodes in the third location (Paderborn) are all in one network group with one node that is also in the *transit* group.

The following diagram shows this configuration.



Regarding runtime and update node groups, there is one of each type for every location.

Policy changes, updates of anti-malware and URL filtering information, as well as sharing of runtime data are handled in the same way as for the smaller sample configuration.

Alternative configuration of nodes with transit group functions

You can configure nodes that perform the functions of nodes in a transit group without formally creating a transit group as a group of its own.

If you have, for example, two groups of nodes, each of which is configured as a network group, you can configure one of the nodes in each group to be a member not only of its own, but also of the other network group.

The nodes that are configured in this way will perform transit group functions. For example, they will distribute policy changes that the administrator applies on one node to all other nodes in the two groups.

Tip:

Best practice: For smaller node groups, configure one node as a member of its own group and of the other group or groups. For larger node groups, configure more than one node with multiple membership for every node group.

Verify the synchronization of nodes

The user interface displays, among other general information, a timestamp for each node in a Central Management Configuration, which allows you to verify whether all nodes are synchronized.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select Appliances (Cluster).

Status and general information about the configuration and its nodes appears on the settings pane.

Under Appliances Information, a list is shown that contains a line with information for every node. The timestamp is the last item in each line.

3. Compare the timestamps for all nodes.
   If they are they same for all nodes, the Central Management configuration is synchronized.

Create a tenant ID

Create a tenant ID, which identifies you as the owner of this instance of Web Gateway and of other McAfee security products that you have purchased.

To create the tenant ID, you work with the options of the user interface and with McAfee® ePolicy Orchestrator® Cloud (McAfee® ePO™ Cloud).

Note: A valid license must have been activated to enable creation of a tenant ID.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, expand Cluster, then click Tenant Info.
   The Tenant Info settings appear in the configuration pane.
3. Click Show Provisioning Key.
   The key is generated based on your customer ID and the current time stamp using SHA256 and base64 encoding. When the key has been generated, it is displayed in the field below the button.
4. Click Copy to copy it for use with McAfee ePO.
5. Create an activation key for the tenant ID, working with McAfee ePO.
   For more information, see the documentation for this product.
6. After creating the activation key with McAfee ePO, copy it and paste it into the lower input field of the Tenant Info settings.
7. Click Set Tenant ID to generate the tenant ID.
   The tenant ID is made known on this appliance and all others that are nodes in the same cluster.
8. Click Save Changes.

Add a scheduled job

You can add a scheduled job to a list of jobs to let them be executed according to a time schedule that you configure.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to add a scheduled job on and click Central Management.
3. On the settings pane, expand the Advanced Scheduled Jobs section.
   The list of scheduled jobs list appears.
4. On the toolbar above the list, click Add.
   The Add Scheduled Job window opens.
5. Configure settings for the scheduled job.
6. Click OK.
   The window closes and the new scheduled job appears on the job list.
7. Click Save Changes.

Update the appliance software in a Central Management cluster

To update the appliance software on Web Gateway appliances that run as nodes in a Central Management cluster, perform the update procedure on one of the nodes and update this node as the last of all.

Tip: We recommend creating a backup of the current configuration before the update.

Task

1. Install a repository with the product version you want to update to on every appliance that is a node in the cluster:
   a. Log on to an appliance from a system console using SSH.
   b. Run this command:

      ```
      yum install yumconf-<version number>-mwg
      ```

      `yumconf-<version number>-mwg` is the repository name. The digits of the version number must be separated by dots.
2. Log on to the interface of one appliance in the cluster for performing the update.
3. Update every appliance in the cluster, except the appliance where you are working:
   a. On the appliances tree, select an appliance.
   b. On the toolbar above the settings pane, click Update appliance software.
4. When all other appliances are running their updates, update the appliance where you are working:
   a. Select the appliance on the appliances tree.
   b. Click Update appliance software.

Results

The appliance software is now updated on all nodes in the cluster.

If the nodes in a cluster are assigned to different network groups, with some nodes being members of more than one group, we recommend that you:

- Perform the update procedure from one of the nodes with multiple membership.
- Update any other node with multiple membership at the end of the procedure.
- Update the node where you are working last.

For example, you have network group A with nodes 1, 2, 3, 4 and network group B with nodes 3, 4, 5, 6, where 3 and 4 are members of both groups:

- Choose node 3 or 4 for performing the update procedure
- Update nodes 1, 2, 5, 6 first, then 4 (if you have chosen 3 to perform the procedure), and finally 3.

Configuring a cluster from the command line interface

You can use the command line interface (CLI) to configure a cluster of Web Gateway appliances.

A usage information command displays a list of the available commands and their parameters. The command uses a path and directory on Web Gateway:

```
/opt/mwg/bin/mwg-coordinator -A usage
```

These activities can be performed:

- Adding a node to a cluster
- Letting an appliance join a cluster
- Deleting a node in a cluster
- Importing a cluster certification authority (CA) and key
- Enabling and disabling message queue logging
- Updating configuration data in several ways, for example, on all nodes in a cluster
- Synchronizing policy configuration data when Web Gateway and McAfee Web Gateway Cloud Service run in a hybrid solution.

Policy configuration

To protect your network against threats arising from the web, Web Gateway enforces a web security policy, which is implemented during the initial setup. You can configure this policy later on to adapt it to your requirements..

When performing this configuration, you are dealing with several fields of web security that your policy should cover. You can also extend the filtering process and make it suitable for cloud use.

- **Web security policy** — A web security policy is made up of rules, which are grouped in rule sets on Web Gateway

  When a situation arises where a rule applies, it performs an action to handle this situation. The situation can be an immediate threat, for example, a virus in a file that a user who works within your network attempts to download. In this case, the rule would block the attempt.

  Other situations might be that a user requests access to an online shopping site during work hours or tries to download a very large streaming file. Both activities could be blocked by suitable rules.

  You can modify all rules in the rule sets on Web Gateway to let them perform the actions that you consider appropriate.

  When a rule performs an action, the user who requested web access can be informed about this action by a message. For example, the user can be told that a request was blocked because a file that was requested for downloading is malware-infected.
- **Cloud use** — The rules of your web security policy are applied to the traffic that is created by the web usage of the users of your organization.

  Unless you configure it differently, however, the rules are only applied to the web usage of those users who access the web from inside your local network. This kind of usage is also known as on-premise use.

  You can, however, enable one or more rule sets for cloud use. This means that the rules in these rule sets are also enforced when users of our organization access the web from outside your local network.

Working with rules

A web security policy is implemented on Web Gateway, which includes various rules. When a situation arises where a rule applies, it performs an action. You can configure this policy by modifying its rules to adapt them to the needs of your organization.

To configure a web security policy, you modify its rules, dealing with them on different levels.

- **Rule sets** — Rules are grouped in rule sets, each of which usually covers a particular field of web security, such as anti-malware filtering, URL filtering, media type filtering, and others.

  A default system of rule sets is implemented on Web Gateway during the initial setup.

  You can enable or disable these rule sets for on-premise and for cloud use, move, copy, and delete them, modify their rules, import rule sets from a built-in or an online library, and create rule sets of your own.

  Two different views are usually provided of a rule set, where you can complete these configuration activities. One is for key activities that are often performed, while the other is for more complex activities.
- **Rules** — You can enable and disable individual rules, move, copy and paste them, delete them, and create rules of your own.

  Individual rules are usually configured in the more complex view.
- **Rule elements** — As default rules are already implemented on Web Gateway, you will usually configure individual elements of rules rather than creating completely new rules. The following are rule elements that you might deal with more often.
  - **Lists of web objects** — Lists of web objects are used within rules, for example, to make sure that access to these objects is not impeded by a particular blocking rule.
  - **Properties** — Every rule contains at least one property. A property in a rule on Web Gateway is usually a property of a web object or an entity that is related to a web object, such as the user who requests access to it.

A property of a web object is, for example, Antimalware.Infected. If this property has the value *true* for a web object that access is requested to, a default rule on Web Gateway, which contains this property, blocks the request and, consequently, denies the user access.

- **Module settings** — Property values are found by modules of the filtering process on Web Gateway. These modules are also known as *filters* or *engines*. You can configure settings for these modules to let them complete their jobs in different ways.

  For example, to find out whether the value of the Antimalware.Infected property is *true* for a requested web object, the object must be scanned for infections. This process is handled by the Anti-Malware module.

  By configuring settings for this module, you can, for example, involve the Gateway Anti-Malware engine in the scanning process combined with additional scanning by Advanced Threat Defense.

For lists and module settings, you can use both rule set views that are provided on Web Gateway. Properties can only be configured in the more complex view.

Access a rule set

Access a rule set on the user interface of Web Gateway to work with its rules and their elements.

Task

1. Select Policy → Rule Sets.
   The Rule Sets tab appears showing the rule sets that are implemented in the navigation pane.
2. Click the rule set that you want to access.
   A view of the rule set appears in the configuration pane.

Results

You can now work with the rules and rule elements of the rule set.

Enable a rule set for cloud use

You can enable a rule set for cloud use.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, navigate to the rule set you want to enable for cloud use and select it.
   A view of the rule set is shown in the configuration pane.
3. Click Enable in cloud to make this rule set available for cloud use.
4. Click Save Changes.

Results

The rules in this rule set are now also used to filter traffic that occurs when cloud users access the web.

Configure a key rule element

Configure a key element of a web security rule.

Note: This task is a sample task that shows how to complete this configuration procedure.

A URL is entered into a URL whitelist. This whitelist is a key element of a rule in the default URL Filtering rule set.

When a request for access to a web object is received on Web Gateway, the rule lets the request skip URL filtering if the URL that is submitted with the request is on the whitelist. This reduces filtering effort and time for requests to access "allowed" web objects.

The URL entry in the sample is *http://www.mcafee.com/**. Due to the wildcard element (*), all requests with URLs that match this entry, for example, *http://www.mcafee.com/us/products/web-gateway.aspx*, will skip URL filtering.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select the URL Filtering rule set.
   Key elements of the rules in this rule set appear in the configuration pane.
3. Under Basic Filtering, click Edit next to URL Whitelist.
   The Edit List window opens.
4. Enter a URL into the whitelist.
   a. Under List content, click the Add icon.
      The Add Wildcard Expression window opens.
   b. In the Wildcard Expression field, type `http://www.mcafee.com/*`.
   c. Click OK.
      The Add Wildcard Expression window closes, and the URL appears in the list of the Edit List window.
5. Click OK.
   The Edit List window closes.
6. Click Save Changes.

Configure a rule element in the complete rules view

The following is a sample task for configuring an element of a web security rule in the complete rules view.

A URL is entered into a URL whitelist. This whitelist is an element of a rule in the default URL Filtering rule set. The steps for accomplishing this are almost the same as for completing this task in the key elements view. Only the way the URL whitelist is accessed is different.

When a request for access to a web object is received on Web Gateway, a rule lets the request skip URL filtering if the URL that is submitted with the request is on the whitelist. This reduces filtering effort and time for requests to access "allowed" web objects.

The URL entry in the sample is *http://www.mcafee.com/**. Due to the wildcard element (*), all requests with URLs that match this entry, for example, *http://www.mcafee.com/us/products/web-gateway.aspx*, will skip URL filtering.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select the URL Filtering rule set.
   Key elements of the rules in this rule set appear in the configuration pane.
3. Click Unlock View to leave the key elements view.

   A message asks you to confirm that you want to leave the key elements view, and also warns you that you cannot return to this view.
4. Click Yes.
   The complete rules view appears.
5. In the rule Allow URLs that match in URL WhiteList, click URL WhiteList.
   The Edit List window opens.
6. Enter a URL into the whitelist.
   a. Under List content, click the Add icon.
      The Add Wildcard Expression window opens.
   b. In the Wildcard Expression field, type, for example, `http://www.mcafee.com/*`.
   c. Click OK.
      The Add Wildcard Expression window closes, and the URL appears in the list of the Edit List window.

7. Click OK.
   The Edit List window closes.
8. Click Save Changes.

Results

For more information about working with rules and rule sets, see the *Rules* chapter.

Import a rule set

You can import a rule set from the library into your rule set system.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, navigate to the position where you want to insert the new rule set.
3. From the Add drop-down list, select Rule Set from Library.
   A window with a list of the library rule sets opens.
4. Select the rule set you want to import, for example, the Gateway Antimalware rule set.
   If conflicts arise when importing this rule set, they are displayed in the window.
   Note: Conflicts arise when a rule set uses configuration objects, such as lists or settings, that already exist in your rule set system.
5. Use one of the following methods to solve conflicts:
   - Click Auto-Solve Conflicts and choose one of the following strategies for all conflicts:
     - Solve by referring to the existing objects — If rules of the imported rule set refer to objects existing in the appliance configuration under the same names, references are made to apply to these existing objects.
     - Solve by copying and renaming to suggested — If rules of the imported rule set refer to objects existing in the appliance configuration under the same names, these objects are also used, but are renamed, so as to avoid conflicts.
   - Click the listed conflicts one after another and solve them individually by choosing either of the two above strategies each time.
6. Click OK.

   The rule set is inserted in the rule sets tree. It is enabled by default.

   List and settings that the rule set needs to perform its filtering job are implemented with the rule set and can be viewed on the lists and settings trees.
7. If necessary, use the blue arrows above the rule sets tree, to move the rule set to where you want it to be.
8. Click Save Changes.

Create a rule set

You can create a rule set and add it to your configuration.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, navigate to the position where you want to insert the new rule set.
3. Click Add above the rule sets tree.
   A drop-down list opens.
4. Select Rule Set.
   The Add New Rule Set window opens.

5. Configure the following general settings for the rule set:
   - Name — Name of the rule set
   - Enable — When selected, the rule set is enabled.
   - Enable in cloud — When selected, the rule set is also enabled for cloud use.
   - [Optional] Comment — Plain-text comment on the rule set
6. In the Applies to section, configure the processing cycles. You can select only one cycle, or any combination of these three:
   - Requests — The rule set is processed when requests from the users of your network are received on the appliance.
   - Responses — The rule set is processed when responses from web servers are received.
   - Embedded objects — The rule set is processed for embedded objects sent with requests and responses.
7. In the Apply this rule set section, configure when the rule set is applied:
   - Always — The rule set is always applied.
   - If the following criteria is matched — The rule set is applied if the criteria configured below is matched.
8. In the Criteria section, click Add.
   The Add Criteria window opens.
9. In the Property area, use the following items to configure a property:
   - Property — List for selecting a property (property types shown in brackets)
   - Search — Opens the Property Search window to let you search for a property.
   - Parameter — Opens the Property Parameters window for adding up to three parameters, see Step 10.

     The icon is grayed out if the property has no parameters.
   - Settings — List for selecting the settings of the module that delivers a value for the property (module names shown in brackets)

     The icon is grayed out if no settings are required for the property and *(not needed)* is added.
     - Add (String, Boolean, or numerical) — Configure it in the Value area. Then click OK.
     - Edit — Opens the Edit Settings window for editing the selected settings.

   If no parameters need to be configured for the property, click OK and continue with Step 11.
10. If you need to add property parameters:
    a. Click Parameter.
       The Property Parameters window opens.
    b. Add as many parameters as needed.
       A parameter can be a:
       - Value (String, Boolean, or numerical) — Configure it in the Value area. Then click OK.
       - Property — Follow the instructions for editing properties, beginning with Step 4.
11. From the Operator list, select an operator.
12. In the Parameter area, add a parameter (also known as operand).
    This can be a:
    - Value (String, Boolean, or numerical) — Configure it in the Value area.
    - Property — Follow the instructions for editing properties, beginning with Step 4.
13. Click OK to close the Add Criteria window.
14. [Optional] Click the Permissions tab and configure who is allowed to access the new rule set.
15. Click OK. to close the Add New Rule Set window.
    The Add New Rule Set window closes and the rule set is inserted into your rule set system.
16. Click Save Changes.

Restrict access to a rule set

To restrict access to a rule set, complete the following procedure.

Task

1. Select Policy → Rule Sets (or Lists or Settings).
2. On the tree structure, navigate to the position where you want to add the new item.
3. Click Add above the tree structure.
   An Add window opens.
4. Complete the steps for adding a new item. Then click the Permissions tab.
   Three modes of access can be configured: *Read and Write*, *Read*, and *No Access*.
5. Click Add under the Read and Write pane.
   The Add Role or User window opens.
6. Select a role or a user (or more than one of each type at once) from the list in the corresponding pane. Or type a wildcard expression as the name of a role or user in the Wildcard field.
7. Add as many entries to the Read and Write list as needed.
   Use the Delete button under the pane to delete entries
8. Fill the Read and No Access panes in the same way.
9. Use the radio buttons under All other roles have to configure access for all roles and users that are not included in one of the lists on the tab.
10. Click OK to close the window.
11. Click Save Changes

Restrict access to configuration items

When creating rule sets, lists, or settings, or working with existing ones, you can restrict access to them.

Task

1. Select Policy → Rule Sets (or Lists or Settings).
2. On the tree structure, navigate to the position where you want to add the new item.
3. Click Add above the tree structure.
   An Add window opens.
4. Complete the steps for adding a new item. Then click the Permissions tab.
   Three modes of access can be configured: *Read and Write*, *Read*, and *No Access*.
5. Click Add under the Read and Write pane.
   The Add Role or User window opens.
6. Select a role or a user (or more than one of each type at once) from the list in the corresponding pane. Or type a wildcard expression as the name of a role or user in the Wildcard field.
7. Add as many entries to the Read and Write list as needed.
   Use the Delete button under the pane to delete entries
8. Fill the Read and No Access panes in the same way.
9. Use the radio buttons under All other roles have to configure access for all roles and users that are not included in one of the lists on the tab.
10. Click OK to close the window.
11. Click Save Changes

Rule sets

Rules are grouped and included in rule sets on the appliance. A rule can never stand on its own, it must be included in a rule set.

A rule set can include just a single rule or several of them. It can also include one or more nested rule sets. If it includes nested rule sets, it can include individual rules on the same level as the nested rule sets.

Rule sets usually include rules that work together to provide a particular function for ensuring web security.

For example, a virus and malware filtering rule set will include a rule that blocks infected rule sets and one or several others that whitelist objects to let them skip the blocking rule and ensure users can access them.

You can modify the implemented rule sets and create rule sets of your own to build functional units in whatever way is suitable for your network.

Rule set criteria

Like rules, rule sets have criteria and are applied if their criteria matches.

Usually, the criteria of a rule set differs from that of its rules. For a rule to apply, both its own criteria and the criteria of its rule set must match.

Rule set cycles

Rule sets are processed, with their rules, in the three cycles of the filtering process.

A rule set can be processed in any combinations of these cycles, for example, only in the request cycle or in both request and response cycles, and also in all three cycles.

The cycles of a rule set are at the same time those of the individual rules it includes. A rule cannot differ with regard to cycles from its rule set.

Nested rule sets

Rule sets can have other rule sets nested within them. A nested rule set has its own criteria.

Regarding cycles, it can only be processed in the cycles of the nesting rule set, but need not be processed in all of them.

This way, a nested rule set can be configured to deal especially with a particular cycle, while another nested rule set deals with a different cycle.

For example, a media type filtering rule set could apply to all cycles, but have nested rule sets that are not processed in all of them.

Media Type Filtering rule set (for requests, responses, and embedded objects)

- Nested rule set *Media Type Upload*(for requests)
- Nested rule set *Media Type Download* for responses and embedded objects)

Default rule set system

Several rule sets that cover important fields of web security are by default implemented in the rule set system of a on the appliance after its initial setup.

The default rule system looks like this (nested rule sets are not shown).

Note: Some of these rule sets are not enabled by default.

Default rule set system

Rule set	Description
Bypass Microsoft (Office 365) Services	Lets requests and responses that are sent to and received from Office 365 and other Microsoft services bypass filtering. **Note:** This rule set is not enabled by default.
HTTPS Scanning	Prepares web traffic that is secured under HTTPS for processing by other filtering functions. **Note:** This rule set is not enabled by default.
Remove Privacy Violation Header	Removes privacy violation headers from requests to prepare them for processing by other filtering functions. **Note:** This rule set is not enabled by default.

Rule set	Description
Global Whitelist	Lets requests for whitelisted URLs or IP addresses skip further filtering.
Common Rules	Provides functions that support the filtering process, such as web caching, progress indication, and opening of archives.
URL Filtering	Controls filtering of individual URLs and URL categories.
Media Type Filtering	Controls filtering of particular types of media.
Gateway AntiMalware	Controls virus and malware filtering using virus signatures and proactive methods.
Dynamic Content Classification	Controls dynamic classification of content. **Note:** This classification is performed in support of URL filtering.

Rule set libraries

The built-in and online libraries provide rule sets for importing into your rule set system.

You can import a rule set, for example, to add a function that is missing in your system or when the default rule sets do not suit your network.

- The built-in rule set library also contains the rule sets that are part of the default rule set system.
- More rule sets are available from an online rule set library. A link to this library is provided in the window of the standard rule set library.

In the built-in rule set library, rule sets are grouped in categories, for example, authentication or URL filtering. The following table shows these categories.

Categories of rule sets in the built-in library

Rule set category	Includes rule sets for ...
Application Control	Filtering applications and individual functions of applications
Authentication	Authenticating users
Coaching/Quota	Imposing quotas and other restrictions on the web access of users
Cloud Services	Implementing single sign-on access to cloud applications
Common Rules	Supporting the filtering process, for example, by web caching, progress indication, or opening of archives
DLP	Implementing data loss prevention
ePO	Enabling use of the ePolicy Orchestrator
Error Handling	Implementing error handling measures
Gateway Anti-Malware	Filtering web objects for infections by viruses and other malware
HTML/Script Filter	Filtering HTML pages and scripts

Rule set category	Includes rule sets for ...
ICAP Client	Running an ICAP client on an appliance
Logging	Logging filtering and other activities
Media Type Filter	Filtering particular types of media
Mobile Security	Filtering mobile traffic
Next Hop Proxy	Using next-hop proxies for data transfer
Privacy	Modifying requests to ensure privacy
SiteAdvisor Enterprise	Using the SiteAdvisor for filtering request
SSL Scanner	Handling SSL-secured web traffic
Troubleshooting	Performing troubleshooting measures
URL Filter	Filtering individual URLs and URL categories
Web Hybrid	Enabling synchronization with McAfee Web Gateway Cloud Service

Rule set views

The user interface provides two kinds of views for a rule set, the *key elements view* and the *complete rules view*.

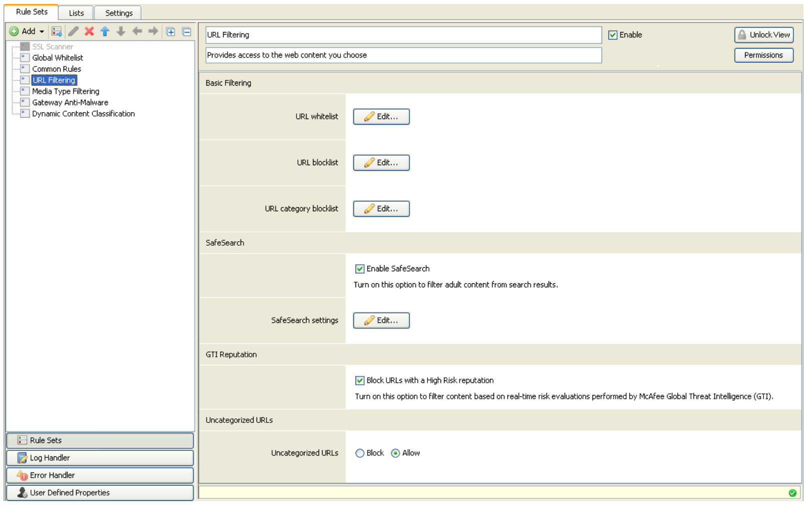
- **Key elements view** — This view shows key elements of the rules in a rule set and allows you to configure them. It appears first when you select a rule set.

  To complete more complex activities regarding individual rules, you can switch to the other view that is provided.
- **Complete rules view** — This view shows the rules that are in a rule set *completely* with all their elements. it allows you to configure all rule elements that are configurable on Web Gateway.

Key elements view

The key elements view shows key elements of the rules in a rule set and allows you to configure them.

Key elements view



URL Filtering

Provides access to the web content you choose

☑ Enable

Basic Filtering

URL whitelist

URL blocklist

URL category blocklist

SafeSearch

☑ Enable SafeSearch

Turn on this option to filter adult content from search results.

SafeSearch settings

GTI Reputation

☑ Block URLs with a High Risk reputation

Turn on this option to filter content based on real-time risk evaluations performed by McAfee Global Threat Intelligence (GTI).

Uncategorized URLs

Uncategorized URLs ○ Block ● Allow

Options of the key elements view

The following table describes the options of the key elements view.

Options of the key elements view

<table>
<tr><th>Option</th><th>Definition</th></tr>
<tr><td>Rule set name field</td><td>Shows the default name of the rule set that key elements are displayed for and lets you edit this name.</td></tr>
<tr><td>Rule set description field</td><td>Shows the default description of the rule set that key elements are displayed for and lets you edit this description.</td></tr>
<tr><td>Enable</td><td>When selected, the rule set with the key elements that you are currently configuring is enabled.</td></tr>
<tr><td>Enable in Cloud</td><td>When selected, the rule set with the key elements that you are currently configuring is enabled for cloud use.</td></tr>
<tr><td>Unlock View</td><td>Leaves the key elements view and displays the corresponding complete rules view.
<b>Note:</b> A confirmation message appears. Be aware that after leaving the key elements view, you cannot return to it unless you discard all changes or re-import the rule set.
On the rule sets tree, icons before the rule set name show which of the two views is currently enabled.
<table>
<tr><td>▣</td><td>Rule set in key elements view</td></tr>
<tr><td>☰</td><td>Rule set in complete rules view</td></tr>
</table>
• To work with nested rule sets, click Unlock View for the nesting rule set.</td></tr>
</table>

Option	Definition
	The nested rule sets appear on the rule sets tree, with the complete rule sets view enabled for each of them. • To display the nested rule sets of the default *Common Rules* rule set, expand this rule set. The complete rules view is already enabled for the last of the nested rule sets, while the others are still displayed in the key elements view. **Note:** You can use the Unlock option of the rule set context menu to leave the key elements view for one or more rule sets at once. 1. Select one rule set or several rule sets at once, then right-click and select Unlock. You can also expand a rule set that includes nested rule sets and select one or more nested rule sets. 2. Confirm that you want to leave the key elements view. The complete rules view is enabled for all selected rule sets
Permissions	Opens a window for configuring who is allowed to access the rule set with the key elements you are currently configuring.
Key elements for a rule set	The key elements vary for each rule set. Key elements for related functions are displayed in a group. Each group is preceded by a group header. For example, on the key elements view for URL filtering, key elements are displayed in the groups Basic Filtering, SafeSearch, and others. These groups contain key elements for basic URL filtering, for additionally using SafeSearch functions in the filtering process, and for other functions.

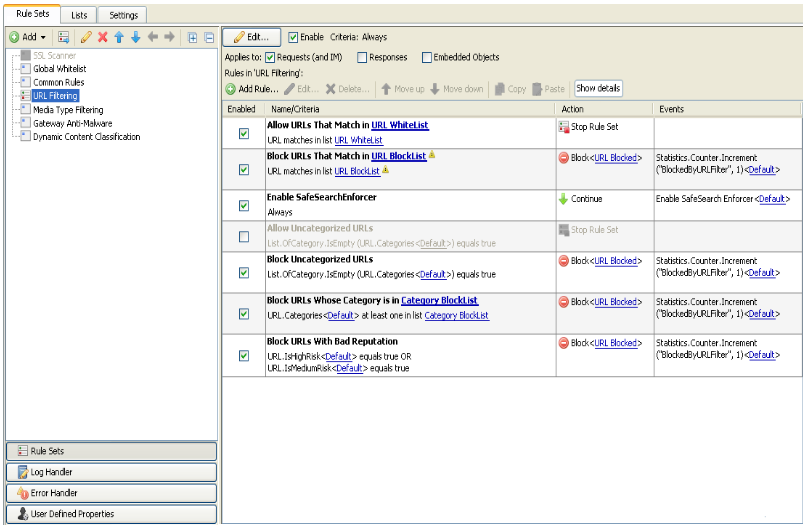
Complete rules view

The complete rules view shows the complete rules that are contained in a rule set. It allows you to work with their elements, including the key elements.

You can edit and delete rules and create rules of your own. You can also edit, delete, and create rule sets. New rule sets can be filled with existing rules, as well as with rules of your own.

You can also import rule sets from a rule set library on Web Gateway and from an online rule set library. You can work with these rule sets and their rules in the same way as with any other rules and rule sets.

Complete rules view



☑ Enable Criteria: Always

Applies to: ☑ Requests (and IM) ☐ Responses ☐ Embedded Objects

Rules in 'URL Filtering':

Enabled	Name/Criteria	Action	Events
☑	**Allow URLs That Match in URL WhiteList** URL matches in list URL WhiteList	Stop Rule Set	
☑	**Block URLs That Match in URL BlockList** ⚠ URL matches in list URL BlockList ⚠	Block<URL Blocked>	Statistics.Counter.Increment ("BlockedByURLFilter", 1)<Default>
☑	**Enable SafeSearchEnforcer** Always	Continue	Enable SafeSearch Enforcer<Default>
☐	**Allow Uncategorized URLs** List.OfCategory.IsEmpty (URL.Categories<Default>) equals true	Stop Rule Set	
☑	**Block Uncategorized URLs** List.OfCategory.IsEmpty (URL.Categories<Default>) equals true	Block<URL Blocked>	Statistics.Counter.Increment ("BlockedByURLFilter", 1)<Default>
☑	**Block URLs Whose Category is in Category BlockList** URL.Categories<Default> at least one in list Category BlockList	Block<URL Blocked>	Statistics.Counter.Increment ("BlockedByURLFilter", 1)<Default>
☑	**Block URLs With Bad Reputation** URL.IsHighRisk<Default> equals true OR URL.IsMediumRisk<Default> equals true	Block<URL Blocked>	Statistics.Counter.Increment ("BlockedByURLFilter", 1)<Default>

Filtering process

A filtering process is performed on the appliance that uses the implemented rules to ensure web security for your network.

The process blocks attempts to access the web that do no comply with your web security policy and allows those that are compliant.

The process is performed in different cycles.

- **Request cycle** — Filters requests for web access submitted by users in your network
- **Response cycle** — Filters responses to requests sent by web servers to your network
- **Embedded object cycle** — Filters embedded objects, for example, files or archives, sent embedded in requests or responses.

Only one filtering cycle is going on at a particular point in time on Web Gateway.

The rule sets of your web security policy can be differently configured with regard to these cycles. A particular rule set can apply to all cycles, or only to one, or to any combination of them.

Properties of filtered objects

The activities that are performed by rules on Web Gateway can be seen as parts of a comprehensive filtering process. This process filters web traffic that is caused by the web usage of the users within your network.

This process filters web traffic. It blocks some objects and lets others pass through, like a tea sieve or strainer that catches the tea leaves and allows the liquid to flow through its perforations.

How does the process tell the tea leaves from the liquid? The tea strainer obviously uses size as a key concept. If something is too big, it cannot pass through.

Similarly, the filtering process on the appliance uses in its rules all kinds of properties that web objects can have or that are related in some way to web objects to make filtering decisions.

A property of a web object checked in the filtering process is, for example, *being virus-infected*. A web object can have the property of being virus-infected, put more simply, it can be virus-infected.

Other examples could be the property of belonging into a particular URL category or the property of having a particular IP address.

The following can then be asked about these and other properties:

- *For a given web object, what value does property p have?*
- *And: If this value is x, what action is required?*

Giving an answer to the second question leads to a rule:

If the value of property p is x, action y is required.

A property is a key element in every rule on the appliance. Understanding the property is essential to understanding the rule.

When you are creating a rule, it is a good idea to begin by thinking about the property you want to use. Using a property of an already existing rule as an example, you might consider something like the following:

I want to filter viruses and other malware. I use the property of being virus-infected and build a rule around it. I let this rule require a blocking action to be taken if a given web object has this property.

The rule could look as follows:

If being virus-infected has the value true (for a given web object), block access to this object.

The web object could, for example, be a file that a web server has sent because a user of your network requested it and that is intercepted and filtered on the appliance.

Properties and rules are explained in this section using normal language. However, the format they have on the user interface of the appliance does not differ from this very much.

For example, the above rule about virus infections could appear on the user interface as follows:

Antimalware.Infected equals true –> Block (Default)

where *Antimalware.infected* is the property and *Block* is the action, which is executed in the default way.

The arrow does not appear on the user interface, it is inserted here to show that the blocking action is triggered if a given web object really has the property in question.

Filtering users

Properties can be related to web objects, but also to the users that request them.

For example, a rule could use the property *user groups that user is member of* to block requests sent by users who are not in an allowed group:

If user groups that user is member of (for a given user) are not on the list of allowed groups, block requests sent by this user.

Filtering cycles

The filtering process on the appliance has three cycles: the request cycle, the response cycle, and the embedded objects cycle. Only one of them can go on at a given moment.

The request cycle is performed for filtering requests that users of your network send to the web, the response cycle is for the responses received upon these requests from the web.

When embedded objects are sent with requests or responses, the embedded objects cycle is performed as an additional cycle of processing.

An embedded object could, for example, be a file sent with a request to upload a file and embedded in this file. The filtering process begins with the request cycle, filtering the request and checking the file that is requested for uploading. Then the embedded objects cycle is started for the embedded file.

Similarly, the response cycle and the embedded objects cycle are started one after another for a file that is sent in response from a web server and has another file embedded in it.

For every rule on the appliance, it is specified in which cycle it is processed. However, the cycle is not specified individually for a rule, but for the rule set that contains it.

A rule set can be processed in just one cycle or in a combination of cycles.

Process flow

In the filtering process, the implemented rules are processed one after another, according to the positions they take in their rule sets.

The rule sets themselves are processed in the order of the rule set system, which is shown on the Rule Sets tab of the user interface.

In each of the three cycles, the implemented rule sets are looked up one after another to see which must be processed in this cycle.

When a rule is processed and found to apply, it triggers an action. The action executes a filtering measure, such as blocking a request to access a web object or removing a requested object.

In addition to this, an action has an impact on the filtering process. It can specify that the filtering process must stop completely, or skip some rules and then continue, or simply continue with the next rule.

Processing also stops after all implemented rules have been processed.

Accordingly, the process flow can be as follows:

All rules have been processed for each of the configured cycles and no rule has been found to apply.	–>	Processing stops.
		In the request cycle, the request is allowed to pass through to the appropriate web server.
		In the response cycle, the response sent from the web is forwarded to the appropriate user.
		In the embedded objects cycle, the embedded object is allowed to pass through with the request or response it was sent with.
		Processing begins again when the next request is received.
A rule applies and requires that processing must stop completely.	–>	Processing stops.
		An example of a rule that stops processing completely is a rule with a blocking action.
		If, for example, a request is blocked because the requested URL is on a blocking list, it is no use to process anything else.
		No response is going to be received because the request was blocked and not passed on to the appropriate web server. Filtering an embedded object that might have been sent with the request is also not needed because the request is blocked anyway.
		A message is sent to the user who is affected by the action, for example, to inform this user that the request was blocked and why.
		Processing begins again when the next request is received.
A rule applies and requires that processing must stop for the current rule set.	–>	Processing stops for this rule set.
		The rules that follow the stopping rule in the rule set are skipped.

		An example of a rule that stops the processing of a rule set is a whitelisting rule followed by a blocking rule in the same rule set.
		When a requested web object is found on a whitelist, the request is allowed to pass through without further filtering. Therefore the rule set is not processed any further and the rule that eventually blocks the object is skipped.
		Processing continues with the next rule set.
		The next rule set can contain rules that, for example, block a request, although it was allowed to pass through the preceding rule set.
A rule applies and requires that processing must stop for the current cycle.	–>	Processing stops for this cycle.
		The rules and rule sets that follow the stopping rule in the cycle are skipped.
		An example of a rule that stops the processing of a cycle is a global whitelisting rule.
		When a requested object is found on a global whitelist, the request is allowed to pass through to the appropriate web server. To ensure the request is not blocked eventually by any of the following rules and rule sets, the request cycle is not processed any further.
		Processing continues with the next cycle.
A rule applies and requires that processing continues with the next rule.	–>	Processing continues with the next rule.
		This can be the next rule in the current rule set or the first rule in the next rule set or cycle.
		An example of a rule that lets the filtering process continue unimpeded is a statistics rule.
		This rule just counts requests by increasing a counter and does otherwise nothing.

Working with Web Gateway using a browser without Java support

You can work with the interface of Web Gateway through a browser that requires no Java support.

Product behavior is mainly the same as when using other methods for working with Web Gateway, with a few differences.

These differences are mostly related to file handling with the purpose of moving files between Web Gateway and your local file system.

Uploading and downloading files to and from Web Gateway, exporting and importing lists or rule sets from and to files, and some other activities require that you work with another dialog window to complete them.

Working with an additional window for file handling

Assume you want to download a file from Web Gateway to your local file system. Two windows are involved in the process:

- The usual Web Gateway window for file downloads
- A second window file handling activities

You browse for and select a file in the first window and execute the download by clicking the appropriate button in the second.

A third window is involved when you upload a file from your local system to Web Gateway.

Options of the additional window

The additional window for file handling provides the following options.

Additional file handling window

Option	Definition
Download selected	Downloads a file from Web Gateway to your local system. Before performing the download, you select the file in the usual window of the Web Gateway interface.
Upload files	Uploads a file from your local system to Web Gateway. You select the file and perform the upload in a third window, which opens after selecting this option.
Delete selected	Deletes a file within Web Gateway. Before deleting this file, you select it in the usual window of the Web Gateway interface. If the file also exists on your local system, it is not deleted.

Activities to be completed in the additional window

The file handling activities that are completed in the additional window are listed here.

They are grouped according to the tabs and pages of the Web Gateway interface that you begin with to perform an activity.

Activities to be completed in the additional window

Top-level menu	Tab or page	File handling activities
Policy	Rule Sets	• Import a rule set from a file • Export a rule set to a file
Policy	Lists	• Import a list from a file • Export a list to a file • Append list content from a file
Troubleshooting	Rule tracing central	• Import a rule trace from a file

Top-level menu	Tab or page	File handling activities
		• Export a rule trace to a file
Troubleshooting	Files	• Upload a file • Download a file • Delete a file within Web Gateway
Troubleshooting	Log files Rule tracing files Feedback Core files Connection tracing Packet tracing	• Download a file • Delete a file within Web Gateway
Troubleshooting	System tools Network tools	• Export tool output to a file
Troubleshooting	Backup/Restore	• Create a configuration backup • Import and restore a configuration

Number of users working on Web Gateway

When using a browser that is not supported by Java, we recommend limiting the number of simultaneous users (administrators) on a Web Gateway appliance to 6. This is also the default limit.

Tip: Best practice: If you are running multiple appliances as nodes in a Central Management cluster, you can distribute users among the appliances without exceeding the limit for each appliance.

Complete rules

Working with complete rules allows you to configure all their elements. You can also delete rules, copy and move them, and also create rules of your own.

For these activities, you need to work on the complete rules view of the rule set where you want to modify a rule or complete other activities regarding individual rules.

This view shows you the individual rules that are contained in a rule set with all the elements that they include, allowing you, for example, to:

- **Modify a rule** — To modify an existing rule, you configure one or more of its elements.
- **Create a rule** — To create a new rule, you can make up new elements, as they are required for a rule, with optional elements as needed.

  You can also copy an existing rule and create a new rule by modifying its elements.

Rule elements

A web security rule on the appliance has three main elements: *criteria*, *action*, and (optionally) *event*.

1. Criteria

   Determines whether a rule applies.

   Note: Other rule syntaxes use the term *condition* instead of *criteria*.

   If the category of a URL is in list x, ...

   The criteria has three elements: *property*, *operator*, and *operand*

   - Property

     Is related to a web object or a user.

     ... the category of a URL ...

   - Operator

     Links the property to an operand.

     ... is in list ...

   - Operand

     Specifies a value that the property can have.

     ... x (list name), ...

     Note: The operand is also referred to as *parameter* on the user interface.

2. Action

   Is executed if the criteria is matched.

   ... block the URL ...

3. Event

   Is executed if the criteria is matched.

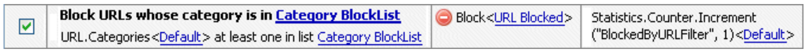
   ... and log this action.

   An event is optional for a rule. A rule can also have more than one event.

Rule format on the user interface

On the user interface, a rule appears in the following format.

Format of a rule on the user interface



☑	**Block URLs whose category is in Category BlockList** URL.Categories<Default> at least one in list Category BlockList	⊖ Block<URL Blocked>	Statistics.Counter.Increment ("BlockedByURLFilter", 1)<Default>

The following table explains the meaning of the rule elements.

Elements of a rule on the user interface

Option	Definition
Enabled	Allows you to enable or disable the rule.
Name	Name of the rule • Block URLs ... — Name text • Category BlockList (in rule name) — List used by the rule **Note:** Clicking on the list name opens the list for editing. • Yellow triangle (next to a list name) — Indicates that the list is initially empty and you need to fill the entries.
Criteria	Criteria of the rule **Note:** The criteria is only visible after clicking the Show details toggle button. • URL.Categories — Property • <Default> — Settings of the module that retrieves a value for the property For example, the *Default* settings that appear here are settings of the URL Filter module. **Note:** Clicking on the settings name opens the settings for editing. The module name is not visible in the rule. It appears, however, in the Edit window for the rule criteria. • at least one in list — Operator • Category BlockList — Operand (also known as parameter) **Note:** Clicking on the list name opens the list for editing. The list name appears both in the rule name and the criteria to let it be available when the criteria is not visible. • Yellow triangle (next to a list name) — Indicates that the list is initially empty.
Action	Action of the rule • Block — Name of the action • <URLBlocked> — Settings of the action **Note:** Clicking on the settings name opens the settings for editing.
Events	One or more events of the rule **Note:** The events are only visible in full after clicking the Show Details toggle button. • Statistics.Counter. Increment — Name of the event

Option	Definition
	• "BlockedByURLFilter, 1" — Parameters of the event • <Default> — Settings of the event **Note:** Clicking on the settings name opens the settings for editing.

Rule representation in the documentation text

When rules are explained in the Web Gateway documentation, different ways of representing them within the documentation text are used.

A rule can be represented in a long or short format, providing more or less explicit information about the structure of a rule. The individual elements of a rule can be marked using different fonts to distinguish them from each other or all appear in the same font.

The long and the short formats can both be combined with different element markup to represent rules as follows:

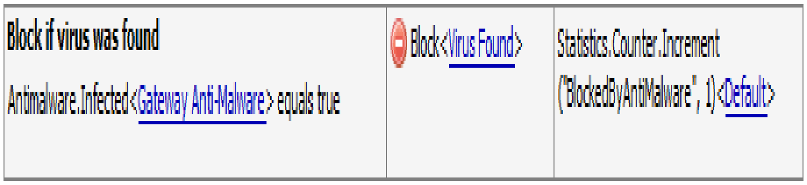
- **Short rule representation** — A rule is represented in a short format with different fonts used for the individual rule elements.
- **Short unified rule representation** — A rule is represented in a short format with the same fonts used for all rule elements.
- **Long rule representation** — A rule is represented in a long format with different fonts used for the individual rule elements.
- **Long unified rule representation** — A rule is represented in a long format with the same fonts used for all rule elements.

All rule representations are followed by explanations of the respective rules in plain text.

Rule representation on the user interface

On the user interface of Web Gateway, a rule looks like this. The three main rule elements (criteria, action, and events) are each shown in a separate column. The rule name appears in bold above the rule criteria.

Rule representation on the user interface



Block if virus was found
Antimalware.Infected<Gateway Anti-Malware> equals true

Block<Virus Found>

Statistics.Counter.Increment
("BlockedByAntiMalware", 1)<Default>

In this sample representation, the rule name and elements are as follows:

- **Name** — Block if virus was found
- **Criteria** — Antimalware.Infected<Gateway Anti-Malware> equals true
- **Action** — Block<Virus Found>
- **Event** — Statistics.Counter.Increment("BlockedByAntiMalware",1)<Default>

The different representation methods used in the documentation text all rely in one way or other on how a rule is represented here.

Short rule representation

The short rule representation shows the main elements of a rule next to each other with the rule name in bold above the criteria. This representation method comes closest to the way that a rule is shown on the user interface.

To distinguish the main rule elements even further than it is done on the user interface, the criteria is shown in italics and the action is preceded by an arrow. The arrow symbolizes the relation between the criteria and the action (if the criteria matches, then the action is performed).

The rule event is always optional. It is also executed if the criteria matches, so it is just added after the action, separated by a dash.

Block if virus was found

Antimalware.Infected<Gateway Anti-Malware> equals true –> Block<Virus Found> – Statistics.Counter.Increment ("BlockedByAntiMalware",1)<Default>

Short unified rule representation

The short *unified* rule representation differs from the short rule representation in that it does not use different fonts to distinguish the name from the rule elements and the rule elements from each other. It rather shows them all in narrow bold font.

Block if virus was found
Antimalware.Infected<Gateway Anti-Malware> equals true – Block<Virus Found> – Statistics.Counter.Increment ("BlockedByAntiMalware",1)<Default>

Long rule representation

The long rule representation shows each rule element in a separate row within a table, preceded by the element name. The rule name appears in red above the table like a section title.

Block if virus was found

Rule element	Definition
Criteria	Antimalware.Infected<Gateway Anti-Malware> equals true
Action	Block<Virus Found>
Event	Statistics.Counter.Increment ("BlockedByAntiMalware", 1)<Default>

Long unified rule representation

The long unified rule representation differs from the long rule representation in that all individual rule elements are marked in narrow bold font.

Block if virus was found

Rule element	Definition
Criteria	Antimalware.Infected<Gateway Anti-Malware> equals true
Action	Block<Virus Found>
Events	Statistics.Counter.Increment ("BlockedByAntiMalware",1)<Default>

Create a rule

Creating a rule includes several activities that are related to the different elements of a rule.

The Add Rule window is provided for creating a rule. It allows you to complete the activities for configuring the rule elements in the order that you prefer.

You can, for example, begin with naming and enabling a rule and then add the criteria, the action, and an event.

Name and enable a rule

Configure name and enabling as general settings for a rule.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select a rule set for the new rule.
3. Click Add Rule above the settings pane.

   The Add Rule window opens with the Name step selected.
4. Configure general settings for the rule:
   a. In the Name field, type a name for the rule.
   b. Select Enable rule to let the rule be processed when its rule set is processed.
   c. [Optional] In the Comment field, type a plain-text comment on the rule.

   Continue with adding the rule elements.

Add the rule criteria

Add the rule criteria to determine when a rule is applied.

Task

1. In the Add Rule window, click Rule Criteria.
2. In the Apply this rule section, select when the rule is applied:
   - Always — The rule is always applied.

     Continue with adding another element, for example, the rule action.
   - If the following criteria is matched — The rule is applied if the configured criteria is matched.

     Continue with the next step.
3. In the Criteria section, click Add and select a criteria group from the drop-down menu.

   The Add Criteria window opens displaying items that are suitable for configuring criteria from the selected group.

   Note: To display items for all criteria, select Advanced criteria.

   The window has three columns:
   - Left column for selecting a property
   - Middle column for selecting an operator
   - Right column for selecting an operand

   The currently selected elements are displayed at the top of each column under Selected property, Selected operator, and Compare with.

   The window supports you in selecting suitable elements by automatically adapting the lists in the other columns after you have selected an item in one column. Then the other columns show only items that are suitable for being configured with the selected item.

   You can begin by selecting an item from the left or right column. Accordingly, steps 4 to 6 could also be completed in a different order.

   Tip: Best practice: If your criteria is to use a list as an operand, begin with selecting this list from the right column.
4. Select a property.
   a. From the list in the left column, select an item or leave the one that is preselected (if there is any).

      Note:

      You can filter the list and add self-configured properties.
   b. [Conditional] If you have selected a property that requires settings, select settings from the Settings drop-down list that is displayed with the property or leave the preconfigured settings.

c. [Conditional] If you have selected a property that requires the setting of parameters, click Parameters below the property name and work with the options in the window that opens to set values for all required parameters.

5. Select an operator from the list in the middle column or leave the one that is preselected (if there is any).
6. Select an operand from the list in the right column or leave the one that is preselected (if there is any). If the list is empty, type a suitable value, for example, a number.

   Note:

   To change the type of operands that are displayed, select a type from the list at the top of the column.

   After selecting an individual operand or a type of operands, the lists in the middle and left columns are adapted to show suitable operators and properties.
7. Click OK to close the Add Criteria window.

   The new criteria appears in the Add Rule window.

   If you want to configure complex criteria, repeat steps 3 to 6 to configure more criteria parts.

   Connect criteria parts by AND or OR, which are then provided as options. For three or more criteria parts, type parentheses to indicate how they are logically connected in the Criteria combination field, which appears then.

   Continue with adding another element, for example, the rule action.

Add the rule action

Add the action that is executed if the rule criteria matches.

Task

1. In the Add Rule window, click Action.
2. From the Action list, select one of the following actions:
   - Continue — Continues with processing the next rule
   - Block — Blocks access to an object and stops processing rules
   - Redirect — Redirects the client that requested access to an object to another object
   - Authenticate — Stops processing the current cycle and sends an authentication request
   - Stop Rule Set — Stops processing the current rule set and continues with the next rule set
   - Stop Cycle — Stops processing the current cycle, but does not block access to the requested object
   - Remove — Removes the requested object and stops processing the current cycle
3. [Conditional] If you have selected an action that requires settings (Block, Redirect, Authenticate), select settings from the Settings list.

   Note: Click Add or Edit before selecting settings to open windows for adding new settings or editing existing settings.
4. If you have created all required rule elements, but do not want to add an event:

   a. [Optional] Click Summary to review what you have configured.

   b. Click Finish.

   The Add Rule window closes and the new rule appears within the rule set you have selected for it.

Add a rule event

Optionally add one or more events that are executed if the rule criteria matches.

Task

1. In the Add Rule window, click Events.
2. In the Events section, click Add and select Events from the drop-down list.

   The Add Event window opens.
3. From the Event list, select an event.

   Note: To filter the list, type a filtering term in the input field above the list.

4. [Conditional] If you have selected an event that requires settings, select settings from the Settings list.

   Note: Click Add or Edit before selecting settings to open windows for adding new settings or for editing existing settings.
5. [Conditional] If you have selected an event that requires the setting of parameters, click Parameters and work with the options in the window that opens to set values for all required parameters.
6. Click OK.

   The Add Event window closes and the new event appears in the Events list.
7. If this is the last of the adding procedures:
   a. [Optional] Click Summary to review what you have configured.
   b. Click Finish.
      The Add Rule window closes and the new rule appears within the rule set you have selected for it.

Working with the Add Criteria window

The window for adding the rule criteria provides several functions to help you with selecting suitable criteria elements.

According to the three elements of the rule criteria, the window is divided into the following columns:

- Left column for selecting a property
- Middle column for selecting an operator
- Right column for selecting an operand

Within a column, properties, operators, and operands are displayed in lists.

For example, you can select the following:

- Left column: *MediaType.EnsuredTypes*
- Middle column: *none in list*
- Right column: *Anti-Malware Media Type Whitelist*

This creates the criteria *MediaType.EnsuredTypes none in list Anti-Malware Media Type Whitelist*. If you add *Block* as the action, you get a rule for blocking access to media of all types that have not been entered in the specified whitelist.

To help you make suitable selections, the window does the following:

- Filters lists according to the filter settings that you provide
- Adapts lists in other columns when you select an item in one column to show only items that are suitable for being configured with the selected item
- Groups lists items in the left and right columns into the categories Recommended, Suggested, and Other if this categorization is possible for the currently displayed items
- Preselects two or three items (one per column) if they can be recommended for being combined with each other

Beginning with the left or right column

You can begin by selecting an item from the left or right column, depending on what you have already in mind about the criteria you are going to add.

For example, if this criteria is to be part of a rule for filtering infected web objects, you might begin by selecting the property *Antimalware.Infected* from the left column and then see what are suitable items to go with it. The result could be: *Antimalware.Infected* (property) *equals* (operator) *true* (operand).

On the other hand, if you want to include the criteria in a rule that prevents the users of your network from accessing drug-selling web sites, you might begin by selecting the URL category list *Drugs* as an operand and then combine it with a suitable operator and property. The result could be: *URL.Categories* (property) *at least one in list* (operator) *Drugs* (operand).

Left column

The list in the left column of the window allows you to select a property. The currently selected property appears at the top of the column in the Selected property field.

You can adapt the list in the following ways:

- Filter the list.
  -

    Using the Filter menu to filter according to:

        - Property type
        - Module (or *engine*) that is called to deliver a value for a property
        - Criteria group, such as Anti-Malware criteria, Media Type criteria, and others

          Note: This part of the menu appears also immediately before the window opens. After selecting a criteria group, the lists in all columns show only items that are suitable for configuring criteria of the selected group.
        - User-defined properties (to show only those properties)
    - Using a filtering term that you type in the input field below the menu
- Add self-configured properties to the list using the Add User-Defined Property button and window.

The list is automatically adapted when you select an operand from the list in the right column. Then it shows only properties that are suitable for being configured with this operand.

After selecting a property, you can configure its settings and parameters if it has any. The Settings and Parameter buttons are then displayed with the property, which open windows for configuring the respective items.

Middle column

The list in the middle column of the window allows you to select an operator. The currently selected operator appears at the top of the column in the Selected operator field.

The list is automatically adapted when you select an item from the list in the left or right column. Then it shows only operators that are suitable for being configured with the selected item.

Right column

The list in the right column of the window allows you to select an operand. The currently selected operand appears at the top of the column in the Compare with field.

An operand can be a single item of different types, a list of items, or another property. Types for single operands include Boolean, String, Number, Category, and others.

You can adapt the list in the following ways:

- Select an operand type (including the list and property types) from the list at the top of the column.

  Only items of this type are then displayed in the main list.
- (Only for lists and properties:) Filter the list using the Filter drop-down menu or the input field below .

If lists are displayed as operands, the Add <list type> and Edit <list type> buttons are provided at the bottom of the column. They open windows for adding and editing lists in the usual way.

The list is automatically adapted when you select a property from the list in the left column. Then it shows only operands that are suitable for being configured with this property.

Complex criteria

The criteria of a rule can be made complex by configuring it with two or more parts.

In complex criteria each of the parts has a property with operator and operand. The parts are linked by AND or OR.

The criteria is matched if a filtered URL belongs to a category that is on any of the two specified category lists (or on both).

If you configure criteria with three or more parts and use both AND and OR between them, you also need to put brackets to indicate how the parts are logically connected. For example, (a AND b) OR c differs in meaning from a AND (b OR c).

When you add a third criteria part on the user interface, lowercase letters appear before the parts and an additional field is inserted at the bottom of the configuration window.

The field displays your criteria parts in short, for example, a AND b OR c. You can then type brackets in the field as needed.

Best practices - Rule configuration

Web Gateway offers many ways to configure rules. However, to achieve an efficient filtering process, some guidelines should be observed.

- **Use rules and rule sets in appropriate filtering cycles** — Some filtering activities are better handled in the request cycle, while others can be left to processing in the response cycle. For example, a rule that blocks requests based on the categories of the submitted URLs should be processed in the request cycle.
- **Use "expensive" properties toward the end of the filtering process** — Some properties require more time and bandwidth to retrieve values for them during the filtering process.

  For example, the *Antimalware.Infected* property is expensive in this sense, so a rule that contains this property should be placed after a rule with, for example, the less expensive *URL.Categories* property. If a request is blocked by the first rule, the effort of processing the second rule is avoided.
- **If possible, do not use more than two properties in the rule criteria —** This guideline does not save processing resources, but makes it easier to understand how rules work.

Using rules and rule sets in appropriate cycles

Rule processing on Web Gateway is performed in different cycles. Perform filtering activities in the cycles that are best suited for them.

The following cycles are available:

- **Request cycle** — For processing requests from clients of Web Gateway

  This cycle works with any data that is available in a request, such as client IP address, URL, user name (if authentication is performed), or browser-related header information.

  If a request is blocked in this cycle, no response cycle is performed, as the request is not forwarded to a web server and therefore no response is received.
- **Response cycle** — For processing responses by web servers, responding to the requests that Web Gateway forwarded to them

  This cycle works with any data that is available in a response, such as the requested data or server-related header information.
- **Embedded objects cycle** — For processing web objects that are embedded in requests or responses

  This cycle is performed when an opener is called in the request or response cycle to allow the filtering modules to look into a web object more deeply. The following two openers are available:

  - **Composite Opener** — The "normal" opener for inspecting files with formats such as .zip, .exe, and others
  - **HTML Opener** — Used very rarely in some advanced configurations

  The embedded objects cycle is performed after the request or response cycle if embedded objects need to be inspected. If there are no embedded objects, the cycle is not performed.

After the request, response, and embedded object cycles are completed, rule sets with logging rules are processed on Web Gateway to let data be written into log files. Processing these rule sets is sometimes also referred to as performing the *logging cycle*.

General guideline for using the request cycle

Let any information that is available in a request be filtered in the request cycle to get any blocked matter out of the way as soon as possible. To understand this guideline, consider cases like the following:

- If filtering based on URL categories is performed in the response cycle, rather than in the request cycle, the requested data might be received from the web server, only to find out that it cannot be passed on to the client because their category is not allowed.
- If filtering based on client IP addresses is performed in the request cycle and a request is blocked, no response cycle is performed, so it is useless to have a rule for filtering this data in the response cycle. If a request is allowed, it is not necessary to filter the data a second time in the response cycle.

Processing cycles and recommended filtering activities

The following table shows the filtering activities that should be performed in the different cycles.

Processing cycles and recommended filtering activities

Request cycle	Response cycle	Embedded objects cycle
Filtering based on whitelists	Filtering based on whitelists	Inspecting the body of a request or response
Filtering based on blocking lists	Filtering based on server-sent headers, such as Content-Length and others	Media type filtering
Filtering based on client-sent headers, such as User-Agent and others	Media type filtering	Anti-malware scanning for embedded objects
User authentication	Anti-malware scanning for downloads	
URL filtering		
Anti-malware scanning for uploads		

As this overview also shows, there are some activities that are only recommended for one cycle, while others, for example, whitelisting or anti-malware scanning, are recommended for two or more cycles.

Using expensive properties at the end of the filtering process

"Expensive" properties require a huge processing effort. Rules with these properties should be placed at the end of the rule set system.

When rules are processed, the modules (or *engines*) on Web Gateway are called to retrieve values for their properties. Some of these modules usually consume more time and bandwidth than others. For example, running the engines for anti-malware scanning usually consumes more resources than letting the URL Filter module retrieve URL category information.

To improve performance, place rule sets containing rules with expensive properties at the end of the rule set system, so that rules with less expensive properties are processed first. If one of these rules already blocks a request or response, the rules with the more expensive properties need not be processed.

Expensiveness of properties

The following table shows the "expensiveness" of some properties that are often used in rules.

Properties marked by an * (asterisk) rely also on external components, for example, on an authentication server, which additionally impacts performance. The table also shows expensiveness for two rule elements that are not properties, but events.

Less expensive	Medium	More expensive
URL	URL.Destination.IP*	Antimalware.Infected
URL.Host	Media.EnsuredTypes	Properties used in DLP (Data Loss Prevention) filtering
URL.Categories*	Properties* used for authenticating users	Using the HTML Opener (enabled by an event)
Client.IP	Using the Composite Opener (enabled by an event)	
Proxy.IP		
Proxy.Port		

Less expensive	Medium	More expensive
System.Hostname		
Properties used to check HTTP header information		

Expensiveness of properties considered for individual rules

The guideline for using properties according to their expensiveness applies not only to a suitable placement of rules and rule sets within the rule set system as a whole, but also to the use of properties in individual rules.

The following rule blocks a request for access to a web server with a particular host name if this request was sent by a client with a particular IP address.

Name		
Block host abcd.com for client with IP address 1.2.3.4		
Criteria	Action	Event
Client.IP equals 1.2.3.4 AND URL.Host matches *abcd.com –>	Block<Default>Continue	

When the rule is processed, the value for *Client.IP* is retrieved first to see where the request comes from. If it does not equal the configured operand, the rule does not apply and processing continues with the next rule. Only if the value for *Client IP* is actually *1.2.3.4*, will the value for *URL.Host* be retrieved as well, to see if the criteria matches completely.

Client.IP is placed first in the criteria because comparing two client IP addresses is less expensive than verifying that a host name matches a wildcard expression.

Using not more than two properties in the criteria of a rule

Using not more than two properties in the criteria of a rule (where possible) makes the rule easier to understand for others and for you when you get back to it after some time.

The following sample rule allows access to destinations within a particular domain and for administrators, but only if they use a particular port for access. There are four different properties in the criteria of this rule for checking the following parameters:

- **Host name of a URL** — Is access requested to the configured domain?
- **User group** — Did authentication show that the user who sent the request is in the user group for administrators?
- **Client IP address range** — Was the request sent from a client with an IP address within the address range that is reserved for administrators?
- **Proxy port** — Is access to the domain requested over the configured port?

The rule looks as follows:

Name/Criteria	Action	Event
Allow only administrators using port 9090 access to test domain		
URL.Host matches "testdomain.com" AND (Authentication.UserGroups does not contain "Administrator" OR –>	Block <Default>	

Client.IP is not in range 192.168.42.0/24 OR Proxy.Port does not equal 9090)		

For a match that lets the rule apply, the first part of the rule criteria requires that a request for access to the test domain is actually submitted.

All other criteria parts are phrased negatively. If the user is *not* an administrator or the client IP address is *not* within the configured range or the proxy port is *not* 9090, then the request is blocked.

In other words, only if a request for access to the test domain is sent by an administrator from a client with an IP address that is within the configured range, using proxy port 9090, does this rule allow access.

The last three criteria parts are included in parentheses, so a combined truth value can be found for them and then checked together with the value for the first criteria part.

The same filtering behavior can be achieved by splitting this rule up into the following three rules.

Name/Criteria	Action	Event
Check whether request is for accessing test domain		
*URL.Host does not match *testdomain.com* –>	Stop Rule Set	
Block access if not over proxy port 9090		
Proxy.Port does not equal 9090 –>	Block <Default>	
Block users who are not administrators based on user name and client IP address		
Authentication.UserGroups does not contain "Administrator" OR Client.IP is not in range 192.168.42.0/24 –>	Block <Default>	

The first of the three rules checks whether a request for access to the test domain is actually submitted. If this is not the case, processing the rules that follow this rule within the same rule set is stopped.

This means the two blocking rules that follow the first rule would not be processed. It is not necessary to process them, however, as there is no attempt made to access the test domain in the first place.

When the two blocking rules are processed, they check the parameters that are involved in deciding whether a request to access the test domain is allowed. The checking is performed in the same way as in the preceding single rule with four properties in its criteria.

The parameters that concern the administrator status of a user are combined within one rule with two properties.

Lists

Lists are used by rules for retrieving information on web objects and users.

There are several types of lists, which differ, for example, with regard to who created them or which type of elements they contain. Accordingly, you work with these lists in different ways.

Lists appear in different places on the user interface, for example, in the criteria of rules and rule sets, on the Lists tab, and within settings.

At the initial setup of the appliance, lists are implemented together with the rule set system.

You can then review the lists of the implemented system, modify and delete them, and also create your own lists.

List types

Web security rules on Web Gateway use several types of lists for retrieving information about web objects and users.

The following are the main list types:

- **Custom lists** — You can modify these lists. They are displayed on the upper branch of the lists tree on the Lists tab, for example, the list of URLs that are exempted from filtering.

  Custom lists can have entries in string, number, category, and other formats. Lists with different formats can require different methods of maintaining them. Some custom lists are initially empty and must have their entries filled by you.

  To the custom lists that Web Gateway provides after the initial setup, you can add lists that you create on your own.
- **System lists** — You cannot modify most of these lists. They are displayed on the lower branch of the lists tree on the Lists tab.

  System lists include category, media type, and application name lists, as well as lists of connectors used for cloud single sign-on. They are updated when an upgrade to a new version of Web Gateway is performed.

  The list of custom connectors is an exception among system lists, as you can change this list by adding connectors to it that you have configured on your own.

  System lists for Data Loss Prevention (DLP), application filtering, and the Dynamic Content Classifier can be included in automatic updates that you schedule.
- **Inline lists** — You can modify these lists, but they do not appear on the Lists tab. They appear inline as part of the settings for a configuration item, for example, a list of HTTP ports as part of the proxy settings.
- **Subscribed lists** — You set up these lists with a name on Web Gateway. They are initially empty and have their content retrieved from a data source that you subscribe to. Subscribed lists are displayed on the lists tree at the end of the custom lists.

  There are two subtypes of subscribed lists:

  - **McAfee-supplied lists** — Content for these lists is retrieved from a McAfee server.

    A number of lists are available on the McAfee server, for example, lists of IP address ranges or media types.
  - **Customer-maintained lists** — Content for these lists is retrieved from a data source that you specify.

    Sources that you can specify are files on web servers running under HTTP, HTTPS, or FTP.

  List content is retrieved from the respective servers. To ensure that newer versions of this content are transferred to your lists on Web Gateway, you can perform updates manually or configure automatic updates.
- **External lists** — These lists reside on external sources under their own names. They have their content transferred to Web Gateway, where they provide the value of a property in a rule.

  External list content is transferred during runtime, which means it is retrieved when the rule with the external list property is processed.

  When the content has been retrieved, it is cached and reused until its date of expiration, which you can configure. After expiration, the transfer is repeated when the rule is processed again.

  Sources that content can be retrieved from include files on web servers running under HTTP, HTTPS, FTP, or LDAP, and in particular types of databases. They also include files that are stored within your local file system.

- **Map type lists** — These lists store pairs of keys and values that are mapped to each other. You can create map type lists and fill list entries on Web Gateway, or retrieve them as subscribed or external lists from other sources.

  Keys and values on map type lists are initially stored in string format, but can be converted into different formats using suitable properties in rules.
- **Common Catalog lists** — These lists can be pushed from a McAfee ePO server to Web Gateway.

  Common Catalog lists can have entries in IP address, domain name, string, or wildcard expression format. They are maintained on the McAfee ePO server.

Access a list

You can access a list on the Lists tab or by clicking a list name in a rule.

Access a list on the Lists tab

To access a list on the Lists tab, you locate it on the lists tree and select the list.

Task

1. Select Policy → Lists
2. On the lists tree, navigate to the branch that contains the list you want to access and click the list name.

   The list entries appear on the settings pane.

Results

You can now work with the list.

Access a list in a rule

To access a list in a rule, you locate the rule on the Rule Sets tab and click the list name.

Task

1. Select Policy → Rule Sets
2. On the rule sets tree, select the rule set that contains the rule with the list you want to access.

   The rules of the rule set appear on the settings pane.
3. Make sure Show details is selected.
4. In the rule with the list you want to access, do one of the following:
   - Click the list name in the rule name if it is contained in this name.
   - Click the list name in the rule criteria.

   An Edit List <Type> window opens, where <*Type*> is the type of the list you are accessing.

Results

You can now work with the list.

Create a list

You can create lists of your own in addition to those that were implemented on the appliance at the initial setup or when you imported a list from the library.

Creating a list includes the following two steps:

- Adding a new list
- Filling the new list with entries

Add a new list

You can add a new list that you fill with entries later.

Task

1. Select Policy → Lists.
2. On the lists tree, navigate to the position where you want to add the list.
3. Click Add on the toolbar.
   The Add List window opens, with the Add List tab selected.
4. Use the following items to configure general settings for the list:
   - Name — Name of the list
   - Comment — [Optional] Plain-text comment on the list
   - Type — List for selecting a list type
5. [Optional] Click the Permissions tab and configure who is allowed to access the list.
6. Click OK.
   The Add List window closes and the new list appears on the lists tree.
7. Click Save Changes.

Results

You can now fill the list with entries.

Fill a list with entries

When you have added a new list on the appliance, you need to fill it with entries.

Task

1. Select Policy → Lists.
2. From the lists tree, select the list you want to add entries to.
3. Click Add on the settings pane.
   The Add <List type> window opens, for example, the Add String window.
4. Add an entry in the way it is done for a particular list type.
5. [Optional] In the Comment field, type a plain-text comment on the list entry.
6. Click OK.
   The Add <List type> window closes and the entry appears in the list.
7. Click Save Changes.

Work with different types of lists

Working with lists is done differently depending on the list type.

For example, if the type is *String*, you can add entries by typing strings in the String field of the Add String window. However, if the type is *MediaType*, you need to select an entry from a media type folder, which is part of a system of folders.

For string and wildcard expression lists, there is the option to add multiple entries at once by clicking Add multiple and typing text for each entry in a new line.

For media type lists, you can select multiple entries or folders at once if you do not want to add them separately.

Add a wildcard expression to a global whitelist for URLs

You can add a wildcard expression to a whitelist used by a global whitelisting rule.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select a rule set that contains rules for global whitelisting, for example Global Whitelist.

   The rules appear on the settings pane.
3. Find the rule that uses a whitelist to exempt requests when they submit URLs for hosts matching the wildcard expressions on the list, for example, URL.Host matches in list Global Whitelist and click on the list name.

   Note: A yellow triangle next to the list name means the list is initially empty and you need to fill the entries.

   The Edit List (Wildcard Expression) window opens.
4. Click Add.

   The Add Wildcard Expression window opens.
5. In the Wildcard expression field, type a wildcard expression.

   To add multiple wildcard expressions at once, click Add multiple and type every wildcard expression in a new line.
6. [Optional] In the Comment field, type a comment on the wildcard expression.
7. Click OK.

   The window closes and the wildcard expression appears on the whitelist.
8. Click Save Changes.

Add a URL category to a blocking list

You can add a URL category to a blocking list to block access to all URLs falling into that category.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select the rule set that contains rules for URL filtering.

   The rules appear on the settings pane.
3. Find the rule that uses a category blocking list, for example, Block URLs whose category is in Category BlockList, and click on the list name.

   Note: A yellow triangle next to the list means that the list is initially empty and you need to fill the entries.

   The Edit List (Category) window opens.
4. Expand the group folder with the category you want block, for example, Purchasing, and select the category, for example, Online Shopping.

   To add multiple categories at once, select multiple categories or one or multiple group folders.
5. Click OK.

   The window closes and the category appears on the blocking list.
6. Click Save Changes.

Add a media type to a media type filter list

You can add a media type to a list for media type filtering.

Task

1. Select Policy → Rule Sets.

2. On the rule sets tree, navigate to a rule set that contains rules for media filtering, for example, the nested Download Media Types rule set of the Media Type Filtering rule set and select it.
   The rules appear on the settings pane.
3. Select the rule Block types from Media Type Blocklist and click on the list name.
   The Edit List (MediaType) window opens.
4. Click Edit.
   An Edit window opens. It displays a list of group folders with media types.
5. Expand the group folder with the media type you want to add, for example, Audio, and select the media type, for example, audio/mp4.
   To add multiple media types at once, select multiple media types or one or multiple group folders.
6. Click OK.
   The window closes and the media type appears on the filter list.
7. Click Save Changes.

External lists

Data can be retrieved from external sources, for example, web servers, and used in rules on the appliance.

This data can be a complete list or a single value. It is generally referred to as *external lists* or *external list data*. Different data types can be used in an external list, such as strings, numbers, IP addresses, and others.

An important feature of external lists is that they are processed dynamically on the appliance. All retrieving and conversion of external list data happens at run time when the data is first used in a rule.

When the data has been retrieved, it is stored in an internal cache for a period of time that you can configure, but not on disk, so it will not persist after a restart of the appliance. Also external lists do not appear on the lists tree of the user interface.

External lists properties

Access to data retrieved from external sources is provided through special properties. The name of an external list property is *ExtLists.<type>*, where *<type>* is the type of elements in the list that is the value of the property. For example, the value of *ExtLists.IntegerList* is a list of integers. Possible list element types include String, Number, Wildcard Expression, and others.

Usually the value of an external list property is a list, but there also external list properties for single values. When an external source delivers more than one value as input for the latter type of property, only the last value is retrieved and stored.

External list data can be filtered, depending on the source type, and converted into a different format, depending on the type of the property used in a given rule.

By configuring parameters for an external list property, you can specify placeholders that are substituted with property parameters at run time. Using these placeholders, you can let the content of an external list depend on criteria such as a user name or user group name.

For logging purposes, you can use the *ExtLists.LastUsedListName* property, which has as its value the name of the settings for the External Lists module that were used last.

External Lists module

To specify which data is to be retrieved from an external source, you need to configure the settings of the *External Lists* module (also known as the External Lists filter or engine), which retrieves the data.

When external data cannot be retrieved successfully, the External Lists module returns an error code, which you can process using Error Handler rules. A separate range of error IDs is available for this purpose.

The External Lists module consumes memory for caching data that it retrieves from external sources. You should take this into account when setting up rules for external list handling.

Sources of external list data

The sources of the content that external lists are filled with can be the following:

- A web service, which is accessed under the HTTP, HTTPS, or FTP protocol
- A file within your local file system

- An LDAP or LDAPS server
- A database:
    - PostgreSQL
    - SQLite3

For performing queries on the databases, the SQL query language is used. However, the particular query format can be different for both database types.

As an SQLite3 database operates file-based, we recommend it for testing, rather than for production environments. However, you might still want to use it if you already have data in a database of this type. Otherwise it is easier to use Web Service or File data sources for retrieving external list content.

Recommended use

Working with the external lists feature is recommended in cases like the following.

You need to handle a large number of lists that are mostly stored in external sources, you are running multiple appliances as nodes in a Central Management configuration, and you need to apply frequent changes to the list data.

Synchronizing all list data on all nodes could then no longer be scalable.

Use of external list data in rules

To handle external list data, you need to configure rules that contain suitable external list properties in their criteria.

Suppose you want to block a request for a web object if its URL has a destination IP address that is within one of the IP address ranges on a list that is stored in an external source.

You can achieve this with the following rule:

Block URLs with IP addresses in forbidden range

URL.Destination.IP is in range ExtLists.IPRangeList(" ", " ", " ")<External Lists> –> Block<URL Blocked>

When the rule is processed, it is checked whether the IP address that is the value of the *URL.Destination.IP* property is within one of the ranges on the list that is the value of *ExtLists.IPRangeList*.

Together with the external list property, the <*External Lists*> settings are specified. These are the settings that the External Lists module uses to retrieve the appropriate data as the value for the external list property.

You need to configure these settings to let the module know where a particular external list can be retrieved from and how the retrieval is performed. For example, if this list is stored in a text file on a web server, you can specify the URL that allows access to the file.

Other information that you can configure as part of these settings includes timeouts and size limits.

The parameters of an external list property are optional. They are empty in this example.

By default, no rules for handling external lists exist on the appliance. If you want to use external list data to restrict web access for the users of your network, you need to set up one or more rules like the above and insert them into a suitable rule set.

Substitution and placeholders

To allow more flexibility in retrieving external list data, placeholders can be used when configuring the settings of the External Lists module, for example, in URLs.

A placeholder is substituted at run time with a value that you provide as a parameter of an external list property.

For example, you want to retrieve data from a web service that delivers lists of media types allowed for individual users. A URL for a particular media type list would then be:

`http://my-web-service.com/ mediatypes?user=` <value>

where <value> is the name of a user.

Configuring separate settings for the External Lists module to cover each user individually would be tiresome, so you can use a placeholder in the following way:

- For the *Web service's URL* parameter in the settings, you specify:

  `http://my-web-service.com/mediatypes?user=${0}`

  where `${0}` is a placeholder for the first of the three parameters of the external list property you are using in a rule.
- For the first parameter of the external list property, you specify the *Authentication.Username* property.

This retrieves a list with the media types that are allowed for an individual user. The user name is the one that this user submitted when required to authenticate after sending a request to access media of a particular type.

You can use the following two types of placeholders:

- *${<n>}* — Placeholder that is substituted with a converted value

  <n> is the position number (0, 1, 2) for a parameter of an external list property. At run time, this placeholder is substituted by the value that you specified when configuring the parameter.

  Before the placeholder is substituted, the value is converted. This process is also known as "escaping". The conversion is performed according to the internal rules of the data source that is involved.

  For example, if the source is a web service, it replaces all characters that are not allowed by *%XX* sequences, as is specified in the corresponding HTTP standard (RFC 2616).
- *$<<n>>* — Placeholder that is substituted with a non-converted value

  As above, but without conversion. This means you need to ensure yourself that the substitution does not lead to unwanted results.

  You can use this type of placeholder when complete URLs, rather than parts of them are to be substituted.

Configure the External Lists module

You can configure settings for the External Lists module to provide the information the module needs to retrieve external list data.

By default, no settings exist for this module on the appliance. You need to add individual settings and configure them for each external list you want to retrieve data from in a rule.

Task

1. Select Policy → Settings.
2. On the settings tree, select External Lists and click Add.

   The Add Settings window opens.
3. In the Name field, type the settings name.
4. [Optional] In the Comment field, type a plain-text comment on the settings.
5. [Optional] Click the Permissions tab and configure who is allowed to access the settings.
6. Configure the other settings parameters as needed.
7. Click OK.

   The window closes and the settings appear under External Lists on the settings tree.
8. Click Save Changes.

Configure general settings for external lists

You can configure settings applying to all external lists that are retrieved for use on the appliance.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure settings for and click External Lists.

   The settings for the external lists appear on the settings pane.
3. Configure these settings as needed.
4. Click Save Changes.

Subscribed lists

Lists for use in web security rules can be filled with content that is retrieved from suitable servers. These lists are known as subscribed lists.

When working with subscribed lists, you only have to configure general settings, such as the list name, yourself. For the list content, for example, IP addresses or URLs, you rely on a server, which can be the McAfee server that is provided for supplying subscribed lists or another server that you specify.

Subscribed lists that retrieve their content from the McAfee server are known as *McAfee-supplied lists*. Lists that retrieve their content from another server are known as *customer-maintained lists*.

After you have created a subscribed list, it appears on the Subscribed Lists branch of the lists tree on the user interface. You can work with a subscribed list in the same way as with other lists on the lists tree.

Note: There is a restriction in size for subscribed lists. A subscribed list must not be larger than 4 MB or contain more than 100,000 entries.

By configuring update schedules or performing updates manually, you ensure that the latest content is made available to the web security rules by a subscribed list.

Retrieving list content from the McAfee server

When the content of a subscribed list is retrieved from the McAfee server that is provided for this purpose, you select the type of content for this list from a catalog.

The content is maintained on the McAfee server. To ensure that McAfee-supplied lists hold the latest content, you perform manual updates on the user interface of your appliance.

Retrieving list content from another server

When the content of a subscribed list is retrieved from a server other than the McAfee server, you specify the URL for the file that holds this content on the server.

The content is maintained on this server. Updates for this kind of subscribed lists are performed according to a schedule that you set up when configuring the list settings.

Create a subscribed list

To create a subscribed list, you configure general list settings and settings for the list content.

Task

1. Select Policy → Lists.
2. Above the lists tree, click the Add icon.

   The Add List window opens.
3. Configure general settings for the list.
   a. In the Name field, type the list name.
   b. From the Type lists, select the list type.
   c. Under Contains, select the type of entry that the list will contain.
   d. [Optional] In the Comments field, type a plain-text comment on the list.
   e. [Optional] Click the Permissions tab and configure who is allowed to access the list.
4. Select List content is managed remotely.
5. Configure settings for the list content.

   - For list content that is retrieved from the McAfee server:

     - Under Source, select McAfee-supplied list.
     -

       Click Choose.

The Choose List Content window opens.

  - Select a content type
  - Click OK to close the window.
- For list content that is retrieved from another server:
  - Under Source, select Customer-maintained list.
  - Click Setup.

    The Setup window opens.
  - Configure settings for the list content.
  - Click OK to close the window.

6. Click OK again.

   The Add List window closes and the list appears on the Subscribed Lists branch of the lists tree.
7. Click Save Changes.

Updating subscribed lists

Updates of subscribed lists content are performed manually or according to a schedule, depending on whether the content is retrieved from the McAfee server that is provided for this purpose or from another server.

For list content that is retrieved from the McAfee server, you must perform updates manually. Each time you perform a manual update, all McAfee-supplied lists are updated together.

The content of McAfee-supplied lists is also updated each time you create a new list of this kind.

For list content that is retrieved from a server other than the McAfee server, updates are performed according to a schedule. Each subscribed list has a schedule of its own. You can set up and modify the schedule when configuring the settings for the list content.

When administering subscribed lists on a node in a Central Management configuration, updates are shared by all other nodes within the update group.

The update group is configured in the section This Node is a Member of the Following Groups of the Central Management settings.

Update subscribed lists supplied by the McAfee server

For subscribed lists that are supplied by the McAfee server, you must perform updates manually.

The content of McAfee-supplied lists is also updated each time you create a new list.

Task

1. Select Configuration → Appliances.
2. On the toolbar above the appliances tree, click Manual Engine Update.

   The content of allMcAfee-supplied lists is updated.

Creating a content file for a customer-maintained list

When a subscribed list has been configured as a customer-maintained list, a content file describing the list structure must be created and stored on the web server that the content for this list is retrieved from.

A content file is created in txt or xml format, depending on whether it describes the structure of a simple or complex customer-maintained list. For simple lists, the content file can be created in both formats, for complex lists in xml format only.

Simple customer-maintained lists can be lists of the following types: Application Name, Category, Dimension, IP, IPRange, MediaType, Number, String, Wildcard Expression.

Complex customer-maintained can be lists of the following types: Certificate Authority, Extended List Element, HostAndCertificate, ICAP Server, NextHopProxy.

Content file for a simple list in txt format

The following is an example of a content file in txt format for a customer-maintained list of the Wildcard Expression type.

```
type=regex "*.txt" "txt file extension" "*.xml" "xml file extension"
```

The example illustrates the following conventions for a content file in txt format.

- The first line in the file specifies the type of the customer-maintained list that the content file is provided for. The format is: `type=`<list type>

  For the list type, one of the following terms must be used: `applcontrol`, `category`, `dimension`, `ip`, `iprange`, `mediatype`, `number`, `string`, `regex`.
- The lines below the first line are for list entries in the customer-maintained list.

  A line contains as many items as a list entry in the customer-maintained list. Each item is included in double quotes.

  An entry in a list of the Wildcard Expression type contains two items. One is the wildcard expression, the other is a comment that describes the wildcard expression.

The following example illustrates some more conventions for a content file.

```
type=string "withoutDescription" "*emptyDescription\"\"\" "" "data with description and more spaces in-between"
"description" "data with spaces* " "description" "Hello\"Michael\" \"Michael!\"" ""
```

- An entry in a list of the String type also contains two items: the string and a comment that describes it. However, the description can be omitted.

  If the description is omitted, the item for it in the content file can also be omitted, which is shown in line 2.
- Alternatively, if the description is omitted, this can be represented by two double quotes with nothing in between, as shown in line 3.

  The line also illustrates the following:

  - Double quotes occurring in a string must be masked by a following backslash.
  - A backslash that does not follow on double quotes represents itself (a backslash).
  - Non-alphanumerical characters, such as the * (asterisk), are allowed at the beginning of a string.

  On the user interface, the list entry specified in line 3 would look as follows: `*emptyDescription\""`
- If multiples spaces are inserted between items in the content file, they are ignored in the list entries of the customer-maintained file.

  On the user interface, the entry specified in line 4 would therefore look as follows: `"data with description and more spaces in-between" "description"`
- Multiple spaces within a string in a content file are also ignored in the list entry of the customer-maintained list.

  So, on the use interface, the entry specified in line 5 would look as follows: `"data with spaces* " "description"`
- Line 6 illustrates several of the already mentioned conventions.

Content file for a simple list in xml format

The following is an example of a content file in xml format for a customer-maintained list of the Wildcard Expression type. The list content is the same as in the first example of the preceding subsection.

```
<content type="regex"> <listEntry> <entry>*.txt</entry> <description>txt file extension</description> </
listEntry> <listEntry> <entry>*.xml</entry> <description>xml file extension</description> </listEntry> </
content>
```

For the content type, the same terms must be used as for a content file in txt format.

Content files for complex lists

Manually creating a content file for a complex customer-maintained list is rather difficult. However, you can use the options of the user interface to export an existing complex list and store it in a file.

In this file, the complex list appears in xml format. If you delete all lines in the file that precede the opening `<content>` tag and follow the closing `</content>` tag, you almost get a content file for that complex list.

Then you only need to modify the opening `<content>` tag to read `<content type="`<file type>`"`, for example, `<content type="nexthopproxy">`.

The terms you can use to specify the file type are: `ca`, `extendedlist`, `icapserver`, `hostandcertificate`, `nexthopproxy`.

Best practice: Working with a McAfee-supplied subscribed list

You can use a subscribed list that is supplied by McAfee in a rule of your web security policy, for example, to let particular traffic bypass SSL scanning.

Web traffic might be sent from the clients of your corporate network to particular destinations, for example, WebEx applications, using SSL-secured connections. When this traffic is received on Web Gateway, you might want to let it skip SSL scanning.

For this purpose, you need a list with the IP address ranges that are used by WebEx. As these addresses change frequently, McAfee supplies an address list, which is updated in intervals, saving you the effort of keeping this list up to date manually.

The list is included in the update schedule that you configure on Web Gateway to make sure that any updates supplied by McAfee are eventually passed on to your Web Gateway appliance or to all the appliances that you are running in a Central Management configuration.

To use this McAfee-supplied list in your web security policy:

- Create an empty list of your own and let this list be filled with WebEx address ranges from the McAfee list
- Set up a rule that uses your list to let requests for accessing WebEx destinations skip SSL scanning

Use a McAfee-supplied subscribed list in a rule

To use a McAfee-supplied subscribed list in a rule that performs a suitable action on web traffic to particular destinations, configure the list as part of the rule criteria.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select the SSL Scanner default rule set and click Unlock View to view the complete rules view.
3. Make sure the rule set is enabled and select the nested Handle CONNECT Call rule set.
4. Click Add Rule and in the window that opens configure a rule as follows.
   a. Under Name, type the rule name, for example, `Bypass SSL scanning for WebEx destinations`.
   b. Under Criteria, configure the following:
      - Property: URL.Destination.IP
      - Operator: is in range list
      - Compare with (operand): WebEx IP Ranges Subscribed Lists

   c. Under Action, select Stop Rule Set.
   d. Click Finish.
      The window closes and the rule appears at the end of the rules in the rule set.
   e. Move the rule into first position.
5. Click Save Changes.

Results

Requests for destinations with the IP addresses on the WebEx list will now bypass SSL scanning on Web Gateway.

Create a McAfee-supplied subscribed list with IP address ranges

To create a subscribed list with IP address ranges for WebEx applications that is maintained by McAfee, create a list of your own and let its content be provided by a McAfee-supplied list.

Task

1. Select Policy → Lists.
2. Above the lists tree, click the Add icon.
3. In the Add List window, configure a list as follows.
   a. Configure general settings for the list:
      - Name: `WebEx Subscribed List` or any other suitable name
      - Type: IPRange
   b. Select List content is managed remotely.
   c. Select McAfee-supplied and click Choose.
   d. In the Choose List Content window, select the list named WebEx IP Ranges.
4. Click OK in both windows.
   The list appears on the Subscribed Lists branch of the lists tree
5. Click Save Changes.

Results

You can now use the list that you have created in a suitable rule.

Map Type lists

Map Type lists, also known as maps, can be used to store pairs of keys and values mapped to each other. Both the keys and their values are of the string type.

Lookup operations can be performed on existing maps, for example, to find out whether a particular key exists in a map or what value is mapped to a key.

Other operations include setting and deleting values for a particular key or converting a complete map into a single string.

You can create and fill Map Type lists on the user interface of Web Gateway or retrieve them from a remote location using the external lists and subscribed lists functions.

If you want to work with other data types for your maps, for example, numbers or IP addresses, you can convert them using properties such as *Number.ToString* or *IP.ToString*.

Create a Map Type list

To create a Map Type list, add a list of this type and fill it with pairs of keys and values.

Task

1. Select Policy → Lists.
2. Above the lists tree, click the Add icon.
   The Add List window opens.
3. Add a MapType list.
   a. In the Name field, type a list name.
   b. From the Type list, select MapType.
   c. Click OK.

   The window closes and the new Map Type list appears on the lists tree under Custom Lists → MapType.

   The settings pane is ready for filling the list with entries.
4. Click the Add icon on the settings pane.
   The Add Map Type window opens.
5. For each pair of entries, you need to fill the list as follows.
   a. In the key field, type a key name.
   b. In the value field, type a value.

c. Click OK.

The window closes and the pair of entries appears in the first row on the settings pane.

6. Click Save Changes.

Using properties to work with Map Type lists

There are several properties for working with Map Type lists. Using these properties in rule criteria, you can retrieve information about a Map Type list, modify a list, create a new list, and also convert a list into a string.

To retrieve information about a Map Type list (map), you can:

- Retrieve a map that you specify a name for
- See whether a particular key exists within a map
- Retrieve the number of key-value pairs in a map
- Retrieve a list of the keys in a map
- Retrieve the value for a given key in a map

The following properties are used to perform these activities.

Property	Description
Map.ByName	Provides a map with the name that you specified.
Map.HasKey	Is true if the specified map includes the specified key.
Map.Size	Provides the number of key-value pairs in a map.
Map.GetKeys	Provides a list of the keys in a map.
Map.GetStringValue	Provides the string that is the value of the specified key in the specified map.

You can, for example, use the *Map.GetStringValue* property in the criteria of a rule to see whether a key in a list has a particular value. The key could be a user name and the value a string that serves as a token for authentication.

The criteria would then be configured as follows:

Map.GetStringValue (testmap, "sampleuser") equals "sampletoken"

If the *sampleuser* key has *sampletoken* as its value, the criteria matches, and the rule executes a particular action, for example, *Continue*.

When a map is modified, the modification is applied to a copy of the original map, while the original map itself remains unmodified. To modify a map in this way, you can:

- Set a key to a particular value
- Delete a key

The following properties are used to perform these activities.

Property	Description
Map.SetStringValue	Provides a map in which the specified value is set for the specified key.
Map.DeleteKey	Provides a map in which the specified key is deleted.

To create a new map or convert a map into a string, the following properties are used.

Property	Description
Map.CreateStringMap	Provides a new map, which is still empty.
Map.ToString	Provides a map converted into a string.

Retrieving map data from external and subscribed lists

Data for Map Type lists (maps) can be retrieved from external and subscribed lists.

External lists

For retrieving map data from an external list, the *ExtLists.StringMap* property is provided, which you can use in the criteria of a suitable rule. The value of this property is a list of maps that have an external list as their source.

For example, to find out whether a particular key is contained in a list that is retrieved from an external source, you can configure the following rule criteria:

Map.GetKeys(ExtLists.StringMap(" ", " ", " ")<External Lists>) contains "samplekeyname"

To specify the external list and where it can be retrieved from, you need to configure the settings of the External Lists module, which is the module that performs the retrieval. In the above criteria, these settings appear under the name of *External Lists*.

External list data can be retrieved from a web service, a file, a PostgreSQL or SQLite3 database, or using LDAP. For these source types, the following must be observed when configuring the retrieval of data for a map:

- Web service or file

  The type of data that is retrieved from a web service or a file must be *Plain Text*.

  To locate the data, a regular expression is used that includes two parts. The first part is for the keys, the second for the values.
- Databases

  The database query for retrieving the data must return two columns. The first column delivers the keys, the second column delivers the values.
- LDAP

  To retrieve the data, a first and a second attribute are configured within the LDAP settings. The first attribute delivers the keys, the second attribute delivers the values.

Subscribed lists

Entries in subscribed lists that map data is retrieved from must have the following format.

```
<listEntry> <complexEntry defaultRights="2"> <configurationProperties> <configurationProperty key="key"
type="com.scur.type.string" value="key"/> <configurationProperty key="value" type="com.scur.type.string"
value="value"/> </configurationProperties> </complexEntry> <description></description> <l/istEntry>
```

Within the *listEntry* element, there's a *complexEntry* embedded. This allows the Subscribed Lists module to process the format.

Common Catalog

The Common Catalog provides lists that can be pushed from a McAfee ePO server to a Web Gateway appliance.

The following types of lists can be pushed: IP address, domain name, string, wildcard expression.

Note: Do not modify the content of the lists on the Web Gateway appliance, because this content is updated in intervals on the McAfee ePO server. These updates will overwrite any changes that you might have applied.

A REST (Representational State Transfer) interface runs internally on both systems to enable the list transfer. A McAfee ePO extension for Web Gateway must also be running on the McAfee ePO server.

This extension includes a help extension, which provides online Help for handling the extension. An extension package is provided on the user interface of Web Gateway under the ePolicy Orchestrator system settings.

To let requests from the McAfee ePO server bypass filtering by web security rules on Web Gateway, you need to import a suitable rule set from the library, place it at the topmost position of the rule sets tree, and enable it.

In addition to this, you need to set up a McAfee ePO user account, as there must be an instance on the appliance that is allowed to handle the list transfer. For setting up this account, the ePolicy Orchestrator system settings are used.

The user of the McAfee ePO account must also appear as an administrator with an account among the internal Web Gateway administrator accounts.

After lists from the Common Catalog have been pushed to Web Gateway, they appear on the Lists tab of its user interface. A prefix in the list name indicates that a McAfee ePO server is the source of a list.

You can use these lists to configure rules like any other lists on the Lists tab.

Prepare the use of Common Catalog lists

To prepare the use of Common Catalog lists that are pushed from a McAfee ePOserver to a Web Gateway appliance, complete the following high-level steps.

Task

1. Set up an account for a McAfee ePO user on Web Gateway.
2. Set up an administrator account with the same user name and password on Web Gateway.
3. Enable use of the REST interface on Web Gateway.
4. Import the *Bypass ePO Requests* rule set from the library on the user interface of Web Gateway, move it to the topmost position of the rule sets tree, and enable it.
5. Download a McAfee ePO extension package for Web Gateway and install it on the McAfee ePO server.
6. On the user interface of the McAfee ePO server, register a new server for communication with Web Gateway, specifying an appliance that Web Gateway runs on.
   On the dashboard of the user interface, you should see, after about 15 minutes, data on web traffic that is processed on Web Gateway.
7. Push lists from the McAfee ePO server to Web Gateway.

Results

You should see the lists that you have pushed to Web Gateway on the lists tree of its user interface.

For more information on how to install a McAfee ePO extension package and perform activities on the McAfee ePO server, refer to the McAfee ePO documentation.

Set up a user account for Common Catalog lists

To enable the use of Common Catalog lists, you must set up a McAfee ePO user account on Web Gateway to create an instance that is allowed to handle the list transfer.

Task

1. Select Configuration → ePolicy Orchestrator.
2. Under ePolicy Orchestrator Settings, configure a user account.
   a. In the ePO user account field, leave the preconfigured value, which is `epo`.
   b. Next to the Password field, click Change.
      The New Password window opens.
   c. Use the window options to set a new password.
3. Make sure Enable data collection for ePO is selected.
4. Click Save Changes.

Results

The user of the McAfee ePO account that you have configured must also appear as an administrator in an administrator account on Web Gateway.

Set up an administrator account for Common Catalog lists

To enable the use of Common Catalog lists, you must set up an administrator account on Web Gateway with the same user name and password as for the McAfee ePO user account.

Task

1. Select Accounts → Administrator Accounts.
2. Under Internal Administrator Accounts, click Add.
   The Add Administrator window opens.
3. Set up an administrator account for using Common Catalog lists.
   a. In the User name field, type `epo`.
   b. In the Password and Password repeated fields, type the password you configured when setting up the user account for the ePO user.
   c. From the Role list, select the ePO Common Catalog Administrator role.
   d. Click Edit to review the current role settings.

      The Edit Role window opens. Enable the following settings if necessary:

      - Policy — Lists accessible
      - Policy — Lists creation
      - REST Interface accessible

   e. Click OK.
      The window closes and the new administrator account appears under Internal Administrator Accounts.

Results

Together with the user account for the McAfee ePO user, this administrator account serves as the instance on Web Gateway that must exist for handling the transfer of lists from a McAfee ePO server.

Enable use of the REST interface for Common Catalog lists

For communication with the McAfee ePO server that Common Catalog lists can be transferred from, you need to enable the internal REST (Representational State Transfer) interface on Web Gateway.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance that you want to transfer Common Catalog lists to and click User Interface.
3. Under UI Access, select both Enable Rest Interface over HTTP and Enable Rest Interface over HTTPS .
4. Under Login Page Options, select Allow multiple logins per login name.
5. Click Save Changes.

Sample settings for registering Web Gateway on a McAfee ePO server

To transfer Common Catalog lists to a Web Gateway appliance, you must register the appliance as a new server on the McAfee ePO server.

The following are sample settings for this registration.

Option	Sample value
Server type	`McAfee Web Gateway 7`
Name	`mwg7-3.sample-lab.local`
Notes	(optional)
Host name	`mwg7-3.sample-lab.local`
Host address	`171.18.19.226`
Administration port	`4712`
Statistics retrieval port	`9090`
User name (for access to the host GUI)	<Initial or current user name for access to the Web Gateway user interface>
Password	<Initial or current password for access to the Web Gateway user interface>
User name (for statistics retrieval and list management)	`epo`
Password	<Same password as the one that was configured for the ePO user and administrator accounts on Web Gateway>
Options	Allow ePO to manage lists on this system (enabled)

Note: The initial user name and password for access to the user interface of Web Gateway are `admin` and `webgateway`.

JavaScript Object Notation data

Data that is encoded in JavaScript Object Notation (JSON) format can be read, modified, and created on Web Gateway.

JavaScript Object Notation is a text-based data-interchange format. It can be read easily by JavaScript, but is not tied to using this language. The notation is used for communication with interactive websites, as well as with NoSQL and document-oriented databases, for example, MongoDB or Couch DB.

JSON-based programming interfaces exist for use in well-known social networks, such as Facebook or Twitter.

On Web Gateway, JSON data is used, for example, in scanning reports that are provided by McAfee® Advanced Threat Defense (Advanced Threat Defense). Lists that are retrieved form external sources and processed on Web Gateway can also be in JSON data format.

JSON data

JSON data is made available in what is called *objects*. A JSON object is a container that includes data of the same or of different ordinary types, such as strings, numbers, and others.

The basic structure of a JSON object can be represented as follows:

```
object: {"key": value, "key": value, ...}
```

For example:

```
Employee: {"First name": "Joe", "Last name": "Miller", "Age": 32}
```

The value of a JSON element can be data of the following types: string, number, Boolean, null.

A JSON object can also include an array:

```
object: {"key": value, "key": value, array: [value, value, ...]}
```

For example:

```
Employee: {"First name": "Joe", "Last Name": "Miller", "Children": [Ian, Lisa]}
```

In original JavaScript Object Notation, only objects and arrays can occur at the top level of a hierarchical data structure. However, when it is supported on Web Gateway, a simple element can also occur in top-level position.

A JSON object can also be embedded in another JSON object.

Using properties to handle JSON data

Several properties are available on Web Gateway for reading, modifying, and creating JSON data.

For example, the *JSON.FromString* property is used to create a JSON element from a string. The string is specified as a parameter of the property. So *JSON.FromString("Miller")* delivers the string "Miller" as the value of a JSON element.

A JSON object is created using the *JSON.CreateObject* property. This object is initially empty. To store a JSON element inside an object, you need to identify both items by giving them names.

An object is given a name by making it a user-defined property, which is always configured with a name.

For example, you can create a user-defined property under the name *User-Defined.myjsonemployee* and then use an event in a rule to give it the value of the JSON.CreateObject property.

Name				
Create JSON object as user-defined property				
Criteria		Action		Event
Always	–>	Continue	–	Set User-Defined.myjsonemployee = JSON.CreateObject

The empty JSON object *User-Defined.myjsonemployee* can be filled using the *JSON.StoreByName* property, which has parameters for object name, element key, and element value.

For example, the following stores an element with the key "Last name" and the value "Miller" in the object:

```
JSON.StoreByName(User-Defined.myjsonemployee, "Last name", JSON.FromString("Miller"))
```

Storing an element inside an object can also be performed in a simpler way:

- You need not create the object before using the JSON.StoreByName property.

  Specifying the object name as a parameter of the property creates the object if it did not exist before.
- You need not use the JSON.FromString property to obtain the element value.

  Specifying a string directly also creates this value. The same applies to the other ordinary data types that the value of a JSON element can have.

So, the following also stores an element inside an object:

```
JSON.StoreByName(User-Defined.myjsonemployee, "Last name", "Miller"))
```

Groups of JSON properties

Many JSON properties are similar to other properties in that they are used to perform the same kind of data handling activity.

The JSON.From<x> properties, for example, JSON.FromString, deliver a JSON element that has the value of a simple data type. The value of the simple data type is specified as the parameter of the JSON property.

The following are some important groups of JSON properties:

- **JSON.From**<x> = Delivers a JSON element that has the value of a simple data format

  Properties: *JSON.FromString*, *JSON.FromNumber*, *JSON.FromBool*, *JSON.FromStringList*, *JSON.FromNumberList*
- **JSON.As**<x> = Delivers the value of a JSON element in a simple data format

  The properties of this group are used to perform an operation that is the reverse of what the JSON.From<x> properties do.

  For these properties to work correctly, the format of the JSON element must match the simple data format.

  For example, the JSON.AsString property will only deliver a (simple) string if the value of the JSON element is a (JSON) string.

Properties: *JSON.AsString*, *JSON.AsNumber*, *JSON.AsBool*

- **JSON.Create**<x> = Creates a JSON object, array, or the element value 0.
  Properties: *JSON.CreateObject*, *JSON.CreateArray*, *JSON.CreateNull*
- **JSON.Get**<x> = Delivers a JSON element from within an object or the data type of an element
  JSON.GetByName delivers an element that is identified by its key from within a JSON object.
  JSON.GetAt delivers an element that is identified by its position within a JSON array.
  JSON.GetType delivers the type of an element.

Using JSON properties in filtering rules

The *JSON.ToString* property delivers the value of a JSON element in string format.

You can use this property, for example, in a simple rule to whitelist a particular client IP address.

In this rule, a given client IP address is compared to the client IP address you want to whitelist to see whether both addresses match.

Name		
Allow client IP address provided as JSON element value		
Criteria		Action
Client.IP equals String.ToIP(JSON.ToString(User-Defined.myjsonipaddress))	–>	StopCycle

The client IP address that is to be whitelisted is provided as the value of the user-defined property *User-Defined.myjsonipaddress*.

The JSON.ToString property delivers this value in string format. The String.ToIP property converts the string back into an IP address, so it can be compared to the address that is the value of the Client.IP property at the beginning of the rule.

Before the UserDefined.myjsonipaddress property can be used in the sample rule, you must create it in JSON data format and set its value to the address that is to be whitelisted.

To set the value, you can use an event in another sample rule, as shown in the following.

Name				
Set value of JSON type user defined property to client IP address				
Criteria		Action		Event
Always	–>	Continue	–	Set User-Defined.myjsonipaddress = JSON.FromString ("10.149.8.34")

The JSON.FromString property in the rule event converts the client IP address, which is specified as a property parameter in string format, into the value of a JSON element.

Retrieving JSON data from an Advanced Threat Defense report

When Advanced Threat Defense is called by a rule on Web Gateway to scan a web object, the scanning result is stored as the value of the *Antimalware.MATD.Report* property.

The result is provided as a string that has the elements of the result arranged in JSON style. It can be converted into a JSON element, using the JSON.ReadFromString property. This property takes the AntiMalware.MATD.Report property as a parameter.

The JSON element can then be set as the value of a user-defined property.

The rule that uses these properties could look as follows:

Name				
Set value of JSON type user defined property to Advanced Threat Defense report				
Criteria		Action		Event
Always	–>	Continue	–	Set User-Defined.myjsonmatdreport = JSON.ReadFromString (Antimalware.MATD.Report)

You can retrieve the data of the result using the JSON.GetByName property and, for example, write it into a log file.

Name				
Write JSON data from Advanced Threat Defense report into log file				
Criteria		Action		Event
Always	–>	Continue	–	FileSystemLogging.WriteLogEntry(Ge Defined.myjsonmatdreport, "Summary")<AdvancedThreat DefenseLog>

In the event of this rule, "Summary" is the key of a JSON element that has the data of a scanning result as its value. This key and its value are contained in a JSON object, which is the value of the Antimalware.MATD.Report property.

The structure of the JSON object is shown in the following.

It contains several embedded objects. The element keys are the ones that are actually used in a report, while the values are examples.

```
Report: {"Summary": {"Selectors": [{"Engine": "GAM engine", "MalwareName": "EICAR test file", "Severity":
"5" }], "Verdict": {"Severity": "5", "Description": "Subject is malicious" }, "Stats": [{"ID": "0", "Category":
"Persistence, Installation Boot Survival", "Severity": "5" }] }
```

Retrieving external lists in JSON data format

For handling JSON data in a list that has been retrieved from an external source, the *Ext.Lists.JSON* property is available. After retrieving the external list, the list content is a JSON element that is the value of this property.

Like all external list properties, *Ext.Lists.JSON* has three parameters in string format, which can be used to identify the external source.

Settings

Settings are used within Web Gateway for configuring modules (engines), rule actions, and system functions.

Settings names appear in different places on the user interface, for example, in the criteria, action, and events of rules or on the Settings and Appliances tabs.

After clicking a settings name, you can access and configure the parameters and values of the settings.

At the initial setup of the appliance, module and action settings are implemented together with the rule set system, as well as settings for the appliance system. Additional module and action settings are implemented when you import a rule set from the rule set library.

You can review and modify the initially implemented or imported settings. You can also completely delete module and action settings and create module and action settings of your own.

Types of settings

Different types of settings are used for the appliance system and in rule processing.

- **System settings** — Settings of the appliance system
- **Module settings** — Settings for the modules (also known as *engines*) that are called by rules to deliver values for properties and perform other jobs
- **Action settings** — Settings for the actions that rules execute

System settings

System settings are settings of the appliance system, for example, network interface settings or domain name system settings

You can access these settings on the Appliances tab of the Configuration top-level menu.

You can modify these settings, but not create new system settings.

Module settings

Module settings are settings for the modules (also known as *engines*) that are called by rules to deliver values for properties and perform other jobs.

The URL Filter module, for example, retrieves information on URL categories to deliver values for the URL.Categories property in a filtering rule.

In a rule, the settings name for a module that is called by the rule appears next to a rule property. For example, in a rule for virus and malware filtering, *Gateway Antimalware* can appear as the settings name next to the Antimalware.Infected property.

This means that when the Anti-Malware module is called to deliver the value *true* or *false* for the property, the module runs with the *Gateway Antimalware* settings. These settings specify, for example, which methods are used in scanning web objects for infections.

You can access module settings in rules and on the lower main branch of the settings tree on the Settings tab.

You can modify these settings and also create new settings.

Action settings

Action settings are settings for the actions that are executed by rules.

They are mainly configured to specify the messages that are sent to users who are affected by rule actions, such as Block or Authenticate. Actions that do not affect users have no settings, for example, Continue or Stop Rule Set.

You can access these settings in rules and on the upper main branch of the settings tree on the Settings tab.

Access settings

You can access settings on the Settings tab or by clicking a settings name in a rule. For accessing system settings, you must work with the Appliance tab of the Configuration top-level menu.

Access action and module settings on the Settings tab

You can use the Settings tab to access settings for actions and modules.

Task

1. Select Policy → Settings.
2. On the settings tree, navigate to the Actions or Engines branch to access the settings you want to work with.
3. To select settings, do one of the following:
    - On the Actions branch, click an action to expand it, and select the action settings you want to access.
    - On the Engine branch, click a module (also known as *engine)* to expand it, and select the module settings you want to access.

    The parameters and values of the settings appear on the settings pane.

Results

You can now work with the settings.

Access action and module settings in a rule

You can click names of settings for actions and modules that appear in rules to access these settings.

Task

1. Select Policy → Rule Sets
2. On the rule sets tree, select the rule set that contains the rule with the settings you want to access.
    The rules of the rule set appear on the settings pane.
3. Make sure Show details is selected.
4. In the rule with the settings you want to access, click the settings name:
    - In the rule criteria to access module settings
    - In the rule action to access action settings

    The Edit Settings window opens with the settings that you selected.

Results

You can now work with the settings.

Access system settings

You can access system settings using the Configuration top-level menu.

Task

1. Select Configuration → Appliances
2. On the appliances tree, select the appliance you want to configure system settings for and click the settings name.
   The parameters and values of the settings appear on the settings pane.

Results

You can now work with the settings.

Create action and module settings

You can create settings for modules and actions.

When creating these settings, you do not create them completely new, but use existing settings that you give a new name and modify as needed.

Task

1. Select Policy → Settings.
2. To select the settings that serve you as the starting point for creating new settings, use one of the following two methods:
   - On the settings tree, select these settings and click Add.

     The Add Settings window opens with the parameters and values of the selected settings.
   - Click Add right away.

     The Add Settings window opens.

     Select settings from the Settings for pane of the window.

     The parameters and values of these settings appear in the window.
3. In the Name field of the window, type a name for the new settings.
4. [Optional] In the Comment field, type a plain-text comment on the settings.
5. Modify the existing values of the settings as needed.
6. [Optional] Click the Permissions tab and configure who is allowed to access the settings.
7. Click OK.
   The window closes and the new settings appear on the settings tree.
8. Click Save Changes.

Anti-malware filtering

Anti-malware filtering ensures that the users of your network cannot access web objects that are infected by viruses and other malware. The filtering process detects infections and blocks access accordingly.

A default process for anti-malware filtering is implemented on Web Gateway after the initial setup.

This process requires no administration, but you can configure it to meet the requirements of your organization. You can also extend the process or create your own process.

Important configuration items used in this process include:

- Gateway Anti-Malware **rule set** — Default rule set for anti-malware filtering
- Anti-malware **settings** — Settings for the Anti-Malware module, which is involved in the anti-malware filtering process, for example, to handle the use of anti-malware engines for scanning web objects.

  The default settings for this module are the Gateway Anti-Malware settings.

To modify or extend the process, or to create your own process, you can use these items as a starting point.

Anti-malware filtering process

The anti-malware filtering process is rule-based like all processes that run on Web Gateway to ensure web security.

The anti-malware filtering rules are usually contained in one rule set. They include a blocking rule that blocks access to infected objects. Other rules in this process whitelist objects to exclude them from anti-malware filtering.

Whitelisting is implemented to ensure that the users of your network can access particular objects that are not considered a risk to web security.

The blocking rule of the default anti-malware filtering process on Web Gateway is executed if a web object that a user tries to access is found to be infected by a virus or other malware. To find out about an infection, the object is scanned.

The scanning is handled by the Anti-Malware module on Web Gateway. The module is called when the blocking rule is processed. It calls one or several anti-malware engines in turn, which perform the scanning.

Anti-malware filtering administration

When administering the anti-malware filtering process, you can use several configuration items that are available by default.

- Gateway Anti-Malware **rule set** — Default rule set for anti-malware filtering

  Using this rule set, you can run the default anti-malware filtering process on Web Gateway without further administrative activities.

  The rule set includes a blocking rule and whitelisting rules. Further rules ensure a high level of filtering quality.

  For example, one rule requires that the complete body of a web object is scanned for infections, even if only a request for accessing the object in parts was submitted.
- **Whitelists** — Used to exclude web objects from further anti-malware filtering
  - Anti-Malware Host Whitelist — Lists the URLs of hosts. Use this list to exclude requests with particular URLs from further anti-malware filtering.

    The list is empty by default and you need to fill the entries.
  - User Agent Whitelist — Lists user agents. Use this list to exclude requests with particular user agent information in its headers from further anti-malware filtering.

    The list is empty by default and you need to fill the entries.

- Gateway Anti-Malware **settings** — Default settings for the Anti-Malware module.

  An option is selected in these settings that enables the McAfee Gateway Anti-Malware (GAM) engine for scanning web objects. You can modify these settings, for example, to involve the Avira engine in the scanning process.

Configure anti-malware filtering

A default process for anti-malware filtering is implemented on Web Gateway after the initial setup. This process requires no administration, but you can configure it to perform anti-malware filtering in a way that suits your requirements.

Task

1. Configure the settings for the Anti-Malware module, which handles activities that are related to anti-malware filtering on Web Gateway.
   a. Select Policy → Settings. On the Engines branch of the settings tree, click an instance of the Anti-Malware settings, for example, the Gateway Anti-Malware settings, which are available by default after the initial setup.
   b. Modify the options of these settings as needed.

      For example:

      - Include the Avira scanning engine, which is not included by default, in the scanning process for web objects or exclude default engines.
      - Shift the focus in analyzing mobile code behavior from accuracy to proactivity, or in reverse direction, to achieve more accurate scanning results or block more suspicious web objects proactively, even if some of this code might actually not be malicious.
      - Modify advanced settings, for example, settings for running a prescan on web objects to reduce workload for the scanning engines proper.

   c. Create your own instance of the settings for the Anti-Malware module or more of them as needed and use them in anti-malware filtering rules.
2. Configure the Gateway Anti-Malware rule set, which is provided as default rule set for anti-malware filtering on Web Gateway after the initial setup..
   a. Select Policy → Rule Sets and on the rule sets tree, click Gateway Anti-Malware.
   b. Modify the options of the key elements view that is provided for this rule set as needed.

      For example:

      - Allow web objects to bypass anti-malware filtering including scanning by the scanning engines.

        You can allow bypassing based on the user-agent information that is sent in the headers of requests for web access or on the hosts that access is requested by entering user agent information and the URLs of these hosts in whitelists.

        You can also allow bypassing for web objects that exceed a size that you configure.
      - Access the settings for the Anti-Malware module from this view and configure them.

   c. Proceed from the key elements view to the complete rules view of the rule set. This allows you to configure more rule elements and also to modify, move, and delete rules, insert rules from other rule sets and create rules of your own.

      You can also create your own rule set for anti-malware filtering and use it along with the default rule set or delete this rule set.
3. Extend the anti-malware filtering process filtering by using options that are not provided as part of the Anti-Malware settings or in the Gateway Anti-Malware rule set.

   You can, for example, retrieve URL filtering and TIE server information for the anti-malware filtering process or add Advanced Threat Defense (ATD) scanning to this process.

Configure the use of the scanning engines

Several scanning engines are available for scanning web objects, depending on what licenses you have purchased. You can configure their use if you do not want to use them in the default way.

Task

1. Select Policy → Rule Sets.
2. Access an instance of the Anti-Malware settings.
   a. Select Policy → Settings.
   b. On the settings tree, navigate to the Anti-Malware settings and select one of the settings instances that are available, for example, the Gateway Anti-Malware settings.

   Note: You can also access an instance of the Anti-Malware settings from within a suitable rule or from the key elements view of the Gateway Anti-Malware default rule set.
3. Under Select Scanning Engines and Behavior, select one of the following options for using scanning engines in different combinations:

   At the end, make a choice on whether scanning will continue or not after one of the scanning engines has detected a virus or other malware.

Option	Definition
Full McAfee coverage: The recommended high-performance configuration	When selected, the McAfee Gateway Anti-Malware engine and the McAfee Anti-Malware engine are active. Web objects are then scanned using: *Proactive methods + Virus signatures* If you are running Web Gateway with a license for McAfee Gateway Anti-Malware in addition to the one for Web Gateway itself, this option is selected by default.
Layered coverage: Full McAfee coverage plus specific Avira engine features — minor performance impact	When selected, the McAfee Gateway Anti-Malware engine, the McAfee Anti-Malware engine, and, for some web objects, also the third-party Avira engine are active. Web objects are then scanned using: *Proactive methods + Virus signatures + Third-party engine functions for some web objects*
Duplicate coverage: Full McAfee coverage and Avira engine — less performance and more false positives	When selected, the McAfee Gateway Anti-Malware engine, the McAfee Anti-Malware engine, and the third-party Avira engine are active. Web objects are then scanned using: *Proactive methods + Virus signatures + Third-party engine functions*
McAfee Anti-Malware without mobile code scanning and emulation	When selected, only the McAfee Anti-Malware engine is active. Web objects are then scanned using: *Virus signatures* This is the option that you must select when running Web Gateway with a license for Web Gateway only, but without a license for McAfee Gateway Anti-Malware. The McAfee Gateway Anti-Malware license includes a license for Avira.
Avira only: Only uses Avira engine — not recommended	When selected, only the Avira engine is active. Web objects are then scanned using:

Option	Definition
	Third-party engine functions
McAfee Advanced Threat Defense only: Send files to an MATD appliance for deep analysis through sandboxing	When selected, only scanning by Advanced Threat Defense is active.
Stop virus scanning right after an engine detected a virus	When selected, all engines stop scanning a web object as soon as one of them has detected an infection by a virus or other malware.

4. Click OK to close the window.

What to do next

If you select the Avira only option, we recommend renaming these settings to indicate that a key value has changed.

You might, for example, change the settings name from *Gateway Anti-Malware* to *Avira Anti-Malware*.

Instead of renaming the settings, you might also create additional settings to have different settings options available for configuring rules.

Use case: Blocking the download of a virus-infected file

The anti-malware filtering functions of Web Gateway protect your network against infections from the web.

An infection might be imported, for example, when a user who works inside this network attempts to download a virus-infected file from a web server.

The infection is detected when the file is intercepted and scanned on Web Gateway.

According to a web security rule, the download is blocked and the user who requested the download is notified.

Note: Blocking the download of a virus-infected file is performed by default on Web Gateway, so no administrator activities are required. You can, however, modify the default process.

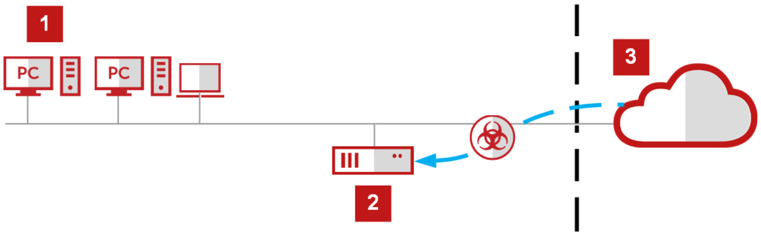
Scenario

1. Working with a web browser that is configured to run as a client of Web Gateway, a user clicks a link on a webpage to request the download of a file from a web server.
2. The request is processed.
   a. The request is redirected to Web Gateway.
   b. The request is processed in the request cycle on Web Gateway.

      The web security rules that are enabled for this cycle are processed to see if they apply.

      The request might not be compliant with your web security policy, which means at least one rule applies that forbids forwarding this request to the web.
   c. Because no rule in the request cycle applies, Web Gateway forwards the request to the web server.
3. The web server accepts the request and sends the file that the user wants to download in response.

Web sends response with infected file



1 – *Your network*	2 – *Web Gateway*	3 – *Web*

4. The response is processed.
   a. The response arrives at Web Gateway.
   b. The response is processed in the response cycle of Web Gateway.

      The web security rules that are enabled for this cycle are processed to see if they apply.

      These rules include the following rule, which is part of the default rule set for anti-malware filtering:

      Antimalware.Infected<Gateway Anti-Malware> equals true –> Block<Virus Found> — Statistics.Counter.Increment ("BlockedByAntiMalware",1)<Default>

      In plain text, this rule says that a virus-infected web object must not be forwarded to your network in response to a download request that a user submitted.
   c. The rule applies if its criteria matches. To find out if this is the case, the following substeps are performed.
      - The value for the Antimalware.Infected property is determined. If this value is *true*, the criteria matches.

        To determine the value, the anti-malware engines that are licensed and enabled process the response. The URL and the response header are checked and the file that was sent as the body of the response is scanned.

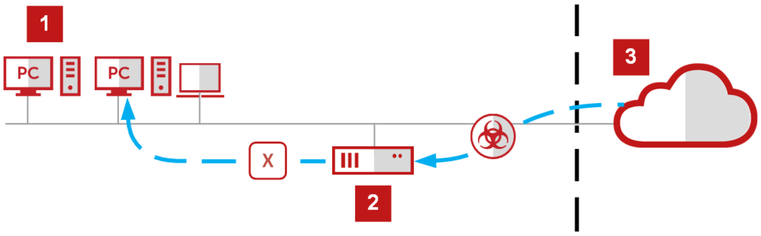
        Enabling or disabling these engines is part of configuring the Gateway Anti-Malware settings, which are shown as applicable after the property name in the rule.

        The following engines are enabled by default (if licensed):
        - McAfee engine (included in the license for Web Gateway)
        - Gateway Anti-Malware engine (GAM engine, requires a separate license)

        A third anti-malware engine, the Avira engine, can also be enabled (included in the license for the GAM engine).
      - The result of scanning the file is that it is virus-infected. So the value of the Antimalware.Infected property is set to *true*, which means that the rule criteria matches.
   d. Because the criteria matches, the rule action, which is Block, is executed.
      - Rule processing is stopped for all other rules on Web Gateway.

        A rule already blocks the download, so there is no need to try out any other rules and see if they also block it.
      - The virus-infected file is *not* forwarded to the user's browser.
      - A block message is sent to this browser instead.

        Web Gateway sends block message to browser

1 – *Your network*	2 – *Web Gateway*	3 – *Web*

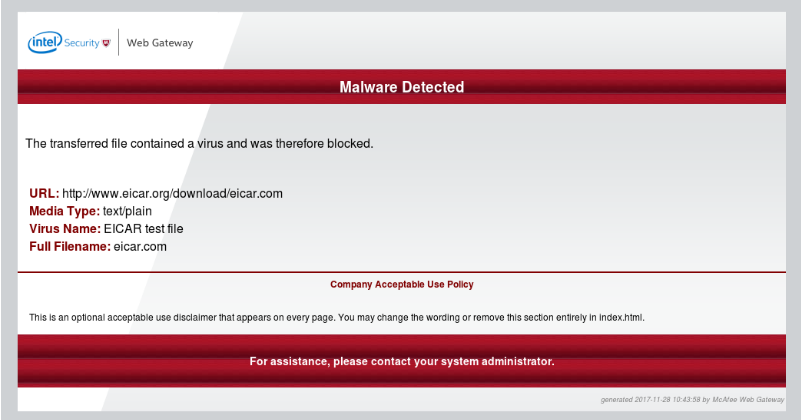
        This message notifies the user that downloading the requested file was blocked because this file was virus-infected.

        The text and layout of the message can be edited by configuring the Virus Found settings, which are shown after the action name in the rule.
      - The Statistics.Counter.Increment event increases the counter for blocking caused by anti-malware filtering.

5. The block message appears in the browser that the user works with.

   The following is an example of a block message in English language.

Block message: Malware Detected



Malware Detected

The transferred file contained a virus and was therefore blocked.

URL: http://www.eicar.org/download/eicar.com
Media Type: text/plain
Virus Name: EICAR test file
Full Filename: eicar.com

Company Acceptable Use Policy

This is an optional acceptable use disclaimer that appears on every page. You may change the wording or remove this section entirely in index.html.

For assistance, please contact your system administrator.

generated 2017-11-28 10:43:58 by McAfee Web Gateway

Anti-Malware settings

The Anti-Malware settings are the settings for the Anti-Malware module, which handles activities related to anti-malware filtering on a Web Gateway appliance.

For more information, see the sections on anti-malware filtering and Anti-Malware settings in the *McAfee Web Gateway Product Guide*.

Gateway Anti-Malware settings

The Gateway Anti-Malware settings are settings for the Anti-Malware module (engine) that are by default available after the initial setup of Web Gateway.

Extending the anti-malware filtering process

You can extend the default process for anti-malware filtering in several ways.

To include more information in the process, which improves the accuracy of its results, the following can be done.

- **Using URL information** — URL information can be used in the anti-malware filtering process. This information includes URL categories and reputation scores.
- **Connecting to a TIE server** — Information retrieved from a TIE server can be used in the anti-malware filtering process. The TIE server is in turn notified of critical filtering results found by anti-malware filtering on Web Gateway.
- **Integrating Advanced Threat Defense** — After having been scanned on Web Gateway, web objects can additionally be scanned by Advanced Threat Defense.

Other measures for extending the process can be taken to ensure a smooth workflow.

- **Using the anti-malware queue** — To avoid overloading of the anti-malware filtering process, user requests for access to web objects can be moved to a queue before being processed.

- **Scanning media streams chunk-by-chunk** — The scanning of media streams, which is done for anti-malware filtering purposes, can be performed chunk-by-chunk instead of in a single long-lasting process. This improves user experience by reducing waiting time.

Extending the process can also be a means to prevent potential issues from occurring.

- **Dealing with a missing ICAP host header** — When messages received in ICAP communication on Web Gateway fail to provide a host header, processing issues can occur. There are several ways to solve these issues.

Media stream scanning

Media streams can be scanned on Web Gateway chunk-by-chunk, which allows users to see or hear downloaded streaming media faster, as they do not have to wait until a stream has been scanned completely.

This scanning method is performed by the Media Stream Scanner, which is provided by the Gateway Anti-Malware engine. Streaming media is scanned and delivered chunk-by-chunk to the client that requested the download. If an infection is detected in a chunk, the download stops, and this chunk and the rest of the streaming media are not delivered.

The scanning that is performed by the Media Stream Scanner uses the proactive functions of the Gateway Anti-Malware engine. The Avira engine, which can also be configured to scan web objects for infections by viruses and other malware, is not involved when the Media Stream Scanner is active.

The scanner is started by an event of a rule in the Gateway Anti-Malware rule set of the default rule set system. The rule applies if the Stream Detector module finds that a web object that was received on Web Gateway in response to a download request is streaming media.

Processing of the rule set stops and the remaining rule in the rule set, which also lets web objects be scanned for infections by viruses and other malware, is not processed.

If a web object is not recognized by the Stream Detector as streaming media, the rule does not apply, processing continues with the remaining rule, and the web object is scanned according to the settings that are configured for this rule.

Using URL information for anti-malware filtering

URL information is important for achieving accurate results in anti-malware filtering. Particular settings of the filter modules are required to make this information available.

When the Gateway Anti-Malware engine scans files within anti-malware filtering on Web Gateway, it uses available information about the URL of a file to achieve a more reliable result. URL categories and reputation scores are an important part of this information.

As URL filtering on Web Gateway is mainly handled by the URL Filter module (or engine), the Anti-Malware module (or engine), which is the module for anti-malware filtering, requests URL information from the URL Filter module. The URL Filter module retrieves this information from several sources, among them the Global Threat Intelligence system, depending on its settings.

To ensure that suitable information for the scanning process is passed on to the Gateway Anti-Malware engine, options that enable queries to the Global Threat Intelligence system must be configured for the Anti-Malware and URL Filter modules.

Ensure the use of URL information for anti-malware filtering

Ensure that information about URL categories and reputation scores from the Global Threat Intelligence system is made available to the scanning process in anti-malware filtering on Web Gateway.

Task

1. Select Policy → Settings.
2. Expand all settings that are configured for the Anti-Malware module (engine) and ensure that in the Advanced Settings section, the Provide GTI web and file reputation queries to McAfee Gateway Anti-Malware option is selected.

3. Select Policy → Rule Sets and ensure that the following applies.

   ◦

     There is a Common Rules rule set with a nested Set URL Filter Internal Settings rule set.

     These rule sets are part of the default rule set system. If they have been deleted or modified, import the default versions of these rule sets from the rule set library.

   ◦

     The nested Set URL Filter Internal Settings rule set contains the Set URL Filter settings to be used by other filters rule with the URLFilter.SetInternalSettings event.

     Note: The event settings, which are named Default, are the default settings of the URL Filter module.

   ◦

     In the Rating Settings section of the event settings, Use online GTI web reputation and categorization services if local ratings yields no result is selected.

     Note:

     This option is selected and grayed out by default. It is still enabled as long as the Enable the Dynamic Content Classifier if local ratings yields no result option is selected, which is also true by default.

     If you deselect Enable the Dynamic Content Classifier if local ratings yields no result, Use online GTI web reputation and categorization services if local ratings yields no result remains selected, but is no longer grayed out.

4. If you have modified any settings options or the rule set system, click Save Changes.

Integrating TIE server information with anti-malware filtering

You can integrate TIE server information with anti-malware filtering on Web Gateway, using this information in filtering rules and notifying the TIE server of critical scanning results found on Web Gateway.

A rule set is available in the library to implement this integrated filtering, providing several rules in addition to the rules in the Gateway Anti-Malware default rule set.

The additional rules integrate anti-malware filtering as performed by the filtering functions that are available on Web Gateway with information retrieved from a TIE server. The TIE server is in turn notified of critical filtering results found on Web Gateway.

Note: The integrated filtering is only applied to files with media type Executables.

DXL messages are used to exchange information between Web Gateway and the TIE server. As parts of the DXL architecture are managed by a McAfee ePO server, Web Gateway must also be configured to connect to this administration device.

Property and event for exchanging information with a TIE server

The following property and event can be used in rules that handle information exchange with a TIE server.

- TIE.Filereputation — The value of this property is set to the reputation score that is queried and retrieved from a TIE server for a particular file.

  Processing of the property is performed by the TIE Filter module, which runs with particular settings.
- TIE: Report file reputation — This event sends a file reputation score to a TIE server. The score is based on the malware probability that the Gateway Anti-Malware (GAM) engine on Web Gateway finds after scanning a file.

  Scores are sent according to the scale of values used on a TIE server, corresponding to ranges of probability grades. For example, for a malware probability between 60 and 80, 30 is sent as a score to the TIE server.

Sample rule for using file reputation retrieved from a TIE server

The following sample rule uses TIE.Filereputation property to find out whether the reputation of a file that is processed on Web Gateway remains below a particular value. Information about the file reputation is retrieved from a TIE server.

If the file reputation actually remains below the configured value, an action is executed to block access to the file.

The blocking action runs with particular settings, which you can configure to provide a message to inform the user who requested the file about the blocking reason.

Name		
Block after retrieving information about bad reputation from a TIE server		
Criteria		Action
TIE.Filereputation less than or equals 30	–>	Block<TIE Reputation>

Sample rule for reporting file reputation to a TIE server

The following sample rule uses the TIE: Report file reputation event to send a file reputation score to a TIE server.

The score is based on the malware probability for a file that is processed on Web Gateway. The Antimalware.Infected and Antimalware.Proactive.Probability are used to find out about this probability.

If the probability exceeds the configured value, an action is executed to block access to the file. The TIE: Report file reputation event then sends a reputation score to a TIE server, which is based on the found probability range.

Name			
Send information about file reputation to a TIE server			
Criteria		Action	Event
Antimalware.Infected<Gateway Anti-Malware with TIE> equals true AND Antimalware.Proactive.Probability<Gateway Anti-Malware> greater than or equals 90	–>	Block<Virus Found>	TIE: Report File Reputation (1)

Configure integrating TIE server information with anti-malware filtering

To integrate TIE server information with anti-malware filtering, import the appropriate library rule set and configure settings for connecting to a McAfee ePO server.

Task

1. Implement the rule set for integrating TIE server information with anti-malware filtering.
   a. Select Policy → Rule Sets.
   b. Import the Gateway Anti-Malware with TIE rule set from the library.
   c. On the rule sets tree, place the imported rule set immediately before the default Gateway Anti-Malware rule set and disable or delete the default rule set.
2. Configure settings for connecting to a McAfee ePO server.
   a. Select Configuration → Appliances.
   b. On the appliances tree, select the appliance that you want to connect to a McAfee ePO server and click ePolicy Orchestrator.
   c. Configure a host name, user account, and password for use by Web Gateway when connecting to a McAfee ePO server.
   d. Click Rejoining ePO for DXL communication to complete the setup.

3. [Optional] Configure DXL message tracing.
   a. On the appliances tree, keep your appliance selected and click Troubleshooting.
   b. In the Troubleshooting section of the configuration pane, select Enable DXL tracing and, optionally, Write full message body into log.
4. Click Save Changes.

Dealing with a missing host header

When the host header is missing from an ICAP request that is received on Web Gateway, additional measures can be required to configure anti-malware scanning for this request.

Note: If you have only purchased a license for the McAfee scanning engine, the problem with the missing host header does not arise since this engine does not require the information that is provided by this header for scanning.

When a client of Web Gateway sends a request under the ICAP protocol, the host header can be missing from this request, which occurs, however, very rarely.

A request with no host header can be sent, for example, in the reqmod mode of ICAP communication. The GET portion of this request contains an empty URL, which means that this URL only consists of an entry for the HTTP protocol, which the ICAP request is embedded in. The URL then just looks as follows: http://.

Importance of the host header

When the Gateway Anti-Malware engine is involved in the scanning process on Web Gateway, it requires a URL to perform behavioral scanning as part of its scanning activities. The URL is used in the scanning, for example, to retrieve reputation scores and category information from the Global Threat Intelligence system.

The URL is assembled on Web Gateway from several sources, one of which is the host header in a request. It is then made available to the Gateway Anti-Malware engine. The URL cannot be assembled and made available, however, without retrieving information from the host header.

A missing host header therefore leads to an error in the scanning process when the Gateway Anti-Malware engine is involved in the usual way. Under the default rules, this error results in blocking the request, which means that it is not forwarded to the requested destination. The error is logged in the mwg-antimalware log.

Combined use of anti-malware engines

For scanning web traffic, the Gateway Anti-Malware engine relies in parts on another scanning engine, which is known as McAfee engine. A reduced range of scanning activities can also be performed by the McAfee engine alone, but only the combined use of the two engines ensures the anti-malware protection that we recommend.

Whether both engines are available on Web Gateway depends on whether you have purchased a license for the Gateway Anti-Malware engine, which also covers use of the McAfee engine, or a license for the McAfee engine only.

Solving the host header problem

There are several ways to solve the problem when a host header is missing from an ICAP request.

- Configure the ICAP client to send a host header.

  This solution is not applied on Web Gateway, but on the client system that sent the incomplete request. The request might have been sent without a host header due to an error within the client configuration.

  For more information, refer to documentation that explains the ICAP protocol.
- Create rules for anti-malware filtering with full and reduced use of the Gateway Anti-Malware engine.

  You can perform scanning with a reduced use of the Gateway Anti-Malware engine that does not require the processing of information from a host header.

  The rule set for anti-malware filtering then includes:

    - A rule for performing the default scanning process with full use of the Gateway Anti-Malware engine if a request includes a host header
    - A rule for performing a scanning process with reduced use of the Gateway Anti-Malware engine if a request does not include a host header

For more information about these two rules, see the *Rules for anti-malware filtering with full or reduced use of the Gateway Anti-Malware engine* subsection.

- Create a rule that adds a host header.

  You can add a rule that sets a value for the host header and place it before the rule that controls anti-malware scanning. When this second rule is executed, the host header value is found by the rule engine and scanning can be performed making full use of the Gateway Anti-Malware engine.

  The rule provides the host header value using an event that sets the URL.Host property. If you know that requests are usually received from a particular host, you can set the property to the value for this host.

  If you do not know such a host, you can set the property to a dummy value. Setting the property in this way is sufficient for letting the scanning process make full use of the Gateway Anti-Malware engine. Inappropriate host information can, however, have an impact on the anti-malware filtering results, which might include an increased number of false positives.

  For more information about the rule, see the *Rule for adding a host header* subsection.

Rules for making full or reduced use of the Gateway Anti-Malware engine

The following are sample rules for performing anti-malware filtering with full or reduced use of the Gateway Anti-Malware engine, depending on whether a host header is sent with a request or missing from it.

Note: When creating these rules, you can use the rule Block if virus was found in the default Gateway Anti-Malware rule set as a starting point. After creating and enabling these rules, the default rule must be deleted.

The first rule blocks a request that includes a host header if scanning the request results in detecting an infection by a virus or other malware.

When the rule engine processes the rule, it calls the Anti-Malware module to provide a value for the AV.Infected property. The module runs with default settings, which means full use of the Gateway Anti-Malware engine is made in the scanning process that is performed to provide the property value.

Name			
Scan with full use of the Gateway Anti-Malware engine			
Criteria		Action	Event
URL.Host does not equal " " AND AV.Infected<Default> equals true	–>	Block<Virus Found>	Statistics. Counter.Increment ("BlockedByAnti Malware">, 1)<Default>

The second rule blocks a request that does not include a host header if scanning the request results in detecting an infection by a virus or other malware.

When the rule engine processes the rule, it calls the Anti-Malware module to provide a value for the AV.Infected property. The module does not run with default settings, but with new settings that let the scanning process be performed with reduced use of the Gateway Anti-Malware engine.

The new settings differ from the default settings in that the option Enable mobile code scanning is disabled.

Name			
Scan with reduced use of the Gateway Anti-Malware engine			
Criteria		Action	Event
URL.Host equals " " AND AV.Infected<Reduced use of Gateway Anti-Malware engine> equals true	–>	Block<Virus Found>	Statistics. Counter.Increment ("BlockedByAnti Malware">, 1)<Default>

Rule for adding a host header

The following is a sample rule for adding a host header to an ICAP request that was sent with this header missing.

Name			
Add a host header			
Criteria		Action	Event
URL.Host equals " "	–>	Continue	Set URL.Host=<value for host that request was sent from>

Configure settings for reduced use of the Gateway Anti-Malware engine

Configure an option in the settings for anti-malware filtering to achieve a reduced use of the Gateway Anti-Malware engine. This reduced use can be required if a request does not contains a host header.

Task

1. Create new settings for anti-malware filtering, based on the default settings, and give them a suitable name, for example, Reduced use of Gateway Anti-Malware engine.
2. Within the new settings, expand Advanced Settings for McAfee Gateway Anti-Malware.
3. Deselect Enable mobile code scanning.
4. Click Save Changes.

Results

You can now insert the new settings in a rule for performing anti-malware filtering with reduced use of the Gateway Anti-Malware engine.

For more information about how to create new settings, see the *Create action and module settings* section of the *Settings* chapter.

Anti-malware queue

To avoid overloading of the modules that scan web objects for infections by viruses and other malware, requests for access to web objects are moved to a queue before being processed.

This queue is known as the *anti-malware queue*. When a request has been received on the appliance, it is moved to this queue by a working thread of the proxy module. It remains there until it is fetched by another thread and forwarded to a thread of one of the scanning modules.

The same applies to responses received from web servers that requests have been forwarded to.

The working threads that deliver requests and responses to the scanning modules, as well as those that are used by the modules to execute scanning activities, are referred to as *anti-malware working threads* or simply as *AV threads*.

When configuring the anti-malware queue, you can specify the following:

- Number of available anti-malware working threads
- Size of the anti-malware queue
- Maximum time for requests and responses to stay in the queue

Note: Moving requests and responses to the anti-malware queue is a solution to avoid load peaks occurring over a short period of time. Permanent overloading should be addressed by other measures.

Configure the anti-malware queue

You can configure settings for the anti-malware queue to avoid overloading of the scanning modules.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure the anti-malware queue on and click Anti-Malware.
   The settings for the anti-malware queue appear on the settings pane.
3. Configure these settings as needed.
4. Click Save Changes.

URL filtering

URL filtering ensures that the users of your network cannot access web objects that are considered a risk for web security or are not allowed because they contain inappropriate subject matter or for other reasons.

The filtering process uses blocking lists, category information, and reputation scores for the URLs of web objects and blocks or allows access accordingly.

A default process for URL filtering is implemented on Web Gateway after the initial setup. Important configuration items used in this process include:

- URL Filtering **rule set** — Default rule set for URL filtering
- Dynamic Content Classification **rule set** — Default rule set supporting the URL filtering process

  The rules in this rule set categorize web objects based on the analysis of the Dynamic Content Classifier component when other URL filtering methods yield no results.
- URL Filter **settings** — Default settings for the URL Filter module, which handles the retrieval of category information and reputation scores from intelligence systems.

  These settings also include options for configuring the Dynamic Content Classifier.

The default process requires that you maintain the block lists used by the rules in the URL Filtering and Dynamic Content Classification rule sets. You can further modify this process to meet the requirements of your organization.

You can also extend the process in several ways or set up a process of your own.

URL filtering process

The URL filtering process includes several elements, which contribute to it in different ways.

- **Filtering rules** — Control the process. There are usually the following types of rules.
  - **Blocking rules** — Block access to web objects with particular URLs.

    The rules apply if a URL has been entered in a list that is used by these rules or falls into a category that is on a list.

    When categories are used in a rule, the URL filter module is called to handle the retrieval of category information from the Global Threat Intelligence (GTI) service.
  - **Whitelisting rules** — Exclude web objects from further URL filtering to ensure they can be accessed by the users in your network.

    Whitelisting rules are placed before the blocking rules in an URL filtering rule set. If a whitelisting rule applies, processing of the following URL filtering rules is stopped to ensure that the blocking rule is not executed.
- **Whitelists and blocking lists** — These lists are used by whitelisting and blocking rule that exist in the URL filtering process.

  Because a URL filtering rule set is only used for URL filtering, multiple whitelists for several types of objects are not needed in the filtering process, in contrast to, for example, anti-malware filtering.
- URL Filter **module** —This module, which is also known as an *engine*, retrieves information on URL categories and reputation scores from the Global Threat Intelligence™ service that is provided by McAfee. Based on this information, blocking rules block access to URLs.

  Various technologies, such as link crawlers, security forensics, honeypot networks, sophisticated auto-rating tools, and customer logs are used to gather this information. An international, multi-lingual team of McAfee web analysts evaluates the information and enters URLs under particular categories into a database.

  To gather information on the reputation of a URL, its behavior on a worldwide real-time basis is analyzed, for example, where a URL shows up in the web, its domain behavior, and other details.

  You can configure settings for this module, for example, to perform a DNS lookup for URLs and include the corresponding IP address in the search for category information.

URL filtering administration

When administering the URL filtering process, you can use several configuration items that are available by default.

- URL Filtering **rule set** — Default rule set for URL filtering

  This rule set includes two nested rule sets, which allow you to run the default URL filtering process on Web Gateway in two different ways.

  - Special URL Filtering Group **rule set** — Nested rule set for performing URL filtering with regard to particular users, user groups, and IP addresses.

    The rule set includes a blocking rule and whitelisting rules. Further rules ensure a high level of filtering quality.

    For example, one rule requires that the complete body of a web object is scanned for infections, even if only a request for accessing the object in parts was submitted.

  - Default **rule set** — Nested rule set for performing URL filtering in general, regardless of any particular users, user groups, or IP addresses

    The rule set includes a blocking rule and whitelisting rules. Further rules ensure a high level of filtering quality.

    For example, one rule requires that the complete body of a web object is scanned for infections, even if only a request for accessing the object in parts was submitted.

- **Whitelists and blocking lists** — Used to allow and block access to web objects with particular URLs

  - URL Whitelist — Lists URLs. Use this list to exclude requests for access to web objects with particular URLs from further URL filtering.

    This way you ensure that users are not prevented from accessing these objects by URL filtering

    The list is empty by default and you need to fill the entries.

  - URL Blocklist — List user agents. Use this list to exclude requests with particular user agent information in its headers from further anti-malware filtering.

    The list is empty by default and you need to fill the entries.

  - Category Blocklist — List user agents. Use this list to exclude requests with particular user agent information in its headers from further anti-malware filtering.

    The list is empty by default and you need to fill the entries.

- **Blocklists** — Used to exclude web objects from further anti-malware filtering

  - URL Host Whitelist — Lists the URLs of hosts. Use this list to exclude requests with particular URLs from further anti-malware filtering.

    The list is empty by default and you need to fill the entries.

  - User Agent Whitelist — List user agents. Use this list to exclude requests with particular user agent information in its headers from further anti-malware filtering.

    The list is empty by default and you need to fill the entries.

- URL Filter **settings** — Default settings for the URL Filter module

  An option is selected in these settings that enables the McAfee Gateway Anti-Malware (GAM) engine for scanning web objects.

  You can change these settings, for example, to involve the Avira engine in the scanning process.

  You can also create your own rule set, lists, and settings for URL filtering.

Configure URL filtering

You can configure URL filtering to adapt this process to the needs of your network.

To configure URL filtering, you can work with the key elements view or the rules view.

Task

1. Review the rules in the rule set for URL filtering.
   By default, this is the *URL Filtering* rule set.
2. Modify these rules as needed.
   You can, for example, do the following.
   - Enable or disable blocking rules and the whitelist rule
   - Edit the lists used by these rules
     Note: A yellow triangle next to a list name means the list is initially empty and you need to fill the entries.
   - Modify the settings of the URL Filter module
3. Save your changes.

Extending the URL filtering process

A default process for URL filtering is implemented after the initial setup. You can extend this process in several ways.

- You can implement your own URL filter database.
- You can work with the Dynamic Content Classifier
- You can use a Private GTI Cloud service for URL filtering.
- You can use an IFP proxy for URL filtering.

Use case: Blocking a URL in a forbidden category

URL filtering on Web Gateway protects your network by preventing users from accessing websites that you don't want to allow. The filtering is based on individual URLs and URL categories.

URL filtering is performed, for example, when a user who works inside your network tries to access an online shopping website.

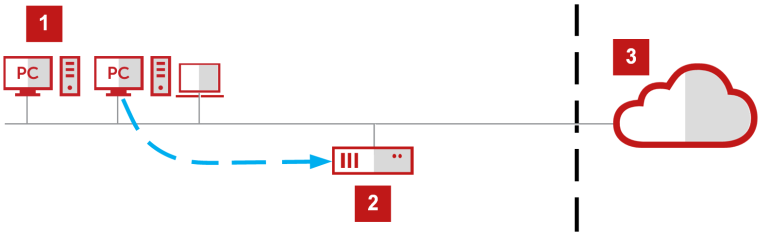
Your web security policy might not allow online shopping or allow it only to a limited extent, for example, during the lunch break. Access to the online shopping website is blocked in line with this, and the user who requested the access is notified.

Note: Online shopping is not among the URL categories that are blocked by default on Web Gateway. But as an administrator, you can add this category to the relevant block list.

Scenario

1. A user clicks a link on a webpage of an online shopping site from inside your network. The user is working with a web browser that is configured to run as a client of Web Gateway.
2. The request is redirected to Web Gateway.

Request for web access is redirected to Web Gateway



1 – *Your network*	2 – *Web Gateway*	3 – *Web*

3. The request is processed in the request cycle on Web Gateway to see if any of the web security rules that are enabled in this cycle apply.

   a. Processing finds the following rule enabled. It is a rule in a default rule set for URL filtering.

      URL.Categories<Default> at least one in list Category Block List for Default Group –> Block<URL Blocked> — Statistics.Counter.Increment (“BlockedByURLFilter”,1)<Default>

      In plain text, this rule says that a request must not be forwarded from your network to the web if the URL within this request is in a category that is on a particular block list.

   b. The rule applies if its criteria matches. To find out if this is so, these substeps are performed:

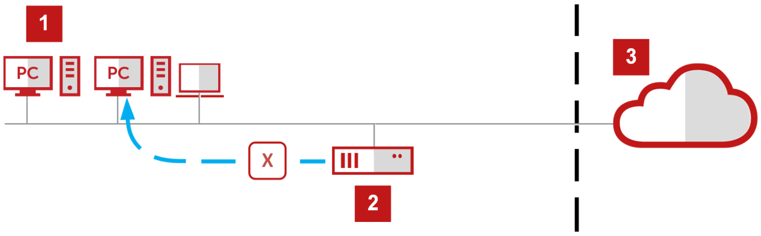
      - To provide a value for the URL.Categories property, the URL Filter module tries to find the category that the URL falls into. For this purpose, the module retrieves information from the Global Threat Intelligence service.
      - When the URL category has been found, processing continues with checking the block list used by the rule to see if it includes the category.
      - The result is that the URL sent with the request is in the Online Shopping category, which is on the Category Block List for Default Group.

   c. Because the criteria matches, the rule action, which is Block, is executed.

      - Rule processing is stopped for all other rules on Web Gateway.

        A rule already blocks the access, so there is no need to try out any other rules and see if they also block it.
      - The user's request for access to the online shopping site is *not* forwarded to the web server of that site.
      - A block message is sent to the user's browser instead.

Web Gateway sends block message to browser



1 – *Your network*	2 – *Web Gateway*	3 – *Web*

The message notifies the user that the request was blocked because the URL sent with it was that of an online shopping site, which falls in a forbidden category

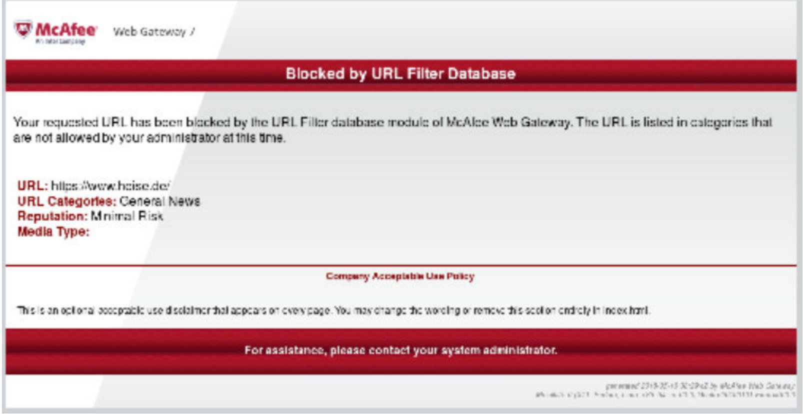
The text and layout of the message can be edited by configuring the BlockedByURLFilter settings, which are shown after the action name in the rule.

- The Statistics.Counter.Increment event increases the counter for blocking caused by URL filtering.

4. The block message appears in the user's browser.

   The following is an example of a block message in English language.

Block message: URL in forbidden category



Blocked by URL Filter Database

Your requested URL has been blocked by the URL Filter database module of McAfee Web Gateway. The URL is listed in categories that are not allowed by your administrator at this time.

URL: https://www.heise.de/
URL Categories: General News
Reputation: Minimal Risk
Media Type:

Company Acceptable Use Policy

This is an optional acceptable use disclaimer that appears on every page. You may change the wording or remove this section entirely in index.html.

For assistance, please contact your system administrator.

[illegible]

Best practices - Using URL properties to whitelist web objects

URL properties, such as *URL*, *URL.Host*, *URL.Host.BelongsToDomains*, and others, can be used in the criteria of rules to whitelist web objects.

When a web object is whitelisted, users are allowed to access it, for example, to view a web page or download a file. Whitelisting rules are inserted into appropriate rule sets within the rule set system of Web Gateway. They usually stop further rule processing with regard to the current request for accessing a web object to prevent other rules from blocking this access.

Different URL properties can be used for different kinds of whitelisting. To allow access to an individual web object, for example, to ensure users can download a particular file, the *URL* property is best used together with a list that contains the full URL for this file.

The following examples explain which URL properties are best used for different kinds of whitelisting and how to do it.

In addition to this, some tips and examples are given regarding the:

- Values that different URLs are set to when a sample URL is processed that has been sent to Web Gateway in a request for web access
- Use of the two operators *is in list* and *matches in list* in the criteria of a rule
- Good and bad entries in the lists that are used with different URL properties

Whitelisting individual web objects – URL

Goal	Allow users to access individual web objects. For example, download the file *Stinger.exe*, which can be accessed using the URL *http://download.mcafee.com/products/mcafee-avert/Stinger/Stinger.exe*.

How to do it	Use the *URL* string property with a list of full URLs in the criteria of a rule.

The rule could, for example, be configured as follows:

URL is in list URLWhiteList –> Stop Rule Set

If you add the URL *http://download.mcafee.com/products/mcafee-avert/Stinger/Stinger.exe* to the list *URLWhiteList*, the file *Stinger.exe* is whitelisted when the rule is processed.

Note:

In a similar way, you can block access to the file using the following rule from the default URL Filtering rule set:

URL matches in list URLBlockList –> Block

If you add the URL in question to the list *URLBlockList*, the file is blocked when the rule is processed.

If the *matches in list* operator is used instead of *is in list*, expressions containing wildcards can be entered into the list that is used by the property. The property can then also be used to whitelist multiple web objects.

However, if all web objects provided by a particular host should be whitelisted, this can be achieved more easily using the *URL.Host* property.

Whitelisting hosts – URL.Host

Goal	Allow users to access the web objects that are provided on particular hosts. For example, download the file *Stinger.exe* or any other file that is provided on the host *download.mcafee.com*.
How to do it	Use the *URL.Host* string property with a list for host names in the criteria of a rule.

A rule that the *URL.Host* property is used in could, for example, be configured as follows:

URL.Host is in list HostWhiteList –> Stop Rule Set

If you add the host *download.mcafee.com* to the list *HostWhiteList*, all web objects that are provided by this host are whitelisted when the rule is processed.

If the *matches in list* operator is used instead of *is in list*, expressions containing wildcards can be entered into the list that is used by the property. The property can then also be used to whitelist multiple hosts.

However, if all hosts within a particular domain should be whitelisted, this can be achieved more easily using the *URL.Host.BelongsToDomains* property.

Whitelisting domains – URL.Host.BelongsToDomains

Goal	Allow users to access the web objects that are provided within particular domains. For example, download the file *Stinger.exe* and any other file that is provided by the host *download.mcafee.com*, as well as any other downloadable file provided by any other host within the domain *mcafee.com*.
How to do it	Use the *URL.Host:BelongsToDomains* Boolean property with a list of domain names in the criteria of a rule.

The rule could, for example, be configured as follows:

URL.Host.BelongsToDomains("Domain List") equals true –> Stop Rule Set

If you add the domain *mcafee.com* to the list *Domain List*, all web objects within this domain are whitelisted when the rule is processed.

The list *Domain List* is configured as a parameter of the *URL.Host:BelongsToDomains* property, which is of the Boolean type.

When, for example, the URL *http://download.mcafee.com/products/mcafee-avert/Stinger/Stinger.exe* is processed, the value of the property (*true* or *false*) depends on whether the *mcafee.com* domain has been entered into the list *Domain List* or not.

The following example shows which entries in the list *Domain List* lead to a match when the property is used for whitelisting:

mcafee.com

dell.com

k12.ga.us

twitter.com

xxx

Then the criteria:

URL.Host.BelongsToDomains("Domain List") equals true

matches for the following URLs:

https://contentsecurity.mcafee.com

https://my.mcafee.com

http://my.support.dell.com

http://www.dekalb.k12.ga.us

http://twitter.com

http://www.twitter.com

any.site.xxx

but not for:

https://www.mymcafee.com

http://www.treasury.ga.us

http://malicioustwitter.com

Using the *URL.Host.BelongsToDomains* property also avoids the effort of creating more complicated solutions to achieve the same, for example:

- Using two entries in a list of wildcard expressions, such as:
  twitter.com
  *twitter.com
- Using a single, complex entry in a list of wildcard expressions, such as:
  regex((.*\.|.?)twitter\.com)

Property values for a sample URL

When the sample URL *http://www.mcafee.com/us/products/web-gateway.aspx* is processed, the URL properties below are set to different values as follows.

Property	Value for sample URL
URL	*http://www.mcafee.com/us/products/web-gateway.aspx*
URL.Host	*www.mcafee.com*
URL.Host.BelongsToDomain	*true* or *false* In the list that is configured as a parameter of this property, the following would have to be entered for the domain: *mcafee.com*.
URL.FileName	*web-gateway.aspx*
URL.Path	*/us/products/web-gateway.aspx*
URL.Protocol	*http*

Use of operators for different types of matches

It makes an important difference whether the *is in list* or *matches in list* operator is used in the criteria of a rule.

Operator	Description
is in list	Requires an exact string match. If there are wildcard characters in a list entry, they are interpreted as literal strings.
matches in list	Allows and evaluates wildcards in list entries.

Good and bad entries in lists for URL properties

Entries in the lists that are used by the different URL properties can be good are bad, according to how they fit in with the intended use of a property. The following are examples of good and bad list entries.

URL property	Good and bad list entries
URL with *is in list* operator	**Good** *http://www.mcafee.com/us/products/web-gateway.aspx* The full URL is entered, as it is required for this property. No wildcards are specified, as these are not evaluated when the *is in list* operator is used. **Bad** *www.mcafee.com/us/products/web-gateway.aspx* The entry does not specify the full URL, as the protocol information, *http://*, is not included.
URL with *matches in list* operator	**Good** _http://www.mcafee.com/*_ This entry contains a wildcard for allowing access to any web object provided by the host *www.mcafee.com*, which is appropriate when the *matches in list* operator is used. **Note:** The entry will not match for *http://mcafee.com/*. _regex(htt(p|ps)://(.*\.|\.?)mcafee.com(\/.*|\/?))_ This entry is more complex, as it uses regular expressions. When matched, it allows access, under the HTTP or HTTPS protocol, to any web object within the domain *mcafee.com* and its subdomains. _regex(htt(p|ps)://(.*\.|\.?)mcafee.(com|co.us)(\/.*|\/?))_ This entry is the same as the previous, but shows how other top-level domains, such as *.com* or *.co.us*, can be whitelisted. **Bad** _*.mcafee.com*_ The entry does not exclude unwanted matches, for example, a match for the URL *http://malicious-download-site.cc/malicious-file.exe?url= www.mcafee.com*.
URL.Host with *is in list* operator	**Good** *www.mcafee.com* A host name is entered, which fits in with the intended use for this property. No wildcards are specified, which is appropriate when the *is in list* operator is used. **Bad** *mcafee.com*

URL property	Good and bad list entries
	The entry specifies a domain name (_mcafee.com_), whereas the value of the property is a host name (_www.mcafee.com_ if, for example, the URL _http://www.mcafee.com/us/products/web-gateway.aspx_ is processed). No match will be produced this way. _*.mcafee.com_ The entry contains a wildcard, which is not evaluated when the _is in list_ operator is used. _*.mcafee.com/us*_ The entry includes path information (_/us_), which does not fit in with the intended use of the property. In addition to this, a wildcard is specified, which is not evaluated when the _is in list_ operator is used.
URL.Host with _matches in list_ operator	**Good** _*.mcafee.com_ The entry matches for on any host within the domain _mcafee.com_, but not for _mcafee.com_ itself. _regex((.*\.|\.?)mcafee.com)_ The entry uses regular expressions to whitelist the domain _mcafee.com_ and any of the hosts within it. **Bad** _*.mcafee.com*_ The entry does not exclude unwanted matches, for example, _http://www.mcafee.com .malicious-download-site.cc/_. _*.mcafee.com/us*_ The entry includes path information (_/us_), which does not fit in with the intended use of the property.
URL.HostBelongsToDomains	**Good** _mcafee.com_ entered in the list _Domain List_, which is configured as a parameter of the property. The entry matches for the _mcafee.com_ domain and all hosts within it, for example, _www.mcafee.com_ or _secure.mcafee.com_. _www.mcafee.com_ The entry does not specify a domain, but is valid. It only whitelists the host _www.mcafee.com_. **Note:** This can also be achieved by adding the entry to a list for the _URL.Host_ property used together with the _is in list_ operator. **Bad** _*.mcafee.com_ The entry contains a wildcard, which does not fit in with the intended use of the property. The property was rather developed to avoid the effort of using wildcards in list entries. Instead it requires an exact domain match, for example, a match for _mcafee.com_.

URL filtering using the Dynamic Content Classifier

URLs can be categorized for filtering by the Dynamic Content Classifier.

The Dynamic Content Classifier (DCC) is another source of category information with regard to URLs, in addition to the local database and the Global Threat Intelligence service.

You can configure use of the Dynamic Content Classifier when lookups for URL category information involving the other two sources yield no results.

Configure use of the Dynamic Content Classifier

You can configure use of the Dynamic Content Classifier for detecting URL categories when other methods of detection yield no results.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select a rule set with rules for URL filtering.
   In the default rule set system, this is, for example, the URL Filtering rule set.
   The rules appear in the settings pane.
3. Make sure Show details is selected.
4. Select the rule for handling URL categories that you want to configure use of the Dynamic Content Classifier for.

   In the URL Filtering rule set, this is, for example, the rule Block URLs whose category is in Category BlockList.
5. Click the settings of the URL Filter module in the rule criteria.

   In the sample rule, these are the Default settings in the criteria URL.Categories <Default> at least one in list Category BlockList.

   The Edit Settings window opens. It provides the settings of the URL Filter module.
6. Under Rating Settings, make sure Enable the Dynamic Content Classifier if GTI web categorization yields no results is selected.
7. [Optional] Edit the list of URL categories the Dynamic Content Classifier should detect.
   1. Above the list Categories that will be dynamically detected, click the Edit icon.
      The Edit window opens.
   2. Under DCC category, expand the Supported Categories folder.
   3. Select or deselect URL categories as needed.
   4. Click OK.

      The Edit window closes and the selected categories appear on the list.

      Note: You can remove a URL category from the list by clicking the Delete symbol and confirming in the window that opens.
8. Click OK to close the Edit Settings window.
9. Click Save Changes.

Results

The Dynamic Content Classifier is now involved in detecting whether a URL that is submitted in a request for web access falls into one of the configured URL categories.

Using your own URL filter database

URL filtering can be performed using information that is retrieved from a database of your own.

URL filtering on a Web Gateway appliance uses information about the categories that URLs fall into and the web reputation scores that are assigned to them. This information is retrieved from the local URL filter database, the Global Threat Intelligence system, or the Dynamic Content Classifier, depending on how the settings of the module for URL filtering are configured.

The information in the local database is the result of storing categories and web reputation scores there after they have been determined for particular URLs by the Global Threat Intelligence system. When a lookup in the local database yields no results, the other two information sources can additionally be used.

Instead of the local database, you can use a database of your own, containing information on URL categories and web reputation scores. To replace the local database, you need to specify the URL of the server that your database resides on when configuring the Central Management settings.

You can use your own database as the source that is searched first to retrieve URL filtering information, but also disable the other two sources and restrict the filtering process to using the information stored in your database.

Configure use of your own URL filter database

To retrieve URL filtering information from a database of your own, configure the use of this database as part of the Central Management settings.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance that should use your database information and click Central Management.
3. Scroll down to Advanced Update Settings.
4. In the Enter a special custom parameter for an update server field, enter the URL of the server that your database resides on.
5. Click Save Changes.

Results

When database information is used to filter URLs on the appliance, it is not retrieved from the local database, but from your own database.

You can additionally disable other sources of URL filtering information to restrict the filtering process to the information stored in your own database.

Restrict URL filtering to using database information

To use only database information for URL filtering, disable use of the Global Threat Intelligence system and the Dynamic Content Classifier.

If you configured the use of your own URL filter database, filtering information is retrieved only from this database.

Task

1. Select Policy → Settings.
2. Under Engines → URL Filter, select the URL Filter settings you want to disable information sources for.
3. Under Rating Settings, deselect the following two checkboxes one after another:
   - Enable the Dynamic Content Classifier if GTI web categorization yields no result
   - Use online GTI web reputation and categorization services if local rating yields no result
4. Click Save Changes.

Using a GTI Private Cloud service for URL filtering

When filtering URLs, you can configure Web Gateway to gather web reputation and category information through Global Threat Intelligence (GTI) lookups that are performed using your own cloud service.

This cloud service is also referred to as GTI Private Cloud service. It runs on-premise within your local network, where you configure and maintain it on your own, using a local database.

To perform web reputation and category lookups for URL filtering, the cloud service connects to a GTI server over a connection that is secured under HTTPS. Web Gateway is not involved when these lookups are performed.

When connecting to the GTI server, the cloud service uses server and client certificates that you create on your own. Server and client certificates of your own are also used when connecting to. Web Gateway. These certificates can be imported on the Web Gateway interface.

When connecting to the cloud service, Web Gateway uses the IP address of the server where the cloud service resides. The IP address of the default server that has been configured for performing GTI lookups is then overwritten.

URL filtering using an IFP proxy

URL filtering can be performed on requests to web access submitted under the IFP protocol.

To perform URL filtering on such requests, you need to:

- Set up an IFP proxy.
- Implement suitable filtering rules.

Filtering activities for IFP requests are displayed on the dashboard of the user interface. Connection tracing can also be performed for these activities.

Setting up an IFP proxy

To process and filter requests for web access that users submit from their client systems under the IFP protocol, the proxy functions of the appliance must be appropriately configured. An IFP proxy must be set up that intercepts these requests and makes them available for URL filtering.

To set up the proxy, you need to specify a number of settings on the user interface under Configuration → Proxies. These settings include:

- Enabling or disabling the proxy
- List of proxy ports, specifying for each proxy:
  - IP address and port number
  - Message mode (Indicates whether a block message is sent as a redirect or as normal message under the IFP protocol)
- Maximum number of concurrent IFP requests

  Using this setting, you can prevent an overloading of the IFP proxy.

Rules for filtering IFP requests

There is no default or library rule set for controlling the process of filtering IFP requests. However, you can create a rule set of your own and also make use of the IFP proxy functions in existing rule sets.

When creating a rule set for filtering IFP requests, you need to specify use of the IFP protocol as the rule set criteria to ensure the rule set is applied to requests that are submitted under this protocol. This is achieved by including the *Connection.Protocol* property in the criteria and configuring the IFP protocol as its operand.

As the IFP protocol covers only requests, you can exclude filtering responses and embedded objects as activities that the rule set should apply to.

The rules in the rule set can be the same as in the default URL Filtering rule set.

Tip: Best practice: If you want to perform URL filtering only for requests sent under the IFP protocol, delete the default URL Filtering rule set and use only the IFP filtering rule set that you have created in the way described here.

Using the IFP proxy functions in existing rule sets can be an option, for example, if you have authentication implemented for requests submitted under various other protocols and want to add authentication for IFP requests.

The Authentication Server (Time/IP-based Session) library rule set contains an embedded rule set with rules that check whether there is already an authenticated session for a client that a request is received from. Otherwise a rule redirects a request to the authentication server.

The embedded rule set covers protocols such as HTTP or HTTPS. Using the *Connection.Protocol* property, you can extend the criteria to include the IFP protocol.

Restrictions for IFP filtering

When using an IFP proxy for filtering URLs, you should be aware of the following restrictions:

- Limited use of SafeSearch Enforcer

  When performing IFP filtering, you the SafeSearch Enforcer will only work for filtering search requests that are carried out using Google.

  The reason for this is that only Google uses URLs for submitting the search criteria while all other search providers use cookies. However, cookies cannot be processed by the IFP proxy on an appliance.
- HTTP proxy required for some functions

  An HTTP proxy must be set up in addition to the IFP proxy if you want to do the following:

  - Redirect IFP requests that are blocked due to a filtering rule to a blocking page to let a block message appear on the client of the user who sent the request.
  - Authenticate users on the appliance by having their credentials verified on the internal authentication server.
  - Restrict web usage of users by implementing the Time Quota library rule set.

IFP filtering activities on the dashboard

The dashboard on the user interface provides information on several IFP filtering activities.

- Number of IFP requests processed

  This information is shown under Web Traffic Summary → Requests per protocol.
- Domains that access to was requested most often (counting the number of requests)

  Among these requests can be such that were submitted under the IFP protocol.

  This information is shown under Web Traffic → Top Level Domains by Number of Requests.
- Websites that were most often the destinations of requests (counting the number of requests)

  Among these requests can be such that were submitted under the IFP protocol.

  This information is shown under Web Traffic → Destinations by Number of Requests.

Connection tracing for IFP filtering activities

Connection tracing can be performed for filtering IFP requests.

When connection tracing is enabled, connection tracing files are created and stored. They can be accessed on the user interface under the Troubleshooting top-level menu.

Configure the IFP Proxy settings

You can configure the IFP proxy settings to set up a proxy that enables the processing of requests for web access submitted under this protocol.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, expand the appliance you want to configure the IFP proxy settings for and click Proxies (HTTP(S), FTP, ICAP, and IM).
3. On the settings pane, scroll down to the IFP Proxy section.
4. Configure the settings in this section as needed.
5. Click Save Changes.

Create a rule set for filtering IFP requests

You can create a rule set with rules that filter requests for web access submitted under the IFP protocol.

Task

1. Select Policy → Rule Sets, then click Add and select Rule Set.

The Add New Rule Set window opens.

2. In the Name field, enter a suitable name for the rule set, for example `Filter IFP Requests`.
3. Under Applies to, deselect Responses and Embedded Objects.
4. Under Apply this rule set, select If the following criteria is matched.
5. Configure the rule set criteria.
   a. Under Criteria, click Add and select Advanced criteria.
      The Add Criteria window opens
   b. From the properties list, select Connection.Protocol.
   c. From the operators list, select equals.
   d. In the input field for operands, type `IFP`.
   e. Click OK.
      The Add Criteria window closes and the criteria appears in the Criteria field.
6. Click OK.
   The Add New Rule Set window closes and the new rule set appears on the rule set tree.

What to do next

When the rule set has been created, you need to insert rules for URL filtering into it. You can, for example, copy rules from the default URL Filtering rule set and adapt them as needed.

Modify an authentication rule set to include the IFP protocol

You can include the IFP protocol in the criteria of an authentication rule set to enable authentication for requests that are submitted under that protocol.

Task

1. Import an authentication rule set from the library.
   a. Select Policy → Rule Sets , then click Add and select Top Level Rule Set.
      The Add Top Level Rule Set window opens.
   b. Click Import rule set from Rule Set Library.
      The Add from Rule Set Library window opens.
   c. From the Rule Set Library list, select the Authentication (Time/IP-based Session) rule set.
   d. In the Import conflicts area, select the conflict that is listed and under Conflict Solution choose a conflict-solving strategy.
   e. Click OK.
      The Add from Rule Set Library window closes and the rule set appears on the rule set tree.
2. Expand the rule set and select the embedded Check for Valid Authentication Session rule set.
   The criteria and rules of the embedded rule set appear on the settings pane.
3. Click Edit. The Edit Rule Set window opens.
4. Modify the rule set criteria.
   a. Under Criteria, click Add and select Advanced criteria.
   b. From the properties list, select Connection.Protocol.
   c. From the operators list, select equals.
   d. In the input field for operands, type `IFP`.
   e. Click OK.
      The Add Criteria window closes and the criteria appears in the Criteria field.
   f. Under Criteria combination remove the closing parenthesis after the letter e and insert one after d.
5. Click OK.
   The Edit Rule Set window closes.
6. Click Save Changes.

URL Filter settings

The URL Filter settings are used for configuring the URL Filter module, which handles activities related to URL filtering on a Web Gateway appliance.

Instances of the URL Filter settings include the following:

- Default settings — Default settings

  These settings are used when working with the default rule set for URL filtering. This rule set is named Default and nested within the URL Filtering rule set.
- Special URL Filtering Group settings — Settings used when working with the nested Special URL Filtering Group rule set

Extended List

Settings for extended lists

Extended List

Option	Definition
Use the extended list	Provides a list for selecting an extended list.
Add	Opens the Add List window for adding an extended list.
Edit	Opens the Edit List (Extended List) window for editing the selected extended list.

Rating Settings

Settings for retrieving rating information on URLs based on categories and reputation scores

Rating Settings

Option	Definition
Search the CGI parameters for rating	When selected, CGI parameters are included in the search for information. CGI (Common Gateway Interface) parameters in a URL trigger scripts or programs when the URL is accessed. Information on CGIs is considered when categorizing a URL.
Search for and rate embedded URLs	When selected, embedded URLs are included in the search for information and rated. Information on an embedded URL is considered when categorizing the embedding URL. **Note:** Searching for embedded URLs can impact performance.
Do a forward DNS lookup to rate URLs	When selected, a DNS lookup is performed for a URL that no relevant information has been found for. The IP address that was looked up is used for another search.
Do a backward DNS lookup for unrated IP-based URLs	When selected, a backward DNS lookup, based on its IP address, is performed for a URL that no relevant information has been found for. The host name that was looked up is used for another search.
Use the built-in keyword list	When selected, the built-in keyword list is included in the search.

Option	Definition
Disable local GTI database	When selected, no information about web reputation and categories is retrieved from the local Global Threat Intelligence database..
Use online GTI web reputation and categorization services if local rating yields no result	When selected, information on URL categories and reputation scores is only retrieved from the Global Threat Intelligence service if the search in the internal database yielded no results.
Use default server for online GTI web reputation and categorization services	When selected, the appliance connects to the default server for retrieving information on URL categories and reputation scores from the Global Threat Intelligence system. • IP of the server — Specifies the IP address of the server used to connect to the Global Threat Intelligence system when the default server is not used. Format: <domain name> or <IPv4 address> or <IPv4 address mapped to IPv6 address> Regular IPv6 addresses cannot be specified here. • Port of the server — Specifies the port number of the port on this server that listens to requests from the appliance. Allowed range: 1–65535
Enabke the Dynamic Content Classifier if GTI web categorization yields no result	When selected, the Dynamic Content Classifier is involved in the URL filtering process if a search performed by the Global Threat Intelligence service yielded no results.

Advanced Settings

Advanced settings for the URL Filter module

Advanced Settings

Option	Definition
Treat connection problems to the cloud as errors	When selected, problems arising on the connection from the appliance to the Global Threat Intelligence server are logged as errors. Properties for error handling are set and eventually rules from an Error Handler rule set are executed.
Do a backward DNS lookup also for private addresses	When selected, private IP addresses are included in the backward DNS lookup. Excluding these addresses from the lookup leads to an increase in performance for URL filtering. This option is disabled by default. The lookup includes the following types of addresses: • IPv4 ◦ Private addresses ◦ Zeroconf addresses • IPv6 ◦ Link local addresses ◦ Site local addresses ◦ Unique local addresses

Proxy Settings

Option	Definition
Use upstream proxy	When selected, the appliance uses a proxy for connecting to the Global Threat Intelligence server on which lookups for URL category information, also known as "in-the-cloud" lookups, can be performed.
IP or name of the proxy	Specifies the IP address or host name of the proxy.
Port of the proxy	Specifies the number of the port on the proxy that listens for lookup requests from the appliance.
User name	Specifies a user name for the appliance when logging on to the proxy.
Password	Sets a password for an appliance.
Set	Opens a window for setting a password.
Connect to GTI cloud via host name also when a proxy is configured	When selected, Web Gateway connects to a cloud service for performing GTI lookups using the host name of the server where the cloud service resides, regardless of whether a proxy is also configured..
Try to bypass the proxy if unreachable	When selected, Web Gateway tries to bypass a proxy that has been set up if this proxy cannot be reached.
Trust server certificate	When selected, a certificate sent under HTTPS by a cloud service for performing GTI lookups is trusted on Web Gateway. • Subject, Issuer, Validity, Extensions, Fingerprint, Private Key — Provide information about the certificate that is sent by the cloud service.. • Import — Opens a window for importing a server certificate..
Provide client certificate	When selected, Web Gateway provides a certificate when connecting as a client under HTTPS to a cloud service for performing GTI lookups. • Subject, Issuer, Validity, Extensions, Fingerprint, Private Key — Provide information about the certificate that Web Gateway sends to the cloud serviice. • Import, Export, Export Key — Open windows for importing a client certificate and for exporting a client certificate and key.

Logging

Option	Definition
Enable logging	When selected, URL filtering activities are logged on the appliance. If this option is not selected, the following logging options are grayed out.

Option	Definition
Log level	Provides a list for selecting the log level. Log levels are as follows: • 00 FATAL — Logs only fatal errors. • 01 ERRORS — Logs all errors. • 02 WARNING — Logs errors and warnings. • 03 INFO — Logs errors, warnings, and additional information. • 04 DEBUG1 ... 013 DEBUG9 — Log information required for debugging URL filtering activities. The amount of logged information increases from level DEBUG1 to DEBUG9. • 14 TRACE — Logs information required for tracing URL filtering activities. • 15 ALL — Logs all URL filtering activities
(Log area)	Provides a set of options for including different areas of URL filtering activities into the logging. • LOG_AREA_ALL — When selected, all URL filtering activities are logged. • LOG_AREA_NETWORK — When selected, activities regarding the network connections used for URL filtering are logged. • LOG_AREA_DATABASE_SEARCH — When selected, activities regarding the retrieval of data for URL filtering from the internal database are logged. • LOG_AREA_DNS — When selected, activities regarding a DNS lookup that is performed for URL filtering are logged. • LOG_AREA_URL — When selected, activities for handling URLs, such as parsing them, are logged. • LOG_AREA_CLOUD — When selected, activities regarding the retrieval of information from the Global Threat Intelligence system are logged.

Cloud Settings

Option	Definition
Connection count (maximum)	Limits the number of connections that can be active at the same time. Maximum number of connections by default: 4
Request timeout	Limits the time between retries of requests on a connection. Maximum time by default: 2000 ms
Request attempts	Limits the number of retries. Maximum number of retries: 3

Troubleshooting

Settings for troubleshooting issues with URL filtering

Air-Gap Mode Setting

Option	Definition
Automatic air-gap mode	An automatic air-gap mode can be enabled for connections from a Web Gateway appliance to a Global Threat Intelligence (GTI) server when issues impacting response time arise. Enabling this mode prevents increased response times on GTI server connections from creating overload issues elsewhere, for example, on the anti-malware or the proxy working queue. Traffic resulting from queries sent to and received from the GTI server is reduced in air-gap mode to the minimum that is required to monitor response times in order to recognize a return to normal. When a return to normal is recognized, the automatic air-gap mode is disabled. What is considered a normal response time here can be configured. While the automatic air-gap mode is enabled, information about URL categories and reputation scores can still be retrieved from the local database on Web Gateway. Monitoring functions can be enabled with or without the automatic air-gap mode. The following can be selected for the automatic air-gap mode: • Off — When selected, no monitoring is performed on GTI server connections and the automatic air-gap mode is never enabled. This option is selected by default. • Monitor only — When selected, GTI server connections are monitored, but the automatic air-gap mode is still never enabled. When these connections are monitored, issues impacting response time are logged like this: ◦ When the maximum average response time exceeds a configured threshold as long as or longer than a time interval that is also configured, a warning message is logged, as a possible trigger to taking appropriate measures. ◦ When response times return to normal again, falling below the threshold as long as or longer than configured, an information message is logged. Default values are configured for the threshold and the time intervals. You can modify these values to adapt them to your network conditions. • Active — When selected, GTI server connections are monitored and the automatic air-gap mode is enabled and disabled depending on how response times on these connections develop. The configured threshold and time intervals are then evaluated for both enabling the air-gap mode and logging warnings and information messages.
Maximum average delay threshold	Sets a threshold value that marks the acceptable maximum average response time (in ms) on connections to a GTI server.

Option	Definition
	Default: 250 ms
Retention time enable air gap	Sets the time interval (in seconds) over which the average response time on GTI server connections must exceed the configured threshold before a warning message is logged and the automatic air-gap mode is enabled if available and activated. Default: 10 seconds
Retention time disable air gap	Sets the time interval (in seconds) over which the average response time on GTI server connections must fall below the configured threshold before a back-to-normal message is logged and the automatic air-gap mode is disabled if previously enabled. Default: 120 seconds
Probing rate if enabled	Sets the percentage of requests for web access submitted by users for which queries are sent to a GTI server to a minimal value that applies when the automatic air-gap mode is enabled. Keeping a minimal amount of traffic on the connections to the GTI server is required to monitor this traffic in order to recognize when response times return to normal, so the automatic air-gap mode can be disabled. Default: 1 %

Media type filtering

Media type filtering ensures that the users of your network can only access media belonging to types that are allowed under your web security policy. For example, access to streaming media might not be allowed because it consumes too many resources.

A default process for media type filtering is implemented on Web Gateway after the initial setup. This process includes the following important configuration item:

- Media Type Filtering **rule set** — Default rule set for media type filtering

The default process requires that you maintain lists of media types, which are used by the rules in the rule set for allowing and blocking access to particular media types.

You can further modify this process to meet the requirements of your organization. You can also extend the process in several ways or set up your own process.

Media type filtering process

The media type filtering process includes several elements that contribute to it in different ways.

- **Filtering rules** — Control the process. There are usually the following types of rules.
    - **Blocking rules** — Block access to media types.

      The rule applies if a user requests access to media of a type that is on a blocking list.
    - **Whitelisting rules** — Exclude web objects from further media type filtering to ensure they can be accessed by the users in your network.

      Whitelisting rules are placed before the blocking rule in an media type filtering rule set. If a whitelisting rule applies, processing of the following media type filtering rules is stopped to ensure that the blocking rule is not executed.

  A media type filtering rule can use a list of media types in its criteria. It can also use a suitable property there, such as MediaType.IsAudio or MediaType.IsVideo .
- **Blocking lists** — List web objects that access is blocked for.

  There can be a blocking list for media that should not be uploaded from within your network to the web, as well as one for media that should not be downloaded from the web to your network.
- **Whitelists** — List web objects that are excluded from further media type filtering.

Media type filtering administration

When administering the media type filtering process, you can use several configuration items that are available by default.

- Media Type Filtering **rule set** — Default rule set for media type filtering

  This rule set has two nested rule sets:
    - Upload Media Types **rule set** — Nested rule set for filtering uploads of media that are performed by the users in your network.
    - Download Media Types **rule set** — Nested rule set for filtering downloads of media that are performed by the users in your network.
- **Blocklists** — Used to block access to media types
    -

Upload Media Type Blocklist — Lists media types. Use this list to block uploads of media belonging to these types. The list is empty by default and you need to fill the entries.

- Download Media Type Blocklist — List media types. Use this list to block download of media belonging to these types. The list is empty by default and you need to fill the entries.

You can also create your own rule set and lists for media type filtering.

Configure key elements for media type filtering

Configure key elements of the rules for media type filtering to adapt important parts of the filtering process to the requirements of your web security policy.

Task

1. Select Policy → Rule Sets.
2. On the rule set tree, select the Media Type Filtering rule set.
   Key elements of the rules for the filtering process appear in the configuration pane.
3. Configure the key elements as needed.
4. Click Save Changes.

Configure media type filtering using the complete rules view

You can configure media type filtering to adapt this process to the needs of your network.

To configure URL filtering, you can work with the key elements view or the rules view.

Task

1. Review the rules in the rule set for URL filtering.
   By default, this is the *URL Filtering* rule set.
2. Modify these rules as needed.
   You can, for example, do the following.
   - Enable or disable blocking rules and the whitelist rule
   - Edit the lists used by these rules
     Note: A yellow triangle next to a list name means the list is initially empty and you need to fill the entries.
   - Modify the settings of the URL Filter module
3. Save your changes.

Modify a media type filtering rule

You can modify a media type filtering rule to filter a different kind of media types by changing the property in the rule criteria. Then you also need to create a new filter list for use by the modified rule.

Create a filter list for a modified rule

You can create a new filter list for use in a modified media type filtering rule.

Task

1. Select Policy → Lists.
2. On the Custom Lists branch of the lists tree, select Media Type and click Add.

   The Add List window opens.
3. In the Name field, type a name for the new list, for example, `Not Ensured Download Media Type Blocklist`.
4. [Optional] In the Comment field, type a plain-text comment on the new list.
5. [Optional] Click the Permissions tab and configure who is allowed to access the list.
6. Click OK.

   The Add List window closes and the new list appears on the lists tree under MediaType.

Results

You can now fill the entries for the new list to let the media type filtering rule know what to block or allow.

Replace a property in a media type filtering rule

You can replace the property in the criteria of a media type filtering rule with a different property to let the rule filter a different kind of media types.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select a rule set for media type filtering, for example, the nested Download Media Type rule set in the Media Type Filtering rule set.
3. Select a rule, for example, Block types from Download Media Type Blocklist, and click Edit.

   The Edit Rule window opens with the Name step selected.
4. Click Rule Criteria and under Criteria select the rule. Then click Edit.

   The Edit Criteria window opens.
5. Edit the rule criteria as follows:
   a. From the list of properties in the left column, select a new property, for example, MediaType.NotEnsuredTypes (instead of *MediaType.EnsuredTypes*).
   b. From the list of operands in the right column, select Not Ensured Download Media Type Blocklist.
6. Click OK.

   The window closes and the modified criteria appears under Rule Criteria.
7. Click Finish.

   The Edit Rule window closes and the modified rule appears within the nested rule set that you selected..
8. Click Save Changes.

Streaming media filtering

Streaming media filtering ensures that media of this type is detected when it is received on Web Gateway and handled according to your web security policy.

You might, for example, want to block access to streaming media to avoid excessive bandwidth consumption.

No default process for streaming media filtering is implemented on Web Gateway after the initial setup, but you can set up your own process.

Important configuration items to be used in this process include:

- StreamDetector.IsMediaStream **property** — Boolean property that is set to *true* when a web object is recognized as streaming media in the filtering process
- Default Streaming Detection **settings** — Default settings for the Stream Detector module, which evaluates web objects and calculates the probability that they are streaming media

When setting up your own process, you can use these items in rules that you insert in an already existing or a newly created rule set.

Process for streaming media filtering

A process for filtering streaming media is based on rules like all other filtering processes that run on Web Gateway.

The most important part of this process is the detection of streaming media among the web traffic that is filtered. Streaming media is usually detected in the response cycle of the filtering process when it is received from web servers that sent it in response to user requests.

The detection of streaming media is the job of the Stream Detector module. This module uses URL categories, content-type headers, source IP addresses, and other information to calculate the probability that a web object is streaming media.

The module is capable of performing this calculation for a large number of streaming media types.

The module is triggered when a rule with the Boolean StreamDetector.IsMediaStream property in its criteria is processed. It sets this property to *true* when the calculated probability reaches or exceeds a given value. You can configure this value in the settings for the Stream Detector module.

When streaming media is detected, suitable rule actions can block or allow access to it. You can, for example, use these actions to:

- Block access to streaming media to avoid excessive bandwidth consumption
- Allow access to streaming media chunk-by-chunk after scanning each chunk for malware

Scanning streaming media and allowing access to it chunk-by-chunk is the job of the Media Stream Scanner, which is a component of the Anti-Malware filtering module. The scanning begins after the Stream Detector has detected that a web object is streaming media.

The default Gateway Anti-Malware rule set contains a rule for enabling this workflow.

Administration of streaming media filtering

To perform streaming media filtering on Web Gateway, you must set up this process on your own because there is no default process.

Using several default configuration items, you can, for example, create rules that block or allow access to streaming media.

Tip: Do not create a separate rule set for streaming media filtering, but insert the rules for this process n suitable other rule sets, for example, in the Media Type Filtering rule set.

Streaming media filtering is usually performed in the response cycle of the filtering process on Web Gateway, where streaming media is received that web servers send in response to user requests. A suitable rule set for use in this cycle and for inserting rules for streaming media filtering into is Download Media Type, which is nested in the Media Type Filtering rule set

Configuration items you can use to create a process for streaming media filtering include:

- StreamDetector.IsMediaStream **property** — Boolean property that is set to *true* when a web object is processed and detected as streaming media

  When this property is set to *true*, related values are set for two additional properties:

  - StreamDetector.Probability — Probability that a web object is streaming media, for example, 60 or 70 percent
  - StreamDetector.MatchedRule — Name of the rule that was processed with the result that streaming media was detected

  You can insert the additional properties, for example, in logging rules to record what happens during the process for streaming media filtering.

- Default Streaming Detection **settings** — Default settings for the Stream Detector module, which calculates the probability that a given web object is streaming media, and sets the StreamDetector.IsMediaStream property accordingly

  These settings include an option to configure the percentage for the probability that must be reached to recognize a web object as streaming media.

The default Gateway Anti-Malware rule set contains a rule that uses the StreamDetector.IsMediaStream property to find out whether a web object is streaming media.

The rule eventually enables the Media Stream Scanner, which scans this media and allows access to it chunk-by-chunk, as long as no malware is detected.

Rules for streaming media filtering

You can create rules for streaming media filtering, for example, to block or allow web objects that belong to this media type.

These rules use the StreamDetector.IsMediaStream property to find out whether a web object is streaming media.

The following rule blocks access to streaming media:

Name		
Block access to streaming media		
Criteria		Action
StreamDetector.IsMediaStream<Streaming Detection> equals true	–>	Block<Streaming Media Blocked>

This rule allows access to streaming media:

Name		
Allow access to streaming media		
Criteria		Action
StreamDetector.IsMediaStream<Streaming Detection> equals true	–>	Continue

Set up a process for streaming media filtering

You can set up your own process to filter streaming media using several default configuration items.

Task

1. Create a rule that blocks web objects if the probability that they are streaming media reaches or exceeds a configured value.
2. Insert this rule in a suitable rule set, for example, in a media type filtering rule set.

3. Save your changes.

List of supported streaming media types

The Stream Detector module on Web Gateway detects streaming media among the web objects that are filtered.

The types of streaming media that can be detected include:

- Flash-based videos
- HLS streams
- ICY-based streams
- MP3 streams
- MS-WMSP
- Multipart streams
- Real media
- RTMP-based streams
- Silverlight-based videos
- WebM media
- YouTube

Best practices - Configuring streaming media scanning

You can perform a special kind of scanning when the Stream Detector has found that a web object is streaming media.

Anti-malware filtering on Web Gateway usually requires that web objects are completely downloaded and scanned by the Anti-Malware module. But completeness can never be achieved for streaming media, so the usual scanning method will not deliver results, but delay processing of this media type endlessly.

Streaming media must therefore be handled in a special way. Two modules on Web Gateway are available for this:

- The Stream Detector module detects that a web object is streaming media.
- The Media Stream Scanner, which is a component of the Anti-Malware module scans streaming media chunk-by-chunk.

Compared to the usual method, the Media Stream Scanner performs a less intensive way of scanning.

Following the progress made by the Media Stream Scanner, streaming media is delivered chunk-by-chunk to the client that requested its download. If an infection is detected in a chunk, the process is stopped, and this chunk and the rest of the streaming media are not delivered.

A suitable rule calls both components to perform their jobs. It is contained in the default Gateway Anti-Malware rule set.

The rule is not available in older versions of McAfee Web Gateway. So we recommend the following:

- Inspect your rule set system.
- If the rule is not included in the default Gateway Anti-Malware rule set or any other rule set you are using for anti-malware filtering, create the rule in one of these rule sets.

  Make sure you place it immediately before the rule that triggers the usual anti-malware scanning.

Rule for detecting and scanning streaming media

The following rule of the default Gateway Anti-Malware rule set detects streaming media and enables the Media Stream Scanner for scanning this media:

Name
Start Media Stream Scanner on streaming media and skip anti-malware scanning

Criteria	Action	Event
Cycle.Name equals "Response" AND		
StreamDetector.IsMediaStream<Default Streaming Detection> equals true	Stop Ruleset –	Enable Media Stream Scanner

In its rule set, this rule is placed immediately before the rule that triggers the usual anti-malware scanning.

When the Stream Detector finds that a web object is streaming media, the rule stops processing for this rule set and starts the Media Stream Scanner, so the special method of scanning streaming media is performed and the rule for the usual scanning is skipped.

The criteria part with the Cycle.Name property ensures that the rule only applies in the response cycle of processing when web objects are received on Web Gateway from the web, in response to a request that was forwarded.

Settings for the Stream Detector

The settings for the Stream Detector module can be accessed on the settings tree under Stream Detector. The name of the default settings is Default Streaming Detection.

The default settings include only this option:

Minimal probability — Sets the probability of being streaming media that is sufficient for recognizing a web object as streaming media.

- The probability is measured in percent and configured as a number from 1 to 100.
- The probability is found by the Stream Detector. If the minimal probability is reached, the StreamDetector.IsMediaStream property, which is used in the default rule for streaming media filtering, is set to *true*.
- The default minimal probability is 60. We recommend leaving this value unchanged.

Application filtering

Application filtering ensures that the users of your network cannot access unwanted applications, which could be, for example, Facebook, Xing, and others. The filtering process application names and reputation scores and blocks access accordingly. Filtering can also be applied to individual functions of applications.

The following elements are involved in this process:

- Filtering rules that control the process
- Application lists that are used by rules to block applications
- Application system lists that are updated in intervals

Update status and statistics of the application filtering process are shown on the dashboard.

Rules for application filtering

The rules that control application filtering are usually contained in one rule set. They block access to applications and individual functions of applications using the following two methods:

- Block applications and individual functions that are on a list
- Block applications that are assigned a particular risk level

To block applications and individual functions according to a list, the *Application.Name* property is used.

The value of this property is the name of an application or an individual function of an application that appears in a request sent by a user who wants to access the application or application function. If this name is on a blocking list, access is blocked, as, for example, the following rule does it.

Name		
Block applications according to list		
Criteria		Action
Application.Name is in list Unwanted Applications	–>	Block<Application Blocked>

To block applications according to their risk levels, properties, such as *Application.IsMediumRisk* or *Application.IsHighRisk* are used, which have *true* or *false* as their values.

Risk evaluation is based on the reputation score for an application that is assigned to it by the Global Threat Intelligence system. If the risk for allowing access to an application is considered to be high, it means it has a bad reputation.

If an application reaches or exceeds this level, access to it is blocked, as, for example, the following rule does it.

Name		
Block high-risk applications		
Criteria		Action
Application.IsMediumRisk equals true OR Application.isHighRisk equals true	–>	Block<Application Blocked>

Both methods rely on the application system lists. Only applications and application functions that are on these lists can appear on a list that is used by an application filtering rule.

The risk levels for applications and application functions are also those that are shown on the application system lists.

For logging purposes, there are the *Application.To String* and *Application.Reputation* properties, which are the name of a requested application converted into a string and a numerical value for its reputation score, respectively.

You can use these properties in rules that record information in log file entries.

Application filtering is not performed by default on an appliance. However, you can import the *Application Control* rule set from the library.

You can then review the rules in this rule set, modify or delete them, and also create your own rules.

Blocking lists

Blocking lists are used by rules to block access to applications that are requested by users. The rules in the library rule set include lists that are already filled with several application names.

You can add application names to a list from the library rule set or remove them and also create your own lists. If you add application names, you must take them from the application system list.

In the same way, you can create and edit lists with names of application functions.

Application system lists

The applications and application functions that can be blocked by application filtering rules appear on lists, which are provided by the appliance system and updated in intervals.

You can view these lists by expanding the *Application Name* folder under *System Lists* on the lists tree of the Lists tab. This folder contains a number of subfolders for different types of applications, for example, *File Sharing* or *Instant Messaging*.

A subfolder contains a list of applications, providing the following information for each of them:

- Application name (or application name with application function)
- Comment
    - Risk level
    - Description of the application (or application function)

A function of an application appears in parentheses after the application name, for example, *Orkut(Orkut Chat)*. If you include an application function in the list of a blocking rule, only this function is blocked, not the complete application.

The following is an example of an entry for an application in a system list:

MessengerFX | Risk: Minimal: A web-based instant messaging service

The next example shows an entry for an application function:

Orkut(Orkut Chat) | Risk: High: Allows users to send instant messages.

Application filtering information on the dashboard

The dashboard provides the following information on application filtering:

- Update status of the application list
- Statistics on applications and application functions that have actually been blocked

Configure application filtering

You can configure application filtering to adapt this process to the needs of your network.

Complete the following high-level steps.

Task

1. Import the *Application Control* rule set.
2. Review the rules in this rule set and modify them as needed.

   You can, for example, do the following.

   - Enable or disable blocking rules
   - Edit the lists used in rules by adding or removing applications
   - Create lists of your own and use them instead of or in addition to the existing lists
   - Change the reputation levels used in rules by replacing the relevant properties, for example, by replacing *Application.IsHighRisk* with *Application.IsMediumRisk*

You can also create blocking rules of your own.

3. Save your changes.

Create a list for application filtering

You can create a list for use in an application filtering rule and fill it with entries for applications or individual functions of applications that should be blocked.

Task

1. Select Policy → Lists and click the Add icon.
   The Add List window opens.
2. Configure general list settings.
   a. In the Name field, type a name for the list, for example, `Unwanted Applications`.
   b. From the Type list, select Application Name.
   c. [Optional] Click the Permissions tab and configure who is allowed to access the list.
   d. [Optional] In the Comments field, type a plain-text comment on the list.
3. Click OK.
   The Add List window closes and the list appears on the list tree under Custom Lists → Application Name.
4. Select the list and, above the settings pane, click the Edit icon.
   The Edit window opens with a collection of folders that contain application names.
5. Fill the list with entries for applications or individual functions of applications.
   a. Expand a folder that contains an application or application function that name you want to add to the list, for example, Instant Messaging Web Applications.
   b. Select an application or application function, for example, MessengerFX or Orkut(Orkut Chat).
   Note: You can select multiple applications or application functions at once, you can select items from multiple folders at once, and you can select complete folders.
   c. Click OK.
   The Edit window closes and the selected applications and application functions appear on the list.
   Note: You can also add a complete folder and afterwards delete the entries for applications or application functions that you do not want to include.
6. Click Save Changes.

Results

You can use the list you created in the criteria of an application filtering rule, for example, to let the criteria match if the name of an application or application function that access is requested to appears on the list.

Modify the risk level in an application filtering rule

You can modify the risk level in a rule that filters applications according to the risk they present to web security, for example, from high to medium. This increases web security because a blocking action can then be triggered even if an application is only a medium risk.

Before you begin

The following procedure assumes that you have imported the Application Control rule set from the library.

Task

1. Select Policy → Rule Sets.
   The Add New Rule Set window opens.
2. Expand the Application Control rule set, then expand the nested Block Applications in Request Cycle rule set.
   The general settings and rules of the nested rule set appear on the settings pane.

3. Make sure Show details is selected.
4. Select the rule Block web applications with high risk and click Edit.
   The Edit Rule window opens.
5. Under Steps, select Rule Criteria and in the Criteria section, select the upper part of the complex criteria (the one that uses the Application.IsHighRisk property), then click Edit.
   The Edit Criteria window opens with the Application.IsHighRisk property selected in the properties list.
6. From the properties list, select Application.IsMediumRisk.
7. Click OK.
   The Edit Criteria window closes and the modified criteria appears in the Criteria section.
8. Click Finish.
   The Edit Rule window closes and the rule with the modified criteria appears on the settings pane.
9. Click Save Changes.

Global whitelisting

Global whitelisting ensures that all further filtering is skipped for the web objects that are whitelisted, so access to them cannot be blocked.

The global whitelisting process includes several elements, which contribute to it in different ways.

- Filtering rules control the process.
- Whitelists are used by rules to let some web objects skip further filtering.

A default process for global whitelisting is implemented on Web Gateway after the initial setup. You can modify this process to adapt it to the requirements of your web security policy.

Filtering rules

The rules that control global whitelisting are usually contained in one rule set.

Whitelisting rules are placed and processed in this rule set. If any of them applies, the following rule sets are skipped and no further filtering is performed for the whitelisted objects.

You can review these rules, modify or delete them, and also create your own rules.

When the default rule set system is implemented, a rule set for global whitelisting is included. Its name is *Global Whitelist*.

Whitelists

Whitelists are used by whitelisting rules to let particular web objects skip further filtering. There can be whitelists for URLs, media types, and other types of objects.

You can add entries to these lists or remove entries. You can also create your own lists and let them be used by the whitelisting rules.

Configure global whitelisting

You can configure global whitelisting to adapt this process to the needs of your network.

Complete the following high-level steps.

Task

1. Review the rules in the rule set for global whitelisting.
   By default, this is the *Global Whitelisting* rule set.
2. Modify these rules as needed.
   You can, for example, do the following:
   - Enable or disable whitelisting rules
   - Edit the lists used by the whitelisting rules
     Note: A yellow triangle next to a list name means the list is initially empty and you need to fill the entries.
   - Create whitelists of your own and let them be used by the whitelisting rules
3. Save your changes.

HTTPS scanning

HTTPS scanning ensures that SSL-secured web traffic can be processed and made available to other filtering functions on Ó. This scanning mode is also known as SSL scanning.

The HTTPS or SSL scanning process includes several elements, which contribute to this it in different ways.

After the initial setup, the following configuration items are available on Web Gateway for running and controlling an HTTPS scanning process:

- HTTPS Scanning **rule set** — Default rule set for HTTPS scanning

  This rule set is part of the default rule set system, but it is not enabled by default. You can enable this rule set and also modify it to meet the requirements of network.

  You can also extend the HTTPS scanning process or create your own process.
- SSL Scanner **settings** — Default settings for the SSL Scanner module, which handles important parts of the HTTP scanning process, such as certificate verification and content inspection.

  More settings for modules that handle certificates used in SSL-secured traffic are also available.

To modify or extend the HTTPS scanning process, or to create your own process, you can use these items as a starting point.

HTTPS scanning process and administration

The HTTPS scanning process ensures that SSL-secured web traffic can be processed and made available to other filtering functions. As an administrator, you can use several configuration items to modify this process.

HTTPS scanning rules

The rules that control HTTPS scanning are usually contained in one rule set that has several nested rule sets. Each of the nested rule sets controls a particular function of the process:

- **Handle the CONNECT call** — There is a rule set with rules for handling the CONNECT call, which is sent at the beginning of SSL-secured communication under the HTTPS protocol.
- **Verify certificates** — There are rule sets for verifying certificates that are submitted by clients and servers in SSL-secured communication, for example, by verifying the common names in these certificates.

  This part of the process allows verification for both explicit proxy and transparent setups.
- **Enable content inspection** — Another rule set contains rules for enabling the inspection of content that is transferred in SSL-secured communication.

To find out whether an object is infected, the rule calls the Anti-Malware module, which scans the object and lets the rule know about the result.

Whitelisting rules can be placed and processed in this rule set before the blocking rule. If any of them applies, the blocking rule is skipped and the whitelisted objects are not scanned.

You can review the rules that are implemented on the appliance for HTTPS scanning, modify or delete them, and also create your own rules.

When the default rule set system is implemented, a rule set for HTTPS scanning is included. Its name is HTTPS Scanning. However, the rule set is not enabled initially.

Whitelists and other lists for HTTPS scanning

Whitelists are used by the HTTPS scanning rules to let web objects skip parts of the process. For example, a certificate whitelist exempts certificates from undergoing verification.

Other lists used in HTTPS scanning contain the port numbers that are allowed in CONNECT calls if these are to be accepted or the servers that require a special kind of certificate verification because a particular method of exchanging keys cannot be applied on them.

You can add entries to these lists or remove entries. You can also create your own lists and let them be used by the SSL scanning rules.

Modules for HTTPS scanning

The following modules (also know as *engines*) are called by the HTTPS scanning rules to perform different parts of the SSL scanning process:

- **SSL Scanner** — Handles certificate verification or the enabling of content inspection, depending on the settings it runs with.

  Accordingly, the module is called by the rules for certificate verification and content inspection with different settings.
- **Modules for setting the client context** — Handle the submitting of a certificate for the appliance to the clients that send requests to it in SSL-secured communication.

  When this certificate is submitted, the Certificate Authority (CA) that issued the certificate can be sent with it or not. Accordingly, there is a module for submitting a certificate *with* and another module for submitting a certificate *without* its certificate authority.

  The HTTPS Scanning (SSL Scanner) rule set of the default system, uses the method of submitting a certificate with its certificate authority.

  Tip: Best practice: Replace the default certificate authority that is provided for use after the initial setup with a certificate authority of your own for further use.
- **Certificate Chain** — Handles the building of a certificate chain

  When building the chain, the module uses a list of certificate authorities for the certificates that are included in the chain. You can add certificate authorities to existing lists and also add new lists.

Configure HTTPS scanning

You can configure HTTPS scanning to adapt this process to the needs of your network.

Complete the following high-level steps.

Task

1. Enable the rule set for HTTPS scanning and review the rules in this rule set.

   By default, this is the *HTTPS Scanning (SSL Scanner)* rule set.
2. Modify these rules as needed.

   You can, for example, do the following:

   - Replace the default root Certificate Authority (CA) for signing certificates that the appliance sends to its clients by a certificate of your own.

     This can be a certificate authority that you create yourself on the user interface or one that you import from your file system.
   - Enable or disable whitelisting rules, for example:
     - The default rule for skipping certificate verification when a certificate that was submitted by a client is on a whitelist
     - The default for skipping content inspection when the host of a requested URL is on a whitelist
   - Edit the lists used by the whitelisting rules

     Note: A yellow triangle next to a list name means the list is initially empty and you need to fill the entries.
   - Create whitelists of your own and let them be used by the whitelisting rules
   - Modify the settings of the modules involved in HTTPS scanning.
     - SSL Scanner module
     - SSL Client Context module
     - Certificate Chain module
3. Save your changes.

Configure the modules for HTTPS scanning

You can configure settings for the modules (engines) that are involved in HTTPS scanning to modify the way SSL-secured web traffic is processed.

The modules for HTTPS scanning include:

- SSL Scanner module
- SSL Client Context with CA module
- SSL Client Context without CA module
- SSL Client Certificate Handling module
- Certificate Chain module

There are several ways of accessing the settings for these modules on the user interface.

Task

1. Access the settings for the HTTPS scanning modules that you want configure in one of the following ways:
   - Select Policy → Settings. Then select the settings you want to configure on the Engines branch of the settings tree.the settings tree.

     On this branch, the settings appear below their modules (engines), which are arranged in alphabetical order.

     For example, the Default CA settings for the SSL Client Context with CA module appear below this module.
   - Select Policy → Rule sets. Then select a rule set on the rule sets tree with rules that use the HTTPS scanning modules to process web traffic.

     For example, the HTTPS Scanning rule set uses the SSL Client Context with CA module to set a client context for HTTPS scanning. You can access the settings for this module in both views that are provided for a rule set.
     - *Key elements view* (this is the view that you see first after selecting a rule set)

       In this view, the SSL Client Context with CA option allows you to edit the settings for this module.
     - *Complete rules view* (this is the view that you see after clicking Unlock View in the former view)

       In this view, you can access the Set Client Context rule, which uses the SSL Client Context with CA to process web traffic. The rule displays the Default CA settings, which allows you to access and configure them.

       The rule is included in the Handle Connect Call rule set, which is embedded in the HTTPS Scanning rule set.
2. Configure any of these settings as needed.
3. Click Save Changes.

Managing certificates

When administering the HTTPS process on Web Gateway, managing certificates is one of the most important activities.

When web traffic is going on under HTTPS using SSL-secured connections, certificates are sent from one partner in this communication to another. For example, Web Gateway sends certificates to its clients when communication is going on under HTTPS.

A certificate is sent to a communication partner to indicate that the sender can be trusted. Certificates are signed by certificate authorities (CAs) to prove that they can rightfully be trusted.

When several certificate authorities are involved in the signing, resulting in a chain of certificates with one certificate proving the trustworthiness of another, the initial certificate authority is also known as root certificate authority (root CA).

Activities performed to manage certificates include:

- **Replacing a root certificate authority** — After the initial setup, a default root certificate authority (root CA) is provided on Web Gateway. You can replace this root CA with one that you import or create on your own.

- **Creating a client certificate list** — You can create a list with certificates that Web Gateway can choose from when sending a certificate to a client in communication going on under HTTPS.
- **Managing certificates for cloud use** — You can make certificates that you create or import on Web Gateway also available for cloud use.

Managing certificates for cloud use

You can manage certificates for handling SSL-secured web traffic on Web Gateway and make them available for cloud use.

Certificates used to prove authorization of Certificate Authorities (CAs) can be added by generating or importing them on Web Gateway. They can then be made available to McAfee Web Gateway Cloud Service (McAfee WGCS) when this service is managed on the MVision Cloud platform.

When managed on this platform, McAfee WGCS is part of the solution known as *McAfee Mobile Cloud Security (MMCS) 2.0,* which allows mobile devices to be included among the end-user systems that McAfee WGCS protects.

You can add CA certificates on the Web Gateway interface. The certificates are stored, along with user and group mapping information as configuration for MMCS.

The synchronization process that is in place to synchronize configuration data between Web Gateway and McAfee WGCS under what is known as the Web Hybrid solution ensures that any MMCS configuration created or modified on Web Gateway is known on McAfee WGCS.

MMCS configuration data is stored in a database on McAfee WGCS and read by dedicated services on globally distributed nodes called *Points of Presence (PoPs)*. These nodes serve as VPN gateways. They provide VPN connectivity for the mobile devices of cloud users who are attempting to gain web access.

Managing CA certificates for cloud use on Web Gateway includes:

- Adding and removing certificates
- Testing certificates

Information about expired certificates can be retrieved when a certificate test is performed.

Certificate pinning

Several mobile apps use certificate pinning to ensure they communicate with authorized web servers. This means that issues can arise when McAfee WGCS replaces a server certificate with a certificate of its own.

You can whitelist the relevant sites, which will exempt traffic going to them from being filtered by the rules for inspecting SSL-secured traffic that are enabled under your web security policy.

A McAfee-maintained list includes websites that certificate pinning is applied to. Its name is Sites Using Pinned Certificates. You can use this list in a whitelisting rule to exempt traffic going to these sites from further filtering.

Rule for whitelisting mobile apps with certificate pinning

A rule is available for exempting traffic going to mobile apps that use certificate pinning from filtering by rules for inspecting SSL-secured traffic. The rule relies on a list with host and domain names to allow bypassing the filtering for this traffic. The list is maintained by McAfee.

The rule is contained in the Handle CONNECT Call rule set, which is nested in the HTTPS Scanning top-level rule set. Its name is Tunneled pinned certificate hosts for IPSec mobiles.

The rule sets are part of the default rule set system on Web Gateway after the initial setup. They are not enabled by default and the rule is not enabled either. You can enable them for both on-premise and cloud use.

The HTTPS Scanning rule set and its nested rule sets are also available in the built-in library within the HTTPS Scanning group.

Test a device certificate for cloud use

You can test a certificate on Web Gateway that is intended for a cloud user device.

A device certificate is used to secure communication on a device that is operated by an end user when working in the cloud. The test checks the validity of a device certificate.

A device certificate is valid if there is at least one certificate among the CA certificates for cloud use stored on Web Gateway with a CA that authorizes the tested certificate.

Task

1. Select Configuration → Appliances.
2. In the navigation pane, select Mobile Cloud Security.
3. Under Device Certificates Test click Test Device Certificates .
4. Provide a device certificate for testing.
   a. In the Certificate Test window, click Browse.
   b. Select a device certificate file from your file system, then click Open to make the certificate available for testing.

      Note: If you are running the HTML-based user interface, you must upload suitable files before, using the particular file management options that are provided.

      You are returned to the Certificate Test window.
5. Click Test to test the file that you have made available.

   The test result is displayed at the top of the window.
6. Click Close to close the test window.

Results

If the test shows that the device certificate is valid, you can use it for securing communication on a cloud user device.

Replace the default root certificate authority

You can replace the default root certificate authority that is provided after the initial setup for signing the certificates that the appliance sends to its clients by a certificate authority of your own.

You can create a new root certificate authority on the user interface or import one from your file system.

Create a root certificate authority

You can create a root certificate authority (CA) for signing the certificates the appliance sends to its clients and use it instead of the default certificate authority.

Task

1. Select Policy → Settings.
2. On the Engines branch of the settings tree, go to SSL Client Context with CA and select the settings you want to use the new certificate authority for.
3. Click Generate New.

   The Generate New Certificate Authority window opens.
4. In the Organization and Locality fields, type suitable information for your own certificate authority.
5. [Optional] In the Organizational unit and State fields, type suitable information. From the Country list, select a country.
6. In the Common name field, type a common name for your own certificate authority.
7. [Optional] In the Email address field, type an email address of your organization.
8. From the Valid for list, select the time that your certificate authority should be valid.
9. [Optional] In the Comment field, type a plain-text comment on the certificate authority.
10. Click OK.

    The new certificate authority is generated.
11. Click Save Changes.

Import a root certificate authority

You can import a root certificate authority (CA) for signing the certificates the appliance sends to its clients and use it instead of the default certificate authority.

Task

1. Select Policy → Settings.
2. On the settings tree, select SSL Client Context with CA and click the settings you want to use the imported certificate authority for.
3. Click Import.
   The Import Certificate Authority window opens.
4. Enter the name of the certificate authority file in the Certificate field by clicking Browse and browsing to a suitable file.
   The file must be encoded in PEM (Privacy-enhanced mail) format.
5. Enter the name of the certificate key file in the Private key key field by clicking Browse and browsing to a suitable file.
   The file must be encoded in PEM format. The key must have a length of at least 2048 bit.
6. [Conditional] If the private key is protected by a password, type it in the Password field.

   Along with unencrypted keys, importing the following key types is supported:

   - AES-128-bit encrypted
   - AES-256-bit encrypted
   - PEM(BASE64-text)-encoded certificates and private key (one per file)
   - multiple PEM(BASE64-text)-encoded certificates for certificate chains
7. [Conditional] If the certificate authority is part of a certificate chain and you want to provide information on this chain with the certificate, enter the name of the file containing the information in the Certificate chain field by clicking Browse and browsing to a suitable file.
   The file must be encoded in PEM format.
8. Click OK.
   The certificate authority is imported.
9. Click Save Changes.

Client certificate list

The client certificate list is a list of certificates that can be sent to a web server when a client request is received on an appliance in SSL-secured communication and passed on to the appropriate web server.

The certificate is sent when the web server asks for it at the initial and subsequent handshakes, as SSL renegotiation is performed.

A rule event tells the appliance to use a client certificate for communication with the web server. The certificate can then be selected from the client certificate list.

In this case, the private key for the certificate must be provided by the client that sent the request.

Alternatively, a preconfigured certificate can be used that is always sent to the web server.

The rule event that triggers the use of a certificate from the client certificate list can belong to rules that apply to CONNECT requests (even in transparent setups) or to rules in rule sets for certificate verification that have CERTVERIFY as value for the *Command.Name* property in their criteria.

You can configure settings for the rule event that include a client certificate list and the instruction to use it. The settings can also specify that the private key for the certificates that the clients of the appliance provide is stored unencrypted.

Create a client certificate list

You can create a list of client certificates that can be sent to web servers in SSL-secured communication.

Task

1. Select Policy → Settings.
2. On the settings tree, select SSL Client Certificate Handling and click Add.
   The Add Settings window opens with the Add Settings tab selected.
3. Configure general settings parameters.
   a. In the Name field, type a name for the settings.
   b. [Optional] In the Comments field, type a plain-text comment on the settings.
   c. [Optional] Click the Permissions tab and configure who is allowed to access the settings.
4. Under Client Certificate Handling, make sure the option Use client certificate from Known client certificates list if client has proven ownership is selected.
5. On the toolbar of the Known client certificates list, click Add.
   The Add Client Certificate window opens.
6. Click Import to import a client certificate.
   The Import Client Certificate window opens.
7. Import a client certificate.
   a. Next to the Certificate field, click Browse, and within the local file manager that opens, browse to a suitable certificate file and select it.
      The file manager closes and the certificate file name appears in the field.
   b. Next to the Private key field, click Browse, and within the local file manager that opens, browse to a suitable key file and select it.
      The file manager closes and the key file name and password appear in the Private key and Password fields.
   c. Click OK.
      The window closes and the certificate file information appears in the Import Client Certificate window.
   d. [Optional] In the Comments field, type a plain-text comment on the certificate.
8. Click OK.

   The Add Client Certificate window closes and the certificate file name and comment (if provided) appear in the Known client certificates list.

   Repeat Steps 5 to 8 for any other certificate you want to add to the list.
9. Click OK to close the Add Settings window.
10. Click Save Changes.

Using Skype for Business

Skype for Business (SfB) is a widely used communication tool. When using it with Web Gateway, you must ensure that no HTTPS (SSL) scanning is applied to web traffic going on over this tool.

Web traffic uses default HTTPS ports under Skype for Business, but does not always follow the relevant protocol. If HTTPS scanning is performed on Web Gateway, it detects a mismatch and closes the connection, which leads to a communication break. Audio and video communication can no longer continue using Skype for Business, nor can its desktop sharing feature be used.

Web Gateway is by default configured to detect web traffic using Skype for Business and exempt it from scanning. An option of the SSL Scanner settings is set for this purpose. Be aware, however, that exempting this traffic creates a security risk.

Using AIA entries for certificate downloads

Certificates missing in a chain of server certificates that a client requires for SSL-secured communication can be downloaded using URLs that are specified in Authority Information Access (AIA) entries. AIA entries are included in the last certificate of a certificate chain.

Note: The information provided here regarding SSL also applies when the newer version of this protocol known as TLS (Transport Layer Security) is used.

AIA downloads are performed by the Certificate Chain module (or *engine*) of Web Gateway, which acts as a proxy in this communication. Only one AIA download is performed per handshake. Downloaded certificates are cached for 24 hours and re-used for other connections.

A certificate chain is considered complete once it ends in a trusted certificate authority (CA). If SSL-secured connections are tunneled, for example, due to whitelisting or for client authentication, the client must also obtain the missing certificates, which can again be achieved using URLs specified in AIA entries.

To enable AIA downloads, you must make sure that the setting for this download is selected on the user interface of Web Gateway. The setting is selected by default.

The setting is part of the Default settings for the Certificate Chain module. It can also be accessed when editing the Certificate chain filters element in the key elements view of the SSL Scanner rule set.

Sending tapped SSL traffic to a monitoring device

You can send tapped SSL traffic in decrypted format through an interface on a Web Gateway appliance to a monitoring device.

Note: The information provided here regarding SSL also applies when the newer version of this protocol known as TLS (Transport Layer Security) is used.

SSL-secured traffic can be tapped on Web Gateway, which means that its content is looked into. Tapping is a "silent" inspection method, as the traffic is only looked into and not interfered with otherwise.

You can configure more than one interface to send copies of the decrypted traffic to different monitoring devices.

Tapping can be applied on Web Gateway to SSL-secured traffic under HTTPS, including any subversion of this protocol. HTTP2 is, however, not supported. When tapping is configured on Web Gateway, HTTP2 traffic is not processed.

The Enable SSL Tap event is provided, which must be included in a suitable rule to enable the tapping. The rule must be applied when the CONNECT call is handled within the process of performing SSL-secured communication.

Sample rule for enabling SSL tapping

The following conditions must be met when using a rule with an event for enabling SSL tapping:

- The rule must be placed in a rule set that has Command.Name equals "CONNECT" as one of its criteria. This is the case in the embedded Handle CONNECT Call rule set of the default SSL Scanner rule set.
- The rule set must be configured for the request cycle.
- Content inspection must be activated. This is the case if you enable the embedded Content Inspection rule set of the default SSL Scanner rule set.

A suitable property for the rule criteria is, for example, Client.IP or URL.Host. A rule that enables SSL tapping for all traffic sent in requests for access to hosts that are on a particular list might look as follows.

Name				
Enable SSL tapping for requests sent to listed hosts				
Criteria		Action		Event
URL.Host is in list SSL Tapping Host List	–>	Continue	–	Enable SSL Tap

Configure the sending of tapped SSL traffic to a monitoring device

Configure the sending of tapped SSL traffic to a monitoring device by configuring an interface to this device on Web Gateway and creating a rule that enables the tapping.

Task

1. Configure at least one interface to a monitoring device.
    a. Select Configuration → Appliances.
    b. On the appliances tree, select the appliance that you want to configure an interface on, then click SSL Tap.
    c. Configure the SSL Tap settings.
2. Create a rule that uses the Enable SSL Tap event.
3. Click Save Changes.

Results

SSL traffic can now be tapped and sent in decrypted format to the monitoring devices that you configured interfaces for. The tapped traffic is sent if the rule that you created applies.

Hardware Security Module

Use of a Hardware Security Module (HSM) enhances security when dealing with private keys for the certificates that are exchanged between clients and servers in SSL-secured communication.

Keys for SSL-certificates can be public or private. If you are using private keys and do not want to expose them, you can store them on a Hardware Security Module.

When a key is required for enabling the use of a certificate, the key is referenced by its ID (also known as *key name*) while remaining protected on the module.

This method of key handling provides greater security than storing private keys in a file within your file system. This file might be read or copied after unauthorized access to a Web Gateway appliance. The private keys on the Hardware Security Module, however, would still remain protected.

Different solutions can be implemented to provide the functions of a Hardware Security Module on Web Gateway.

Using a Hardware Security Module

Several components are involved when a Hardware Security Module is used on Web Gateway. These include the HSM Agent and other components that differ depending on the particular solution that is implemented.

HSM Agent

The HSM Agent runs as a daemon within the Web Gateway appliance system. This agent enables the handling of the Hardware Security Module.

Depending on the solution that is implemented, the agent addresses the component that provides the module functions, for example, a module card or a remote server.

The agent provides a command line interface for performing activities on the module, such as generating, storing, or unlocking keys.

Module card

A module card can be installed as a hardware component on a Web Gateway appliance to provide the functions of a Hardware Security Module.

The module card that is available for use with Web Gateway is the *nShield Solo HSM* card. It is provided by a McAfee partner (Entrust).

When the module card is installed, you can access it by logging on to the appliance from a system console.

For more information, see the documentation of the McAfee partner (Entrust).

Appliance

The functions of a Hardware Security Module can be provided on a Web Gateway appliance using an additional appliance. The appliance that is used within this solution is the *nShield Connect* appliance. It is provided by a McAfee partner (Entrust).

Note: When the nShield Connect appliance solution is implemented, we recommend configuring Web Gateway as an unprivileged client of nShield Connect. This means remote administration of other clients cannot be performed from this client.

For more information, see the documentation of the McAfee partner (Entrust).

Remote server

The functions of a Hardware Security Module can be provided on a Web Gateway appliance using a remote server. The remote server that is used within this solution is the *Luna Network HSM* server. It is provided by a McAfee partner (Thales).

When the remote server has been set up and connected to the appliance, you can access the module by logging on to the appliance from a system console.

For more information, see the documentation of the McAfee partner (Thales).

Emulation

An emulation can be run on Web Gateway, which provides the functions of a Hardware Security Module using *OpenSSL*.

As this solution does not include a module card, additional appliance, or remote server for storing private keys, you must store these keys manually in a directory of the Web Gateway appliance system.

This solution is not considered as secure as the module card solution. When implemented on a standalone Web Gateway appliance, however, it compares to the remote server solution with regard to security. An emulation is preferably used for demos, tests, and training.

Client-server model for multiple appliances

When multiple Web Gateway appliances are part of an HSM solution, they can be configured to follow the client-server model.

This means, for example, that you need not install a module card on every Web Gateway appliance in your network to use the functions of a Hardware Security Module. Appliances that have no HSM solution of their own implemented can connect to an appliance with this solution to use its functions.

The appliance that has an HSM solution implemented then takes the server role towards the other appliances, which connect to it as clients.

Note: On the user interface of Web Gateway, an appliance that has an HSM solution of its own implemented is referred to as *HSM server* even if no other appliances are configured as its clients.

The client-server model can be configured for Web Gateway appliances regardless of the particular solution (module card, appliance, remote server, or emulation) that is implemented on one of them.

Web Gateway as client of a remote server (Thales)

When the HSM solution on a Web Gateway appliance uses the remote server that is provided by a McAfee partner (Thales), the client-server model also applies. The Web Gateway appliance then connects as client to the remote server.

When a Web Gateway appliance has the HSM solution implemented that uses a remote server (Thales) and other appliances connect to it to use the functions of this solution, this Web Gateway appliance takes both the roles of a client and a server.

The appliance then acts as a client towards the remote server (Thales) and as a server towards the other appliances.

Key handling with a Hardware Security Module

Using a Hardware Security Module allows you to perform several activities for enhancing private key security, such as generating, storing, and referencing keys.

When an HSM solution is implemented, all cryptographic operations related to using a private key for a certificate are performed on the Hardware Security Module.

Keys can be generated on the module, but can also be imported to it. To be available on Web Gateway, they are loaded by the HSM Agent. To enable key loading, the key IDs must be made known to the agent.

Generating private keys often includes the use of passwords or an Operator Card System (OCS) to create additional security. Keys can also be generated, however, without any of these additional options.

To enhance security in key handling, responsibilities can be assigned to different administrators. For example, one administrator might be responsible for generating private keys on a Hardware Security Module,

The Web Gateway administrator then references the keys to configure certificates on the user interface of Web Gateway.

Note: The Web Gateway administrator must know the key IDs that are generated, as well as the passwords that might be set for the keys.

Private key operations involving the Hardware Security Module are logged on Web Gateway. Information about these operations is displayed on the dashboard of the user interface.

Connection traces can also be generated for traffic on the connections between the components of a Hardware Security Module solution.

Work with a Hardware Security Module

Complete these high-level steps to enhance private key security with a Hardware Security Module. Different administrators can be responsible for different steps.

Task

1. Implement a solution for using a Hardware Security Module on a Web Gateway appliance.
2. Generate private keys on the Hardware Security Module.

   For information about how to generate these keys, see the following documentation:

   - If you have installed a module card or an additional appliance, see the documentation of the McAfee partner who provides these devices (Entrust).

     For more enhanced security, this documentation also provides information on creating a Security World and an Operator Card Set.
   - If you have set up a remote server, see the documentation of the McAfee partner who provides this server (Thales).

   If you use an emulation, generate private keys on the user interface of Web Gateway.
3. Store private keys on the Hardware Security Module.

   For information on storing these keys, see the documentation of the McAfee partners if you are working with a module card or appliance (Entrust) or a remote server (Thales).

   Note: You can also store private keys on a module card, appliance, or remote server that you have generated on the user interface of Web Gateway.

   If you use an emulation, store the private keys in a directory that is provided within the Web Gateway appliance system. The path to this directory is: /opt/mwg/data/hsmagent.
4. Make private keys available for use on Web Gateway.
   This is done by entering them in a list on the user interface.
5. Reference private keys on a Hardware Security Module for use on Web Gateway.

   If you have protected the keys using an Operator Card Set, you must unlock them first.

   For information about how to unlock keys, see the documentation of the McAfee partner who provides the Operator Card Set (Thales).

Implement an HSM solution on an appliance

To implement an HSM solution on a Web Gateway appliance, complete some preparatory activities. Then use the options of the user interface to select and configure a solution.

Note: The following steps can be in the responsibility of different administrators.

Task

1. Prepare a Web Gateway appliance for running an HSM solution.

   - If using a module card:
     Install the module card on the appliance.
     For information about how to install the module card, see the *McAfee Web Gateway Installation Guide*.
   - If using an additional appliance:
     Install the appliance.
     For more information, see the documentation of the Intel partner who provides the appliance (Entrust).
   - If using a remote server:

Set up the remote server and connect it to the Web Gateway appliance.

For more information, see the documentation of the Intel partner who provides the remote server (Thales).

- Using an emulation requires no preparatory activities.

2. On the user interface of Web Gateway, configure the HSM solution for the appliance that you have prepared.
   a. Select Configuration → Appliances.
   b. On the appliances tree, select the appliance and click Hardware Security Module.
      The Hardware Security Module settings appear in the configuration pane.
   c. Under HSM Server, select Start local HSM server.
   d. From the Crypto module list, select one of the following HSM solutions according to what you have prepared:
      - Entrust nShield Solo/Connect — Enables use of the module card or the appliance
      - SafeNet Network HSM (formerly Luna SA) — Enables use of a remote server (Thales)
      - OpenSSL — Enables use of an emulation
3. In the Keys to be loaded list, add entries for the private keys that you want to be available for loading.
   For every key, enter its key ID in string format.
4. Make sure that Allow local connections is selected.
5. Click Save Changes.

Results

You can now reference private keys when importing certificates for SSL-secured communication on the user interface of Web Gateway.

You can also configure other Web Gateway appliances that have no HSM solution of their own implemented to use the functions of the solution on this appliance.

Enable use of the nShield Connect appliance solution

When using an nShield Connect appliance, you must start the nShield daemon, also known as *hardserver* daemon, manually to enable this solution.

The daemon is started within the MLOS operating system of the Web Gateway appliance that you are using for the solution.

Task

1. Log on to the MLOS shell.
2. Run a command like the following to add a line for enabling a start script to the appropriate configuration file:

   ```
   echo NFP_ALWAYS_ENABLED=1 >> /etc/nfast.conf
   ```

   The NFP_ALWAYS_ENABLED parameter enables or disables the */etc/init.d/nfast* script, which is used for starting the daemon.
3. Start the daemon:

   ```
   /etc/init.d nfast start
   ```

   Whenever the appliance system is restarted in the following, the daemon is also started.

   To stop the daemon, run */etc/init.d nfast stop*.

   To disable the daemon permanently, set NFP_ALWAYS_ENABLED to zero or remove the line with this parameter from the configuration file.

Sample procedure for configuring a remote server (Thales)

Configure the remote server (Thales) by running suitable commands from a system console.

For information about how to configure this remote server, you should generally refer to the partner documentation (Thales).

The following procedure provides some sample steps and commands for completing this configuration.

Task

1. From a system console, connect to the remote server (Thales), then run the following commands to configure the server for connecting to the Web Gateway appliance.

   ```
   client register -c <label> -ip <IP address of the Web Gateway appliance>
   ```

   ```
   client register -c <label> -ip <IP address of the Web Gateway appliance>
   ```

2. Connect to the Web Gateway appliance, then complete these substeps.
   a. Verify the connection to the remote server.

      ```
      /opt/gemato/lunaclient/bin/vtl verify
      ```

   b. Create a client certificate for connecting to the remote server.

      ```
      /opt/gemato/lunaclient/bin/vtl createCert -n <IP address of the Web Gateway appliance
      ```

   c. Copy the client certificate to the remote server.

      ```
      scp/opt/gemato/lunaclient/cert/client/<IP address of the Web Gateway appliance>.pem admin@<domain name>
      ```

3. On the remote server, copy a server certificate to the Web Gateway appliance.

   ```
   scp admin@<domain name>:server.pem
   ```

Configure the use of an HSM solution by multiple appliances

Configure Web Gateway appliances that have no individual HSM solution implemented as clients of a Web Gateway appliance with an HSM solution. These appliances can then use the functions of the solution.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select a Web Gateway appliance that has an HSM solution implemented, then click Hardware Security Module.
   The Hardware Security Module settings appear in the configuration pane.
3. Under HSM Server, configure this appliance to act as a server allowing clients to use a Hardware Security Module remotely.
   a. Click Allow remote connections.
   b. In the HSM server port definition list, add one or more ports that listen to client requests.
   c. Under Server identification, generate or import a certificate for the server. Then export it to a location where you can import it from when configuring the clients.
   d. Click Save Changes.
4. Complete the following substeps for every appliance that you want to configure as a client.
   a. On the appliances tree, select an appliance that has no HSM solution implemented, then click Hardware Security Module.
   b. Under HSM Client, select Use remote HSM server.
   c. In the Remote server list, add a host name with a listener port and import the server certificate.
      You can add entries for more than one server to allow this client the use of Hardware Security Modules on different servers.
   d. Under Client identification, generate or import a certificate for the client. Then export it to a location where you can import it from when configuring the list of appliances that are permitted as clients of the server.
5. Add the clients of the server.
   a. On the appliances tree, select the appliance that you have configured as server, then click Hardware Security Module.
   b. In the Permitted clients list under HSM Server, add a host name and import a certificate for every appliance that you have configured as client.

6. Click Save Changes.

Make private keys available on an appliance

Make private keys available for referencing in a list on a Web Gateway appliance that has an HSM solution implemented.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select an appliance that has an individual HSM solution implemented and click Hardware Security Module.
   The Hardware Security Module settings appear in the configuration pane.
3. Under HSM Server, add entries for private keys in the Keys to be loaded list.
   For every key, enter its key ID in string format.
4. Click Save Changes.

Results

The private keys are now available for referencing.

Reference a private key on a Hardware Security Module

Reference a private key on a Hardware Security Module to make the key information available for enabling the use of a certificate.

Certificates are involved when settings for filtering SSL-secured communication are configured. This sample procedure describes one scenario for referencing a private key.

Task

1. Select Policy → Settings.
2. On the Engines branch of the settings tree, expand SSL Client Context with CA and select the Default CA settings.
3. Under Define SSL Client Context, click Import next to Certificate Authority.
   The Import Certificate Authority window opens.
4. Click Browse next to Certificate, then locate and import a certificate file.
5. Select HSM next to Private key source.
   A list with the IDs for the available private keys opens.
6. Select a key ID, then click Import to import the certificate with its key information.
7. Click Save Changes.

Using private keys from an Azure Key Vault

You can use private keys stored in an Azure Key Vault for certificates that are required to enable cloud communication with clients over SSL-secured connections.

Azure Key Vault is a device for storing private keys. These keys can be used in a hybrid environment where you are working with both Web Gateway and Web Gateway Cloud Service to protect your network against threats arising from the web.

An instance of an Azure Key Vault with an application for private key handling and the private keys are created on the Azure platform. A certificate that requires a key when a signature is generated to enable its use is imported on Web Gateway.

The certificate is used when Web Gateway Cloud Service controls web access over SSL-secured connections by clients that cloud users of your organization work with. These users do not access the web on premise, but from outside your local network.

Private key handling

A private key that is stored in an Azure Key Vault does not become embedded in any settings that are maintained on Web Gateway.

When a private key is needed for a certificate, Web Gateway submits an application ID, tenant ID, key ID, and a password to obtain a token for access to the Azure Key Vault instance that has been set up to store private keys for use by Web Gateway and Web Gateway Cloud Service.

After receiving the token, Web Gateway sends an HTTPS request to obtain the private key data for generating the signature that is required to make use of the certificate.

Monitoring private key handling

The dashboard on Web Gateway shows the number of private key operations and their average duration as seen from a Web Gateway. The dashboard on Azure has a similar view.

There are no error log entries written to record failures of private key operations, but the response that Web Gateway sends to the client contains an error string, which is mainly extracted from the response that Web Gateway receives from Azure.

You can also use the *testKeyVault.sh* script and run connection traces for troubleshooting..

Create an Azure Key Vault and a private key

Create an Azure Key Vault to store private keys for use with SSL certificates that protect network connections.

Complete the usual steps for creating these items on the Azure portal and provide suitable values for the options of each step to enable use of the private keys by Web Gateway.

Task

1. Log on to the Azure portal.
2. Register an application for private key handling.
   a. On the portal, navigate to Azure Active Directory → App registrations.
   b. Select suitable values on the registration page to register an application.

      When the application has been registered, note down its Application ID and Tenant ID (Directory ID).
   c. Add an API permission to allow the application full access to the Azure Key Vault service.
   d. Set a client secret for the application, which is a password that is required when Web Gateway completes private key operations.
3. Create an Azure Key Vault.
   The name for this key vault might be, for example, `mcafeedoc`.
4. Generate a private key.
   a. Select RSA as the key type and generate the private key.

      When the private key has been generated, note down its Key ID, which is a URL.
   b. Select Sign and Verify as operations that are permitted when using the private key.
   c. Add an access policy for the private key with these parameters.
      - Get, Sign and Verify as key permissions.
      - The application that you registered in step 2 as principal application.

Results

An Azure Key Vault is now available for storing private keys and using their data for certificate handling on Web Gateway.

Configure use of a private key from an Azure Key Vault

To configure use of a private certificate key that is stored in an Azure Key Vault, provide settings for the module that handles communication with Web Gateway clients over an SSL-secured connection.

To provide these settings you import a certificate with the private key assigned to it.

Task

1. Select Policy → Settings.
2. On the Engines branch of the settings tree, expand SSL Client Context with CA,, then select the Default CA settings.
3. Import a certificate with a private key from an Azure Key Vault assigned to it.
   a. Under Define SSL Client Context, click Import next to Certificate Authority.
   b. In the window that opens, click Browse next to Certificate, and locate a certificate file.
   c. Under Private key source, select Azure Vault .
   d. Fill these input fields with the values you noted down when setting up the Azure Key Vault with the private key.
      - **App ID**
      - **Tenant ID**
      - **Key ID**
   e. Under **Password**, enter the password that you set when creating an Azure Key Vault with a private key.
   f. Click Import.
      The certificate is imported. Its key ID and other properties are shown in the settings pane.
4. Click Save Changes.

Results

A private key is now available in an Azure Key Vault for use with an SSL certificate.

Advanced Threat Defense

After a web object has been scanned by Web Gateway for infections by viruses or other malware, it can additionally be scanned by the McAfee® Advanced Threat Defense (Advanced Threat Defense) web security product.

Advanced Threat Defense uses a sandboxing approach for scanning, which means that the behavior of a particular web object in a "sandbox" environment is analyzed. The scanning result is recorded in a report and delivered to Web Gateway.

The additional scanning performed by Advanced Threat Defense is also referred to as *offline scanning* or *background scanning*.

To enable the use of Advanced Threat Defense, suitable rules must be implemented on Web Gateway. You can import rule sets that contain such rules from the rule set library.

Options for configuring the use of Advanced Threat Defense

You can configure different options to implement an additional scanning by Advanced Threat Defense.

- **Forwarding a web object depending on the additional scanning** — When this option is configured, the result of the additional scanning by Advanced Threat Defense determines whether a web object is forwarded to the user who requested it.

  If a web object is found to be safe, it is forwarded, otherwise not.
- **Forwarding a web object before the additional scanning** — When this option is configured, a web object is forwarded to the user who requested it. before the additional scanning by Advanced Threat Defense.

  If a web object is found to be infected, a warning message is sent to the administrator of the network that the user sent his request from.

You can also configure that a web object is not scanned a second time by Advanced Threat Defense if it has been scanned before. In this case, the existing report that was produced after the first scanning is evaluated once again.

Availability of Advanced Threat Defense

For use with Web Gateway, the Advanced Threat Defense web security software is delivered pre-installed on the same hardware platform, where it runs as an appliance on a separate server.

Several instances of the product can also run on different servers and be used to support Web Gateway. Each instance of the product must be installed on its own hardware platform.

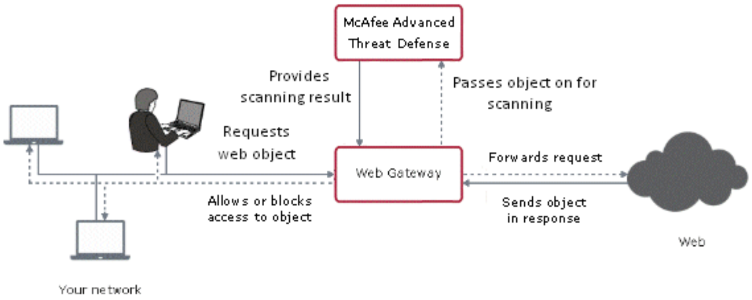
Workflows for using Advanced Threat Defense

Different workflows can be configured when Advanced Threat Defense is used to perform an additional scanning of web objects.

Forwarding a web object depending on the additional scanning

The following diagram shows the workflow that forwards a web object to a user depending on the scanning result of Advanced Threat Defense.

Web object is forwarded depending on additional scanning result



1. A user sends a request to access a web object, for example, a file, from a system within your network that is a client of Web Gateway.
2. If the request passes filtering according to the configured rules, Web Gateway forwards it to the appropriate web server.

   A progress page is sent to the client, telling the user to wait while the request is processed.
3. The web server sends the object to Web Gateway.
4. If the criteria for using Advanced Threat Defense are met, Web Gateway passes the object on for scanning.

   To retrieve information on the scanning progress, Web Gateway queries Advanced Threat Defense in regular intervals.
5. When Advanced Threat Defense has completed the scanning, it lets Web Gateway know whether the object is malicious or not.
6. Depending on this information, Web Gateway allows the user to access the requested object or sends a block page, which states that access is blocked and gives a reason for the blocking.

Criteria for additional scanning by Advanced Threat Defense

Web Gateway uses the functions of Advanced Threat Defense for scanning a web object after the object has been scanned by the anti-malware engines on Web Gateway.

The Advanced Threat Defense library rule set uses this probability in its criteria. The default value that must be reached for the criteria to match is 60. This means that only if scanning a web object on Web Gateway results in a malware probability of 60 percent or more, is it passed on to Advanced Threat Defense.

When configuring the use of Advanced Threat Defense, you can increase or lower this value and, consequently, let this product support Web Gateway more or less frequently.

It is *therefore* important that, on the rule sets tree, the rule set for Advanced Threat Defense is placed behind the rule set for the normal anti-malware functions on Web Gatewayy, which is usually the *Gateway Anti-Malware* default rule set.

The Anti-Malware module (or *engine*) runs with two different settings, when Web Gateway and Advanced Threat Defense work together: one for the Web Gateway part and one for the part of the supporting product.

The default names of the two settings are *Gateway Anti-Malware* and *Gateway ATD*.

One important point in which the settings differ from each other is that the Gateway ATD settings have the option for using Advanced Threat Defense selected, whereas this option is deselected in the other settings.

Configuration elements for using Advanced Threat Defense

To enable the additional scanning of web objects by Advanced Threat Defense, suitable rules must be implemented on Web Gateway. You can import rule sets that contain such rules from the rule set library. After importing this rule set, a list and settings are also implemented.

Rule sets for the additional scanning

There is a rule set for forwarding a web object depending on the additional scanning, as well as a rule set for forwarding a web object before the additional scanning and delivering any warning information afterwards.

- **Advanced Threat Defense library rule set** — This rule set implements the workflow that lets a web object additionally be scanned by Advanced Threat Defense and forwarded to the user depending on the scanning result.

  After importing this rule set, a list and settings are also implemented.
- **ATD - Init Offline Scan nested library rule set** — This nested rule set has the same criteria as the rule set that forwards a web object to the user depending on the result of the additional scanning.

  The rule set applies if previous scanning by Web Gateway has resulted in a configured degree of probability that a web object is infected, the web object is on the list of web objects that can be scanned, and a particular object size is not exceeded.

  The rule set contains only one rule that uses the *Antimalware.MATD.InitBackgroundScan* property in its criteria. The value of this property is *true* by default.

  In this case, data for the current transaction is recorded. This includes all data that is related to a request for web access and the response to it from a web server, such as the IP address of the client, authentication information, the URL of the web server, and the requested web object that was sent as the body of the response message.

  An internal request is sent to initiate scanning by Advanced Threat Defense. After this has been completed, the requested web object is forwarded to the user while the scanning is performed later on, using the data that was recorded.

  If the value of the *Antimalware.MATD.InitBackgroundScan* property is *false*, scanning by Advanced Threat Defense could not be initiated and a rule event is used to display an error message.
- **ATD - Handle Offline Scan nested library rule set** — This nested rule set has the *Antimalware.MATD.IsBackgroundScan* property for its criteria. The value of this criteria is *true* by default.

  In this case, the data that was recorded by the rule in the *ATD - Init Offline Scan* rule set, is used by Advanced Threat Defense to scan the web object specified by the data.

  The rule set has a rule that uses an event to increase a counter if a scanned web object has been found to be infected, a rule that uses another event to create and send a message about the infected web object to the administrator, and finally a rule that stops the processing cycle.

List and settings for the additional scanning

The *Advanced Threat Defense* library rule set provides rules for enabling the use of Advanced Threat Defense on Web Gateway and forwarding a requested web object to the user depending on the scanning result.

After importing this rule set, a list and settings are also implemented.

- **Advanced Threat Defense Supported Types list** — This list is used within the criteria of the library rule set. Only web objects belonging to the media types on this list are passed on to Advanced Threat Defense for scanning.

  The list contains several media types by default. You can add media types to the list or remove them.
- **Gateway ATD settings** — These are settings for the Anti-Malware module (or *engine*) on Web Gateway, which handles virus and malware filtering, including the additional use of Advanced Threat Defense.

  The settings include mainly options for configuring the following:

  - Communication between Web Gateway and the server that Advanced Threat Defense runs on
  -

    Severity grade that lets a web object, for example, a file, be classified as malicious

    When an object is scanned by Advanced Threat Defense, the result is a severity grade on a scale from 0 to 5 (very high severity).

    You can set a value on this scale, for example, 3, which means all objects with a scanning result of 3 or higher are considered to be malicious.

    For these objects, the Antimalware.Infected property is set to *true*, so a rule that uses this property in its criteria will block a web object and prevent it from being passed on to the user who requested access to it.

Using an existing Advanced Threat Defense scanning report

A report that is generated by Advanced Threat Defense after scanning a web object can be used by Web Gateway to evaluate this object and handle access to it.

When using an existing report, Web Gateway will not trigger a new scanning run on Advanced Threat Defense. If more than one report exists, the latest report is used for evaluation. Hash values are calculated internally on Web Gateway to determine whether a web object is the same as another object, so the same report can be used.

To use an existing scanning report on Web Gateway, you need to implement a rule with the *Antimalware.ATD.GetReport* property. If the value of this Boolean property is *true*, it means that a particular web object has been found to have already been scanned by Advanced Threat Defense and a report for this scan has been retrieved.

This report can be made available to other rules, for example, to a rule with the *Antimalware.Infected* property, which evaluates the report to find out whether an object is infected.

Options for using an existing scanning report

There are several options for using an existing scanning report to handle access to web objects.

- **Allow a file when a scanning report shows that it is not infected** — There are files that are uploaded manually to Advanced Threat Defense where they are scanned and a report is generated. Web Gateway then allows users to download such a file if a report exists for it and this report shows that the file is not infected.

If a scanning report does not exist for a web object, the *Antimalware.ATD.GetReport* property can still be used in suitable rules. In these rules, the value of this property is *false*, as no scanning report was retrieved.

- **Allow a file if no scanning report is available and scan this file offline** — If no scanning report exists for a file that was requested for downloading, are rule can allow a user to download the file and let an offline scan be performed. After the scanning, a report is generated and forwarded to the administrator of the user's network.
- **Block a file if no scanning report is available and scan this file offline** — If no scanning report exists for a file that was requested for downloading, a rule can block access to the file and let an offline scan be performed. After the scanning, a report is generated and forwarded to the administrator of the user's network .

Sample rules for using an existing scanning report

There is no preconfigured rule set for using an existing Advanced Threat Defense scanning report in the default rule set system or the rule set library. You can, however, create suitable rules and a rule set for them on your own.

The following sample rules implement the solution that lets files be uploaded manually to Advanced Threat Defense. Downloading a file is allowed if the report that was generated by Advanced Threat Defense shows that the file is not infected.

The name of the rule set might be *Use Existing Advanced Threat Defense Scanning Report*. It must have the same criteria regarding media types as the *Advanced Threat Defense* library rule set and apply for all processing cycles.

The rule set should contain the following rules:

- A rule that uses the *Antimalware.ATD.GetReport* property to retrieve an existing scanning report
- A rule that evaluates files using this report and blocks access if the report shows that they are infected

The rule that retrieves the report might look as follows:

Name			
Allow files that have been scanned before			
Criteria	Action		Event
Antimalware.ATD.GetReport> equals false	Block <BlockedByMATD>	–	Statistics.Counter.Increment" (BlockedByMATD", 1)<Default>

The rule blocks access to a file if no report exists for it. In this case, the next rule is not processed. This rule evaluates a report. It might look as follows:

Name		
Block infected files		
Criteria	Action	Event
Antimalware.Infected <Gateway ATD> equals true –>	Block <BlockedByMATD> –	Statistics.Counter.Increment ("BlockedByMATD", 1)<Default>

In both rules, a counter records how often files were blocked when Advanced Threat Defense functions were used.

Using an ongoing Advanced Threat Defense scanning run

While a scanning run is being performed by Advanced Threat Defense, the results of this run can be used not only for processing the request that it was started for, but also for other requests to access the same web object.

To let the results of one scanning run be used for processing multiple requests, the requests must be received on Web Gateway while the scanning is still going on. Hash values are calculated internally on Web Gateway to determine whether a web object is the same as another object, so it can be decided whether requests for the same object are received.

To use the results of one scanning run for multiple requests to access the same object, you need to enable an option within the *Gateway ATD* settings for the Anti-Malware module (or *engine*). must be enabled. The name of this option is Re-use running task if same sample is being analyzed.

There is no preconfigured rule set in the default rule set system or the rule set library for using the results of one scanning run when multiple requests for the same object are received. You can, however, create suitable rules and a rule set for them on your own.

Limiting object sizes for scanning by Advanced Threat Defense

The size of objects that are additionally scanned by Advanced Threat Defense must be checked for compliance with the size limits that exist for this product.

There are some restrictions for Advanced Threat Defense regarding the size of web objects that can be scanned. The general size limit is 128 MB, which means that web objects of any type must not exceed this limit.

Other size limits exist for particular types of web objects. This is mainly due to the fact that sandboxing is performed on Advanced Threat Defense, which only allows, for example, a size of 10 MB for executable files.

Impact on the user experience

Web Gateway and Advanced Threat Defense communicate with each other over a REST API, which accepts files up to the general size limit by default. An end user who sent, for example, a request for downloading a 30 MB file is therefore first led to believe that this size is allowed.

When the sandboxing functions start operating, however, the file is rejected as too large. The end user receives a block message from Web Gateway, and the Advanced Threat Defense administrator sees an error message.

Configuring size limits on Advanced Threat Defense

File size limits can be set on Advanced Threat Defense using the *set filesizes* command.

Tip: Best Practice: Set all file size limits on Advanced Threat Defense to the same value.

We also recommend implementing this value on Web Gateway, for example, by creating a rule that only forwards files for scanning to Advanced Threat Defense if they do not exceed the size limit.

For more information about default size limits on Advanced Threat Defense and the methods of changing them, see the *McAfee Advanced Threat Defense Product Guide*.

Configuring size limits on Web Gateway

On Web Gateway, you can configure a rule that blocks files if they exceed a particular size limit. By inserting this rule in a rule set for handling Advanced Threat Defense scanning activities, you can make sure that only files with suitable sizes are passed on to Advanced Threat Defense.

If you have imported the library rule sets for Advanced Threat Defense, you can insert the size limiting rule there. Some of these rule sets contain a rule for uploading web objects to Advanced Threat Defense.

By inserting the size limiting rule before this rule, files that exceed the size limit are blocked and the rule for uploading to Advanced Threat Defense is not executed.

Rule for setting a size limit

The following sample rule assumes that files must not exceed a size limit of 10 MB if they are to be scanned by Advanced Threat Defense. It blocks files that exceed this limit.

Name	
Limit file size for scanning by Advanced Threat Defense	
Criteria	Action
Body.Size greater than 10000000) –>	Block<ATD size limit>

To let the size check only apply to particular file types, suitable parts must be added to the rule criteria. For example, if you only want to cover executable files, you can add a criteria part that uses the MediaType.IsExecutable property.

To let the user who sent a request involving an over-sized object to the web know that and why this request was blocked, you can configure appropriate settings for the block action. In the sample rule, these settings are named ATD size limit.

Configure the use of Advanced Threat Defense

You can configure the use of Advanced Threat Defense for additionally scanning web objects after they have been scanned by Web Gateway. Another option is to let a scanning report that has been generated for a web object by Advanced Threat Defense be evaluated on Web Gateway to handle access to this object.

If an existing scanning report for a web object is evaluated, Web Gateway will not trigger a new additional scanning run by Advanced Threat Defense for this object.

Configure scanning by Advanced Threat Defense

Configure additional scanning by Advanced Threat Defense after a scanning run by Web Gateway has been completed.

Task

1. Configure Advanced Threat Defense to integrate it into your network.

   For more information, see the *McAfee Advanced Threat Defense Product Guide*.
2. On the user interface of Web Gateway, complete the following activities:
   a. Import the rule set for one of the two additional scanning workflows from the rule set library.

These rule sets are located in the *Gateway Anti-Malware* rule set group.

  - *Advanced Threat Defense* — For forwarding web objects depending on the additional scanning

    On the rule sets tree, place this rule set after the rule set for scanning by Web Gateway. By default, this is the *Gateway Anti-Malware* rule set.

  - *ATD - Offline Scanning with Immediate File Availability* — For forwarding web objects before the additional scanning

    After importing this rule set, the following two rule sets appear on the rule sets tree.

    - *ATD - Init Offline Scan* — This rule set that initiates the additional scanning.

      On the rule sets tree, place this rule set after the rule set for scanning by Web Gateway. By default, this is the *Gateway Anti-Malware* rule set.

    - *ATD - Handle Offline Scan* This rule set handles the additional scanning once it has been initiated.

      On the rule sets tree, place this rule set after the rule sets that perform global or common activities and before the rule sets that perform particular filtering activities.

      For example, on the default rule sets tree, place this rule set after the *Common Rules* rule set and before the *Media Type Filtering* rule set.

b. To enable monitoring of Advanced Threat Defense scanning activities on Web Gateway, import the *ATD Scanning Log* and *Block on ATD Errors* rule sets from the rule set library and add them to the existing Log Handler and Error Handler rule sets, respectively.

c. Add media types to the list for supported media types or remove them as needed. After importing either of the library rule sets, the name of this list is *Advanced Threat Defense Supported Types*.

   Note: After importing a rule set, you can work with this list on the key elements view of the rule set.

d. Configure the settings for scanning by Web Gateway.

   By default, the name of these settings is *Gateway Anti-Malware*.

   Note: After importing a rule set, you can work with these settings on the key elements view of the rule set.

e. Configure the settings for scanning by Advanced Threat Defense.

   After importing either of the library rule sets, the name of these settings is *Gateway ATD*.

   Note: After importing a rule set, you can work with these settings on the key elements view of the rule set.

f. Save your changes.

Configure use of an existing Advanced Threat Defense scanning report

If you do not want a new scanning run to be performed on a web object, you can let an existing Advanced Threat Defense scanning report be used to evaluate the web object.

There are several options for using an existing scanning report. The following task assumes that:

- Scanning reports were generated for web objects that were uploaded manually to Advanced Threat Defense and scanned.
- Web Gateway allows access if a report shows that a web object is not infected and blocks it if no report exists.

Complete the following high-level steps:

Task

1. Create a rule set for the rules that handle the use of an existing Advanced Threat Defense scanning report.
2. In this rule set, create the following.
   - A rule that retrieves a scanning report for a file and blocks access to a file if no report exists for it
   - A rule that evaluates a scanning report and blocks a file that is infected according to the report
3. Configure the *Gateway ATD* settings of the Anti-Malware module.

   a. Make sure Re-use previous detection ... is selected.
   b. [Optional] Under Maximum detection age, modify the time limit for excluding older reports as needed. This limit is 30 minutes by default.
4. Save your changes.

Configure key elements for using Advanced Threat Defense

Configure key elements for additional scanning by Advanced Threat Defense to adapt important parts of the scanning process to the requirements of your web security policy.

Task

1. Import the Advanced Threat Defense or the ATD - Offline Scanning with Immediate File Availability rule set from the rule set library.
2. On the rule sets tree, select the rule set that you have imported.
   Key elements of the rules for the scanning process appear in the configuration pane.
3. Configure the key elements as needed.
4. Click Save Changes.

Configure settings for using Advanced Threat Defense

You can configure settings for the Anti-Malware module (or *engine*) on Web Gateway to enable the use of Advanced Threat Defense for scanning web objects.

Task

1. Select Policy → Settings.
2. On the Engines branch of the settings tree, expand Anti-Malware and select the settings for configuring the use of Advanced Threat Defense.
   After importing the Advanced Threat Defense library rule set, the name of these settings is Gateway ATD.
3. Configure these settings as needed.
4. Click Save Changes.

Gateway Anti-Malware settings

The Gateway Anti-Malware settings are settings for the Anti-Malware module (engine) that are by default available after the initial setup of Web Gateway.

Monitoring the use of Advanced Threat Defense

Several methods are available for monitoring the scanning activities that are performed by Advanced Threat Defense when it is used to support Web Gateway.

The monitoring can be done on Web Gateway and on McAfee Content Security Reporter.

Monitoring the use of Advanced Threat Defense on Web Gateway

On Web Gateway, you can implement rule sets with rules for logging information about the scanning jobs that Advanced Threat Defense performs and for handling errors that occur during these jobs.

You can also review Advanced Threat Defense activities on the dashboard of the user interface.

- **Log Handler** — The *ATD Scanning Log* rule set can be imported from the *Logging* group of rule sets in the rule set library.

  The rule set contains a logging rule that records information about each scanning job Advanced Threat Defense performs on a web object that was passed on to it by Web Gateway.

  This information includes:

  - Severity grade that is the result of scanning
  - Server that Advanced Threat Defense runs on
  - Task ID for a scanning job
  - Hash value for a scanning job

  To create the log entries that provide this information, the rule uses suitable properties.
- **Error Handler** — The *Block on ATD Errors* rule set can be imported from the Error Handling group of rule sets in the rule set library.

  It contains blocking rules for handling errors that occur when Advanced Threat Defense performs a scanning job.

  The rules use the appropriate error IDs in their criteria. The error IDs range from 14010 to 14012.

  A rule in the Block on Anti-Malware Engine Errors rule set covers the range from 14002 to 14050. The Block on ATD Errors rule set should, therefore, be placed before this anti-malware rule set.

  Otherwise, the blocking rules in the Block on ATD Errors rule set would never be processed and only block messages with text that is related to anti-malware errors in general would be sent to users.
- **Anti-Malware properties** — Several properties are available for monitoring the activities of Advanced Threat Defense. Their names begin with *Antimalware.MATD*, for example, *Antimalware.MATD.Server* or *Antimalware.MATD.Report*.

  These properties are used in the logging rules of the ATD Scanning Log rule set.

  When a scanning job has been performed by Advanced Threat Defense, the value of the *Antimalware.MATD.Report* property is a report on this job. The report is provided as a string that represents the data structure of a JavaScript Object Notation (JSON) object.

  Using JSON properties together with the *Antimalware.MATD.Report* property, you can extract report information.
- **Dashboard** — The dashboard charts and tables show how the following data evolved during a particular time interval.

  - Under Executive Summary: Number of requests for web objects that were blocked due to the scanning results found by Advanced Threat Defense.
  - Under Malware Statistics: Number of requests for web objects that were passed on to Advanced Threat Defense for scanning, number of requests that were blocked due to the scanning results, and the time consumed for the scanning.

Monitoring the use of Advanced Threat Defense on Content Security Reporter

With McAfee® Content Security Reporter, you can collect data about the scanning activities that Advanced Threat Defense performs when it is used to support Web Gateway.

- To collect the data, configure both Web Gateway and Advanced Threat Defense as log sources.
- To view the data, register the server that Advanced Threat Defense runs on. You can then view the data on the dashboard monitor.

For more information, see the *McAfee Content Security Reporter Product Guide*.

Data loss prevention

Data loss prevention (DLP) ensures that sensitive content is not allowed to leave your network. The prevention process detects this content and blocks traffic going out to the web accordingly.

The following elements are involved in this process:

- Data loss prevention rules that control the process
- Default classifications and a dictionary that you fill with entries for data loss prevention
- Data loss prevention modules, which are called by the rules that are processed to find out about sensitive content

You can also use data loss prevention rules to keep inappropriate content from entering your network. However, this can have an impact on performance.

The data loss prevention process can be applied to text contained in the body that is sent with a request or response or to any other text that is contained in requests or responses, for example, URL parameters or headers.

When you are running the appliance together with a DLP solution that uses an ICAP server for the filtering process, you can implement a rule set to ensure the smooth flow of data between the appliance and the ICAP server.

Data loss prevention rules

Data loss prevention is not implemented by default on the appliance, but you can import the *Data Loss Prevention* rule set from the library.

You can then review the rules of this rule set, modify or delete them, and also create your own rules.

A data loss prevention rule blocks, for example, a request if the text that is sent as its body includes sensitive content. To find out whether this is true for a given request body, the rule calls a module that inspects the body. To know what is considered sensitive, the module refers to the default classifications on the system lists or to dictionary entries, according to what is configured.

When a request or response is processed, its body text is stored as the value of the *Body.Text* property. Before body text can be stored and inspected, it must be extracted. The Composite Opener module performs the opening jobs. A rule in a rule set of the *Common Rules* rule set enables the opener by default.

A request body could, for example, be a text file that uploading to the web is requested for. The value of a suitable body-related property in the rule criteria would then have to be true for the rule to apply and execute the blocking.

The following rule uses the *DLP.Classification.BodyText.Matched* property in this way. If a request includes sensitive content in its body, this is detected by the data loss prevention module. The value of the property is set to *true*, and the request is blocked.

Name	
Block files with SOX information	
Criteria	Action
DLP.Classification.BodyText.Matched<SOX> –> equals true	Block<DLP.Classification.Block>

When this rule is processed, the data loss protection module knows, due to its settings, that it has to look for content that is sensitive with regard to the SOX (Sarbanes-Oxley) regulations, which deal with responsibilities of public companies.

Events can be added to the rule to log information on data loss prevention or to increment a counter that counts how often it has occurred that a request is blocked due to this rule.

Default classifications and dictionary entries

Default classifications and dictionary entries are used in data loss prevention to specify sensitive content that should be prevented from leaving your network.

However, you can also use system lists and dictionary entries to specify inappropriate content, such as discriminatory or offensive language, that should not be allowed to enter your network. Inappropriate content could, for example, be specified this way to let a rule block content sent from web servers in response to requests.

The library rule set for data loss prevention contains a nested rule set for processing body text in the response cycle.

Default classifications and dictionary entries differ in the following ways:

- **Default classifications** — Provide information for detecting different kinds of sensitive or inappropriate content, for example, credit card numbers, social security numbers, or medical diagnosis data.

  Default classifications are contained in folders and subfolders on system lists and updated by the appliance system. You can view the system lists under DLP Classification in the System Lists branch of the lists tree, but you cannot edit or delete them.

  When you edit the settings of the module that handles classifications, you can select suitable subfolders from the folders on these lists and create a list with classifications for data loss prevention in your network.
- **Dictionary entries** — Specify sensitive or inappropriate content, for example, names of persons or keywords indicating content that should not leave your network

  The dictionary is created as part of the settings for the module that handles this list.

  Creating a dictionary and filling it with entries for sensitive or inappropriate content is a means to configure the data loss prevention process beyond what is possible by using the default classifications on the system lists. This way you can adapt the process to the requirements of your network.

Data loss prevention modules

The job of the data loss prevention modules (also known as *engines*) is to detect sensitive or inappropriate content in the body text of requests and responses and also in any other text that is sent with requests and responses.

When composite objects, such as archive documents, bodies of POST requests, and others, are sent with requests or responses, they are also included in the data loss prevention process. To account for such objects, the data loss prevention rules are also processed in the embedded objects cycle.

Depending on what the data loss prevention modules find out, body-related properties in rule criteria are set to *true* or *false*, so web traffic is eventually blocked or allowed.

There are two modules that differ in their use of lists for detecting relevant content:

- **Data Loss Prevention (Classifications)** — Uses default classifications on system lists for data loss prevention
- **Data Loss Prevention (Dictionaries)** — Uses dictionaries with entries for sensitive and inappropriate content that you provide yourself for data loss prevention

When configuring settings for the modules, you let them know which content to look for. The default classifications and dictionary entries that specify the content are among the settings parameters.

Search methods for data loss prevention

There are different methods of searching content that should be prevented from leaving or entering your network.

- A search can aim at finding out whether a given request or response body includes portions of content that are specified as sensitive or inappropriate.
- A search can begin with a portion of content, for example, an URL parameter or header, and find out whether it is sensitive or inappropriate according to what you have configured.

For the first method, you can use the *DLP.Classification.BodyText.Matched* property that was already shown in a sample rule.

For the second, you can use the *DLP.Classification.AnyText.Matched* property. This property takes a string parameter for the content portion that is checked for being on a system list or in a dictionary.

Depending on what you are working with, you would use the two already mentioned parameters with system lists and *DLP.Dictionaries.BodyText.Matched* and, *DLP.Dictionaries.AnyText.Matched* with the dictionary.

Logging data loss prevention

Additional properties are provided for logging the results of the data loss prevention process. They allow you to log this data, for example, using an event in a rule.

When the value of *DLP.Classification.BodyText.Matched* is *true* for the body text of a request or response that was processed, the following applies for the relevant logging properties:

- *DLP.Classification.BodyText.MatchedTerms* contains a list of the matching terms from the body text
- *DLP.Classification.BodyText.MatchedClassifications* contains a list of the matching classifications

When the value of *DLP.Dictionary.BodyText.Matched* is *true*, *DLP.Dictionary.BodyText.MatchedTerms* contains a list of all matching terms.

Similarly, matching terms and classifications can be logged for the search method that looks for matches of a given text string.

When the value of *DLP.Classification.AnyText.Matched* is *true*:

- *DLP.Classification.AnyText.MatchedTerms* contains a list of matching terms found in text other than body text.
- *DLP.Classification.AnyText.MatchedClassifications* contains a list of matching classifications found in text other than body text.

When the match is in a dictionary, *DLP.Dictionary.AnyText.Matched* is *true* and *DLP.Dictionary.AnyText.MatchedTerms* contains a list of matching terms.

Information on data loss prevention results is also shown on the dashboard.

Preventing loss of medical data

The following is an example of data loss prevention that assumes medical data must be prevented from leaving the network of an American hospital.

Default classifications for preventing the loss of medical data are contained in the HIPAA (Health Insurance Portability and Accountability Act) folder. In addition to this default information, the names of the doctors who are working in the hospital are entered in a dictionary to ensure they also do not leave the network.

The following activities need to be completed for configuring data loss prevention in this example:

- Configure settings for the Data Loss Prevention (Classifications) module that include the default HIPAA classifications
- Configure settings for the Data Loss Prevention (Dictionaries) module that include the doctors' names as dictionary entries
- Make sure the rule that activates the Composite Opener is enabled

  In the default rule set system, this rule is contained in the Enable Opener rule set, which is nested in the Common Rules rule set.
- Create a rule that checks content according to the configured settings

  The rule must be included in a rule set that applies in the request cycle for request to upload data from the hospital network to the web.

  This rule set can be a nested rule set of the default rule set for data loss prevention or a rule set that you create yourself.

  In this example, the rule checks only text contained in the body of a request. It could look as follows:

Name	
Prevent loss of HIPAA data and doctors' names	
Criteria	Action
DLP.Classification.BodyText.Matched<HIPAA>> *equals true* AND *DLP.Dictionary.BodyText.Matched<Doctors'Names>* *equals true*	Block<DLP.Classification.Block>

Configure data loss prevention

You can configure data loss prevention to keep sensitive content from leaving your network. You can also use it to keep inappropriate content from entering.

Complete the following high-level steps.

Task

1. Import the Data Loss Prevention rule set from the library.
2. Review its rules and modify them as needed.

You can, for example:

- Configure settings for data loss prevention using default classifications.
- Configure settings for data loss prevention using dictionary entries.
- Modify other settings parameters.
- Create rules of your own.

You can also create your own rule set for data loss prevention instead of using the library rule set.

3. Make sure the Composite Opener is enabled, so the body text sent with requests and responses can be inspected.
   In the default rule set system, this rule is contained in the Enable Opener rule set, which is nested in the Common Rules rule set.
4. If you want to run data loss prevention with ICAP, you can import another rule set from the library and modify its rules as needed.
5. Save your changes.

Configure data loss prevention using default classifications

You can configure data loss prevention by selecting default classifications from system lists and entering them in a list that is included in the settings of the data loss prevention module for processing classifications.

Task

1. Select Policy → Settings.
2. On the settings tree, select Data Loss Prevention (Classifications) and click Add.
   The Add Settings window opens.
3. Configure general settings parameters:
   a. In the Name field, type a name for the settings.
   b. [Optional] In the Comment field, type a plain-text comment on the settings.
   c. [Optional] Click the Permissions tab and configure who is allowed to access the settings.
4. On the toolbar of the DLP Classifications inline list, click the Edit icon.
   An Edit window opens with a tree structure of folders containing subfolders with default classifications.
5. Expand a folder, for example, SOX Compliance, and select a subfolder, for example, Compliance Reports. Then click OK.
   You can also select several subfolders of a folder at once, select folders from different subfolders, or select complete folders with all their respective subfolders.
   The Edit window closes and the subfolder or subfolders appear in the DLP Classifications inline list.
6. Click Save Changes.

Configure data loss prevention using dictionary entries

You can enter text and wildcard expressions that specify sensitive or inappropriate content into as entries in a dictionary for data loss prevention.

After importing the library Data Loss Prevention rule set, use of a dictionary with entries specifying sensitive or inappropriate content is not yet implemented. You need to create appropriate settings to implement it and fill the dictionary with entries.

Create settings with a dictionary

For data loss prevention that uses dictionary entries, you must create settings that include a dictionary.

Task

1. Select Policy → Settings.

2. On the settings tree, select Data Loss Prevention (Dictionaries) and click Add.
   The Add Settings window opens.
3. Configure general settings parameters:
   a. In the Name field, type a name for the settings.
   b. [Optional] In the Comment field, type a plain-text comment on the settings.
   c. [Optional] Click the Permissions tab and configure who is allowed to access the settings.

Results

You can now fill the dictionary with entries.

Fill the dictionary with entries

After creating settings with a dictionary, you can fill the dictionary with entries.

Task

1. Within the settings you have created for data loss prevention using dictionary entries, click the Add icon on the toolbar of the Dictionary inline list.
   The Add DLP Dictionary Entry window opens.
2. Under Type of data to search, select Text or Wildcard expression.
3. In the Text or wildcard expression field, enter a text string or a wildcard expression.
4. [Optional] Specify additional information for an entry:
   - If you have entered a text string, select one of the following options or any combination of them:
     - Case-sensitive
     - At start of word
     - At end of word
   - If you have entered a wildcard expression, select Case-sensitive or leave it deselected as needed.
5. [Optional] In the Comment field, type a plain-text comment on an entry.
6. Click OK.

   The Add DLP Dictionary Entry window closes and the new entry appears in the dictionary.

   Repeat Steps 1 to 6 to add more entries.
7. Click OK in the Add Settings window.
   The window closes and the new settings appear on the settings tree under Data Loss Prevention (Dictionaries).

Best practice: Set a size limit of your own for DLP filtering

When implementing the cloud version of Data Loss Prevention (DLP), you can modify the default rules of the relevant rule set. You can, for example, replace the default size limit for the filtering process with one of your own.

This task also shows how the two views of a rule set can be used for different purposes. In the key elements view, you can disable the default size limit or enable it. But you cannot set a size limit of your own.

To complete the latter activity, which is a bit more complex, you must work with the complete rule sets view.

Task

1. Import the Data Loss Prevention (DLP) with ICAP for Cloud rule set from the library.
2. In the key elements view of the rule set that appears after the import, click Unlock View and Yes in the confirmation window.

   The complete rules view of the rule set appears.
3. Click Show Details.
4. Set a size limit that replaces the default value.
   a. Select the Skip Body That Is Greater Than 50 MB rule and click Edit.

b. In the Edit Rule window, select Rule Criteria. Then select the line with the criteria and click Edit again.
c. Under Compare with in the Edit Criteria window, type a new size limit.
   For example, type 80000000 instead of 52428800, which is the default value.
d. Click OK to close the Edit Criteria window.
e. In the Edit Name window, select Name. Then modify the rule name in the Name field by typing `80` instead of `50` (if you chose that as the new size limit).
f. Click Finish to close the Edit Rule window.

5. Click Save Changes.

Results

Requests with a body greater than the default size limit, but still below the newly configured limit, are now involved in the filtering process.

Preventing data loss using an ICAP server

When you have implemented data loss prevention with an ICAP server that handles the filtering process, you can configure settings and implement a rule set to ensure the smooth flow of data between the appliance and the ICAP server.

You can use a solution called nDLP for data loss prevention. Within this solution, data that users want to upload from your network to the web is filtered to prevent data loss. The filtering is done on an ICAP server. The data flow is as follows:

- Data sent from the client systems of your users is forwarded to the appliance.
- The appliance provides an ICAP client that sends REQMOD requests with the user data to the ICAP server.
- The requests are filtered on the server by modifying them according to the ICAP protocol and passed on to the web servers that are the destinations of the requests.

After importing the *Data Loss Prevention with ICAP* rule set from the library, rules that are implemented on the appliance control the sending of requests to the ICAP server.

According to these rules, a request is not forwarded if:

- The body of the request contains no data and the request does not include URL parameters.
- The body of the request exceeds a given size (default: 50 MB).

Together with the rule set, settings are imported that you need to configure. These include a list of the ICAP servers that the appliance can forward requests to.

You can also configure the ICAP client on the appliance not to open more connections for sending requests than a particular ICAP server can handle at the same time.

Create an ICAP server list for data loss prevention

When running the nDLP solution for data loss prevention, which uses an ICAP server for filtering data, you need to configure a list of these servers.

Task

1. Select Policy → Settings.
2. On the settings tree, select ICAP Client and click the ReqMod settings.
3. Configure the the ICAP server list that is provided under these settings as needed.
4. Click Save Changes.

Using an on-premise DLP server from the cloud

You can perform DLP filtering using an on-premise DLP server with an ICAP client that runs in the cloud.

You can, however, only implement this method of DLP filtering if you are using the hybrid solution for Web Protection, which includes both Web Gateway and McAfee Web Gateway Cloud Service.

The ICAP client is made available by importing a rule set on Web Gateway, which is already configured for cloud use. Settings for this solution are imported with the rule set.

By modifying the rules in the rule set or the settings that are imported with it, you can adapt this solution to your requirements.

ICAP configuration

The DLP server takes the role of the ICAP server in this configuration. The server must be placed in a DMZ where it can have a public IP address, as the ICAP client connects to it using this type of address.

Internal and other protected addresses, for example, internal IP addresses of McAfee Web Gateway Cloud Service, must not be used in the cloud and are therefore excluded by a check that the ICAP client performs.

ICAP client and server also send health check messages to each other in regular intervals.

Note: We recommend using ICAPS (Secure ICAP), as data is transferred in plain text format when normal ICAP is used. The ICAPS client does, however, not validate the certificate that the DLP server sends in its role as ICAPS server.

ICAP-related properties for cloud use

The properties listed in the following can be used in rules for the filtering process on McAfee Web Gateway Cloud Service.

For example, the ICAP.ReqMod.Satisfaction property is used in a rule of the library rule set for Data Loss Prevention filtering using a cloud ICAP client.

- Properties for the ReqMod mode:
  - ICAP.ReqMod.Satisfaction
  - ICAP.ReqMod.ResponseHeader.Exists
  - ICAP.ReqMod.ResponseHeader.ExistsMatching
  - ICAP.ReqMod.ResponseHeader.Get
  - ICAP.ReqMod.ResponseHeader.GetMatching
  - ICAP.ReqMod.ResponseHeader.GetMultiple
  - ICAP.ReqMod.ResponseHeader.GetMultipleMatching
- Properties for the RespMod mode:
  - ICAP.RespMod.ResponseHeader.Exists
  - ICAP.RespMod.ResponseHeader.ExistsMatching
  - ICAP.RespMod.ResponseHeader.Get
  - ICAP.RespMod.ResponseHeader.GetMatching
  - ICAP.RespMod.ResponseHeader.GetMultiple
  - ICAP.RespMod.ResponseHeader.GetMultipleMatching

Configure the use of an on-premise DLP server from the cloud

To configure the use of an on-premise DLP server from the cloud, complete the following procedure.

Task

1. Configure the network components that run in this solution with Web Gateway.
   a. Place the DLP server in a Demilitarized Zone (DMZ), so that it can have a public IP address.
      The ICAP client in the cloud must be able to connect to the DLP server in its role as ICAP server using the public IP address of this server.
   b. Configure the firewall to accept requests from McAfee Web Gateway Cloud Service on a dedicated port and a specific set of IP addresses. The port which must be the ICAP server port that is also configured on the ICAP client.

      The port which must be the ICAP server port that is also configured on the ICAP client. Use port 1344 when working with ICAP and port 11344 when working with ICAPS.

      There are several public IP addresses of McAfee Web Gateway Cloud Service that must be whitelisted on the firewall. For these IP addresses, see this Knowledge Center article: KB87232.

Refer the following link to know the public facing IP addresses of McAfee's WGCS which needs to be whitelisted on the firewall:

2. On Web Gateway, import the Data Loss Prevention (DLP) with ICAP for Cloud rule set from the library.

   Note: The rule set is by default configured for use in the cloud.
3. Review the Reqmod for Cloud settings, which are imported with the rule set.

Authentication

Users can be “filtered” on an appliance, which means you can allow web access only for those who are able to authenticate.

Authentication is not implemented by default, but there are preconfigured authentication rule sets, which you can use.

The types of authentication that you can implement include:

- **Standard authentication** — You can configure authentication for users who send requests for web access under a standard protocol, such as HTTP, HTTPS, or FTP.

  When the authentication rule set of the default rule set system is enabled, user information is by default retrieved from an internal user database.

  You can change this setting and configure a different method, such as NTLM, LDAP, Kerberos, and others.
- **Instant messaging authentication** — You can configure authentication for users who send requests for web access under XMPP, which is the protocol used for several instant messaging services such as Jabber, Google Talk, Facebook Chat, and others.

You can also control administrator access to an appliance by setting up and maintaining administrator accounts and roles.

Authenticating users

Authenticating the users of your network ensures that they cannot access the web if they do not submit appropriate information about themselves. The authentication process looks up user information, for example, in an internal database or on a web server and blocks or allows access accordingly.

This process includes several elements:

- Authentication rules, which control the process
- Authentication module, which is called by the rules to retrieve user information

An authentication process is not implemented by default on Web Gateway after the initial setup. You can implement a process by importing suitable rule sets from the rule set library and modify it to meet the requirements of your organization.

Authentication rules

Authentication rules usually include a rule that asks an unauthenticated user to authenticate and blocks requests from users who are not successfully authenticated.

There can also be whitelisting rules that allow users to skip authentication. Skipping might be allowed, for example, depending on the user group that a user belongs to or on the URL of a requested web object.

Rule sets for several authentication types and methods are available in the rule set library.

Authentication module

The Authentication module (engine) retrieves information about users from databases. The module is called by the rules that need to know whether a user who requests access to a web object is authenticated.

Methods of retrieving this information are:

- NTLM — Uses a database on a Windows domain server.
- NTLM Agent — Uses an external agent on a Windows-based system for applying the NTLM authentication method.
- User Database — Uses an internal database on the appliance.
- LDAP — Uses a database on an LDAP server.
- Novell eDirectory — Uses data from a directory on a server that takes the role of an LDAP server.
- RADIUS — Uses a database on a RADIUS server.
- Kerberos — Uses a database on a Kerberos server.
- Authentication Server — Uses a database on another external server.

To select the authentication method and set other parameters of the authentication process, you configure the settings of the Authentication module.

Configure authentication

You can implement authentication and adapt it to the needs of your network.

Complete the following high-level steps.

Task

1. Enable the Authenticate and Authorize rule set of the default rule set system.
2. Review the nested Authenticate with User Database rule set.

   This rule set contains a single rule, which asks unauthenticated users to authenticate.

   The rule criteria includes settings for the Authentication module, which specify use of the User Database authentication method. This means information for authenticating users is retrieved from an internal database on the appliance.
3. Modify the default rule set as needed.
   You can, for example, do the following:
   - Modify the common parameters of the Authentication module
   - Modify the specific parameters for the User Database method
   - Implement a different authentication method, for example, NTLM or LDAP
   - Modify the specific parameters for the new authentication method
4. Consider importing a rule set from the library to implement authentication for a different type of communication, for example, instant messaging authentication.
5. Save your changes.

Configure the Authentication module

You can configure the Authentication module to modify the way user information is retrieved to authenticate users.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select the rule set for authentication.
   In the default rule set system, this is the *Authenticate and Authorize* rule set.
3. Select a rule that controls user authentication and click the settings that are specified in the rule criteria.
   In the rule set of the rule set system, this is, for example, the rule Authenticate with User Database in the nested Authenticate with User Database rule set and the settings name is User Database.
   The Edit Settings window opens. It provides the settings for the Authentication module.
4. Configure these settings as needed.
5. Click OK to close the window.
6. Click Save Changes.

Implement a different authentication method

If you do not want to use the User Database authentication method of the default rule set, you can implement a different method, such as NTLM, LDAP, and others.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, navigate to the rule set that contains rules for authenticating users, for example, the default Authentication and Authorize rule set and select the nested Authenticate with User Database rule set.
   The rules of the nested rule set appear on the settings pane.
3. Select the rule Authenticate with User Database and in the rule criteria click User Database.

The Edit Settings window opens.

4. From the list provided under Authentication Method, select an authentication method, for example, NTLM.
5. Configure common and specific parameters for the selected method as needed.
6. Click OK to close the window.
7. Click Save Changes.

What to do next

We recommend that after changing the authentication method, you rename the settings of the Authentication module, the authentication rule, and the nested rule set, accordingly.

For example, after selecting NTLM, rename the settings to `NTLM` and both the rule and the nested rule set to `Authenticate with NTLM`.

Instead of renaming the default settings, you can also keep several settings with different names and parameter values for the Authentication module.

Using system settings to configure authentication

For some authentication methods, you need to configure settings that are not settings of the Authentication module, but of the appliance system.

This applies when you are implementing NTLM as the authentication method. In this case, you need to join the appliance to a Windows domain and configure the *Windows Domain Membership* settings, which are system settings.

It applies also for the Kerberos authentication method, which is implemented using the *Kerberos Administration* system settings.

Join the appliance to a Windows domain

When using the NTLM authentication method, you need to join an appliance to a Windows domain to let the authentication module retrieve user information stored on the domain server.

An appliance can be joined to more than one domain.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to join and click Windows Domain Memberhship.

   A list of domains appears on the settings pane. It is initially empty.
3. Click Join to enter a domain into the list.

   The Join Domain window opens.
4. Configure a domain name, a domain controller, and other settings in the window.
5. Click OK.

   The window closes and the new domain appears in the list. The appliance is now a member of this domain.

   Repeat Steps 3 to 5 to add multiple domains.

What to do next

Use the other icons on the toolbar to work with the list, for example. to modify a list entry or to let an appliance leave a domain.

NTLM Agent authentication

NTLM Agent authentication uses a separate software product, known as the NTLM Agent, for authenticating users on Web Gateway.

This authentication methods is an option, for example, when the connection between Web Gateway and the domain controller that is involved in the authentication process is blocked by a firewall. The NTLM Agent only requires a single freely definable port to be opened for connecting to the domain controller.

Configuring settings for the NTLM Agent

The NTLM Agent is installed on one or several WIndows systems in your network. On these systems, it runs as a service that performs its authentication tasks in the background.

This service can, however, not be accessed at the system desktop. After the NTLM Agent has been installed on a system, it is therefore available as an application, which can be accessed in a directory of that system. This directory is usually the program files directory.

When the application is accessed, it opens a configuration window at the system desktop, which allows you to configure settings for the NTLM Agent.

The NTML Agent service and its application communicate with each other, so the configuration settings that you implement using the application are applied to the service.

In addition to configuring settings using the configuration window of the NTML Agent, you must also configure NTLM Agent settings on the user interface of Web Gateway.

Configure NTLM Agent authentication

Configure NTLM Agent authentication both in the user interface of Web Gateway and the configuration window of the NTLM Agent.

Task

1. Download and install the NTLM Agent software.
   a. Go to the Cloud & Content Security portal at https://contentsecurity.mcafee.com/software_mwg7_download.
   b. Navigate to Products → McAfee Web Gateway 6 → Downloads → Tools.
   c. In the NTLM Agent section, click the .exe icon.
   d. Follow the instructions of the installation program.
2. Configure the settings that are provided by the NTLM Agent.
   a. After the NTLM Agent software has been installed on a system in your network, navigate to the NTLMAgent.exe file in the directory where the software was installed.
   The path to this file might be, for example, *C:\\Program Files\Secure Computing\NTLMAgent*.
   b. Click the .exe file
   A menu with basic options for working with the NTLM Agent opens.
   c. Click Configure.
   The NTLM Agent configuration window opens.
   d. Use the NTLM Agent window to configure the NTLM Agent settings.
3. Configure the settings that are provided on the user interface of Web Gateway.
   a. Select Policy → Settings.
   b. On the settings tree, expand Authentication and click one of the settings, for example, the User Database settings.
   The settings appear in the configuration pane.
   c. From the list under Authentication methods, select NTLM Agent.
   The NTLM Agent Specific Parameters section appears below the Common Authentication Parameters section.
   d. Work with the options of this section to configure NTLM Agent settings.

LDAP digest authentication

The LDAP digest authentication method, which is based on the LDAP authentication method, uses a shared secret known by both sides of the authentication process: a user requesting web access, using a browser on a client of Web Gateway, and Web Gateway.

Web Gateway uses its proxy functions to intercept the request to enable authentication and further filtering under the configured web security policy.

Unlike simpler authentication methods, such as basic authentication, no password is sent directly from the browser to Web Gateway. Instead the password is a part of the shared secret that is known on both sides of the authentication process.

A hash value is calculated for the shared secret and several additional parameters on the client and transmitted to Web Gateway, which calculates the hash again, using its instance of the shared secret, to see if the result is identical. If it is, the user is authenticated.

The hash value that is transmitted from the client to Web Gateway is also referred to as *digest*. Web Gateway retrieves the shared secret that it requires for recalculating the hash from an LDAP server.

Calculating a hash for LDAP digest authentication

The MD5 method for calculating a hash is used when LDAP digest authentication is performed in an authentication process with Web Gateway.

Before the client sends the hash, Web Gateway sends a request for authentication to the client, including a so-called *nonce* (number only once), which is a number that is randomly created on Web Gateway and is one of the parameters that must be used in addition to the shared secret for calculating the hash.

The complete list of parameters that is used for calculating the hash includes the following:

- User name (part of the shared secret)
- Realm name (part of the shared secret)
- Password (part of the shared secret)
- Nonce
- HTTP request that was sent from the client
- URL of the requested destination in the web

Configuring LDAP digest authentication on Web Gateway

LDAP digest authentication on Web Gateway requires the following:

- LDAP authentication must have been configured as the general authentication method on Web Gateway.
- The realm name must be configured as part of the common authentication settings on Web Gateway. This name must also be used for the shared secret.
- You must configure the following parameters for LDAP digest authentication:
    - Enabling of LDAP digest authentication
    - Name of the attribute on the LDAP server that stores the authentication hash
    - Maximum number of times that a nonce can be used
    - Maximum time that a nonce can be used

Optionally, you can do the following.

- Allow only LDAP digest authentication as an authentication method under the current settings

  When configuring other authentication settings, you could, however, still allow other authentication methods, for example, the User database method with basic authentication.
- Let a check be performed for the URL that a client sends as a parameter for calculating the hash

  This URL should be te same as the URL that this client sends in its request for accessing a particular destination in the web. Otherwise successfully passing digest authentication, based on identical hash values, might allow a user to access a destination that was not requested. So if the result of the check is that both URLs are not the same, the request is blocked.

  As the browsers that are used on clients for sending this information use different URL formats, this check might fail, however, due to the formatting problem, even if two URLs are really the same. For this reason, the URL check is optional.

The realm name that is used for the shared secret is configured under Common Authentication Parameters, which is a section that is available under every authentication method at the beginning of the Authentication settings.

The parameters for LDAP digest authentication are configured on Web Gateway as part of the settings for the Authentication module (or *engine*).

When LDAP is selected as the general authentication method at the beginning of these settings, a section named Digest Authentication becomes available after the section for other LDAP specific parameters.

Enforce RADIUS authentication through use of a pam-radius module

You can enforce RADIUS authentication for users who access Web Gateway remotely with SSH or run sudo commands in an unprivileged mode.

To enforce this authentication method you can install a module on Web Gateway, which is known as the pam-radius module, and configure it in a suitable manner.

When this module is installed and configured, RADIUS authentication will be mandatory for users who attempt to do one of the following:

- Log on to Web Gateway from a remote system console using SSH
- Run sudo commands for Unprivileged Users from a system console

Installation and configuration of the module is completed using a local system console. After installing the module, check whether the pam_radius_auth.so system file has been created on Web Gateway.

For troubleshooting issues with SSH access, you can review the /var/log/secure system file on Web Gateway.

You can also add the *-v* parameter when running SSH to increase the output verbosity.

Install and configure a pam-radius module

You can install and configure a pam-radius module on Web Gateway to enforce RADIUS authentication for users when logging on to Web Gateway remotely with SSH or running sudo commands in an unprivileged mode.

Installing and configuring the module is provided as an option on Web Gateway and can be completed using a local system console.

Task

1. Log on to the Web Gateway appliance where you want to install and configure the pam-radius module.
2. Run this command to install the module:
   ```
   yum install pam_radius
   ```
3. Edit the /etc/pam_radius.conf system file to configure the server where the information for completing the RADIUS authentication method is stored.
   a. Comment out these lines:
      ```
      #127.0.0.1 secret 1
      ```
      ```
      #other-server other-secret 3
      ```
   b. Add information about the RADIUS server:
      ```
      <IP address of the RADIUS server> <shared secret> <timeout in seconds>
      ```
4. Edit the /etc/pam.d/sshd system file to configure use of the RADIUS authentication method when logging on with SSH.
   a. Comment out this line:
      ```
      #auth substack password-auth
      ```
   b. Add this line to enforce RADIUS authentication:
      ```
      auth required pam_radius_auth.so
      ```
5. Edit the /etc/pam.d/sudo system file to configure use of the RADIUS authentication method when running sudo commands in an unprivileged mode.
   a. Comment out this line:

```
#auth include system-auth
```

   b. Add this line to enforce RADIUS authentication:

```
auth required pam_radius_auth.so
```

6. Create unprivileged users and enable them to run sudo commands.
   a. Run this command to create a user:

```
useradd <user name>
```

   b. Add this line to the /etc/sudoers system file in order to enable a user to run sudo commands:

```
<user name> ALL=(ALL) ALL
```

Note: We recommend that you use the visudo command to edit this file.

7. On the RADIUS server, add a user name and password to submit for authentication.

   The user name must be the same as the user name that you have configured in step 6a.
8. On the local system console that is connected to Web Gateway, restart SSHD.

```
service sshd restart
```

Results

You have now enforced RADIUS authentication for users who connect to Web Gateway with SSH or run sudo commands in an unprivileged mode.

To verify if the enforcement works:

- Log on to Web Gateway from a remote system console with SSH.

  You should be prompted to authenticate under RADIUS.
- On the remote system console Web Gateway, run a sudo command as follows:

```
sudo <command name>
```

  You should be prompted to authenticate under RADIUS.

Retrieving user group lists from an Azure AD

Lists of user groups can be retrieved from an Azure Active Directory (Azure AD) for authentication purposes when a web security policy is enforced for cloud users through McAfee Web Gateway Cloud Service.

The lists are retrieved in string format to provide a value for an authentication property, which can be used in web security rules to allow or block web access depending on user groups.

You can create these rules on Web Gateway and enable them for cloud use, so that they also apply to McAfee Web Gateway Cloud Service.

To retrieve information from an Azure AD, you must configure options for communication between it and Web Gateway.

User requests for web access from outside your local network

When users of your organization request web access from outside your local network, for example, while traveling or working at home, you can enforce a web security policy for this access using McAfee Web Gateway Cloud Service (WGCS). McAfee Client Proxy then redirects these requests to WGCS.

Client Proxy also adds information about the name and group of the user who sent the request that is redirected. It retrieves the user group information from lists of domain groups that are provided by Windows. But Windows caches these lists only for a short time, so this information cannot be used in rules for a web security policy on WCGS or Web Gateway.

To let WCGS filter requests from users working outside your local network, you can use group information that is stored in an Azure AD. WCGS can, however, not access information stored in a Windows AD, as it is used on-premise by Web Gateway. So, when creating rules on Web Gateway that you also want to enable for WCGS, you must retrieve any user group information from an Azure AD.

Authentication property for retrieving Azure AD user groups

The Authentication.GetAzureUserGroups property is used in rules that require the retrieval of user group information from an Azure AD.

For example, if you want to allow web access only for users belonging to allowed groups that are listed in an Azure AD, a suitable rule might look like this:

Authentication.GetAzureUserGroups<Azure AD> none in list Allowed User Groups –> Block<Authorized only>

The property has settings, which you must configure as part of the activities that are required to enable communication between Web Gateway and the Azure AD.

The property also has a parameter, which is the user name that Web Gateway submits when attempting to access the Azure AD.

Communication between Web Gateway and an Azure AD

To let Web Gateway retrieve user group information from an Azure AD, you must complete several activities for enabling communication between the two devices.

This includes registering Web Gateway as an application (app) at the Microsoft Application Registration Portal and providing credentials for obtaining permission to read user group information in an Azure AD.

On Web Gateway, you must configure settings that specify the credentials for the read access and other information that is required for the communication process.

Enable communication between Web Gateway and an Azure AD

To retrieve user group lists from an Azure AD, you must enable communication between it and Web Gateway.

Task

1. Make Web Gateway known to the Microsoft environment of the Azure AD.
   a. Register Web Gateway as an application (app) at the Microsoft Application Registration Center.
      The registration includes obtaining an application ID and setting up a password for Web Gateway.
   b. Configure permissions for the newly registered application granting read access to the user group lists in the Azure ID.
2. On Web Gateway, configure settings to connect to the Azure AD.
   a. Create a new instance of the Azure Directory settings and name it appropriately, for example, `Azure AD`.
      Create these settings under Policy → Settings and add them as settings for the Authentication.GetAzureUserGroups property.
   b. Configure options for the following:
      - Application that you registered for Web Gateway
      - Search for user group information in the Azure ID
      - Network setup
   c. Save your changes.

Results

You can now create rules to perform authentication based on lists of user groups that are retrieved from an Azure AD.

Best practices - Configuring authentication for deployment types

When configuring authentication, you need to consider the type of deployment that is configured for handling the traffic between Web Gateway and its clients, such as the explicit proxy mode or a transparent mode. For each type, there is a rule set in the rule set library that is best suited to handle authentication.

The following two questions are important with regard to the authentication process:

- How are the user credentials that are evaluated during this process obtained by Web Gateway?

  Note: This part of the authentication process is sometimes referred to as the *authentication front-end*.

  The method for obtaining user credentials depends on whether the explicit proxy mode (also known as *direct* proxy mode) or a transparent mode (transparent router or bridge mode) is configured for handling the traffic between Web Gateway and its clients.

  For the explicit proxy mode, you can configure that clients use a service under the WCCP protocol to send requests as an additional option.

  The rule set library provides suitable rule sets for each of these modes.
- How should credentials be evaluated once they have been obtained?

Note: This is sometimes referred to as the *authentication back-end*.

The evaluation of credentials depends on the authentication method that is configured, for example, LDAP or NTLM.

Library rule sets for authentication

The rule sets for configuring authentication are located in the *Authentication* rule set group of the rule set library.

The following table shows which of these rule sets are recommended for particular types of deployment.

Library rule sets for authentication

Deployment type	Recommended library rule set
Explicit proxy mode	Direct Proxy Authentication and Authorization
Transparent router or bridge mode	Authentication Server (Time/IP Based Session)
Explicit proxy mode with WCCP	If traffic is processed in: • Explicit proxy mode — Direct Proxy Authentication and Authorization • WCCP mode — Authentication Server (Time/IP Based Session)

After importing a rule set from the library, you can modify its rules to adapt them further to the needs of your network.

Position in the rule sets tree

An authentication rule set should be placed after the Global Whitelist rule set, but before the Common Rules rule set (if you keep these items from the default rule sets tree).

Placing an authentication rule set in this way ensures that a user needs not be authenticated when sending a request for accessing a web object that is on the global whitelist.

Authentication for the explicit proxy mode

When configuring authentication for the explicit proxy mode, a suitable rule set must be implemented on Web Gateway.

Library rule set for the explicit proxy mode

The recommended library rule set for the explicit proxy mode is Direct Proxy Authentication and Authorization.

This rule set has two nested rule sets:

- Authenticate with User Database
- Authorize User Groups

When this rule set is implemented, the authentication process is performed for each request that is received from a client of Web Gateway unless an exception rule applies.

Using this rule set is also the preferred way of handling authentication when Citrix is installed or workstations are shared in a configuration.

Direct Proxy Authentication and Authorization rule set

This rule set contains rules for making exceptions that allow a request to be processed on Web Gateway without authenticating the user who sent the request.

Exceptions can be based on:

- The IP address of the client that a request was sent from
- The URL of the web object that is the destination of the request

Using these rules you can ensure that requests coming in from trusted clients or going out to trusted destinations are spared the effort of performing an authentication process for their users, which increases performance.

You can also create rules of your own and add them to this rule set to allow for more exceptions.

Authenticate with User Database nested rule set

This rule set contains a rule that lets authentication be performed for a user who sends a request for web access from a client of Web Gateway. The user is asked to submit credentials, which are evaluated based on information that is stored in the internal user database.

The rule set applies if the user in question has not yet been authenticated and not tried unsuccessfully to authenticate before. The *Authentication.Is.Authenticated* and *Authentication.Failed* properties are used to check this.

Instead of using information from the internal user database to evaluate the credentials, you can configure a different authentication method, for example, LDAP or NTLM.

Authorize User Groups nested rule set

This rule set contains a rule that allows only requests of authorized users, which means a request is blocked if the user who sent it is not a member of one of the user groups on a particular list. The request is blocked, even if the user has successfully passed the evaluation that was performed before.

This rule allows you to implement an additional security check. If you want to use it, you need to fill the list that is used in this rule set with user groups. If you do not want to use it, you can disable or delete the rule set.

Modifying the rule set for the explicit proxy mode

When configuring authentication for the explicit proxy mode, you can modify the library rule set to adapt it to the needs of your network.

This includes:

- Changing the authentication method
- Modifying, disabling, or deleting user authorization
- Configuring more exception rules

Changing the authentication method

By default, the method used for evaluating credentials is comparing them to the information stored in the internal user database.

To change this authentication method (authentication back-end), you need to configure the settings that appear next to the *Authentication.Authenticate* property in the only rule of the Authenticate with User Database rule set.

Under Authentication method, a list of authentication methods is provided to let you select a method that is better suited to the needs of your network, for example, LDAP or NTLM.

Modifying, disabling, or deleting user authorization

The nested Authorized User Groups rule set allows only requests from authorized users. You can fill the list that is provided in the only rule of this rule set with user groups as needed.

If you do not want to use this rule as an additional security check, you can disable or delete the rule set.

Configuring more exception rules

You can add rules to the Direct Proxy Authentication and Authorization rule set to cover more exceptions from the authentication process.

If any of these rules applies, processing of the rule set is stopped, which means it is not executed for the nested rule sets that handle authentication.

For example, you can add a rule to allow requests when the browser on the client they were sent from runs with a particular user agent. Information about the user agent is taken from the request header.

The rule might look as follows:

Skip authorization for user agents that are in list Allowed User Agents
Header.Request.Get ("User-Agent") matches in list Allowed User Agents –> Stop Rule Set

Another rule could allow requests for access to objects on web servers with IP addresses that are on a particular list. The IP address is taken from the URL that was submitted with a request.

This rule might look as follows:

Skip authorization for destination IPs that are in list Allowed Destination IPs
URL.Destination.IP is in range list Allowed Destination IPs –> Stop Rule Set

Authentication for transparent modes

When configuring authentication for the transparent modes, the settings on the browsers that are used for sending requests to Web Gateway need to be modified. A suitable rule must also be implemented on Web Gateway.

Modifying the browser settings

To enable authentication for the transparent router or bridge mode, the settings of each web browser that is used for sending requests must be configured to let it trust Web Gateway.

If NTLM or Kerberos is also configured as the authentication method on Web Gateway, the authentication process is handled internally, without asking the user to authenticate.

- When using Microsoft Internet Explorer, you need to modify the security settings by:
    - Configuring your local intranet as a security zone
    -

      Adding Web Gateway as a website to this zone

      This is done by specifying a URL with an IP address or a fully qualified domain name, for example, `http://10.10.69.73` or `http://*.mcafee.local`.
    - Configuring automatic logon for all websites in the zone as the security setting for user authentication

  You can configure this under Internet Options, using the Local Intranet and Security Settings - Local Intranet Zone windows.

  If group policies can be configured for a browser, you can also use the Group Policy Management Editor together with the Site to Zone Assignment List and the Logon Options window.
- When using Mozilla Firefox, you need to configure an IP address or a fully qualified domain name for Web Gateway under about:config as the value of the network.automatic-ntlm-auth.trusted-uris parameter, for example, `10.10.69.73` or `mwgappl.yourdomain.local`.

For more information, refer to the documentation of the respective web browser.

Library rule set for transparent modes

The recommended library rule set for the transparent router or bridge mode is Authentication Server (Time/IP Based Session).

It has two nested rule sets:

- Check for Valid Authentication Session
- Authentication Server

Differing from the authentication process that is performed for the explicit proxy mode, this rule set handles authentication by creating an authentication session when a user who sent a request for web access is successfully authenticated.

Subsequent requests that this user sends are processed without requiring authentication again as long as this session is still valid. The default session length is 600 seconds.

Using this rule set in a configuration where Citrix is installed or workstations are shared can lead to the following situation: User A sends a request, is authenticated, and an authentication session is created. Later on, user B sends a request from the same workstation and is still allowed to continue with user A's session.

Authentication Server (Time/IP Based Session) rule set

This rule set serves as a container for the two nested rule sets and has no rules of its own.

Check for Valid Authentication Session nested rule set

This rule set contains a rule that checks whether a valid session exists for a user who sends a request from a client. Session information is stored in an internal session database. It includes the user name, the IP address of the client, and the session length.

If a valid session exists, processing of the request is continued for the remaining rules and rule sets that are configured. If no valid session exists, the request is redirected to the authentication server.

Authentication Server nested rule set

This rule set contains a rule that lets authentication be performed for a user whose request has been redirected to the authentication server. If the authentication was successful, a session is created for this user in the session database.

The method used by default for evaluating the user's credential is comparing them to the information that is stored in the internal user database. You can replace this method with a different method, for example, LDAP or NTLM.

Modifying the rule set for transparent modes

When configuring authentication for transparent modes, you can modify the library rule set to adapt it to the needs of your network.

This includes:

- Modifying the authentication server URL
- Changing the authentication method
- Enabling the ideal conditions rule
- Increasing the session TTL

Modifying the authentication server URL

If you have modified the security settings of the browsers that are used for sending requests to Web Gateway by configuring your local domain as a security zone, you can include Web Gateway as a website in this zone by specifying a URL with an IP address or fully qualified domain name for it.

In this case, you also need to modify the URL of the authentication server, which by default contains an IP address for Web Gateway, by inserting the name of your local domain.

The authentication server URL is dynamically generated for an appliance that Web Gateway runs on. As there can be several Web Gateway appliances in a configuration, the IP address cannot be static, but must be configured dynamically, which is done using internal configuration properties.

You can modify this URL under the IP Authentication Server settings, which appear next to the *Authentication.Authenticate* property in the *Redirect clients that do not have a valid session to the authentication server* rule of the Check for Valid Authentication Session rule set.

By default, the URL looks like this:

```
http://$<propertyInstance useMostRecentConfiguration="false" propertyId="com.scur.engine.system.proxy.ip"/>$:
$<propertyInstance useMostRecentConfiguration="false" propertyId="com.scur.engine.system.proxy.port"/>$
```

Shown in human readable format, a particular authentication server URL could be:

http://10.10.69.71:9090

After adapting the URL to the browser settings that have your local domain configured within a security zone, it looks like this:

```
http://$<propertyInstance useMostRecentConfiguration="false" propertyId="com.scur.engine.system"/>
$.yourdomain.local:$<propertyInstance useMostRecentConfiguration="false"
propertyId="com.scur.engine.system.proxy.port"/>$
```

with `"com.scur.engine.system.proxy.ip"/>$` having been replaced by `"com.scur.engine.system"/>$.yourdomain.local`.

In human readable format, this could be, for example

http://mwgappl.yourdomain.local:9090

where *mwgappl* is the host name of an appliance that Web Gateway runs on.

Changing the authentication method

By default, the method used for transparent modes to evaluate credentials is comparing them to the information stored in the internal user database.

To change this authentication method (authentication back-end), you need to configure the settings that appear next to the *Authentication.Authenticate* property in the *Authenticate user against user database rule* of the Authentication Server rule set.

Under Authentication method, a list of authentication methods is provided to let you select a method that is better suited to the needs of your network, for example, LDAP or NTLM.

Enabling the ideal conditions rule

The *Revalidate session under ideal conditions* rule in the Check for Valid Authentication Session rule set lets a user authenticate again under "ideal" conditions, which means authentication will not be asked for at a time when the session has already expired.

In more detail, these conditions are by default:

- The remaining session time is less than 400 seconds.
- The network protocol is HTTP.
- The request that the user sends is a GET request.

Enabling this rule avoids a situation like the following:

1. A user sends a request from a client of Web Gateway and authenticates (600 seconds are allowed for the session time).
2. The user wants to send a ticket to the help desk and begins filling out a data form (300 seconds are used up).
3. The user needs more information to fill out the form and browses the web for this information, which lets some GET requests be received on Web Gateway (200 more seconds are used up).
4. The user completes the data form and submits it, which lets a POST request be received on Web Gateway (200 more seconds elapse, session time expires after the first 100 seconds).
5. As the session time has expired, the user is asked to authenticate again before the POST request is processed. However, due to the session expiration, all filled-out data is lost.

If the ideal conditions rule is enabled, the user is already asked when browsing for information at step 3 to authenticate again, which leaves enough time to complete the form and submit it.

Increasing the session TTL

You can increase the allowed time for an authentication session, for example, from the default 600 seconds (10 minutes) to an hour.

You can also modify the time condition in the criteria of the *Revalidate session under ideal conditions* rule, for example, by increasing it from 400 to 600 seconds.

This way, the rule will ask a user, upon receiving a GET request, to authenticate when session expiration is still 10 minutes away.

Authentication for the explicit proxy mode with WCCP

Configuring authentication for the explicit proxy mode with WCCP includes import and modification of two rule sets, as well as specifying ports for incoming traffic to trigger the use of the appropriate rule set.

When the explicit proxy mode with WCCP is configured, clients send requests to Web Gateway in explicit proxy mode or using a service under the WCCP protocol.

To handle authentication for the explicit proxy mode, the Direct Proxy Authentication and Authorization rule set is recommended, for the WCCP mode, which is a transparent mode, it is the Authentication Server (Time/IP Based Session) rule set.

This means you should import both rule sets and complete additional activities as needed for both modes, including the modification of the browser settings for the WCCP mode.

To let traffic for each mode be handled by the appropriate authentication rule set, you can configure different ports for both types of traffic and specify the respective port in the criteria of each rule set.

Configuring different ports for the explicit proxy and WCCP modes

The ports for the explicit proxy and WCCP modes could, for example, be 9090 and 9091. You need to specify the port for the WCCP mode when configuring a WCCP service and both ports in the list of HTTP ports.

A WCCP service is configured by entering it in the WCCP Services list. This list appears after selecting WCCP in the Transparent Proxy section of the Proxies (HTTP(S), FTP, ICAP, and IM) system settings.

The section appears within these settings when you begin to configure the explicit proxy mode with WCCP by selecting Proxy (optional WCCP) under Network Setup.

The entry for a WCCP service that is used for traffic coming in on port 9091 could, for example, look as follows:

No	Service ID	WCCP router ...	Ports ...	Ports ...	Proxy listener ...	Proxy listener port	MD5 ...	AssignmentComment
1	91	10.10.69.7	80, 443	false	10.10.69.73	9091	oooooo	1000

The HTTP Port Definition List can be configured in the HTTP Proxy section, which is located below the Transparent Proxy section.

The entries for the explicit proxy and WCCP modes could look as follows:

No	Listener address	Serve ...	Ports ...	Transparent ...	McAfee ...	Comment
1	0.0.0.0:9090	true	443	false	true	Explicit proxy traffic
2	0.0.0.0:9091	true	443	false	true	WCCP traffic

Adapting the criteria of the authentication rule sets

After configuring different ports for traffic coming in under the explicit proxy mode or using a WCCP service, for example, `9090` and `9091`, you need to adapt the criteria of the rule sets for handling the two kinds of traffic.

The adapted rule criteria of the Direct Proxy Authentication and Authorization rule set would then look as follows:

```
Proxy.Port equals 9090 AND (Connection.Protocol equals "HTTP" OR Connection.Protocol equals "HTTPS")
```

For the Authentication Server (Time/IP Based Session) rule set, the adapted criteria would be:

```
Proxy.Port equals 9091
```

Best practices - Configuring LDAP authentication

LDAP authentication is one of the methods that can be configured on Web Gateway for authenticating users.

LDAP stands for Lightweight Directory Access Protocol. Under this protocol, the authentication process on Web Gateway can be integrated with an existing directory service in a network. The directory holds user information, which can be queried and used for authentication.

In addition to authenticating a user, a directory can be queried to find other pieces of information about a user and the groups that a user belongs to. These pieces of information are called *attributes*.

An entry for a user in, for example, the Microsoft Windows Server Active Directory (Active Directory) usually includes a *memberOf* attribute holding information about the groups that the user belongs to. An entry for a group usually has a *member* attribute to hold the group members' user names.

The results returned by lookups for both user and group attributes are stored on Web Gateway as the value of the *Authentication.UserGroups* property.

LDAP authentication process

The process that integrates user authentication on Web Gateway and a directory on an LDAP server includes the following main steps.

- Web Gateway sends an initial bind request with administrator credentials to the LDAP server.
- If the request is successful, Web Gateway sends a query with the user name that the user submits.

  The purpose of this query is to find a distinguished name that the user name is mapped to in the directory on the LDAP server.
- If a distinguished name is found, the LDAP server sends it back.

  The distinguished name (DN) is a combination of information about a user, a user group, and a network domain provided in an LDAP-style syntax.

  For example, for the user name *jsmith*, the LDAP server sends back the distinguished name `cn=John Smith,cn=users,dc=ldap,dc=local`.
- Web Gateway sends a second bind request to the LDAP server with the purpose of authenticating the user.

  This request includes the distinguished name and the password that the user submitted.
- If the request is successful, the user is authenticated.

Note: You can record the steps of the authentication process in a tcpdump to review them.

Rule for authenticating a user under LDAP

To configure LDAP authentication on Web Gateway, you must implement a rule that authenticates a user in an integrated process with Web Gateway and a directory on an LDAP server.

The rule set library provides a rule set with a default rule that you can modify and use for this purpose. The modified rule looks as follows:

Name		
Authenticate with LDAP		
Criteria		Action
Authentication.Authenticate<LDAP> equals false	–>	Authenticate<Default>

The rule applies if a user has not yet been authenticated using the LDAP authentication method.

The settings of the *Authentication.Authenticate* property in this rule are configured to provide the information that is necessary to run the authentication process successfully, including the IP address of the LDAP server and the administrator credentials for Web Gateway.

Configure the LDAP method for authenticating a user

To configure the LDAP method for authenticating a user, you can adapt an already existing authentication rule. Modify the names and settings within this rule in a way that makes them suitable for LDAP authentication.

Task

1. Import the Explicit Proxy Authentication and Authorization rule set from the rule set library.

   Note: This rule set is for authentication in explicit proxy mode. For a transparent mode, import the Authentication Server rule set.

2. Adapt the authentication rule in the nested Authenticate with User Database rule set to make it suitable for LDAP authentication.

   Note: For a transparent mode, adapt the authentication rule in the nested Authentication Server rule set.

   a. Rename the current rule name to `Authenticate with LDAP`.

   b. Rename the settings of the Authentication.Authenticate property to a name that is appropriate for LDAP-related settings, for example, `LDAP`.

   c. Modify the settings to make them suitable for LDAP authentication.

3. Rename the nested rule set to `Authenticate with LDAP`.

   Note:

   Instead of adapting the nested library rule set, you can also disable or delete it and create a new nested rule set for LDAP authentication.

   The second nested rule set of the Explicit Proxy Authentication and Authorization library rule set, Authorize User Groups, is not needed for LDAP authentication.

   If you delete this nested rule set, you should rename the nesting rule set or have only one rule set named, for example, `Explicit Proxy Authentication with LDAP`.

4. Click Save Changes.

Configure the settings for the LDAP authentication method

Configure the settings for the LDAP authentication method by modifying the settings in the rule for authenticating a user that you have imported from the rule set library.

Task

1. In the imported rule, click the settings of the Authentication.Authenticate property that you have renamed to `LDAP` or a similar name.

   The Edit Settings window opens.

2. Under Authentication Method, select LDAP.

   The LDAP Specific Parameters section appears next to Common Authentication Parameters.

   Note:

   You can leave the common parameters as they are, as well as the LDAP-specific parameters that are not mentioned in the following.

3. In the LDAP server(s) to connect to list, add an entry for the LDAP server that the directory with the user information resides on.

   The syntax for an entry is as follows:

   {LDAP | LDAPS}://<IP address>[:<port number>]

   For example: `LDAP://10.205.67.8:389`

   Note:

   LDAP is an insecure protocol, as it transmits information in clear text. We recommend using LDAPS (secure LDAP) if possible.

   The default LDAP port is 389 while LDAPS uses 636.

4. Provide the administrator credentials that Web Gateway submits when trying to connect to the LDAP server.

   a. Under Credentials, type a common name and a domain controller name in LDAP style, for example:

   ```
   cn:administrator,cn:users,dc:ldap,dc:local
   ```

   b. Under Password, type an administrator password.

5. If the directory on the LDAP server is an Active Directory, deselect Allow LDAP directory to follow referrals.

6. Provide information for the query to find the distinguished name of the user who is to be authenticated.

   a. Under Base distinguished name to user objects, specify a starting point for the query.

   The starting point is specified in LDAP style, for example:

   ```
   cn:users,dc:ldap,dc:local
   ```

   b. Select Map user name to DN.

   Selecting this option lets the query search for a distinguished name that the submitted user name is mapped to in the directory.

c. Under Filter expression to locate a user object, specify a user attribute that allows the distinguished name to be found.

   Specifying this filter expression enables the search to find the entry for a user in the directory. The filter expression is the user name that the user submitted. The user name is stored in the directory as the value of an attribute that is part of the entry for a user.

   In an Active Directory, the name of the attribute that stores the user name is *sAMAccountName*. On Web Gateway, the user name is stored in a variable named *%u*.

   The filter expression must therefore be specified as follows if an Active Directory is used:

   ```
   samaccountname=%u
   ```

   Using this filter expression, the query will find the user entry and, consequently, try to map the user name to a distinguished name that might have been entered into the directory for a user with that user name.

7. Click OK to close the window.
8. Click Save Changes.

Results

These settings enable Web Gateway to authenticate a user under the LDAP authentication method. To retrieve information stored in other attributes within a directory, additional settings are required.

Configure queries for user and group attributes

Configure additional settings to perform queries that retrieve ("pull") more information about users and user groups from a directory on an LDAP server.

The settings for these queries are part of the settings that you configure for the Authentication module (engine) on Web Gateway to handle the integrated process for authenticating a user.

Task

1. Configure a query for user attributes.
   a. Select Get user attributes.

      Note: You need not configure any special values for the Base distinguished name to user objects option, as these values are the same as those that you already configured for the purpose of authenticating a user.
   b. In the User attributes to retrieve list, add the name of the attribute that the query should find a value for. You can also add multiple names here.

      For example, to retrieve information about the group or groups that a user belongs to, add `memberof`.
   c. Under Attributes concatenation string, type a character for separating multiple resulting values, for example, a comma.
2. Configure a query for group attributes.
   a. Select Get group attributes.
   b. Under Base distinguished name to group objects, provide a starting point for the query using LDAP syntax, for example, `ou=groups,dc=ldap,dc=local`.
   c. Under Filter expression to locate a group object, specify an attribute of a group that allows the group to be found.

      For example, specify `member=%u`, which has `member` as the attribute name and the `%u` variable that holds the user's user name on Web Gateway as the attribute value.
   d. In the Group attributes to retrieve list, add the name of the attribute that the query should find a value for. You can also add multiple names here

      For example, to find the so-called common name of a group, add `cn`.
   e. Under Attributes concatenation string, type a character for separating multiple resulting values, for example, a comma.

Storing an attribute in a separate property

You can store a user or group attribute in a separate User-Defined property for logging and other purposes.

When a query for an attribute of a user or user group is performed in a directory on an LDAP server, the resulting information is stored on Web Gateway as the value of the *Authentication.UserGroups* property.

If you are interested in a particular piece of information, for example, the email address of a user, you can also retrieve it separately and store it in a User-Defined property.

For this purpose, you must create an additional rule, as well as additional settings named, for example, `LDAP Email Lookup`, for the Authentication module (engine). In this rule, the Authentication module runs with the additional settings to retrieve the information that is stored within the entry for a user as the value of the email attribute.

Options must be especially configured in the additional settings as follows:

- Get user attributes must be enabled.
- The User attributes to retrieve list must contain a single entry for the email attribute. When an Active Directory is running on the LDAP server, the attribute name is `mail`.
- Map user to DN must be disabled.

  Not disabling the option produces an error, as the user name has already been mapped when the Authentication module was running with the `LDAP` settings to authenticate the user.

All other options can be configured in the same way as the settings within the rule that authenticates the user.

The complete rule should look as follows:

Name		
Get email information and store separately		
Criteria	Action	Event
Authentication.IsAuthenticated –> equals true AND AuthentIcation.GetUserGroups <LDAP_Email_:Lookup> does not contain "no-group"	Continue	Set User-Defined.Email= List.OfString.ToString (Authentication.UserGroups," ")

The rule must be added to the rule set for LDAP authentication and placed after the rule that authenticates the user.

Storing the original user name for logging

The original user name can be stored for logging purposes.

When a user has been authenticated using the LDAP method, the value of the *Authentication.Username* property is set to the user's distinguished name. If the property is used for creating a log entry, the part of the log entry that identifies the user will look, for example, as follows:

```
CN=John Smith,CN=Users,DC=LDAP,DC=local
```

To let the log entry show the original user name, which might be *jsmith*, rather than the distinguished name, you can modify the rule set for LDAP authentication in a suitable manner.

Instead of having only a rule that authenticates a user under LDAP, the rule set should contain the following:

- A rule that handles LDAP authentication for a user and stores the original user name in a User-Defined property
- One or more rules that perform other LDAP-related activities, for example, retrieving information about the group that a user belongs to
- A rule that restores the original user name as the value of the Authentication.Username property after all LDAP-related activities have been completed

Rule for authenticating a user and storing the user name

The following rule stores the original user name after authenticating the user. An event in this rule sets the value of a User-Defined property accordingly.

Name			
Authenticate user and store user name			
Criteria		Action	Event
Authentication.IsAuthenticated equals false AND Authentlcation.Authenticate<LDAP> equals true	–>	Continue	Set User-Defined.UserName= List.OfString.ToString (Authentication.UserGroups," ")

The user name is retrieved by querying the directory on the LDAP server for this name. The settings of the *Authentication.Authenticate* property, which is responsible for authenticating the user, are configured accordingly.

When the query has been performed, the user name is stored as the value of the *Authentication.Groups* property. It is converted into a string, using the *List.OfString.ToString* property.

Note: The original value of the converted property is a list of strings, as it might include not only the user name, but also other pieces of information, after all LDAP-related activities have been completed.

Rule for retrieving user group information

The following rule is an example for an additional LDAP-related activity. It retrieves information about the groups that a user belongs to.

Name		
Get user group information		
Criteria		Action
Authentication.IsAuthenticated equals true AND Authentlcation.GetUserGroups<LDAP_Group_:Lookup> does not contain "no-group"	–>	Continue

To identify the user, the rule still needs to know the user's distinguished name, so the original user name can not yet be restored as the value of the Authentication.Username property.

Note:

You must create different settings and configure them for the Authentication module (engine) to run and retrieve a value for the Authentication.GetUserGroups property.

The name of these settings might, for example, be `LDAP Group Lookup`, as in this sample rule.

Within these settings, the Map user to DN option must be disabled.

Rule for restoring the original user name

The following rule restores the original user name as the value of the *Authentication.UserName* property.

Name		
Restore user name		
Criteria	Action	Event

Authentlcation.Authenticate<LDAP>> equals false	Stop Rule Set	Set Authentication.UserName= User-Defined.Authentication.Username

An event in this rule sets this property to the value of the User-Defined property that you created to store the original user name in a preceding rule. The distinguished name that has temporarily been the value of this property is overwritten.

When the original user name has been restored, the property can be used for logging purposes.

Testing and troubleshooting LDAP authentication

Several activities can be completed for testing and troubleshooting the LDAP authentication process.

A tool for testing the configured authentication process with a given user name and password is available on the user interface of Web Gateway.

If running the tool shows that the process failed, carefully review what you have configured. If no errors can be found, you can create a debug log using another tool. If this does not explain the failure either, create a tcpdump using a third tool.

Test authentication for a given user name and password

The settings for the Authentication module include a section for testing purposes. You can enter a user name and password and let Web Gateway attempt to authenticate the user.

Task

1. Select Policy → Settings.
2. On the Engines branch of the settings tree, click the settings for the Authentication module (engine) that you have modified or newly created, for example, the `LDAP` settings.
3. Under Common Authentication Parameters, deselect Use authentication cache.

   Otherwise no changes in the directory on the LDAP server are detected until the cache expiration time has elapsed.
4. Expand Authentication Test and type a user name and password in the fields that are provided.
5. Click Authenticate User.

   The result of the authentication process is shown under Test result.

   - If the process is successfully performed, an OK message appears.

     The testing tool also displays any attribute values that you have configured queries for.
   - If the process fails, the following message appears: `Error: Authentication failed`.

Create a debug log file for troubleshooting authentication

You can create a debug log file to record the authentication process and review it for troubleshooting purposes.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance that you want to create a debug log file on, then click Troubleshooting.
3. In the Authentication Troubleshooting section, select Log authentication events.

Note: We recommend that you also select Restrict tracing to one IP and specify a client IP address to prevent the log file from becoming too large.

4. Reproduce the authentication process.

   A debug log file is created for the process.
5. Locate the debug log file.
   a. Select Troubleshooting
   b. On the troubleshooting tree, select the appliance that you created the debug log file on, then click Log files.
   c. Open the debug folder and look for the mwg-core.Auth.debug.log file with the appropriate time stamp.

Results

The log file contains log lines showing failure IDs for the authentication process. The meaning of these IDs is as follows:

- 0 – NoFailure: Authentication was successful
- 2 – UnknownUser: Cannot map user name to user DN
- 3 – WrongPassword: Bind with user password failed
- 4 – NoCredentials: Credentials are missing or have invalid format
- 5 – NoServerAvailable: Could not get a server connection
- 6 – ProxyTimeout: Request is being processed longer than the configured timeout
- 8 – CommunicationError: Communication with server failed, for example, due to a timeout

Create a tcpdump for troubleshooting authentication

If the reason for a failed authentication process cannot be found by reviewing a debug log file, create a tcpdump to retrieve more information.

Task

1. Select Troubleshooting.
2. On the troubleshooting tree, select the appliance that you want to create a tcpdump on, then click Packet tracing.
3. In the Command line parameters field, type the following:

   ```
   "-s 0 -i any port 389"
   ```

   Note: The port parameter lets Web Gateway connect to the LDAP server over an unencrypted port, which is required for troubleshooting purposes.
4. Click tcpdump start.
5. Reproduce the problem, then click tcpdump stop.
6. Open the trace using the *wireshark* tool. Then work with the *ldap.bindResponse* display filter to find a response from the LDAP server.

Results

The server response usually includes LDAP, Active Directory, and other error codes. For example, in the following line from a server response:

```
"invalidCredentials (80090308: LdapErr: DSID-0c09030f, comment: AcceptSecurityContext error, data 773, vece)"
```

the 773 error code is an Active Directory error code meaning that the user password must be changed.

Instant messaging authentication

Instant messaging authentication ensures that users of your network cannot access the web through an instant messaging service if they are not authenticated. The authentication process looks up user information and asks unauthenticated users to authenticate.

The following elements are involved in this process:

- Authentication rules that control the process
- The Authentication module, which retrieves information about users from different databases

An authentication rule can use an event to log information on the authentication of users who requested access to the web.

In this case, a logging module is also involved in the process.

Authentication rules

Instant messaging authentication is not implemented by default on the appliance, but you can import the *IM Authentication* rule set from the library.

This rule set contains a rule that looks up user information to see whether users who request web access are already authenticated. The method used for looking up the information is the User Database method.

Unauthenticated users that no information can be found for in the user database are asked to submit their credentials for authentication.

Another rule looks up information using the Authentication Server method to see whether users are authenticated and asks unauthenticated users for their credentials.

The Authentication module is called by these rules to retrieve the user information from the appropriate databases.

You can review the rules in the library rule set, modify or delete them, and also create your own rules.

Authentication module

The Authentication module (also known as *engine*) retrieves information that is needed to authenticate users from internal and external databases. The module is called by the authentication rules.

The different methods of retrieving user information are specified in the module settings. Accordingly, two different settings appear in the rules of the library rule set for instant messaging communication:

- User Database at IM Authentication Server
- Authentication Server IM

These settings are implemented with the rule set when it is imported from the library.

You can configure these settings, for example, to specify the server that user information is retrieved from under the Authentication Server method.

Logging module

The library rule set for instant messaging authentication includes a rule that logs authentication- related data, such as the user name of a user who requested web access, or the URL of the requested web object.

The logging is handled by the FileSystemLogging module, which you can also configure settings for.

Configure instant messaging authentication

You can implement instant messaging authentication and adapt it to the needs of your network.

Complete the following high-level steps.

Task

1. Import the IM Authentication rule set from the library.
2. Review the rules in the rule set and modify them as needed.

   You can, for example, do the following:

   - Modify the settings of the Authentication module for the User Database or the Authentication Server method.
   - Modify the settings of the logging module that handles the logging of information about instant messaging authentication.
3. Save your changes.

Configure the Authentication module for instant messaging authentication

You can configure the Authentication module to specify how it retrieves the information that is needed to authenticate users of an instant messaging service.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select the rule set for instant message authentication.
   If you have imported this rule set from the library, it is the *IM Authentication* rule set.
   The rules of the rule set appear on the settings pane.
3. Make sure Show details is selected.
4. Find the rules that call the Authentication module.
   In the library rule set, these are the rules *Authenticate Clients against the User Database* and *Redirect Not Authenticated Clients to the Authentication Server*.
5. In the rule criteria, click the settings name of the settings you want to configure.

   This name appears next to the *Authentication. Authenticate* property.

   In the library rule set, it is the *User Database at IM Authentication Server* or the *Authentication Server IM* settings.

   The Edit Settings window opens. It provides the settings for the Authentication module.
6. Configure these settings as needed.
7. Click OK to close the window.
8. Click Save Changes.

Configure the File System Logging module for instant messaging authentication

You can configure the File System Logging module to specify how it logs information that is related to instant messaging authentication.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select the rule set for instant message authentication.
   If you have imported this rule set from the library, it is the *IM Authentication* rule set.
   The rules of the rule set appear on the settings pane.
3. Make sure Show details is selected.
4. Find the rule that calls the File System Logging module.
   In the library rule set , this is the rule *Show Authenticated page* .
5. In the rule event, click the name of the settings for the module.

   In the library rule set, this name is *IM Logging*.

   The Edit Settings window opens. It provides the settings for the File System Logging module.
6. Configure these settings as needed.
7. Click OK to close the window.
8. Click Save Changes.

IM Authentication rule set

The IM Authentication rule set is a library rule set for instant messaging authentication.

Library rule set – IM Authentication
Criteria – *Always*
Cycles – Requests (and IM), responses, embedded objects

The following rule sets are nested in this rule set:

- IM Authentication Server
- IM Proxy

IM Authentication Server

This nested rule set handles authentication for instant messaging users. It applies the User Database method for retrieving user information.

Nested library rule set – IM Authentication Server
Criteria – *Authentication.IsServerRequest equals true*
Cycles – Requests (and IM), responses, embedded objects

The rule set criteria specifies that the rule set applies when authentication has been requested for a user of an instant messaging service.

The rule set contains the following rules.

Authenticate clients against user database
Authentication.Authenticate<User Database at IM Authentication server> equals false–> Authenticate<IM Authentication>
The rule uses the *Authentication.Authenticate* property to check whether a user who sends a chat message or file under an instant messaging protocol is authenticated. The settings that follow the property in the rule criteria specify the User Database method for this authentication.
If a user is not authenticated under this method, processing stops and a message is displayed asking the user to authenticate.
The action settings specify that the IM Authentication template is used for displaying the authentication message to the user.
Processing continues when the next user request is received.

Show Authenticated page
Always–> Redirect<Show IM Authenticated> —
Set User-Defined.logEntry =
“[”
+ DateTime.ToISOString
+ “]””
+ URL.GetParameter (“prot”)

+ ““auth””
+ Authentication.Username
+ ““ ””
+ URL.GetParameter (“scrn”)
+ “””
FileSystemLogging.WriteLogEntry (User-Defined.logEntry)<IM Logging>
The rule redirects a request sent from a client by an instant messaging user to an authentication server and displays a message to inform the user about the redirect.
The action settings specify that the Show IM Authenticated template is used for the message.
The rule also uses an event to set values for a log entry on the authentication request. It uses a second event to write this entry into a log file. A parameter of this event specifies the log entry.
The event settings specify the log file and the way it is maintained.

IM Proxy

This nested rule set handles authentication of instant messaging users. It applies the Authentication Server method to retrieve user information.

Nested library rule set – IM Proxy
Criteria – *Connection.Protocol.IsIM equals true AND IM.MessageCanSendBack is true*
Cycles – Requests (and IM), responses, embedded objects

The rule set criteria specifies that the rule set applies when a user sends a chat message or a file on a connection under an instant messaging protocol and a message can already be sent back from the appliance to the user.

The rule set contains the following rule.

Redirect not authenticated users to the authentication server
Authentication.Authenticate<Authentication Server IM> equals false–> Authenticate<IM Authentication>
The rule uses the *Authentication.Authenticate* property to check whether a user who sends a chat message or file under an instant messaging protocol is authenticated. The settings that follow the property in the rule criteria specify the Authentication Server method for this authentication.
If a user is not authenticated under this method, processing stops and a message is displayed, asking the user to authenticate.
The action settings specify that the IM Authentication template is used for displaying the authentication message to the user.
Processing continues when the next user request is received.

One-time passwords

One-time passwords (OTPs) can be processed on Web Gateway to authenticate users. This includes the use of passwords for authorized overriding when a web session has terminated due to quota expiration.

When a user sends a request for web access, authentication is first performed using one of the other authentication methods that are available on Web Gateway, for example, authentication based on information stored in the internal user database.

If the use of one-time passwords is configured, this authentication method is performed as a second step. Web Gateway informs the user that a one-time password is also needed for web access and upon the user's request for such a password, it forwards the user name to a McAfee® One Time Password (McAfee OTP) server and asks the server to provide a password.

If the request is granted, the McAfee OTP server returns a one-time password, which is, however, not exposed to Web Gateway. In its response, the McAfee OTP server also includes what is called "context" information in a header field.

The context information lets the password field and submit button in the page that was presented to the user be activated, so the user can click the button, which submits the one-time password and lets the user access the requested web object.

To implement the use of one-time passwords on Web Gateway, you can import a rule set from the rule set library. After importing the rule set, default settings are provided, which you can configure to adapt them to the needs of your network.

The settings that need to be configured include the IP address or host name of the McAfee OTP server and the port on this server that listens to requests from Web Gateway.

A user name and password for Web Gateway to authenticate to the McAfee OTP server are also required.

If the communication between Web Gateway and the McAfee OTP server should be SSL-secured, you need to import a certificate for use in this communication.

The McAfee OTP server must be configured for working with Web Gateway to handle the authentication process.

One-time passwords for authorized overriding

When quota restrictions are imposed on web usage from within your network, a one-time password can be used as the password that is required to override the termination of a web session due to quota expiration.

To implement the use of one-time passwords for authorized overriding, you can import a different rule set from the library, which also allows you to configure the settings for the authentication process.

Using one-time passwords from a McAfee Pledge device

One-time passwords for authenticating users or performing an authorized override can be provided by a McAfee® Pledge device.

To enable this method of using one-time passwords for the authentication process, you need to implement suitable rule sets, which you can import from the rule set library. Settings for the authentication process are implemented with the import.

For more information on working with a McAfee Pledge device, refer to the documentation for this product.

Configure one-time passwords for authenticating users

To configure the use of one-time passwords for authenticating users, complete the following high-level steps.

Task

1. Import the *Authentication Server (Time/IP Based Session with OTP)* rule set from the rule set library.

   When using one-time passwords from a McAfee Pledge device, import the *Authentication Server (Time/IP Based Session with OTP and Pledge)*

   The rule sets are located in the *Authentication* rule set group.
2. Configure the settings for one-time passwords.
3. Save your changes.

Results

For information on how to configure the McAfee OTP server for working with Web Gateway, refer to the McAfee OTP server documentation.

Configure one-time passwords for authorized overriding

To configure the use of one-time passwords for authorized overriding. complete the following high-level steps.

Task

1. Import the *Authorized Override with OTP* rule set from the rule set library.

   When using one-time passwords from a McAfee Pledge device, import *Authorized Override with OTP and Pledge*.

   The rules sets are located in the *Coaching/Quota* rule set group.
2. Configure the settings for one-time passwords.
3. Save your changes.

What to do next

For information on how to configure the McAfee OTP server for working with Web Gateway, refer to the McAfee OTP server documentation.

Configure the settings for one-time passwords

Settings for one-time passwords are implemented with default values after importing the rule sets that handle the use these passwords. Configure these settings to adapt them to the requirements of your network.

You need to configure different settings for authentication and authorized overriding with one-time passwords.

Task

1. Select Policy → Settings.
2. On the Engines branch of the settings tree, expand Authentication.
3. To configure settings for using one-time passwords to authenticate users, complete the following substeps. Otherwise continue with step 4.
   a. Click OTP.
      The OTP settings appear on the settings pane.
   b. Configure the settings in the One-Time Password Specific Parameters section and the settings in the other sections, which are common authentication settings, as needed.
   c. Click IP Authentication Server.
      The IP Authentication Server settings appear on the settings pane.
   d. Configure the settings in the IP Authentication Server Specific Parameters section and the settings in the other sections, which are common authentication settings, as needed.
   e. Click User Database at Authentication Server.
      The User Database at Authentication Server settings appear on the settings pane.
   f. Configure the settings in the User Database Specific Parameters section and the settings in the other sections, which are common authentication settings, as needed.
      Then continue with step 5.
4. To configure settings for using one-time passwords in authorized overriding, complete the following substeps.
   a. Click OTP.
      The OTP settings appear on the settings pane.
   b. Configure the settings in the One-Time Password Specific Parameters section and the settings in the other sections, which are common authentication settings, as needed.
5. Click Save Changes.

Authentication Server (Time/IP Based Session with OTP) rule set

The Authentication Server (Time/IP Based Session with OTP) rule set is a library rule set that enables the use of one-time passwords for authenticating users.

Library rule set – Authentication Server (Time/IP Based Session with OTP)
Criteria – *Always*
Cycles – Requests (and IM)

The following rule sets are nested in this rule set:

- Check for Valid Authentication Session
- Authentication Server

Check for Valid Authentication Session

This nested rule redirects a user's request sent from a client to the authentication server if the user has not yet been successfully authenticated on that server.

Nested library rule set – Check for Valid Authentication Session
Criteria – *Authentication.IsServerRequest equals false AND* *(Connection.Protocol equals "HTTP" OR* *Connection.Protocol equals "SSL" OR* *Connection.Protocol equals "HTTPS" OR* *Connection.Protocol equals "IFP")*
Cycles – Requests (and IM)

The rule set criteria specifies that the rule set applies if the request that is currently processed is not requesting a connection to the authentication server and the protocol used in this communication is one of the four that are specified..

The rule set contains the following rules:

Fix hostname
Command.Name equals "CERTVERIFY" AND SSL.Server.Certificate.CN.HasWildcards equals false –> Continue – Set URL.Host = SSL.Server.Certificate.CN
The rule uses an event to set the host name that is submitted with the URL of a request to a particular value, which is required when communication is going on under the SSL protocol. This value is the common name of the certificate that is provided in this communication.
The rule applies if the request that is processed contains the CERTVERIFY command and no wildcards are allowed for the common name.

Redirect clients that do not have a valid session to the authentication server
Authentication.Authenticate<IP Authentication Server> equals false AND Command.Name does not equal "CONNECT" –> Authenticate<Default>

The rule uses the *Authentication.Authenticate* property to check whether the user who sends a request is successfully authenticated at the user database of the authentication server. For this purpose, the IP address of the client that the request was sent from is evaluated.
The *Command.Name* property is used to check whether the request is a connection request in SSL-secured communication.
If neither is the case, the user is asked to submit credentials for authentication. This action is executed with the specified settings.

Revalidate session under ideal conditions
Authentication.CacheRemainingTime less than 400 AND *Connection.Protocol equals "HTTP" AND* *Command.Name equals "GET"* –> Authenticate<Default>
Under particular conditions (which could be termed "ideal"), a user is asked to authenticate again after sending a request to ensure the current web session is prolonged before the time quota has elapsed completely.
This is done if communication is going on under the HTTP protocol and the request contains the GET command.
The rule is not enabled by default.

Authentication Server

This nested rule set forwards a request for web access when a user submitted a valid one-time password. A user who could not submit a valid one-time password is asked to authenticate.

Authentication is first performed using information from the user database on an authentication server. A successfully authenticated user is then informed that web access also requires a one-time password, which is sent by Web Gateway upon the user's request.

Nested library rule set – Authentication Server
Criteria – *Authentication.IsServerRequest equals true*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies when a user who sent a request must be authenticated using information from an authentication server.

The rule set contains the following rules:

Redirect if we have a valid OTP
Authentication.Authenticate<OTP> equals true –> Redirect <Redirect Back from Authentication Server>
The rule uses the *Authentication.Authenticate* property to check whether a user who submitted a one-time password with a request for web access could be successfully authenticated.
If this is the case, web access is allowed and the user is redirected from the authentication server to the requested web object.

Stop after providing an invalid OTP

Authentication.Failed equals true –> Block<Authorized Only>
The rule uses the *Authentication.Failed* property to check whether a user who submitted a one-time password with a request for web access could not be successfully authenticated.
If this is the case, the request is blocked and a message informs the user about the blocking and the block reason.

Authenticate user against user database
Authentication.Authenticate<User Database at Authentication Server> equals false –> Authenticate<Default>
The rule uses the *Authentication.Authenticate* property to check whether a user who sent a request and submitted an invalid one-time password could be successfully authenticated at the user database on the authentication server.
If this is not the case, the user is asked to authenticate.

Send OTP if requested
Header.Exists(Request.OTP) equals true –> Continue – Authentication.SendOTP<OTP>
If none of the preceding rules in this rule set has applied, it means no valid one-time password was submitted by a user who sent a request for web access, but authentication at the user database on the authentication server was successful.
Then this rule is processed, which uses the *Header.Exists* property to check whether the request has a header providing the information that sending a one-time password is requested.
If this is the case, the rule uses an event to send a one-time password to the user.

Return authentication data to client
Header.Exists("Request.OTP") equals true –> Block<Authentication Server OTP> – Header.Block.Add("OTP Context", Authentication.OTP.Context<OTP>)
The rule uses the *Header.Exists* property to check whether there is a header in a request with information that sending a one-time password is requested.
If this is the case, the request is blocked and a message sent to inform the user who sent the request that a one time password is required for access.
An event is also triggered that adds a header with context information about the one-time password authentication process to the block message.
The first of the two event parameters specifies the header information that is added. The second parameter is a property that has information about the one-time password authentication process as its value, which is the source of the added information.

Block request and offer sending OTP
Always –> Block<Authentication Server OTP>
If none of the preceding rules in this rule set have applied, the Block action of this rule is always executed.

The action stops rule processing and the request is not forwarded.
The action settings specify that a message is sent to inform the user that a one-time password is required for web access, which can be obtained from Web Gateway.

Authorized Override with OTP rule set

The Authorized Override with OTP rule set is a library rule set for enabling the use of one-time passwords in authorized overriding.

Library rule set – Authorized Override with OTP and Pledge
Criteria – *SSL.ClientContext.IsApplied equals true OR Command.Name does not equal "CONNECT"*
Cycles – Requests (and IM)

The rule criteria specified that the rule set applies when SSL-secured communication is configured or the request that is currently processed is not a CONNECT request, which is usually sent at the beginning of this communication.

The following rule sets are nested in this rule set:

- Verify OTP
- OTP Needed?

Verify OTP

This nested rule checks whether a user who sends a one-time password with a request for authorized overriding is successfully authenticated and performs a redirect to the requested web object if this is true.

Nested library rule set – Verify OTP
Criteria – *Quota.AuthorizedOverride.IsActivationRequest.Strict<Default> equals true*
Cycles – Requests (and IM)

The rule set criteria specifies that the rule set applies when a user sends a request to override the termination of a web session due to quota expiration and to continue with the session.

The rule set contains the following rules:

Verify OTP
Authentication.Authenticate<OTP> equals false –> Block<Authorized Only>
The rule uses the *Authentication.Authenticated* property to check whether the user who submitted a one-time password when sending an authorized overriding request has been successfully authenticated.
If this is not the case, the request is blocked and the user is informed about the blocking and the reason for it.
The Block action is executed with the specified settings.

The session is validated. Redirect to the original page
Always –> Redirect<Default>

If authentication of a user who submitted a one-time password with a request for authorized overriding did not fail, the preceding rule in this rule set does not apply and processing continues with this rule.
The rule always allows the user to continue with the current session and performs a redirect to the requested web object.
The Redirect action is executed with the specified settings.

OTP Needed?

This nested rule set provides a one-time password for a user who sends a request for authorized overriding if the requested web object is located on a host within the corporate domain of McAfee.

Nested library rule set – OTP Needed?
Criteria – *URL.Host matches *mcafee.com**
Cycles – Requests (and IM)

The rule set criteria specifies that the rule set applies when the host of the URL sent in a request is located within the corporate domain of McAfee.

The rule set contains the following rules:

Send OTP if requested
Header.Exists(Request.OTP) equals true –> Continue – Authentication.SendOTP<OTP>
If none of the proceeding rules in this rule set have applied when processing a request, it means no valid one-time password was submitted by the user who sent the request, but authentication at the user database of the authentication server was successful.
Then this rule is processed, it uses the *Header.Exists* property to check whether the request has a header provides the information that sending a one-time password is requested.
if this is the case, an event is triggered that send a one-time password to the user.

Return authentication data to client
Header.Exists(Request.OTP) equals true –> Block<Authentication Server OTP> – Header.Block.Add("OTP Context", Authentication.OTP.Context<OTP>)
The uses the *Header.Exists* property to check whether the request has a header providing the information that sending a one-time password is requested.
If none of the proceeding rules in this rule set have applied when processing a request, it means no valid one-time password was submitted by the user who sent the request, but authentication at the user database of the authentication server was successful.
If this is the case, the request is not forwarded and an event is triggered that sets a particular property to a value that provides information about the authentication of the user.
The Block action is executed with the specified settings, which require that a message is sent to inform the user about the reason of the blocking.
The information that the event provides is specified by the *OTP.Context* event parameter. The property that has its value set to this information is specified in a second parameter.

Block request and offer sending OTP
Always –> Block<Authentication Server OTP>
If none of the preceding rules in this rule set have applied when processing a request, the action of this rule is always executed.
It stops rule processing and the request is not forwarded. The action settings specify that a message is sent to inform the user that a one-time password can be obtained from Web Gateway.

Authentication Server (Time/IP Based Session with OTP and Pledge) rule set

The Authentication Server (Time/IP Based Session with OTP and Pledge) rule set is a library rule set for authenticating users through one-time passwords that are provided by a McAfee Pledge device.

Library rule set – Authentication Server (Time/IP Based Session with OTP and Pledge)
Criteria – *Always*
Cycle – Requests (and IM)

The following rule sets are nested in this rule set:

- Check for Valid Authentication Session
- Authentication Server

Check for Valid Authentication Session

This nested rule redirects a user's request sent from a client to the authentication server if the user has not yet been successfully authenticated on that server.

Nested library rule set – Check for Valid Authentication Session
Criteria – *Authentication.IsServerRequest equals false AND* *(Connection.Protocol equals "HTTP" OR* *Connection.Protocol equals "SSL" OR* *Connection.Protocol equals "HTTPS" OR* *Connection.Protocol equals "IFP")*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies if the request that is currently processed is not requesting a connection to the authentication server and the protocol used in this communication is one of the four that are specified.

The rule set contains the following rules:

Fix hostname
Command.Name equals "CERTVERIFY" AND SSL.Server.Certificate.CN.HasWildcards equals false –> Continue – Set URL.Host = SSL.Server.Certificate.CN

The rule uses an event to set the host name that is submitted with the URL of a request to a particular value, which is required when communication is going on under the SSL protocol. This value is the common name of the certificate that is provided in this communication.
The rule applies if the request that is processed contains the CERTVERIFY command and no wildcards are allowed for the common name.

Redirect clients that do not have a valid session to the authentication server
Authentication.Authenticate<IP Authentication Server> equals false AND Command.Name does not equal "CONNECT" –> Authenticate<Default>
The rule uses the *Authentication.Authenticate* property to check whether the user who sends a request is successfully authenticated at the user database of the authentication server. For this purpose, the IP address of the client that the request was sent from is evaluated.
The *Command.Name* property is used to check whether the request is a connection request in SSL-secured communication.
If neither is the case, the user is asked to submit credentials for authentication. This action is executed with the specified settings.

Revalidate session under ideal conditions
Authentication.CacheRemainingTime less than 400 AND *Connection.Protocol equals "HTTP" AND* *Command.Name equals "GET"* –> Authenticate<Default>
Under particular conditions (which could be termed "ideal"), a user is asked to authenticate again after sending a request to ensure the current web session is prolonged before the time quota has elapsed completely.
This is done if communication is going on under the HTTP protocol and the request contains the GET command.
The rule is not enabled by default.

Authentication Server

This nested rule set forwards a request for web access by a user who submitted a valid one-time password that was retrieved from a McAfee Pledge device.

A user who did not submit a valid one-time password is asked to authenticate. Authentication is first performed using information from the user database of the authentication server.

A successfully authenticated user is then informed that web access also requires a one-time password from a McAfee Pledge device.

Nested library rule set – Authentication Server
Criteria – *Authentication.IsServerRequest equals true*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies when a user who sent a request must be authenticated using information from an authentication server.

The rule set contains the following rules:

Authenticate user against user database
Authentication.Authenticate<User Database at Authentication Server> equals false –> Authenticate<Default>
The rule uses the *Authentication.Authenticate* property to check whether a user who sent a request and submitted an invalid one-time password could be successfully authenticated at the user database on the authentication server.
If this is not the case, the user is asked to authenticate.

Show block template
URL.GetParameter(pledgeOTP) equals " " –> Block<Authentication.Server OTP with PledgeOTP>
The rule uses the *URL.GetParameter* property to check whether a one-time password from a McAfee Pledge device was sent as a parameter of the URL in a request.
If the parameter is empty, the request is blocked and the user is informed that authentication using a one-time password from a McAfee Pledge device is also required for web access.

Retrieve OTP context
Always –> Continue – Authentication.SendOTP<OTP>
The rule uses an event to send context information on the one-time password authentication process to an authenticated user.
This way the information is retrieved that is required to validate a one-time password on a McAfee OTP server.

Redirect back if we have a valid OTP
Authentication.Authenticate<OTP> equals true –> Redirect<Redirect Back from Authentication Server>
The rule uses the *Authentication.Authenticate* property to check whether a user who submitted a one-time password with a request for web access could be successfully authenticated.
If this is the case, web access is allowed and the user is redirected from the authentication server to the requested web object.

Stop after providing an invalid OTP
Authentication.Failed equals true –> Block<Authorized Only>
The rule uses the *Authentication.Failed* property to check whether a user who submitted a one-time password with a request for web access could not be successfully authenticated.
If this is the case, the request is blocked and a message informs the user about the blocking and the block reason.

Authorized Override with OTP and Pledge rule set

The Authorized Override with OTP and Pledge rule set is a library rule set for authorized overriding using one-time passwords that are provided by a McAfee Pledge device.

Library rule set – Authorized Override with OTP and Pledge
Criteria – *SSL.ClientContext.IsApplied equals true OR Command.Name does not equal "CONNECT"*
Cycles – Requests (and IM)

The rule criteria specified that the rule set applies when SSL-secured communication is configured or the request that is currently processed is not a CONNECT request, which is usually sent at the beginning of this communication.

The following rule sets are nested in this rule set:

- Verify OTP
- OTP Needed?

Verify OTP

This nested rule checks whether a user who sends a one-time password with a request for authorized overriding is successfully authenticated and performs a redirect to the requested web object if this is true.

Nested library rule set – Verify OTP
Criteria – *Quota.AuthorizedOverride.IsActivationRequest.Strict<Default> equals true*
Cycles – Requests (and IM)

The rule set criteria specifies that the rule set applies when a user sends a request to override the termination of a web session due to quota expiration and to continue with the session.

The rule set contains the following rules:

Verify OTP
Authentication.Authenticate<OTP> equals false –> Block<Authorized Only>
The rule uses the *Authentication.Authenticated* property to check whether the user who submitted a one-time password when sending an authorized overriding request has been successfully authenticated.
If this is not the case, the request is blocked and the user is informed about the blocking and the reason for it.
The Block action is executed with the specified settings.

The session is validated. Redirect to the original page
Always –> Redirect<Default>
If authentication of a user who submitted a one-time password with a request for authorized overriding did not fail, the preceding rule in this rule set does not apply and processing continues with this rule.
The rule always allows the user to continue with the current session and performs a redirect to the requested web object.
The Redirect action is executed with the specified settings.

OTP Needed?

This nested rule set provides a one-time password for a user who sends a request for authorized overriding if the requested web object is located on a host within the corporate domain of McAfee.

Nested library rule set – OTP Needed?
Criteria – *URL.Host matches *mcafee.com* AND Quota.AuthorizedOverride.SessionExceeded<Default> equals true*
Cycles – Requests (and IM)

The rule set criteria specifies that the rule set applies when the host of the URL sent in a request is located within the corporate domain of McAfee and the time quota for a session that can be continued after an authorized override has been exceeded.

The rule set contains the following rules:

Retrieve OTP context
Always –> Continue – Authentication.SendOTP<OTP>
The rule uses an event to send a one-time password to an authenticated user.
This way the context information is obtained that is required for authenticating a user through a one-time password that is validated on a McAfee OTP server.

Block request and offer sending OTP
Always –> Block<OTP Required with Pledge>
The rule blocks a request for web access.
The action settings specify that a message is sent to inform the user web access can be allowed after submitting a one-time password that an be obtained from a McAfee Pledge device.

Client Certificate authentication

Submitting a client certificate can be configured as a method of accessing the user interface of the appliance. This method is known as *Client Certificate authentication* or *X.509 authentication*.

Client Certificate authentication is one of the methods you can choose for the authentication procedure when configuring the proxy functions of the appliance.

The following applies to the method when using it in proxy configuration.

- No user name and password is required to authenticate a user who sends a request, as is the case with other methods such as NTLM or LDAP.
- The method can be implemented for requests that are sent in SSL-secured communication from a web browser on a client to an appliance that is configured in explicit proxy mode.
- The protocol used for this communication is HTTPS.

A client certificate is submitted when the SSL handshake is performed as one of the initial steps in the communication between the appliance and a client. The request is then redirected to an authentication server to validate the certificate.

If it is valid, authentication is successfully completed for the client and the request is eventually forwarded to the appropriate web server.

When running multiple appliances as nodes in a configuration, it is important that the authentication server resides on the node that a request was originally directed to.

Also forwarding to the web after successful authentication must be done from the same node.

Use of an authentication server for Client Certificate authentication is controlled by rules. You can import an authentication server rule set and modify the rules in its nested rule sets to enable the use of appropriate certificates.

You must also implement a way to let Client Certificate authentication be applied. A recommended way of doing this is using cookie authentication.

If this method is implemented, authentication is required for a client that a request was sent from, but a cookie is set for this client after a certificate has been submitted and recognized as valid once. Submitting a certificate is then not required for subsequent requests from that client.

You can import and modify a rule set for having Client Certificate authentication handled in this way.

Use of certificates for Client Certificate authentication

Different types of certificates are required for performing authentication under the Client Certificate authentication method, which can be implemented for SSL-secured communication.

Client certificate

A client certificate is needed to certify the identity of a client that sends a request to the appliance.

Only if the client is trusted will a request that it sends be accepted. A client is trusted if the certificate that is submitted with the request has been signed by a Root CA (certificate authority) that is trusted.

Under the Client Certificate authentication method, the client certificate is also used for authentication. Authentication is successfully completed if the client certificate that is submitted with a request has been signed by a trusted certificate authority.

Server certificate

A server certificate is needed to certify the identity of a server that is involved in SSL-secured communication.

A server is trusted by a client if the certificate that it sends during the initial steps of the communication has been signed by a Root CA (certificate authority) that is also trusted by the client.

Under the Client Certificate authentication method, a server certificate is needed for the authentication server.

Root CA

A Root CA (certificate authority) is an instance that signs other certificates.

In SSL-secured communication, a Root CA appears itself as a certificate that can be viewed in the communication process.

If a Root CA is trusted by a client or server, certificates that have been signed by it are trusted as well, which means that if a client or server submits such a signed certificate, it is trusted.

Rule sets for Client Certificate authentication

Rule sets for implementing the Client Certificate authentication method are available in the rule set library.

Authentication Server (for X509 Authentication) rule set

The Authentication Server (for X509 Authentication) rule set uses several nested rule sets to handle use of the authentication server under the Client Certificate authentication method.

- **SSL Endpoint Termination** — Prepares the handling of requests in SSL-secured communication
    - **Accept Incoming HTTPS Connections** — Provides the certificates that can be submitted for the authentication server
    - **Content Inspection** — Enables inspection of the content that is transmitted with a request
- **Authentication Server Requests** — Redirects requests back to the proxy on the appliance for further processing after authentication on the authentication server was completed successfully

  Requests are also redirected if a cookie has been set for a client that a request was sent from.

If authentication could not be completed successfully on the authentication server, the user is asked to submit credentials for authentication on the user database.

- **Block All Others** — Blocks requests for which authentication was not completed successfully

Cookie Authentication (for X509 Authentication) rule set

The Cookie Authentication (for X509 Authentication) rule set uses several nested rule sets to initiate use of the Client Certificate authentication method and handle the setting of cookies.

- **Cookie Authentication at HTTP(S) Proxy** — Contains nested rule sets that handle Client Certificate authentication with cookies
  - **Set Cookie for Authenticated Clients** — Sets a cookie after authentication has been successfully completed once for a client and redirects the request that the client sent back to the proxy on the appliance for further processing
  - **Authenticate Clients with Authentication Server** — Redirects requests sent from clients for which no cookie has been set to the authentication server

Redirecting requests to an authentication server

Under the Client Certificate authentication method, a request is redirected to an authentication server for validating the client certificate that was submitted with it. The redirecting can be done using a special listener port on the appliance or a unique host name.

Using a special listener port

Requests can be redirected to an authentication server using a special listener port, for example, port 444. Suppose the IP address of an appliance is 192.168.122.119, then a request will be redirected to the authentication server by:

```
https://192.168.122.119:444/
```

However, it is important to consider whether exceptions from using a proxy have been configured for the web browser on a client that sends the request.

- **No proxy exceptions configured** — If no proxy exceptions have been configured, all requests are sent to the proxy port that is listening for them on the appliance, which is port 9090 by default.

  Even a request to `https://192.168.122.119:444/` will arrive on port 9090 if this is the configured proxy port.

  If a firewall is part of your network configuration, no exceptions from the firewall rules are needed because there is no connection from the client to port 444.

  To ensure requests are redirected to the authentication server, 444, or another value that you want to use for this purpose, must be configured for the *URL.Port* property in the criteria of the Authentication Server (for X509 Authentication) rule set.

  The value of the URL.Port property is the port contained in the URL that is specified by a request. It can be, for example, 444, even if the request actually arrives at port 9090.
- **Proxy exceptions configured** — Proxy exceptions can be configured for various reasons. For example, a web browser could be configured not to use proxies for accessing local hosts.

  A request to `https://192.168.122.119:444/` will then not arrive at port 9090.

  Because the browser is configured to access its destination directly, it will try to connect to the appliance on port 444. This means that you need to set up a listener port with port number 444.

  If firewall rules are in place, an exception is also needed to allow requests to arrive at port 444.

  To ensure requests are processed by the appropriate rules, 444, or another value that you want to use for this purpose, must be configured for the *Proxy.Port* property in the criteria of the Authentication Server (for X509 Authentication) rule set.

  The value of the Proxy.Port property is the port that a request actually arrives at. It is, for example, 444 if you have set up a port with this number for receiving requests that are to be redirected to an authentication server.

Using a unique host name

Requests can be redirected to an authentication server using a unique host name, for example, authserver.local.mcafee. Using this name, requests are redirected to the authentication server by:

```
https://authserver.mcafee.local
```

The client that the request was sent from must not try to look up the host name using DNS, as the URL will most likely not resolve and the client will be unable to connect.

To ensure that requests are processed by the appropriate rules, this host name must be configured as the value for the *URL.Host* property in the criteria of the Authentication Server (for X509 Authentication) rule set.

Implement Client Certificate authentication

The Client Certificate authentication method uses client certificates that are sent with requests for authentication. To implement this method on the appliance, complete the following high-level steps.

Task

1. Import the Authentication Server (for X509 Authentication) rule set.
2. Modify the nested rule sets to configure the use of appropriate certificates.
3. Configure a listener port for requests sent by web browsers that are not using the proxy port on the appliance.
4. Configure a way to let Client Certificate authentication be applied.
   You can import and modify the Cookie Authentication (for X509 Authentication) rule set to use a cookie for authentication after Client Certificate authentication has been applied once and successfully been completed.
5. Make sure a suitable client certificate is available on a web browser that is used for sending requests to the appliance.

Import the Authentication Server (for X509 Authentication) rule set

To implement the Client Certificate authentication method on the appliance, there must be a rule set that handles authentication in this way. You can import the Authentication Server (for X509 Authentication) rule set for this purpose.

We recommend that you insert the rule set at the top of the rule sets tree.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, navigate to the position where you want to insert the rule set and click Add.
3. Click Top Level Rule Set, then click Import Rule Set from Library.
   The Add from Rule Set Library window opens.
4. Select the Authentication Server (for X509 Authentication) rule set and click OK.
   If conflicts arise from the import, they are displayed next to the list of rule sets. Follow one of the suggested procedures for solving them before clicking OK.
   The rule set is inserted with its nested rule sets in the rule sets tree.
5. Review the rule set criteria and modify them if necessary.

   After the import, the criteria is:

   URL.Port equals 444 or Proxy.Port equals 444.

   This ensures that the rule set is applied to all requests coming in on that port. If you want to use a different port, specify its port number here.

Modify a rule set to configure the use of server certificates

The Authentication Server (for X509 Authentication) rule set needs to be modified to ensure appropriate server certificates are submitted for the authentication server. The modification is done in a nested rule set.

Because it is possible to reach the authentication server under different host names and IP addresses, you can let the appliance submit a different server certificate each time, so that the host name or IP address is matched by the common name in the certificate.

To achieve this, you need to import a server certificate for each host name or IP address and add it to the list of server certificates.

Task

1. Select Policy → Rule Sets and expand the Authentication Server (for X509 Authentication) rule set.
2. Expand the nested SSL Endpoint Termination rule set and, within this rule set, select the nested Accept Incoming HTTPS Connections rule set.
3. In the Set client context rule, click the Proxy Certificate event settings.
   The Edit Settings window opens.
4. In the Define SSL Context section, review the list of server certificates.
5. To add a server certificate to the list:
   a. Click the Add icon above the list.
      The Add Host to Certificate Mapping window opens.
   b. In the Host field, enter the host name or IP address that the certificate should be submitted for.
   c. Click Import.
      The Import Server Certificate window opens.
   d. Click Browse and browse to the certificate you want to import.
   e. Repeat this activity to import a key and certificate chain with the certificate.
   f. Click OK.
      The window closes and the import is performed. The certificate information appears in the Add Host to Certificate Mapping window.
6. [Optional] In the Comment field, type a plain-text comment on the server certificate.
7. Click OK.
   The window closes and the server certificate appears in the list.
8. Make sure the SSL-Scanner functionality applies only to client connection checkbox is selected.
   This lets the appliance accept requests from its clients without contacting other servers of the network, which is not required in this communication.
9. Click OK to close the Edit Settings window.
10. Click Save Changes.

Modify a rule set to configure the use of certificate authorities

The Authentication Server (for X509 Authentication) rule set needs to be modified to ensure appropriate Root CAs (certificate authorities) are configured. The modification is done in a nested rule set.

A client certificate is trusted if signed by a certificate authority from the list that is maintained on the appliance. You need to import all certificate authorities into the list that you want to be signing instances for trusted client certificates.

Task

1. Select Policy → Rule Sets and expand the Authentication Server (for X509 Authentication) rule set.
2. Expand the nested SSL Authentication Server Request rule set.
3. In the Ask user for client certificate rule, click the X509 Auth module settings.
   The Edit Settings window opens.
4. In the Client Certificate Specific Parameters section, review the list of certificate authorities.
5. To add a certificate authority to the list:
   a. Click the Add icon above the list.
      The Add Certificate Authority window opens.
   b. In the Host field, enter the host name or IP address that the certificate should be submitted for.
   c. Click Import.

    A window providing access to your local file system opens.

    d. Browse to the certificate authority file you want to import.
    e. Click OK.

       The window closes and the import is performed. The certificate appears in the Add Certificate Authority window.

6. Make sure the Trusted checkbox is selected.
7. [Optional] In the Comment field, type a plain-text comment on the certificate authority.
8. Click OK.

   The window closes and the certificate authority appears in the list.
9. Click OK to close the Edit Settings window.
10. Click Save Changes.

Configure a listener port for incoming requests on the appliance

Requests that are sent to the appliance can be received on the proxy port or a special listener port. The proxy port is port 9090 by default.

You need to configure a listener port if proxy exceptions have been created that prevent requests from arriving at the proxy port.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure a listener port on and click Proxies (HTTP(S), FTP, ICAP, and IM).

   The proxy settings appear on the settings pane.
3. Scroll down to the HTTP Proxy section.
4. Make sure Enable HTTP proxy is selected.
5. On the toolbar of the HTTP port definition list, click the Add icon.

   The Add HTTP Proxy Port window opens.
6. Configure a listener port as follows:
    a. In the Listener address field, type `0.0.0.0:444`.

       If you want to use a different port for listening to incoming requests, type it here.
    b. In the Ports treated as SSL field, type `*`.
    c. Make sure all other checkboxes are selected.
7. Click OK to close the Edit Settings window.
8. Click Save Changes.
9. Restart the appliance to make the configuration of the listener port effective.

Import the Cookie Authentication (for X509 Authentication) rule set

When the Client Certificate authentication method is used on the appliance, use of this method can be initiated by the Cookie Authentication (for X509 Authentication) rule set.

We recommend that you insert this rule set after the rules sets for functions that do not require authentication, but before the rule sets that handle the filtering functions.

This ensures the filtering functions are not executed when a request is blocked because authentication failed, which saves resources and improves performance.

If your rule set system is similar to the default system, you can insert the rule set after the SSL Scanner and Global Whitelist rule sets, but before the Content Filtering and Gateway Antimalware rule sets.

Task

1. Select Policy → Rule Sets.

2. On the rule sets tree, navigate to the position where you want to insert the rule set and click Add.
3. Click Top Level Rule Set, then click Import Rule Set from Library.
   The Add from Rule Set Library window opens.
4. Select the Cookie Authentication (for X509 Authentication) rule set and click OK.
   If conflicts arise from the import, they are displayed next to the list of rule sets. Follow one of the suggested procedures for solving them before clicking OK.
   The rule set is inserted with its nested rule sets in the rule sets tree.

Modify a rule set to change the listener port for incoming requests

You can modify the Cookie Authentication (for X509 Authentication) rule set to configure a listener port for incoming requests that you want to use instead of port 444, which is the default port. The modification is done in a nested rule set.

A special listener port must be used for receiving incoming requests if proxy exceptions are in place that prevent requests from arriving at the proxy port of the appliance. Requests that arrive at port 444 or a different port you have configured for this purpose are redirected to the authentication server.

Task

1. Select Policy → Rule Sets and expand the Cookie Authentication (for X509 Authentication) rule set.
2. Expand the nested Cookie Authentication at HTTP(S) Proxy rule set and, within this rule set, select the nested Authenticate Clients with Authentication Server rule set.
3. In the Set client context rule, click the Proxy Certificate event settings.
   The Edit Settings window opens.
4. In the Authentication Server Specific Parameters section, review the URL in the Authentication server URL field.

   The URL is by default as follows:

   ```
   https://$<propertyInstance useMostRecentConfiguration="false" propertyId=
   "com.scur.engine.system.proxy.ip"/>$:444
   ```

   When the rule is processed, the `$...$` term is replaced by the IP address of the appliance.
5. To configure a different listener port, type the number of this port here.
6. Click OK to close the Edit Settings window.
7. Click Save Changes.

Import a client certificate into a browser

A suitable client certificate must be available on a web browser to be sent with a request to an appliance in SSL-secured communication.

Procedures for importing certificates vary for different browsers and are subject to change. Browser menus can also vary depending on the operating system you are using.

The following are two possible procedures for importing a client certificate into Microsoft Internet Explorer and Mozilla Firefox.

Import a client certificate into Microsoft Internet Explorer

You can import a client certificate and make it available on Microsoft Internet Explorer for presenting it in SSL-secured communication.

Before you begin

To import the certificate file, you must have stored it within your local file system.

Task

1. Open the browser and on the top-level menu bar, click Tools, then click Internet Options.
   The Internet Options window opens.
2. Click the Content tab.
3. In the Certificates section, click Certificates.
   The Certificates window opens.
4. Click Import.
   The Certificate Import Wizard appears.
5. On the wizard pages, proceed as follows:
   a. On the Welcome page, click Next.
   b. On the File to Import page, click Browse and navigate to the location where you stored the certificate file.
   c. In the File Name field, type `*.pfx`, then press Enter.
   d. Select the certificate file and click Open, then click Next.
   e. On the Password page, type a password in the Password field. Then click Next.
   f. On the Certificate Store page, click Place all certificates in the following store.
   g. In the Certificate Store section on the same page, select Personal, then click Next.
   h. On the Completing the Certificate Import Wizard page, click Finish.
6. Confirm the message that appears by clicking OK.
7. Click Close, then click OK to close the Certificates and Internet Options windows.

Import a client certificate into Mozilla Firefox

You can import a client certificate and make it available on Mozilla Firefox for presenting it in SSL-secured communication.

Before you begin

To import the certificate file, you must have stored it within your local file system.

Task

1. Open the browser and on the top-level menu bar, click Tools, then click Options.
   The Options window opens.
2. Click Advanced, then click Encryption.
3. In the Certificates section of the Encryption tab, click View Certificates.
   The Certificate Manager window opens.
4. Click Import.
   Your local file manager opens.
5. Navigate to the certificate file that you have stored and click Open.
6. When prompted, submit a password, then click OK.

Quota management

Quota management is a means of guiding the users of your network in their web usage. This way you can ensure that resources and performance of your network are not impacted in excess.

Quotas and other restrictions can be imposed in several ways:

- **Time quotas** — Limit the time that users are allowed to spend on their web usage
- **Volume quotas** — Limit the volume that users are allowed to consume during their web usage
- **Coaching** — Limits the time that users can spend on their web usage, but allows them to exceed the configured time limit if they choose to do so
- **Authorized override** — Limits the time that users can spend on web usage in the same way as coaching

  However, the time limit can only be exceeded by an action of an authorized user, for example, a teacher in a classroom.
- **Blocking sessions** — Blocks access to the web for a configured period of time after a user attempted to access a web object, for which access was not allowed

Quotas and other restrictions can be imposed separately or in a combination of measures.

Imposing quotas and other restrictions on web usage

Imposing quotas and other restrictions you can guide the users of your network in their web usage and limit their consumption of resources.

The quota management process includes several elements:

- Quota management rules, which control the process
- Quota management lists, which rules use to impose restrictions depending on listed objects, such as URLs, IP addresses, and others
- Quota management modules (engines), which are called by rules to handle quotas, session time, and other restrictions

A quota management process is not implemented by default on Web Gateway after the initial setup.

You can implement a process by importing suitable rule sets from the rule set library and modify it to meet the requirements of your organization.

Quota management rules

The rules that control quotas and other restrictions are contained in various rule sets, for example, in a time quota, or a volume quota, or a coaching rule set.

These rules check whether the configured time and volume of web usage is exhausted and eventually block requests for further web usage.

Quota management rule sets are not part of the default rule set system, but can be imported from the rule set library.

You can review the rules that are implemented with the library rule sets, modify or delete them, but also create your own rules.

Quota management lists

Rules for quotas and other restrictions use lists of web objects and users to impose restrictions on them.

For example, a time quota rule set uses a list with URLs of particular websites to record the time a user spends visiting these websites. When the time configured for weekly usage is exhausted, further access is blocked.

You can add and remove entries to and from these lists. You can also create your own lists.

Quota management modules

The quota management modules (engines) handle time and volume parameters of the quota management process. They are checked by rules to find out, for example, about consumed and remaining times and volumes.

By configuring settings for these modules, you specify times and volumes, for example, how many hours and minutes per day users are allowed to access particular web objects.

Session time

Session time is the time allowed for a single session of web usage by a user. It is configured in different ways:

- **Session time for time quotas** — When configuring time quotas, you also configure a session time. When session time is exhausted for a user, it is deducted from the user's time quota.

  As long as the overall time quota is not exhausted, the user can start a new session. Otherwise, any request sent by the user is blocked and a block message appears.
- **Session time for volume quotas** — Session time has no impact on the volume quota for a user.

  You can still configure a session time to inform the user about the time that has been used up. When this session time is exhausted, the user can start a new session, as long as the volume quota is not exhausted.

  If you set the session time to zero, no session time is configured.
- **Session time for other quota management functions** — You can also configure session time for other types of restrictions, such as coaching, authorized override, and blocking sessions.

  When session time is exhausted for coaching and authorized overriding, a request that a user sends is blocked.

  A message appears, stating the block reason. The user can start a new session unless time quota has also been configured and is exhausted.

  The session time for a blocking session is the time during which requests sent by a user are blocked. When this time has elapsed, requests from the user are again accepted unless time quota has also been configured and is exhausted.

Combining quota management functions

A particular quota management function that is configured to restrict web usage does not impact other quota management functions. But you can combine these functions in meaningful ways.

For example, you can impose coaching on users when accessing some URL categories, while requesting authorized override credentials for accessing others.

Time quota

By configuring time quotas, you can limit the time that users of your network are allowed to spend for web usage.

Time quotas can be related to different parameters:

- **URL categories** — When time quotas are related to URL categories, users are allowed only a limited time for accessing URLs that fall into particular categories, for example, Online Shopping.
- **IP addresses** — When time quotas are related to IP addresses, users who send requests from particular IP addresses are allowed only a limited time for web usage.
- **User names** — When time quotas are related to user names, users are allowed only a limited time for web usage. Users are identified by the user names they submitted for authentication on the appliance.

These parameters are used by the rules in the library rule set for time quotas. You can create rules of your own that use other parameters in relation to time quotas.

The time that users spend on web usage is stored on the appliance. When the configured time quota has been exceeded for a user, a request that this user sends is blocked. A message is displayed to the user stating why the request was blocked.

Users are identified by the user names they submitted for authentication. If no user name is sent with a request, web usage is recorded and blocked or allowed for the IP address of the client system that the request was sent from.

Web usage can be limited to time spent per day, per week, or per month.

Configure time quotas

You can configure time quotas to limit the time users of your network spend on web usage.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, expand the rule set that contains rules for time quotas, for example, the Time Quota library rule set.

The nested rule sets appear.

3. Select the appropriate nested rule set.
   For example, to configure time quotas with regard to URL categories, select Time Quota With URL Configuration.
   The general settings and rules of the rule set appear on the settings pane.
4. In the rule set criteria, click the URL Category Block List for Time Quota list name.
   Note: A yellow triangle next to a list name means the list is initially empty and you need to fill the entries.
   The Edit List (Category) window opens.
5. Add URL categories to the blocking list. Then click OK to close the window.
6. In the criteria for one of the rules, click the URL Category Configuration settings name.
   The Edit Settings window opens.
7. Configure session time and the time quota per day, week, and month. Then click OK to close the window.
8. Click Save Changes.

Volume quota

By configuring volume quotas, you can limit the volume of web objects, measured in GB and MB, that the users of your network are allowed to download from the web.

Volume quotas can be related to several parameters:

- **URL categories** — Users are allowed to download only a limited volume of web objects through URLs that fall into particular categories, for example, Streaming Media.
- **IP addresses** — Users who send download requests from particular IP addresses are allowed only a limited volume.
- **User names** — Users are allowed to download web objects only up to a limited volume. Users are identified by the user names they submitted for authentication on the appliance.
- **Media types** — Users are allowed to download web objects belonging to particular media types only up to a limited volume.

These parameters are used by the rules in the library rule set for volume quotas. You can create rules of your own that use other parameters in relation to volume quotas.

Information on the volume that users download from the web is stored on the appliance. When the configured volume quota has been exceeded for a user, a request that this user sends is blocked. A message is displayed to the user stating why the request was blocked.

Users are identified by the user names they submitted for authentication. If no user name is sent with a request, web usage is recorded and blocked or allowed for the IP address of the client system that the request was sent from.

Web downloads can be limited to volume downloaded per day, per week, or per month.

Configure volume quotas

You can configure volume quotas to limit the volume that user of your network consume during their web usage.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, expand the rule set that contains rules for the volume quota , for example, the Volume Quota library rule set.
   The nested rule sets appear.
3. Select the appropriate nested rule set, for example, Volume Quota With IP Configuration.
   The general settings and rules of the rule set appear on the settings pane.
4. In the rule set criteria, click the appropriate blocking list name, for example, IP Block List for Volume Quota.
   Note: A yellow triangle next to the list name means the list is initially empty and you need to fill the entries.
   The Edit List (Category) window opens.
5. Add the appropriate entries to the blocking list, for example, IP addresses. Then click OK to close the window.

6. In the criteria for one of the rules, click the appropriate settings name, for example, IP Configuration.
   The Edit Settings window opens.
7. Configure the appropriate parameters, for example, session time and the volume quota per day, week, and month. Then click OK to close the window.
8. Click Save Changes.

Coaching

By configuring coaching quotas, you can limit the time that users of your network are allowed to spend for web usage, but allow them to continue if they choose to do so.

To coach the web usage of your users, you configure a coaching session with a particular length of time. When this session time has elapsed for a user, a block message is displayed. The user can then choose to start a new session.

You can configure coaching in relation to the parameters used in the Coaching library rule set, such as URL categories, IP addresses, and user names. You can also create rules of your own using other parameters.

Configure coaching

You can configure coaching to restrict web usage for the users of your network, but allow them to continue when they choose to do so after the configured time limit has been exceeded.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, expand the rule set that contains rules for coaching, for example, the Coaching library rule set.
   The nested rule sets appear.
3. Select the appropriate nested rule set, for example, Coaching With IP Configuration.
   The general settings and rules of the rule set appear on the settings pane.
4. In the rule set criteria, click the appropriate blocking list name, for example, IP Block List for Coaching.
   Note: A yellow triangle next to the list name means the list is initially empty and you need to fill the entries.
   The Edit List (Category) window opens.
5. Add the appropriate entries to the blocking list, for example, IP addresses. Then click OK to close the window.
6. In the criteria for one of the rules, click the appropriate settings name, for example, IP Configuration.
   The Edit Settings window opens.
7. Configure the appropriate parameters, for example, the session time. Then click OK to close the window.
8. Click Save Changes.

Authorized override

You can configure session time for a session that allows authorized overriding.

When this session time has elapsed, a user request is blocked and a block message is displayed. The message also asks for submission of a user name and password to start a new session.

These credentials must be those of an authorized user. For example, in a classroom situation, a user who gets blocked after termination of an authorized override session could be a student, while the teacher is the authorized user.

Authentication of this user is performed according to the configured authentication method. However, when configuring this method, you cannot let it include an integrated authentication mode.

The block message also provides an option to specify the time length of the authorized override session for the user who was blocked.

The time length that is configured for this user should not exceed the time length configured for all other users as part of the module settings for authorized overriding.

You can configure authorized overriding in relation to the parameters used in the library rule set, such as URL categories, IP addresses, and user names. You can also create rules of your own using other parameters.

Configure authorized overriding

You can configure authorized overriding to restrict the web usage of your users, but allow the configured time limit to be passed by through the action of an authorized user.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, expand the rule set that contains rules for authorized overriding, for example, Authorized Override library rule set.
   The nested rule sets appear.
3. Select the appropriate nested rule set, for example, Authorized Override With IP Configuration.
   The general settings and rules of the rule set appear on the settings pane.
4. In the rule set criteria, click the appropriate blocking list name, for example, IP Block List for Authorized Override.
   Note: A yellow triangle next to the list name means the list is initially empty and you need to fill the entries.
   The Edit List (Category) window opens.
5. Add the appropriate entries to the blocking list, for example, IP addresses. Then click OK to close the window.
6. In the criteria for one of the rules, click the appropriate settings name, for example, IP Configuration.
   The Edit Settings window opens.
7. Configure the appropriate parameters, for example, the session time. Then click OK to close the window.
8. Click Save Changes.

Blocking sessions

By configuring blocking sessions you can block requests sent by a user for a configured period of time.

A blocking session is imposed after a user has sent a request that is blocked according to a configured rule, for example, a request for a URL that falls into a category that is not allowed.

This is a means of enforcing a web security policy that handles unwanted access to web objects with more strictness.

You can configure blocking sessions in relation to the parameters that are used in the library rule set. You can also create rules of your own using other parameters.

Configure blocking sessions

You can configure blocking sessions to block session for a user over a configured period of time after an attempt to access a web object that is not allowed.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, expand the rule set that contains rules for the blocking session, for example, the Blocking Sessions library rule set.
   The nested rule sets appear.
3. Select the appropriate nested rule set, for example, Blocking Sessions With IP Configuration.
   The general settings and rules of the rule set appear on the settings pane.

4. In the rule set criteria, click the appropriate blocking list name, for example, IP Block List for Blocking Sessions.

   Note: A yellow triangle next to the list name means the list is initially empty and you need to fill the entries.

   The Edit List (Category) window opens.
5. Add the appropriate entries to the blocking list, for example, IP addresses. Then click OK to close the window.
6. In the criteria for one of the rules, click the appropriate settings name, for example, IP Configuration.

   The Edit Settings window opens.
7. Configure the appropriate parameters, for example, the period of time over which sessions are blocked. Then click OK to close the window.
8. Click Save Changes.

Troubleshooting an issue with activating a coaching session

When a user clicks the button for continuing with a coaching session on Web Gateway and the request is allowed, the session should continue immediately.

To address an issue that required a second to continue with the session, you can add a coaching quota setting and set an environment variable to activate it.

This setting is part of a coaching rule, specifying the time that a coaching setting is to last. It must be added to the property that is included in the template for continuing with a session when the user clicks the appropriate button.

Task

1. On the Web Gateway user interface, identify the quota setting that is to be added.
   a. Select Policy → Rule Sets.
   b. Navigate to the rule that terminates a coaching session when the configured time has been exceeded.

      The name of this rule is Check if coaching session has been exceeded. It is contained in rule sets for coaching, for example, in Coaching with URL configuration.

      To view this rule set and the rule, you must enable the complete rules view by selecting a coaching rule set on the rule set tree and clicking Unlock View.
   c. If details are not visible, click Show Details.
   d. Note the name of the setting for the Quota.Coaching,SessionExceeded property, for example, URL.Category.Configuration.
2. Add the coaching quota setting to the property used in the coaching template to continue with a session.
   a. Select Policy → Templates.
   b. Under Templates, navigate to the template for the page that is displayed to the user when the time configured for a coaching session has been exceeded, for example, to Default Schema → Coaching Session Exceeded.
   c. Expand the template entry, then expand en, and click html.
   d. In the HTML Editor area on the right, locate and click the Quota.Coaching.JS.Session property.
   e. In the window that opens, select the setting you identified in step 1 from the list under Settings, for example, URL.Category.Configuration. Then click OK.

      The setting is added to the property.
3. Click Save Changes.
4. Connect to the Web Gateway appliance where you added the coaching quota setting from a local system console or remotely.
5. Set an environment variable to activate the setting you added.
   a. Go to the /etc/sysconf/mwg system file and open it for editing.
   b. Append this line:

      ```
      MWG_USE_QUOTA_FULL_NAME="YES"
      ```
   c. Run this command to export the variable:

      ```
      export MWG_USE_QUOTA_FULL_NAME
      ```
6. Restart the appliance to let the setting of the environment variable take effect.

Results

Due to the coaching quota setting you added to the template for an exceeded coaching session, users can continue with the session at once when the configured session time is exceeded after clicking the appropriate button in the template.

Supporting functions

Web Gateway also provides functions, such as web caching or progress indication that do not themselves filter web objects, but support the filtering process in different ways.

Functions supporting web filtering

Some of the functions that support web filtering are implemented by rule sets of the default rule set systems. Rule sets for other functions can be imported from the library.

The default rule sets for supporting functions are all embedded in the Common Rules rule set. Rule sets for web caching, progress indication, use of file openers, and other functions can be found here.

The library provides rule sets for bandwidth throttling and the use of next-hop proxies for web access.

Web caching

A web cache is provided on the appliance for storing web objects to speed up responses to client requests.

Use of the appliance web cache is controlled by rules in a rule set.

To find out whether a web cache rule set is implemented on your appliance, review the system of rule sets on the Rule Sets tab of the Policy top-level menu.

If none is implemented, you can import the Web Cache library rule set. After importing this rule set, you can review and modify it on the Rule Sets tab to make it suit your network. Alternatively, you can create a rule set with rules of your own.

A web cache rule set typically contains rules for reading objects from the cache and writing them to it. The Enable.Cache event is used in these rules for enabling the web cache.

Additionally, there can be bypass rules that exclude objects from being read or written.

You can configure the Cache HTTP module settings to modify the caching behavior, for example, if you want to increase the caching rate.

Cache settings

You can configure the Cache HTTP module settings to modify the caching behavior, for example, if you want to increase the caching rate. These settings are provided with the Enable.Cache event.

Modifying the caching behavior can, however, lead to unfavorable results, for example, if requests that include web server authentication or responses with a Vary header are cached.

We therefore recommend using a default rule with the Always criteria and additional rules that enable web caching depending on more specific criteria. For example, the criteria might use the URL.Host property to specify a particular host.

Verify the enabling of the web cache

You can verify whether the web cache is enabled.

Task

1. Select Configuration Appliances.
2. On the appliances tree, select the appliance that you want to verify enabling of the web cache for and click Proxies (HTTP(S), FTP, ICAP, and IM).
3. Scroll down to the Web Cache section and see whether Enable Cache is selected. If necessary, enable this option.

4. If necessary, click Save Changes.

Progress indication

Progress indication is a process that shows a user who has started the download of a web object the progress made in downloading the object.

The following elements are involved in this process:

- Progress indication rules that control the process
- Progress indication modules that are called by the rules to handle the different methods for progress indication

Progress indication rules

The rules that control progress indication are usually contained in one rule set. Different rules control the use of different methods for progress indication. Accordingly, they call different modules to handle these methods.

Two methods for progress indication are available on the appliance. Which method is appropriate for a download depends on the browser that a user sends the download request with.

- **Progress page** — For Mozilla browsers

  Under this method, a page with a progress bar is shown to the user who started a download and then another page for download completion.
- **Data trickling** — For all other browsers

  Under this method, a web object is forwarded to the user in chunks and at a particular forwarding rate.

Progress indication is not implemented with the default rule set system. A library rule set provides these functions. It's name is *Progress Indication*.

You can implement this rule set, review its rules, modify or delete them, and also create your own rules.

Progress indication modules

Two progress indication modules (also known as *engines*) are available for handling different methods of progress indication:

- **Progress Page module** — For the progress page method
- **Data Trickling module** — For the data trickling method

You can configure settings for these modules to modify the way they handle these methods.

Templates are provided for configuring the two pages used for the progress page method. The configuration is done in the same way as for user message templates.

Configure progress indication

You can implement progress indication and configure it to adapt it to the needs of your network.

Complete the following high-level steps.

Task

1. Import the Progress Indication rule set from the library.
2. Review the rules in this rule set and modify them as needed.

   You can, for example, do the following:

   - Configure settings for the Progress Page module:
     - Choose a particular language for the progress page
     - Modify the text of the progress page
     - Specify timeouts for the downloaded page, for example, a timeout for the time that a page is available after the download
   -

Configure settings for the Data Trickling module:

  - Size of the first chunk in the trickling process
  - Forwarding rate

3. Save your changes.

Configure the progress indication modules

You can configure the progress indication modules to modify the way progress made in downloading a web objects is shown to users.

There are two different modules for progress indication: the Progress Page and the Data Trickling modules.

Task

1. Select Policy Rule Sets.
2. On the rule sets tree, select the rule set for progress indication.
   If you have implemented the library rule set for this function, this is the *Progress Indication* rule set.
   The rules of the rule set appear on the settings pane.
3. Make sure Show details is selected.
4. Find the rule that calls the Progress Page module or the rule that calls the Data Trickling module, according to what you want to configure.
   In the library rule set, these are the rules *Enable progress page* and *Enable data trickling*.
5. In the rule event of the appropriate rule, click the settings name.
   The Edit Settings window opens. It provides the settings for the Progress Page or the Data Trickling module.
6. Configure these settings as needed.
7. Click OK to close the window.
8. Click Save Changes.

Best practice: Working with progress indication methods

To provide progress indication for a user who requested a file upload or download, you can work with suitable progress indication methods according to your working environment.

The following methods are available:

- **Progress pages** — Progress pages keep the user informed about downloading and scanning times and provide a link for obtaining the completely processed file.

  We recommend using progress pages as the default method. Apply other methods only if progress pages are not eligible, for example, when the web browser used for downloading is not Mozilla-compatible.

  Note: When the default Progress Indication rule set is implemented, use of progress indication methods follows this recommendation.
- **Data trickling** — Data trickling informs the user about the estimated overall processing time without indicating the portion that is required for anti-malware scanning.

  This method can, however, be used for any kind of download regardless of the web browser type.
- **FTP upload timeout prevention** — This method can only be used for uploading files when Web Gateway is configured to run as an FTP proxy.

  The upload is not performed using a web browser, but requires a standalone FTP client, which can be implemented, for example, using Filezilla.

Working with progress pages

You can use progress pages with web browsers that are Mozilla-compatible. Taking packet captures allows you to track the progress indication workflow and detect issues.

Mozilla-compatible browsers

Progress pages only work for web browsers that announce their compatibility with Mozilla in the User-Agent headers of HTTP requests. This includes, for example, Mozilla Firefox, Microsoft Internet Explorer, Google Chrome, and Safari, but not Opera.

A packet capture of an HTTP request created, for example, using Wireshark, shows whether your browser is Mozilla-compatible. The capture contains a line about the User-Agent, such as the following:

```
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.2; Trident/4.0; .NET CLR 1.1.4322; .NET CLR
2.0.50727; .NET CLR 3.0.04506.648; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729)
```

This line shows that your browser is Microsoft Internet Explorer (MSIE) and that it is indeed Mozilla-compatible. For other browsers, the User-Agent lines might look as follows.

Firefox:

```
User-Agent: Mozilla/5.0 (Windows NT 5.2; rv:23.0) Gecko/20100101 Firefox/23.0
```

Chrome:

```
User-Agent: Mozilla/5.0 (Windows NT 5.2) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/29.0.1547.57 Safari/
537.36
```

Safari:

```
User-Agent: Mozilla/5.0 (Windows NT 5.2) AppleWebKit/534.57.2 (KHTML, like Gecko) Version/5.1.7 Safari/534.57.2
```

Opera:

```
User-Agent: Opera/9.80 (Windows NT 5.2) Presto/2.12.388 Version/12.15
```

The line shows that this browser is not Mozilla-compatible.

Workflow for progress pages

When progress pages are enabled on Web Gateway, a page showing the download and scanning progress is provided, as well as a page that announces download completion and offers a link for obtaining the fully downloaded file.

By creating packet captures with, for example, Wireshark, you can track the workflow as needed to detect issues. In the following example, three devices with the following IP addresses appear in the workflow:

- A client of Web Gateway: 10.10.80.1
- Web Gateway: 10.10.80.57
- A web server: 10.10.80.200

The workflow includes the following main steps.

1. A client sends a request for downloading a large file from a web server.

   ```
   <message number and timestamp> 10.10.80.1 3365 10.10.80.57 9090 HTTP 596 GET http://10.10.80.200/
   big_archive.zip
   ```

2. Web Gateway redirects the client to the progress page, sending the following HTTP status: 307 Moved Temporarily.

   The redirection specifies a location with the same IP address as in the request that was originally sent by the client, but with the subdirectory information changed to enable redirection to the progress page.

   ```
   Location: http://10.10.80.200/mwg-internal/de5fs23hu73ds/progress?id=kw0Rd85RXX
   ```

3. The client opens a new TCP connection to Web Gateway, sends a request for the location that it was redirected to, and starts downloading the progress page.

   The download begins with a message that contains the following request.

   ```
   GET http://10.10.80.200/mwg-internal/de5fs23hu73ds/progress?id=29qNbsp9oR HTTP/1.1
   ```

   Due to the mwg-internal subdirectory information in the request, Web Gateway knows that the requested page is locally available. It provides this page to the client, rather than querying the web server that is identified by the IP address in the request.

4. After downloading the progress page, the client requests an update from Web Gateway every five seconds, for example, as follows.

   ```
   GET http://10.10.80.200/mwg-internal/de5fs23hu73ds/progress?id=29qNbsp9oR&a=1&13771061 54955 HTTP/1.1
   ```

5. Web Gateway responds according to the progress made, sending the following HTTP status: 200 OK. With this status, line-based text data is sent to indicate the progress.

   The text data includes five values with particular meanings.

   -

     For example, in the following response, the text data indicates that Web Gateway has started downloading the requested file.

     ```
     16.3 MB; 204.5 MB; 7; 0; 0
     ```

     The meaning of each value in this response is as follows.

     - 1 – Amount of data downloaded so far from web server
     - 2 – Total amount of data to be downloaded
     - 3 – Percentage of download completion
     - 4 – Anti-malware scanning complete? (0 = No, 1 = Yes)
     - 5 – Time (in seconds) consumed so far by anti-malware scanning

     So, here the text data indicates that Web Gateway has already downloaded 16.3 MB from a file that is 204.5 MB large, which amounts to 7 percent, and that anti-malware scanning has not started yet.

   -

     In this response, the text data indicates that Web Gateway has downloaded the file and started anti-malware scanning, which is not yet complete.

     ```
     204.5 MB; 204.5 MB; 100; 0; 153
     ```

   -

     In this response, the text data finally indicates that Web Gateway has completed anti-malware scanning.

     ```
     204.5 MB; 204.5 MB; 100; 1; 4512
     ```

   The web browser now presents the user with a link for downloading the requested file to the client.

Workflow for data trickling

When data trickling is enabled, Web Gateway downloads a requested file and sends tiny pieces of it to the requesting client.

This keeps the connection alive while Web Gateway downloads the whole file and scans it for infections. This means that while the scanning process is going on, the data moves at a very slow rate.

If Web Gateway detected an infection after the file had been completely downloaded, only a small amount of the file would have been passed on. So the client would not have received enough malicious data to let any harm be caused.

In more detail, the workflow for data trickling is as follows.

1. Web Gateway sends an initial chunk, which is 4,096 bytes long, followed by 1 byte for every 1,000 that are downloaded, as long as the file is scanned for infections.

   As estimations of the download time are provided by the web browser, which is not aware of any Web Gateway activities, estimated download times look extremely long to the user at the beginning.

   For example, about 58 hours are estimated for downloading a 204 MB file, as shown in the following lines within the web browser.

   ```
   big_archive.zip from 10.10.80.200 Estimated time left 58 hr 10 min (4.80 KB of 204 MB copied) Download to: C:
   \Documents and Settings\big_archive.zip
   ```

2. When Web Gateway has finished anti-malware scanning and found no infections, it sends the remaining portion of the file to the client at full network speed.

Workflow for FTP upload timeout prevention

When FTP upload timeout prevention is enabled, Web Gateway sends progress indication messages to the FTP client that sent a file for uploading to an FTP server.

The messages are sent every five seconds. This keeps the TCP connection alive and gives Web Gateway time to scan the file for infections.

The messages can be viewed on the FTP server using, for example, the Filezilla tool that was also used to install the FTP client.

In more detail, the workflow for FTP upload timeout prevention is as follows.

1. Web Gateway uploads a file from the client and starts anti-malware scanning. While the scanning is in progress, Web Gateway sends a progress indication message to the client every five seconds.

   ```
   Response: 150 File status OK; about to open data connection. Response: 226-data processing in progress
   Response: 226-data processing in progress ...
   ```

   When the file has been uploaded to Web Gateway, this is indicated by a completion message.

   ```
   C:\Documents and Settings ... --> /home admin small_archive.zip 876,241 Normal Transferring 00:00:00 elapsed
   -:-:-left 100% 876,241 bytes
   ```

   The file is not yet visible, however, on the FTP server.
2. After Web Gateway has finished anti-malware scanning and found no infection, it uploads the file to the FTP server and sends a status message to the FTP client.

   ```
   ... Response: 226-data processing in progress Response: 226-data processing in progress Response: 226 File
   receive OK Status: File transfer successful, transferred 876,241 bytes in 26 seconds
   ```

   The file is now visible on the FTP server.

File opening

File opening is performed on Web Gateway to make files available for inspection and filtering that cannot immediately be accessed, for example, because they are compressed or nested in an archive.

The component that handles file opening on Web Gateway is known as the *Composite Opener*. This opener is capable of extracting compressed content, opening archives, and making other multipart files available for inspection and filtering.

The opener provides these functions depending on the file formats that it supports.

The Composite Opener also detects for various formats whether a file is protected by a password or otherwise encrypted. It cannot open these files, but based on the categorization as encrypted, you can handle them through rules. For example, you can block files that the Composite Opener detects as encrypted.

Corrupted archives are also detected. They cannot be opened, but you can likewise handle them through rules.

File opening process

The Composite Opener is enabled by the following rule, which is included in the Enable Opener rule set. This rule set is a default rule set that is nested in the Common Rules rule set.

The rule itself is also by default enabled.

Name		
Enable Composite Opener		
Criteria	Action	Event
Always	Continue	Enable Composite Opener <Default>

Other rules are provided in the Common Rules rule set, which you can enable to block files that have been detected as archives or other types of multipart files or as encrypted or corrupted.

You can configure the Composite Opener settings to set a limit to the number of levels that nesting can include within an archive. When this limit is exceeded, no file opening is performed.

A size limit can also be set to the amount of uncompressed data that the Composite Opener extracts, as well as to the compression ratio that is accepted.

Extracting metadata

Files usually have metadata associated with them in addition to the ordinary data that they contain. The two types of data can be organized in a complex structure. The Composite Opener also works on structures of this kind and makes the complete data available for further inspection.

Metadata mainly provides information on the properties of a file. In addition to standard properties with default values, some file formats allow users to configure customized properties.

After extracting both default and customized metadata, the Composite Opener makes them available in a text format that can be utilized by the Body.Text property within another rule on Web Gateway.

The rule can then be used to form a part of, for example, Data Loss Prevention (DLP) filtering.

Formats supported by the Composite Opener

The Composite Opener opens files that are not immediately accessible, for example, because they are compressed or nested in an archive. Files that are encrypted or corrupted are detected by the opener even if they cannot be opened.

The opener provides these functions depending on the file formats that it supports. The following lists show the formats that are supported with regard to:

- File opening
- Detecting file encryption

Files of the same format can be encrypted or not. If they are encrypted, they cannot be opened.

List of file formats supported for opening
7 Zip
Adobe PDF
Arj
application/vnd.ms-sync.wbxml
application/vnd.ms-windows-imaging-file
application/x-git
BinHex
Brotli encoding
Bzip2
CPIO
DMG
EML
Git
GZIP default encoding

HTML
IWork
ISO 9660
LZH
LZMA
Microsoft ActiveSync WebXML
Microsoft Cab, Compress, Excel, Power Point Presentation, Word
Microsoft Open Office XML, including docx, pptx, xlsx, and others
MultipartForm
ODF
OLE2
PlainText
Rar
RPM
RTF
Tar
UnixArchiv
URLEncodedForm
UUE
WIM
XML
XZ
Zip
Zoo

List of file formats supported for detecting encryption
Microsoft Word (when password-protected)
Microsoft Power Point Presentation (when password-protected)
Microsoft Excel (when password-protected)
Adobe PDF (when password-protected)
Microsoft Word (with restricted access)

Microsoft Power Point Presentation (with restricted access)
Microsoft Excel (with restricted access)
Adobe PDF (with restricted access)
Microsoft Office Open XML
Open Packaging Conventions containers (when password-protected)
Open Document Formula (when password-protected)
Zip archives (when password-protected)
7 Zip archives (when password-protected)
Rar archives (when password-protected)
Arj files (when password-protected)
PGP messages
Open PGP

Bandwidth throttling

You can limit the speed for uploading and downloading data to the appliance in a process known as bandwidth throttling.

Note: Bandwidth throttling is also known as *bandwidth control*.

You can use bandwidth throttling, for example, to avoid a situation where the network performance you need for completing a particular task is impacted by other users who are uploading objects to the web or are requesting large downloads from the web.

Two methods of bandwidth throttling are available on Web Gateway.

- **Basic bandwidth throttling** — Using this method you can limit the speed of data transfer from a client to a Web Gateway appliance and from the appliance to a web server.

  Under this method, you configure suitable rules that specify a maximum transferring speed. When an upload or download request matches the criteria of a rule, the speed of transferring data for this request is limited as specified in the rule.
- **Bandwidth throttling using classes** — Using this method you can limit the speed of data transfer from a client to a Web Gateway appliance and from the appliance back to the client, as well as from an appliance to a web server and from the web server back to the appliance.

  Under this method, you configure classes of transferring speed and suitable rules that make use of them. A class is specified by a speed range, for example, 1001 to 2000 Kbps.

  When an upload or download request matches the criteria of a rule, the speed of transferring data for this request is limited as specified by the class used in the rule. This speed must not exceed the maximum value, nor fall below the minimum.

  You can, for example, create a rule that applies to downloads of Windows updates and prevents them from consuming all of the bandwidth that is available on your system by limiting it to the transferring speed range of a particular class.

  You can also limit the speed of data transfer performed for traffic that does not use the proxy functions of Web Gateway, which means that the rules of your web security policy are not applicable.

  This traffic originates, for example, when Web Gateway log files are uploaded to an external server or data is downloaded from an external server to update information required for performing filtering functions on Web Gateway.

Basic bandwidth throttling

Basic bandwidth throttling limits the transferring speed when user upload objects to the web or download them.

Events in bandwidth throttling rules

Two events are available for use in rules that control bandwidth throttling:

- **Throttle.Client** — Limits the speed of data transfer from a client to the appliance

  This is the case when a client sends a request for uploading an object to a web server and the request is intercepted on the appliance together with the object.
- **Throttle.Server** — Limits the speed of data transfer from a web server to the appliance

  In this case, there has been a client request to download an object from a web server, and after this request has been filtered on the appliance and forwarded, the web server sends the object in response.

Bandwidth throttling rule for uploads

The following is an example of a rule that can execute bandwidth throttling rule for uploads.

Limit upload speed for hosts on throttling list

URL.Host is in list Upload Throttling List –> Continue – Throttle.Client (10)

The rule uses the *Throttle.Client* event to limit the speed with which uploads are performed to 10 Kbps if the web server that the data should be uploaded to is on a particular list.

In the criteria of the rule, the URL.Host property is used to retrieve the host name of the web server that is specified in the uploading request.

If the Upload Throttling List contains this name, the criteria is matched and the rule applies. The throttling event is then executed.

The Continue action lets rule processing continue with the next rule.

Bandwidth throttling rule for downloads

The following is an example of a rule that can execute bandwidth throttling rule for downloads.

Limit download speed for media types on throttling list

MediaType.EnsuredTypes at least one in list MediaType Throttling List –> Continue – Throttle.Server (1000)

The rule uses the *Throttle.Server* event to limit the speed with which downloads are performed to 1000 Kbps if the web object that should be downloaded belongs to a media type on a particular list.

In the criteria of the rule, the MediaType.EnsuredTypes property is used to detect the media type of the web object that the web server sends. An object can also be found to belong to more than one type.

If any of these types is on the Media Type Throttling List, the criteria is matched and the rule applies. The throttling event is then executed.

The Continue action lets rule processing continue with the next rule.

Bandwidth throttling rules and rule sets

We recommend that you create an overall rule set for bandwidth throttling rules and embed two rule sets in it, one for throttling uploads and another for throttling downloads. You can then let the embedded upload rule set apply for the request cycle and the embedded download rule set for the response cycle.

Within each embedded rule set, you can have multiple throttling rules that apply to different kinds of web objects.

The overall rule set for bandwidth throttling should be placed at the beginning of your rule set system. If this is not done, rules in other rule sets can start unthrottled downloads of web objects before your throttling rules are executed.

For example, a rule for virus and malware filtering could trigger the download of a web object that has been sent by a web server in response to a user request. The web object then needs to be completely downloaded to the appliance to see whether it is infected.

If your bandwidth throttling rule set is placed and processed after the rule set with the virus and malware filtering rule, bandwidth throttling is not applied to that download.

Configure bandwidth throttling

You can implement bandwidth throttling and configure it to adapt it to the needs of your network.

Complete the following high-level steps.

Task

1. Create lists of web objects for use by the bandwidth throttling rules.

   You can, for example, create the following:

   - A list of hosts that transferring speed is limited for when objects are uploaded to them
   - A list of media types that transferring speed is limited for when they an object that belongs to one of these types is downloaded

2. Create a rule set for bandwidth throttling.
3. Within this rule set, create rules for bandwidth throttling.

   You can, for example, create the following:

   - A rule for limiting transferring speed when objects are uploaded to particular hosts.
   - A rule for limiting transferring speed when an object that belongs to a particular media type is downloaded.

4. Design these rules as needed.

   You can, for example, do the following:

   - Configure a particular transferring speed for the *Throttle.Client* event that enables bandwidth throttling for uploading objects to the web.
   - Configure a particular transferring speed for the *Throttle.Server* event that enables bandwidth throttling for downloading objects from the web.

5. Save your changes.

Bandwidth throttling using classes

Bandwidth throttling can be performed on Web Gateway using classes, which cover ranges of data transferring speed.

Note: Bandwidth throttling is also known as *bandwidth control*.

A bandwidth class can, for example, cover the speed range between 1 and 1000 Kbps, while another class covers 1001 to 2000 Kpbs. Using an event in a rule, you can configure that particular web objects are uploaded or downloaded with the speed of either of these two classes or any other class that you have created.

You can also group classes and subordinate them to parent classes, assigning different priorities to the classes in a group.

Bandwidth throttling using classes can be configured for all directions that web traffic flows into. The speed of downloads requested by clients of Web Gateway can be throttled, as well as the speed of uploads.

This throttling method can be used for throttling transferring speed, but also for ensuring a minimum speed when web objects are uploaded or downloaded. In most cases, however, its main purpose will be limiting the speed of voluminous downloads from the web.

Bandwidth throttling using classes is available for most of the different network modes that can be configured for the proxy functions of Web Gateway. These modes include the explicit proxy mode, the proxy HA mode, and the transparent modes.

Bandwidth throttling using classes can also be applied to traffic that is not using the proxy functions of Web Gateway.

Events in rules for bandwidth throttling using classes

The following events are available for use in rules that control bandwidth throttling using classes:

- **Bandwidth.FromClient** — Limits the speed of data transfer from a client to the appliance
- **Bandwidth.ToServer** — Limits the speed of data transfer from the appliance to a web server
- **Bandwidth.FromServer** — Limits the speed of data transfer from a web server to an appliance
- **Bandwidth.ToClient** — Limits the speed of data transfer from the appliance to a client

When these events are used in rules, they take the name of a bandwidth class as a parameter. This way traffic can be throttled in any direction according to the limits that are configured for the respective class.

Bandwidth throttling rule for downloads

The following is an example of a bandwidth throttling rule. It applies bandwidth throttling using classes to downloads of large files from the web.

Name		
Limit transferring speed for large downloads		
Criteria	Action	Event
Body.Size greater than 10000000" –>	Continue	Bandwidth.FromServer ("Large")

The rule uses the Bandwidth.FromServer event to limit the speed at which downloads from a web server are performed.

If the size of a web object that is sent as the body of a response from the server exceeds 10 MB, the transferring speed is limited to a particular speed range. This speed range is the range that you configured for the bandwidth class named Large, for example, between 1 and 1000 Kbps.

Configure bandwidth throttling using classes

To configure bandwidth throttling using classes, create bandwidth classes and suitable rules that use these classes.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance that you want configure bandwidth throttling on, then click Bandwidth Control.
   The Bandwidth Control settings appear in the configuration pane.
3. Under Bandwidth Control, make sure that Enable Bandwidth Control is selected.
   Note: Make sure that any other bandwidth throttling functions on Web Gateway are disabled.
4. Under Bandwidth Classes, configure the classes that you want to use for bandwidth throttling.
   a. In the Bandwidth classes list, add entries for bandwidth classes.
   b. In the Interface names list, add entries for the network interfaces on Web Gateway that bandwidth throttling using classes are performed on.
5. Create suitable rules that use the bandwidth throttling events and classes to limit the transferring speed of data in incoming and outgoing web traffic on Web Gateway.
   Note: When creating a rule, make sure that you only reference classes at the lowest hierarchy level, as only the speed limits configured for these classes are applied to data transfers.

Configuring bandwidth throttling for non-proxy traffic

You can apply bandwidth throttling to limit the speed of data transfer that is not using the proxy functions of Web Gateway.

This kind of traffic originates, for example, when Web Gateway log files are uploaded to an external server or data is downloaded from an external server to update information required for performing filtering functions on Web Gateway.

As the usual web security rules are not applicable to this traffic, specific rules, which are static, must be configured here for bandwidth throttling.

Configuring bandwidth throttling for different network modes

Bandwidth throttling using classes can be configured for most of the different network modes that Web Gateway can run in as a proxy.

These modes include:

- Explicit proxy
- Transparent proxy using WCCP
- Proxy HA
- Transparent Router

When configuring bandwidth throttling using classes for any of these modes, you specify bandwidth classes and the network interfaces for the web traffic that these classes are used on.

For the proxy HA mode and the Transparent Router mode, the configuration of network interfaces differs depending on whether you want to configure a particular Web Gateway appliance as a director node or a scanning node.

When configuring a director node, you must specify network interfaces for both incoming and outgoing web traffic. This applies both to configuring an active and a fail-over director node. A scanning node only requires a network interface for outgoing web traffic.

Note: IP spoofing must be disabled in the Proxies settings for the Transparent Router mode, as well as for the Transparent Proxy mode using WCCP.

Configure bandwidth throttling for the Transparent Router and the Proxy HA mode

Configure bandwidth throttling using classes for the Transparent Router mode, as well as for the Proxy HA mode, by creating bandwidth classes and specifying network interfaces depending on whether a Web Gateway acts as a director or scanning node.

Note: IP spoofing must be disabled in the Proxies settings when configuring bandwidth throttling using classes for these network modes.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance that you want to configure bandwidth throttling on, then click Bandwidth Control.
3. Make sure that Enable bandwidth control is selected.
4. In the Bandwidth classes list under Bandwidth Classes, create the bandwidth classes that will be used for bandwidth throttling when running Web Gateway in these network modes.
5. In the Interface names list under Bandwidth Classes, specify the network interfaces for bandwidth throttling.
   - When configuring an appliance as an active or fail-over director node:
     - Click the Add icon.
     - In the Add Interfaces Name window that opens, select the name of a network interface name for inbound web traffic, for example, eth0, then click OK.
     - In the same window, select the name of a network interface name for outbound web traffic, for example, eth1, and click OK.
     - Click Save Changes.
   - When configuring an appliance as a scanning node:
     - Click the Add icon.
     - In the Add Interfaces Name window that opens, select the name of a network interface name for outbound web traffic, for example, eth1, then click OK.
     - Click Save Changes.

Next-hop proxies

Next-hop proxies can be used as an additional means of forwarding requests received from the clients of an appliance to their destinations.

When next-hop proxies are implemented, a rule in a corresponding rule set uses a module (also known as *engine*) to call next-hop proxies that have been entered into a list for forwarding requests.

For example, you can forward requests that have internal destinations using internal next-hop proxies. IP addresses of destinations that are internal are then entered into a list, which the forwarding rule relies on. In addition to this, there is a list of internal next-hop proxies for use by the rule.

A rule set with a rule for using next-hop proxies is not implemented on the appliance after the initial setup. You can import a rule set from the library and modify it according to your needs or create a rule set of your own.

When you import the next-hop proxy rule set, a list of servers that can be used as next-hop proxies is also imported. The list is initially empty and must be filled by you. You can also create more than one list and use these lists for routing in different situations.

Settings for the next-hop proxy module are imported with the library rule set. You can configure these settings to let the module use a particular next-hop proxy list and to determine the mode of calling the next-hop proxies (round-robin or failover).

Next-hop proxy modes

When multiple servers are available as next-hop proxies for routing requests, the next-hop proxy module can use several modes to call them: Round-robin, failover, and stickiness.

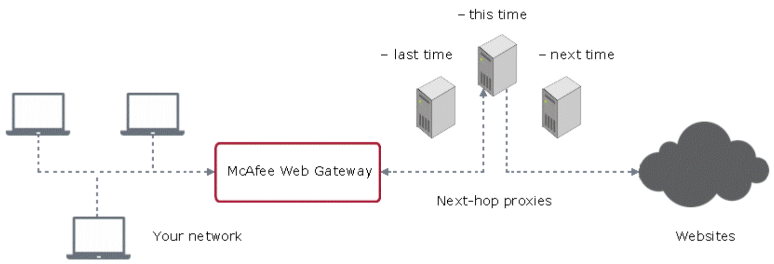
Round-robin mode for next-hop proxies

When routing a request in round-robin mode, the next-hop proxy module calls the next-hop proxy that is next on the list to the one that was called last time.

For the next request, this is handled in the same way, so all servers on the list will eventually have been used as next-hop proxies.

The following diagram shows a next-hop proxy configuration in round-robin mode.

Next-hop proxies in round-robin mode



The round-robin mode is configured as part of the settings for next-hop proxies.

Failover mode for next-hop proxies

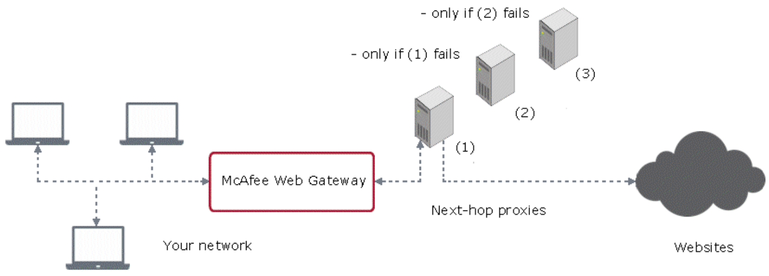
When routing a request in failover mode, the next-hop proxy module calls the first next-hop proxy on the list.

If this next-hop proxy fails to respond, the call is repeated until the configured number of retries is reached. Only then is the second next-hop proxy in the list tried. It is called in the same way as the first, and eventually the third next-hop proxy in the list is tried.

This is continued until a next-hop proxy responds or all next-hop proxies in the list were found to be unavailable.

The following diagram shows a next-hop proxy configuration in failover mode.

Next-hop proxies in failover mode



The failover mode is configured as part of the settings for next-hop proxies.

Next-hop proxy stickiness

A next-hop proxy can also be selected according to what is known as the "sticky" mode. In this mode, requests of a particular kind, for example, requests coming in from the same client of Web Gateway are directed to the same next-hop proxy.

The part of a request that qualifies it for being handled in sticky mode is configured as the value of a property on Web Gateway. An event in a rule sets the property to this value.

The name of the property that is configured to enable next-hop proxy stickiness is NextHopProxy.StickinessAttribute. If you want, for example, to let requests from the same client be directed to the same next-hop proxy, you can use the IP address of a client as the value for this property.

In addition to creating a rule, you must also select stickiness as an option within the settings for handling next-hop proxies. The settings also include an option for limiting the time that the next-hop proxy stickiness mode is applied.

Rule for configuring next-hop proxy stickiness

The following sample rule sets the NextHopProxy.StickinessAttribute property to the value of the Client.IP property to let requests with the same client IP address be directed to the same next-hop proxy.

Name			
Set next-hop proxy stickiness attribute			
Criteria		Action	Event
Always	–>	Continue	Set NextHopProxy.StickinessAttribute = IP.ToString(Client.IP)

The rule uses an event to set the NextHopProxy.StickinessAttribute property. As this property is of the string type, the value for the Client.IP property must be converted into a string before it can be used for setting the NextHopProxy.StickinessAttribute property.

Configure next-hop proxies

You can implement the use of next-hop proxies and configure it to adapt it to the needs of your network.

Complete the following high-level steps.

Task

1. Import the Next Hop Proxy rule set from the library.
2. Review the rules in this rule set and modify them as needed.

   You can, for example, do the following:

   - Edit the lists used by the next-hop proxy rule.

     Note: A yellow triangle next to a list name means the list is initially empty and you need to fill the entries.
   - Configure the settings of the Next Hop Proxy module
3. Save your changes.

Add a next-hop proxy to a list

To add a next-hop proxy to a list, complete the following steps.

Task

1. Open the Edit Settings window for settings of the Next Hop Proxy module.
2. Under Next Hop Proxy Server, select a next-hop proxy list from List of next-hop proxy servers and click Edit.
   The Edit List (Next Hop Proxy Server) window opens.
3. Under List Content, click the Add icon.
   The Add Next Hop Proxy window opens.
4. Configure settings for a next-hop proxy as needed.
5. Click OK for all open windows.
6. Click Save Changes.

Results

The next-hop proxy is added to the list that you selected.

Configure the Next Hop Proxy module

You can configure the Next Hop Proxy module to modify the way next-hop proxies are used for forwarding requests to the web.

Task

1. Select Policy → Rule Sets.
2. On the rule sets tree, select the rule set for next-hop proxies.
   If you have implemented the library rule set for this function, this is the *Next Hop Proxy* rule set.
   The rules of the rule set appear on the settings pane.
3. Make sure Show details is selected.
4. Find the rule that calls the Next Hop Proxy module.
   In the library rule set, this is the rule *Use internal proxy for internal host*.
5. In the rule event, click the settings name.
   In the library rule set, this name is *Internal Proxy*.

The Edit Settings window opens. It provides the settings for the Next Hop Proxy module.

6. Configure these settings as needed.
7. Click OK to close the window.
8. Click Save Changes.

Configure next-hop proxy stickiness

To configure next-hop proxy stickiness, select this mode in the Next Hop Proxy settings and add a stickiness rule to a rule set for handling next-hop proxies.

Task

1. Select the stickiness mode for next-hop proxies.
   a. Select Policy → Settings.
   b. On the Engines branch of the settings tree, expand Next Hop Proxy and select the settings that you want to configure next-hop proxy stickiness for.
   c. Under Next Hop Proxy Server select Sticky .
   d. Under Minimum time for stickiness, modify the time period during which the stickiness mode is applied as needed.
2. Add a rule for next-hop proxy stickiness.
   a. Select Policy → Rule Sets.
   b. Open a rule set for next-hop proxy handling, for example, the Next Hop Proxy library rule set.
   c. Add a rule that sets the NextHopProxy.Stickiness.Attribute property to the value for identifying the requests that are directed to the same proxy.
3. Click Save Changes.

Results

Requests that contain the part specified in the additional rule are now directed to the same next-hop proxy during the configured time period.

Configure next-hop proxies for cloud use

You can configure next-hop proxies to redirect access to websites using additional proxies and make them also available for cloud use.

When configuring IP addresses for next-hop proxies, internal and other protected addresses must not be used as cloud addresses. So, for example, do not use internal IP addresses for McAfee Web Gateway Cloud Service here.

Addresses that must not be used are also excluded by a check that the proxy functions on Web Gateway perform.

Task

1. Select Policy → Rule Sets.
2. Import the Next-Hop Proxy rule set.
   a. On the rule sets tree, navigate to the position where you want to insert this rule set.
   b. From the Add drop-down list, select Rule Set from Library.
   c. From the Next-Hop Proxy group of rule sets select the Next-Hop Proxy rule set.
      If conflicts arise when importing this rule set, they are displayed in the window. Methods for solving them are also suggested.
   d. Click OK.

   The rule set is inserted in the rule sets tree. A view of the rule set is shown in the configuration pane. It is enabled by default.

   Settings for the rule set are also implemented.
3. In the configuration pane, click Enable in cloud to make this rule set available for cloud use.
4. To configure settings for the rules in this rule set, click Edit under Next-Hop Proxy Settings.

The Edit Settings window opens with the settings for next-hop proxies displayed.

5. Edit these settings as needed.

   Note: For cloud use, only the round-robin mode is available to call proxies from the selected list of next-hop proxies.

6. Click Save Changes.

Results

Next-hop proxies can now be used to redirect access to websites for on-premise and cloud users.

Next-hop proxies for SOCKS traffic

A next-hop proxy can be configured to forward web traffic under the SOCKS (Sockets) protocol.

Under this protocol, web traffic also follows an embedded protocol, which can be detected on Web Gateway. If the embedded protocol is HTTP or HTTPS, web traffic can be filtered according to the configured rules.

Versions 4 and 5 of the SOCKS protocol can be used for forwarding web traffic. When setting up a next-hop proxy, you can configure the SOCKS version to use. By default, the version of the incoming traffic is also used when it is forwarded.

Configure a next-hop proxy for SOCKS traffic

To configure a next-hop proxy for SOCKS traffic, let Web Gateway run as a SOCKS proxy and implement suitable rule sets for enabling a next-hop proxy and filtering the traffic.

Enable a SOCKS proxy

Enable Web Gateway to run as a SOCKS proxy by configuring the proxies settings accordingly.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the Web Gateway appliance that you want to configure for running as a SOCKS proxy and click Proxies.
3. Scroll down to the SOCKS Proxy section and select Enable SOCKS proxy.
4. Click Save Changes.

Configure a next-hop proxy rule set for SOCKS traffic

To configure a rule set for SOCKS traffic, modify the criteria of the Next Hop Proxy library rule set and add a rule that enables a next-hop proxy under the SOCKS protocol.

Task

1. Import the *Next Hop Proxy* library rule set from the library.
2. On the rule sets tree, move the rule set up and let it follow immediately after the rule set that you are using for authenticating users, for example, the *Explicit Proxy Authentication and Authorization* rule set.
3. Replace *Always* as the rule set criteria by *Connection.Protocol equals "SOCKS"*.
4. Add a rule that enables a next-hop proxy.
   a. Configure the rule criteria to let the rule apply for particular requests.

      For example, use *Client.IP matches in list Client IP* as the rule criteria to let the rule apply only for requests sent from clients with an IP address that is on a particular list.

   b. Configure *Continue* as the rule action.
   c. Configure *Enable Next Hop Proxy* as the rule event.
   d. Configure the settings of the rule event.
      - Add a next-hop proxy to the list of next-hop proxies.
        When adding the next-hop proxy, make sure that you specify the SOCKS parameters as needed.
      - Configure the remaining options as needed.
5. Click Save Changes.

Results

You can add more rules to the Next Hop Proxy rule set, using different criteria each time for setting up a next-hop proxy.

Configure the SOCKS Proxy rule set

Configure a setting in the SOCKS Proxy rule set that is required for filtering traffic that is forwarded to next-hop proxies under the SOCKS protocol.

Task

1. Import the *SOCKS Proxy* rule set from the library.
   The rule set can be found under *Common Rules*.
2. On the rule sets tree, let the rule set follow immediately after the *Next Hop Proxy* rule set.
3. In the nested *Protocol Detection* rule set of the *SOCKS Proxy* rule set, click the settings for the *Protocol Detector* module.
   The default name of these settings is *Default*.
   The Edit Settings window opens.
4. Under Protocol Detector Options, select Determine next-hop proxy after receiving embedded data.
5. Click OK to close the window.
6. Click Save Changes.

Results

For more information about the SOCKS Proxy rule set, see the *Proxies* chapter.

Rules for enabling next-hop proxies for SOCKS traffic

You can add various rules to the Next Hop Proxy rule set, using different criteria for setting up a next-hop proxy.

The following rule enables a next-hop proxy for a request that was received from a client of Web Gateway with an IP address that is on a particular list.

Name		
Enable next-hop proxy for SOCKS traffic if received from listed client		
Criteria	Action	
Client.IP matches in list –> *Client IPs*	Continue	Enable Next Hop Proxy<SOCKS Next Hop Proxy>

The rule uses the *Client.IP* property to check whether the IP address of the client that a request was received from is on the list.

In this case, an event enables a next-hop proxy for this traffic. The event is executed with particular settings that you can configure to specify, for example, the version of the SOCKS protocol that should be used.

The next rule enables a next-hop proxy if the embedded protocol under the SOCKS protocol is HTTP.

Name		
Enable next-hop proxy for SOCKS traffic with embedded HTTP protocol		
Criteria	Action	
ProtocolDetector.Detected->Protocol<Default> equals "HTTP"	Continue	Enable Next Hop Proxy<Embedded Protocol HTTP Next Hop Proxy>

The rule uses the *ProtocolDetector.DetectedProtocol*< property to check whether the embedded protocol is HTTP.

In this case, an event enables a next-hop proxy for this traffic. The event is executed with particular settings that you can configure to specify, for example, the version of the SOCKS protocol that should be used.

When using this rule, you also need to enable the option Determine next-hop proxy after receiving embedded data in the settings for the Protocol Detector module (or *engine*).

The next rule enables a next-hop proxy if the embedded protocol under the SOCKS protocol is HTTPS.

Name		
Enable next-hop proxy for SOCKS traffic with embedded HTTPS protocol		
Criteria	Action	
ProtocolDetector.Detected->Protocol<Default> equals "HTTPS"	Continue	Enable Next Hop Proxy<Embedded Protocol HTTPS Next Hop Proxy>

The rule uses the *ProtocolDetector.DetectedProtocol*< property to check whether the embedded protocol is HTTP.

In this case, an event enables a next-hop proxy for this traffic. The event is executed with particular settings that you can configure to specify, for example, the version of the SOCKS protocol that should be used.

When using this rule, you also need to enable the option Determine next-hop proxy after receiving embedded data in the settings for the Protocol Detector module (or *engine*).

The next rule enables a next-hop proxy for any embedded protocol under the SOCKS protocol.

Name		
Enable next-hop proxy for SOCKS traffic with any embedded protocol		
Criteria	Action	
Connection.Protocol.Parent-> equals " SOCKS"	Continue	Enable Next Hop Proxy<Embedded Protocol Next Hop Proxy>

The rule uses the *Connection.Protocol.Parent* property to check whether the SOCKS protocol appears as the parent protocol in a request for forwarding SOCKS traffic to the web. If SOCKS appears as the parent protocol, it means that there must be an embedded protocol.

In this case, an event enables a next-hop proxy for this traffic. The event is executed with particular settings that you can configure to specify, for example, the version of the SOCKS protocol that should be used.

The next rule is very similar to the preceding rule. It enables a next-hop proxy for traffic under the SOCKS protocol or traffic that goes on under the HTTP protocol directly, without being embedded in the SOCKS protocol.

Name		
Enable next-hop proxy for SOCKS traffic with any embedded protocol		
Criteria	Action	
Connection.Protocol.Parent –> *equals " SOCKS" OR Connection.Protocol equals "HTTP"*	Continue	Enable Next Hop Proxy<Embedded Protocol Next Hop Proxy>

Using a TCP proxy to forward traffic to a SOCKS next-hop proxy

Web traffic coming in over the listener port of a TCP proxy on Web Gateway can be forwarded to a next-hop proxy that follows the SOCKS protocol.

The forwarding is then performed even for traffic that is not coming in under SOCKS. Handling web traffic in this way is sometimes referred to as *SOCKSification*.

To implement the forwarding, you need to create a suitable rule. A suitable rule set for this rule is the Next-Hop Proxy rule set, which you can import from the rule set library.

Next-hop proxy settings, which you can use for completing the next-hop proxy part of the configuration, are imported with the rule set.

Rule for forwarding traffic from a TCP proxy to a SOCKS next-hop proxy

The rule that forwards the traffic might look as the example shown here. It assumes that the listener port of the TCP proxy is 9102.

If traffic is coming in on this port, the rule event enables a next-hop proxy that runs under the SOCKS protocol. The event settings specify the details of this proxy.

Name				
Forward traffic from a TCP proxy to a SOCKS next-hop proxy				
Criteria		Action		Event
Proxy.Port equals 9102	–>	Continue	—	Enable Next-Hop Proxy<SOCKS Next-Hop>

Settings for a SOCKS next-hop proxy

The settings for the next-hop proxy that runs under SOCKS are configured as the settings of the event that enables this proxy. You can use the Next-Hop Proxy settings that are imported with the rule set as a starting point.

When creating the settings for the SOCKS next-hop proxy, you can keep the existing default list of next-hop proxies, but you need to add a proxy to this list.

The following values must be specified for this proxy:

- Name
- IP address and port
- SOCKS protocol version

For the remaining next-hop proxy options, you can keep the default values.

Troubleshooting

When issues with proxies occur, connection traces or TCP dumps are helpful. TCP dumps can also be loaded into an analyzing tool, such as Wireshark, that "understands" SOCKS.

If an error occurs on the connection from the TCP proxy to the SOCKS next-hop proxy, for example, if the next-hop proxy can't be reached after performing the number of configured retries, the connection is closed.

The user who works with a client of Web Gateway might notice the closing. But because TCP and SOCKS don't support reasonable error reporting to the client, this is usually all that is noticed on the client.

If the next-hop proxy can detect an embedded protocol within SOCKS, for example, HTTP, a block page or a similar message might be sent to the client.

Forward all traffic from a TCP proxy to a SOCKS next-hop proxy

You can forward all web traffic coming in over a TCP proxy listener port on Web Gateway to a next-hop proxy that follows the SOCKS protocol, even if this traffic is not coming in under SOCKS.

To implement the forwarding of web traffic in this way, you need to create a rule.

Task

1. Import the Next-Hop Proxy rule set from the library.
2. Prepare the creation of a new rule in this rule set.
   a. Select the imported rule set and click Unlock View to access the complete rules view.
   b. Copy the rule that is contained by default in this rule set and paste the copy next to the original rule as a starting point for configuring a new rule.
   c. Select the copied rule and click Edit.
3. Create the rule in the Edit Rule window.
   a. Specify a name for the new rule, for example, `Forward traffic from a TCP proxy to a SOCKS next-hop proxy`.
   b. Configure the following rule criteria: `Proxy.Port equals 9012`.
   c. Leave Continue as the rule action.
   d. Configure settings for the Enable Next-Hop Proxy event.

      Using these settings, the event enables a new next-hop proxy that runs under the SOCKS protocol.

      - Select the event and click Edit.
      - In the Edit Event window under Settings, click Add.
      - In the Add Settings window, specify a name for the new settings, for example, `SOCKS Next-Hop`.
      - Under Settings Content, select the default Internal Proxies list and click Edit.
      - In the Edit List window, click the Add icon under List Content to add a next-hop proxy.
      -

        In the Add Next-Hop Proxy window, configure the new next-hop proxy:

        - Identifier, for example, `Lara`
        - IP address and port, for example, `10.140.39.233` and `1080`
        - SOCKS protocol version, for example, SOCKS v5.

You can leave the default values for the remaining settings options.

e. Close all windows.

The new rule appears in the Next-Hop Proxy rule set with the values that you configured.

4. Click Save Changes.

Results

If any web traffic is coming in on the TCP proxy listener port, it is now forwarded to the SOCKS next-hop proxy.

Best practices - Troubleshooting next-hop proxy issues

Reviewing the settings for next-hop proxies can help solve issues with connection delays or unavailability.

Next-hop proxy issues are indicated by alerts on the dashboard. Settings for next-hop proxies can be reviewed to troubleshoot these issues and also to ensure that next-hop proxies have appropriately been configured to enable cloud lookups for URL filtering and regular updates of other filtering information.

Next-hop proxy alerts

Alerts on the dashboard indicating next-hop proxy issues look like this.

- Next hop proxy 10.44.44.44 has been marked as down for 10 seconds

  This alert appears if, after trying to connect to a next-hop proxy, Web Gateway detects that the next-hop proxy is down and 10 seconds are configured as the waiting time until the next retry.

  The waiting time begins after the configured number of retries, which are performed immediately, have been completed unsuccessfully.

- Connection to next hop proxy 10.44.44.44 failed

  This alert appears, if after trying to connect to a next-hop proxy, Web Gateway detects that the next-hop proxy is down and no waiting time (0 seconds) is configured. After unsuccessfully completing the configured number of retries, Web Gateway immediately performs the next retry.

Connection retry settings for next-hop proxies

If you notice slowness on next-hop proxy connections, we recommend reviewing the connection retry settings, which are part of the Next Hop Proxy settings.

The settings include the number of retries Web Gateway performs after a failed connection attempt, and the waiting time before performing the next retry after the configured number of retries has been completed unsuccessfully.

We recommend configuring a low number of retries, for example, 3, and no waiting time at all.

Configuring the settings in this way does not prevent the alerts from appearing, but avoids unnecessary delay with connection retries.

Avoiding delay is also important, as sometimes a next-hop proxy can erroneously be marked as down on Web Gateway, which would make waiting until the next retry even less appropriate.

Next-hop proxies for URL filtering

Slowness or failure in URL filtering can also be related to the next-hop proxy configuration.

The settings for URL filtering are by default configured to let categorizations of URLs be looked up on a cloud server of theMcAfee Global Threat Intelligence McAfee Global Threat Intelligence system if no category for a given URL can be found in the local database on a Web Gateway appliance.

Next-hop proxies can be configured for connecting to these servers as part of the URL Filter settings. If no next-hop proxies are configured or if the configuration settings are faulty, attempts to perform cloud lookups can fail or be slow.

Next-hop proxies for updates

You can also use next-hop proxies for connecting to the update servers, which provide regular updates for anti-malware filtering, URL filtering, and other activities. Next-hop proxies for updates are configured as part of the Central Management settings.

Review the connection retry settings for next-hop proxies

Review the connection retry settings for next-hop proxies to troubleshoot connection issues.

The connection retry settings can cause additional delay on connections if not appropriately configured.

Task

1. Navigate to the connection retry settings for next-hop proxies.
   a. Select Policy → Settings.
   b. On the settings tree, expand Next Hop Proxy and click the settings that you want to review.
      The settings appear in the configuration pane.
   c. Under Next Hop Proxy Server, select a list of next-hop proxy servers and click Edit.
      The Edit List (NextHopProxy) window opens.
   d. From the List Content list, select a next-hop proxy and click Edit.
      The Edit NextHopProxy window opens.
2. Under Next Hop Proxy Definition, configure the following.
   a. Set Number of retries to `3`.
   b. Set After final failure wait to `10` (seconds).
   Close all open windows when you are done.
3. Click Save Changes.

Review the settings of next-hop proxies for URL filtering

Review the settings of next-hop proxies that are used in URL filtering to troubleshoot connection issues.

Next-hop proxies are used in URL filtering to connect to the cloud servers of the McAfee Global Threat Intelligence system.

Task

1. Select Policy → Settings.
2. On the settings tree, expand URL Filter and click the particular settings that you want to review.
   The settings appear in the configuration pane.
3. Scroll down to Advanced Settings and check whether one or more next-hop proxies are correctly configured under Proxy Settings.
4. Correct the settings as needed. If no next-hop proxy is configured, add settings for one or more of them.
5. Click Save Changes if you have modified or made additions to the settings.

Review the settings of next-hop proxies for updates

Review the settings of next-hop proxies used for updates to troubleshoot connection issues.

Next-hop proxies are used for updates to connect to the various update servers.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance that you want to review settings on and click Central Management.
3. Scroll down to Automatic Engine Updates and verify that the following applies:
   - Enable Update Proxies is selected.
   - One or more next-hop proxies are correctly configured under Update proxies.
4. Correct the settings as needed. If no next-hop proxy is configured, add settings for one or more of them.
5. Click Save Changes if you have modified or made additions to the settings.

User messages

Messages can be sent to users when a filtering rule blocks their requests for web access or affects them in other ways.

When you are administering this process, you are mainly dealing with the following:

- **Messages** — Messages are sent to users to inform them that their requests for web access are blocked, or redirected, or that they need to authenticate.
- **Action settings** — Messages to users are part of the settings for the action that is explained in a message.
- **Templates** — Messages to users are based on templates, which can be edited using the Template Editor.

Default settings apply for user messages and their templates after the initial setup of the appliance, which you can review and modify as needed.

Sending messages to users

Messages are sent to users to inform them about actions of the filtering rules that affect them.

User messages belong to different types and are based on templates.

Message types

There are different types of user messages, according to the action that a message informs a user about.

- **Authenticate message** — Informs a user that authentication is required to access a URL
- **Block message** — Informs a user that a request was blocked for various reasons, for example, because a virus was detected in the requested object
- **Redirect message** — Informs a user that redirecting to another URL is needed for accessing the requested object

Message templates

Messages that are sent to users are based on templates. To modify what a message looks like, you need to adapt these templates. You can do this under the settings for an action.

Message templates contain standard text with variables. The variables are filled with values as needed in a given situation.

All variables used in message templates are also properties used by rules. For example, *URL* is a variable in a message text and a property when used in a rule to exempt URLs from filtering.

Different versions can exist of a particular template regarding:

- **Language** — English and other languages
- **File format** — html or txt

Activities that you can complete when editing a message template include the following:

- Select a language for the message
- Edit the message text
- Replace the variables in the template
- Specify a block reason for logging purposes (only for Block action templates)
- Specify a URL for redirecting (only for Redirect action templates)

Message text and variables

The following text and variables could be contained in a Block message that is sent to a user when access to a requested object has been block due to a virus infection of the object.

- **Standard text** — *The transferred file contained a virus and was therefore blocked.*
- **Variables** — as follows:
  - **URL** — URL that the user requested to access the file

    The variable used to display a URL is *$URL$*.
  - **Virus name** — Name of the found virus that caused the blocking of the file

    The variable used to display a virus name is *$List.OfString.ByName(String)$*.

When editing a message template, you can select and insert variables from a list of properties. To serve as variables in message templates, these are converted into strings (if they are not strings already).

For this reason, it makes no sense to select "string converter" properties here, which are properties whose job it is to convert other data types into strings, for example, the NumberToString(String) property.

Template Editor

The Template Editor is a component of the user interface that allows you to work with templates for messages to users. You can access it in several ways.

- Select it under the Policy top-level menu.
- Select the settings for an action:
  - Under Policy → Settings

    OR
  - In a rule of a rule set, after enabling the complete rules view and Show details.

  Then click Edit within these settings for a template collection or an individual template.

Edit the text of a user message

You can edit the text of a user message to adapt it to the requirements of your network.

Task

1. Select Policy → Rule Sets.
2. Select the rule set of a rule that includes the action with the user message you want to edit.
   For example, select the Gateway Antimalware rule set.
   The rules of the rule set appear on the settings pane.
3. Make sure that Show Details is enabled.
4. In the appropriate rule, click the settings of the action with the user message.
   For example, in the rule Block if virus was found, click the Virus Found settings of the Block action.
   The Edit Settings window opens.
5. Next to the Template Name field, click Edit.
   The Template Editor opens.
6. On the templates tree, expand the appropriate action template folder, for example, Virus Found.
   The available language versions of the template appear.
7. Expand a language version, for example, en for English.
   The available message formats of the language version appear.
8. Select a format, for example, html.

   The content of the template appears on the configuration pane in the selected format. It contains the text of the user message.

   For example, in the HTML format of the English Virus Found template, this text reads initially:

   The transferred file contained a virus and was therefore blocked.
9. Edit the text as needed.
10. Click OK to close the Edit Settings window.
11. Click Save Changes.

Modifying a block page

You can modify a block page to adapt it to your corporate design and to provide additional information, for example, for debugging purposes.

A block page appears on a client of a Web Gateway when a request for web access is blocked. Use the Template Editor to modify what the user sees on this page.

Note: Other pages sent to users can be modified in the same way as block pages.

Modifying the parts of a block page

A block page consists of header and footer with general information and the middle with issue-related information. These parts should be modified in different ways.

Unlike the issue-related information, header and footer information is shown on all block pages, for example, the corporate logo.

Information on a block page is in many cases provided by inserting suitable properties in the page template. For example, to show the media type of a web object, the MediaTypeEnsuredTypes property is inserted, displaying the particular value that applies to a request.

Retrieving a value for a property requires considerable effort in some cases, for example, when Web Gateway must perform a database lookup to retrieve a value, such as the one for Authentication.Username. Other properties, for example, Client.IP requires less effort.

Tip: Best practice: Do not insert properties requiring a larger effort to retrieve their values in the header or footer of a block page to avoid an unnecessary impact on performance.

Modify the footer of a block page

Modify the footer of a block page by inserting additional information for debugging purposes.

Inserting the following properties in the footer of a template for a block page provides useful debugging information. The effort for retrieving the values of these properties is rather small.

- System.Hostname — Host name of the Web Gateway appliance that handled a request for web access
- Proxy.IP — IP address of the Web Gateway appliance that acted as a proxy to handle the request
- Client.IP — IP address of the Web Gateway client that the request was sent from
- Rules.CurrentRule.Name — Name of the rule that was applied when the request was handled
- Rules.CurrentRuleSet.Name — Name of the rule set that the rule belongs to

Task

1. Select Policy → Templates.
2. In the Templates pane, expand Default Scheme, then expand the language folder that you want to work with, for example, en.
3. Click html.

   In the File System pane, the index.htm file of the en language folder within the default folder is selected. The content of this file is displayed in the HTML Editor pane.
4. Locate this section of the file content:

   ```
   <tr> <td class='footData'> generated at <span id="time">$DateTime.ToISOString$/<span> by McAfee Web Gateway
   <br /> $Get.Header$ </td> </tr>
   ```

5. Replace this section with the following:

   ```
   <tr> <td class='footData'> generated at $<propertyInstance useMostRecentConfiguration="false"
   propertyId="com.scur.engine.datetimefilter.datetime.toisostring"/>$ by $<propertyInstance
   useMostRecentConfiguration="false" propertyId="com.scur.engine.system.hostname"/>$ ($<propertyInstance
   useMostRecentConfiguration="false" propertyId="com.scur.engine.system.proxy.ip"/>$) <br /> Client IP Address:
   $<propertyInstance useMostRecentConfiguration="false" propertyId="com.scur.engine.system.client.ip"/>$ <br />
   Rule Name: $<propertyInstance useMostRecentConfiguration="false"
   propertyId="com.scur.engine.system.rules.currentrulename"/>$ ($<propertyInstance
   useMostRecentConfiguration="false" propertyId="com.scur.engine.system.rules.currentrulesetname"/>$) </td>
   </tr>
   ```

   After replacing the section, the terms representing properties are converted to display the property names. The result should look like this:

   ```
   <tr> <td class='footData'> generated at $DateTime.ToISOString$ by $System.Hostname$ ($Proxy.IP$) <br /> Client
   IP Address: $Client.IP$ <br /> Rule Name: $Rules.CurrentRuleName$ ($Rules.CurrentRuleSetName$) </td> </tr>
   ```

In the footer of a real block page, the current property values will be shown, for example, as follows:

```
<tr> <td class='footData'> generated at 2016-09-30 by testhost (10.120.233.2) <br /> Client IP Address:
10.120.233.10 <br /> Rule Name: Allow URLs that match in URL WhiteList (URL Filtering) </td> </tr>
```

6. Click Save Changes.

Modify the issue-related part of a block page

Insert additional information in the issue-related part of a block page.

The additional information can, for example, be information about a user who sent a request for web access and has successfully passed the authentication process.

The Authentication.Username property must be inserted in the block page to provide this information. Retrieving a value for this property requires a lookup, but the effort can still be considered as reasonable, as his lookup is not performed each time a block page is displayed.

Task

1. Select Policy → Templates.
2. In the Templates pane, expand Default Scheme, then expand the template and language folders that you want to work with, for example, URL.Blocked, and then en.
3. Click html.

   In the File System pane, the URL.Blocked.htm file of the en language folder within the default folder is selected. The content of this file is displayed in the HTML Editor pane.
4. Locate this section of the file content:

```
<tr> <td class='infoData'> <b>URL</b><script type="text/javascript">break_line("$URL$");</script><br />
<script type="text/javascript">writeToDocument("<b>URL Categories: </b>" + "$Categorylist.ToString"></
script><br /> <b>Reputation</b>: $URLReputationString$<br /> <b>Media Type</b>: $MediaType.EnsuredTypes$<br />
```

5. Copy the following line and paste it below the original line.

```
<b>Media Type</b>: $MediaType.EnsuredTypes$<br />
```

6. Replace the words `Media Type` by `User` in the pasted line.
7. Delete the property name in this line and place the cursor inside the area that is now empty, then click Add.
8. In the Choose Property window that opens, select Authentication.Username and click OK.

   The property is inserted in the line. Its current value will now be shown when the block page is displayed.
9. Click Save Changes.

Change the logo on a block page

Replace the default logo file with a file of your own to change the logo on a block page.

Task

1. Store an image file with your logo in a location of your file system.
2. Navigate to the default image file.
   a. Select Policy → Templates.
   b. In the File System pane, expand default, and then img.
      All image files that are stored in the file system appear.
3. Replace the default image file with your file.
   a. Copy the file named mwg_logo.png and rename it to mwg_logo.png-bak.
   b. Close the img folder, select the closed folder, and click Add.
   c. From the menu that opens, select Existing file or directory, then select your image file in the file system window that opens and click Open.
      The file appears in the img folder.

d. Rename the file to mwg_logo.png .

Note:

You can give the file a different name, but then you must also replace the default name in the following line of the index.html file in the default folder with the new name.

```
<img src='$Proxy.EndUserURL$/files/default/img/logo_mwg.png'>
```

4. Click Save Changes.

Change the color of the background bars on a block page

Replace the default image file for the background bars with a file of your own to change the color of those bars.

Task

1. Store an image file with background bars of your preferred color in a location of your file system.
2. Navigate to the default image file.
   a. Select Policy → Templates.
   b. In the File System pane, expand default, and then img.
      All image files that are stored in the file system appear.
3. Replace the default image file with your file.
   a. Copy the file named bg_navbar.png and rename it to bg_navbar.png-bak.
   b. Close the img folder, select the closed folder, and click Add.
   c. From the menu that opens, select Existing file or directory, then select your image file in the file system window that opens and click Open.
      The file appears in the img folder.
   d. Rename the file to bg_navbar.png .

Note:

You can give the file a different name, but then you must also replace the default name in the following line of the index.html file in the default folder with the new name.

```
<td class='helpDeskData' background='$Proxy.EndUserURL$/files/$Proxy.Message.Collection$/img/
bg_navbar.jpg'>
```

4. Click Save Changes.

Results

You can change more colors by modifying the stylessheet.ccs file in the default folder.

For example, by replacing the color code in the following line, you can modify the general background color of the template.

```
body { background-color: #ced1d4; ... }
```

Administrator accounts

Accounts can be set up and managed on a Web Gateway appliance to grant administrators access rights that are required to perform administrator activities.

To grant these rights, the following must be configured:

- **Accounts** — Accounts are set up for users who work as administrators. They can be set up on a Web Gateway appliance or an external server.
- **Roles** — Roles are created and configured to include the access rights that are needed by an administrator in each role. Roles are then allotted to accounts.

Add an administrator account

You can add administrator accounts to the account that is created by the appliance system at the initial setup.

Task

1. Select Accounts → Administrator Accounts.
2. Under Internal Administrator Accounts, click Add.
   The Add Administrator window opens.
3. Add a user name, a password, and other settings for the account. Then click OK.
   The window closes and the new account appears in the accounts list.
4. Click Save Changes.

Edit an administrator account

You can edit administrator accounts including the one that is created by the appliance system at the initial setup.

Task

1. Select Accounts → Administrator Accounts.
2. Under Internal Administrator Accounts, select an account and click Edit.

   Before selecting an account, you can type a filtering term in the Filter field to display only accounts with matching names.

   The Edit Administrator window opens
3. Edit the settings of the account as needed. Then click OK.
   The window closes and the account appears with your changes in the accounts list.
4. Click Save Changes.

Delete an administrator account

You can delete any administrator account, as long as there is at least one that remains.

Task

1. Select Accounts → Administrator Accounts.
2. Under Internal Administrator Accounts, select an account and click Delete.

   Before selecting an account, you can type a filtering term in the Filter field to display only accounts with matching names.

   A window opens to let you confirm the deletion.

3. Click Save Changes.

Administrator account settings

The administrator account settings are used for configuring credentials and roles for administrators.

Administrator account settings

Settings for administrator accounts

Administrator account settings

Option	Definition
User name	Specifies the user name of an administrator.
Password	Sets an administrator password.
Password repeated	Lets you repeat the password and confirm it. In the Edit Administrator window, you need to select Set a new password before the two password fields become available.
Role	Provides a list for selecting an administrator role. You can use the Add and Edit options to add and edit roles. The added and edited roles appear in the list of administrator roles.
Name	Specifies the real name of the person that an account is set up for. Configuration of this name is optional.

Test with current settings

Settings for testing whether an administrator with given credentials would be admitted on the appliance

Test with current settings

Option	Definition
User	Specifies a user name that is tested.
Password	Specifies the tested password.
Test	Executes the test. The Authentication Test Results window opens to display the outcome of the test.

Manage administrator roles

You can create roles and use them for configuring administrator accounts.

Note: One administrator role is already created by the appliance system at the initial setup.

Task

1. Select Accounts → Administrator Accounts.
2. To add an administrator role:

a. Under Roles, click Add.
   The Add Role window opens.
b. In the Name field, type a role name.
c. Configure access rights for the dashboard, rules, lists, and other items.
d. Click OK.
   The window closes and the new role appears in the list of administrator roles.

3. Use the Edit and Delete options in similar ways to edit and delete roles.
4. Click Save Changes.

Results

The newly added or edited role is now available for being assigned to an administrator account.

Administrator role settings

The administrator role settings are used for configuring roles that can be assigned to administrators of Web Gateway.

Administrator role settings

Rights granted to particular administrator roles

Administrator role settings

Option	Definition
Name	Provides a name for an administrator role.
Dashboard accessible	Allows access to the Dashboard tab on the user interface.
Policy - Rules accessible	Allows access to the rules and rule sets on the Rule Sets tab. If this access is allowed, the following access right can also be granted. • Top-level move and create — Allows the administrator to move top-level rule sets and to create new top-level rule sets.
Policy - Lists accessible	Allows access to the Lists tab. If this access is allowed, the following access rights can also be granted. • List creation — Allows an administrator to create lists. • SSO catalog — Allows an administrator to access the SSO catalog.
Policy - Settings accessible	Allows access to the Settings tab. If this access is allowed, the following access right can also be granted. • Settings creation — Allows an administrator to create settings.
Configuration accessible	Allows an administrator access to the Configuration tab to configure settings of the appliance system. If this access is allowed, the following access right can also be granted. • File editor — Allows an administrator to use the File Editor for editing files of the appliance system.

Option	Definition
	Note: Granting this access right also gives an administrator root privileges when accessing an appliance on the command line of a system console.
Accounts accessible	Allows access to the Accounts tab for administrator accounts.
Troubleshooting accessible	Allows access to the Troubleshooting tab to carry out troubleshooting measures. If this access is allowed, the following access rights can also be granted. • Files — Allows access to files of the appliance system. • Log files — Allows access to log files, which record events and use of appliance functions, for example, web access by users. • Rule tracing — Allows access to the options for rule tracing in order to detect and resolve issues with rule processing. • Feedback — Allows access to feedback files, which record processes that were running before a function failed. • Core files — Allows access to core files, which record memory content at the time when a function failed and caused the appliance to finish operation. • Connection tracing — Allows access to files that record activities occurring on connections between an appliance and other network components. • Packet tracing — Allows access to files that record network activities performed by an appliance. • Network tools — Allows access to network tools, which are used to retrieve status information about network components. These tools include, for example, *ping*, *nslookup*, and *ipneigh*. • System tools — Allows access to system tools, which are used to carry out activities related to the appliance system, for example, performing a restart or displaying the anti-malware filtering threads that are currently running. • Sync to cloud — Allows access to the option that enables rule sets that are implemented on Web Gateway also for cloud use. • Backup/Restore — Allows access to options for creating a backup of an appliance configuration and for restoring it to an appliance. • Reset password — Allows access to options for resetting the root password that is required when accessing an appliance over the command line on a system console. **Note:** Granting this access right also gives an administrator root privileges when accessing an appliance on the command line of a system console.
Permissions accessible	Allows access to the Permissions tabs that are provided when rules, lists, and settings for a web security policy are created. Creating these items includes giving permission for reading or writing access to any of them.
Read-only admin	Allows only reading access to the user interface.
REST Interface accessible	Allows access to the REST Interface.

Configure external account management

You can let administrator accounts be managed on external authentication servers and map externally stored user groups and individual users to roles on an appliance.

Task

1. Select Accounts → Administrator Accounts.
2. Click Administrator accounts are managed in an external directory server.
   Additional settings appear.
3. Under Authentication Server Details, configure settings for the external server.
   These settings determine the way the Authentication module on the appliance retrieves information from that server.
4. Use the settings under Authentication group = role mapping, to map user groups and individual users stored on the external server to roles on the appliance:
   a. Click Add.
      The Add Group/User Role Name Mapping window opens.
   b. Select the checkboxes next to the field for group or user matching as needed and type the name of a group or user in this field.
   c. Click OK.
   d. Under Role to map to, select a role.
   e. Click OK.
      The window closes and the new mapping appears on the mappings list.
   f. Click Save Changes.

   You can use the Edit and Delete options in similar ways to edit and delete mappings.

Monitoring

Several methods are available for monitoring performance on an appliance.

Dashboard

The dashboard on the user interface of the appliance allows you to monitor key events and parameters, such as alerts, filtering activities, status, web usage, and system behavior.

Information is provided on the following two tabs:

- Alerts — Shows status and alerts
- Charts and Tables — Shows web usage, filtering activities, and system behavior

If the appliance is a node in a Central Management configuration, statuses and alerts are also shown for the other nodes.

Access the dashboard

You can access the dashboard on the user interface of an appliance.

Task

1. Select the Dashboard top-level menu.
2. Select one of the following two tabs, according to what you want to view:
    - Alerts — Shows status and alerts
    - Charts and Tables — Shows web usage, filtering activities, and system behavior

Alerts tab

The Alerts tab displays information on the status and alerts for an appliance and, in case the appliance is a node in a Central Management configuration, also of the other appliances.

View status and alerts information

On the alerts tab, you can view information on the status of an appliance and on alerts that occur.

Task

1. Select Dashboard → Alerts.
2. [Optional] Refresh information on alerts using one of the following two options:
    - Automatic refresh — Performs an automatic refresh in regular intervals

      This option is enabled by default.
    - Refresh now — Performs an immediate refresh

Overview of status information

Information about the status of an appliance is displayed under Appliances Status on the Alerts tab of the dashboard.

If an appliance is a node in a Central Management configuration, information on the the other nodes is also displayed.

The following table provides an overview of this information.

Overview of status information

Information	Description
Appliance	Provides basic appliance information. • Name — Specifies the name of an appliance.
Performance	Provides key performance parameters. • Alert peaks, last 7 days — Indicates the most severe alert on an appliance for each of the last seven days. A colored field is displayed for each day (right-most field is today): ◦ Gray — No alert during the day ◦ Green — Most severe alert during the day was an information ◦ Yellow — Most severe alert during the day was a warning ◦ Red — Most severe alert during the day was an error • Requests per second — Provides a diagram showing how number of web requests in HTTP and HTTPS mode received on the appliance evolved over the last 30 minutes The value to the right of the diagram is the average number of requests per second over the last ten minutes.
McAfee Anti-Malware Versions	Provides update and version information on modules used in virus and malware filtering. • Last update — Shows the number of minutes since the modules were last updated. • Gateway Engine — Shows the version number of the McAfee Web Gateway Anti-Malware engine. • Proactive Database — Shows the version number of the Proactive Database. • DATs — Shows the version number of the DAT files (containing virus signatures).
URL Filter	Provides update and version information for the module used in URL filtering. • Last update — Shows the number of days since the module was last updated. • Version — Shows the version number of the module.
Vulnerabilities	Provides information about recently detected CVE vulnerabilities and measures for mitigation.

Charts and Tables tab

The Charts and Tables tab displays statistical data on web usage, filtering activities, and system behavior of an appliance. If the appliance is a node in a Central Management configuration, it displays also statistical data for the other nodes.

View charts and tables information

On the Charts and Tables tab, you can view information on web usage, filtering activities, and system behavior.

Task

1. Select Dashboard → Charts and Tables.
2. From the Appliance drop-down list, select the appliance you want to view chart and tables information for.
3. [Optional] Click Update to ensure you see the most recent information.
4. From the list on the navigation pane, select the type of information you want to view, for example Web Traffic Summary.

Logging

Logging enables you to record web filtering and other processes on an appliance. Reviewing the log files that contain the recordings allows you to find reasons for failures and solve problems.

The following elements are involved in logging:

- Log files that entries recording web filtering and other processes are written into
- System functions that write entries into log files
- Modules that write entries into log files
- Logging rules that write entries into log files
- Log file management modules that rotate, delete, and push log files

Log files

Log files contain entries on web filtering and other processes. Log files with the same kind of content are stored in folders, which are called *logs*. You can view all logs and log files on the user interface of an appliance.

Depending on their content, log files are maintained by functions of the appliance system, modules, or logging rules. Accordingly, you can perform some or all kinds of activities for these log files, such as viewing, editing, rotating, and others.

Logging by system functions

For some content, log file entries are written by functions of the appliance system. You can view these files on the user interface, but not edit or delete them. The files are also rotated in regular intervals by system functions.

Logging by modules

For some content, log file entries are written by particular modules, such as the proxy module or the Anti-Malware module.

You can view these files on the user interface, but not edit or delete them. Rotation and deletion of these files and pushing them to another location is handled by the Log File Manager, which you can configure settings for.

Logging by rules

A logging rule uses events to create a log file entry and write it into a log file if its criteria matches.

Like other rules, logging rules are contained in rule sets. Logging rule sets are nested in top-level rule sets, which are known as *Log Handlers*. A default Log Handler rule set is available after the initial setup of an appliance. This rule set includes the following nested rule sets by default.

- Access Log — Contains a rule that writes entries about access to a Web Gateway appliance into the log
- Access Denied Log — Contains a rule that writes entries about attempts to access a Web Gateway appliance that were denied into the log

- Found Viruses Log — Contains a rule that writes entries about viruses that were found when requests were processed on a Web Gateway appliance into the log

To these default rule sets, you can add rule sets that you import from the rule set library, for example, the Proxy Error Log rule set. These rule sets are located in the Logging rule set group of the library.

Logging rules are processed in a separate logging cycle after the request, response, and embedded object cycles have been completed for a request that is received on an appliance.

Rotation and deletion of these files and pushing them to another location is handled by the File System Logging module, which you can configure settings for.

Log file management modules

There are two modules for performing management activities on log files, including rotation, deletion, and pushing to other locations.

These modules are the Log File Manager for log files that are maintained by modules and the File System Logging module (also known as *engine*) for log files maintained by logging rules.

You can configure settings for these modules to adapt the rotation, deletion, and pushing of log files to the requirements of your network.

Administer logging

You can administer the logging functions of an appliance to monitor how it performs filtering and other activities that ensure web security for your network.

Complete the following high-level steps.

Task

1. View the log files that are maintained on an appliance.
2. Modify the implemented log file system as needed.

   You can, for example, do the following:

   - Enable, disable, or delete logging rules
   - Modify logging rules
   - Add logging rules of your own
   - Configure the settings of the logging modules for:
     - Log file rotation
     - Log file pushing
     - Log file deletion
3. Save your changes.

View log files

You can view log files on the user interface of an appliance.

Task

1. Select the Troubleshooting top-level menu.
2. On the appliances tree, select the appliance you want to view log files for and click Log Files.

   A list of log file folders appears, some of which contain subfolders.
3. Double-click the folder or subfolder with the log files you want to view.

   The folder opens to display its log files.
4. Select the log file you want to view and click View on the toolbar above the list.

Log file types

There are several types of log files on an appliance. They differ in the kind of content that is recorded and in the way the recording is done.

Log files that record the same kind of content are stored in the same folder. A folder for storing log files with the same kind of content is called a *log*.

Depending on their content, log files are maintained by system functions, modules, or logging rules.

System-maintained log files

Some log files are maintained by functions of the appliance system, which includes the operating system and several system-related services.

You can view these files on the user interface, but not edit or delete them. However, when system log files are unreadable, they are not displayed on the user interface.

The files are also rotated in regular intervals by system functions. There are no options for configuring this rotation.

Module-maintained log files

Other log files are maintained by particular modules of the appliance, such as the proxy module or the Anti-Malware module.

You can view these files on the user interface, but not edit or delete them. The files are stored in subfolders that are located on the appliance under:

/opt/mwg/log

Rotation, deletion, and pushing of these files in regular intervals is handled by the Log File Manager, which you can configure settings for.

All files in these folders are handled by the Log File Manager, except those that have *mwgResInfo* as a part of their names.

The folders with the following names are also not handled by the Log File Manager: *cores, feedbacks, tcpdump, migration, system, ruleengine_tracing, connection_tracing, message_tracing*.

Logs for module-maintained log files include the following:

- **Audit log** — Stores log files that record changes to the appliance configuration
- **Debug log** — Stores log files that record debugging information
- **Migration log** — Stores log files that record migration activities
- **MWG errors logs** — Store log files that record errors occurring in appliance components

  There are separate errors logs for the core and coordinator subsystems, the Anti-Malware module, the user interface, and the system configuration daemon.
- **Update log** — Stores log files that record updates of modules and files

Rule-maintained log files

There are also log files that are maintained by logging rules. The recording of data is executed by events that are triggered when these rules apply.

For example, a rule triggers an event when an object that a user requested is infected by a virus. The triggered event writes an entry with information on the user, the infected object, date and time of the request, and other parameters, to the log file.

You can work with the rules for this type of log files in the same way as with any other rules.

Rotation, deletion, and pushing of these files in regular intervals is handled by the File System Logging module, which you can configure settings for.

The following rule-maintained log files are provided on an appliance by default:

- **Access log** — Stores log files that record requests and related information, including date and time, user name, requested object, infection of an object, blocking of an object
- **Found viruses log** — Stores log files that record the names of viruses and other malware that were found to infect requested objects

  The log also records date and time, user name, requested URL, and the IP address of the client a request was sent from.
- **Incident logs** — Store log files that record incidents concerning various functions, such as licensing, monitoring, or updates

To these default logs, you can add logs that you have created yourself.

Configure log file settings

You can configure settings for the log file management modules to modify the way log files are rotated, deleted, and pushed.

The log file management modules handle rotation, deletion, and pushing for module-maintained and rule-maintained log files. Log file management for system-maintained log files cannot be configured.

Configure settings for module-maintained log files

You can configure settings for rotation, deletion, and pushing of module-maintained log files. These activities are handled by the Log File Manager.

The settings for module-maintained log files are system settings that are configured for the Log File Manager.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to configure log file settings for and click Log File Manager.
   The Log File Manager settings appear on the settings pane.
3. Configure these settings as needed.
4. Click Save Changes.

Configure settings for rule-maintained log files

You can configure settings for rotation, deletion, and pushing of rule-maintained log files. These activities are handled by the File System Logging module.

Task

1. Select Policy → Settings.
2. On the settings tree, expand File System Logging and select the log file settings you want to configure, for example, Found Viruses Log.
3. Configure these settings as needed:
4. Click Save Changes.

Create a log

You can create a log that can be used by a logging rule to write entries into its log files.

When you create a log, you do not create it separately, but as a part of creating new settings for the File System Settings module.

Task

1. Select Policy → Settings.
2. Expand File System Logging and select one of the existing settings, for example, Access Log Configuration.
   This will serve as a starting point for creating new settings, including the creation of a new log.
3. Click Add above the settings tree.
   The Add Settings window opens.
4. In the Name field, type a name for the settings.
5. [Optional] In the Comment field, type a plain-text comment on the settings.

6. [Optional] Select the Permission tab and configure who is allowed to access the settings.
7. Under Name of the log, type the name of the new log.
8. Configure other settings items, such as rotation or deletion, as needed.
9. Click OK.

   The Add Settings window closes and the new settings appear under File System Logging on the settings tree.
10. Click Save Changes.

Create a log handler

When you create new logging rules, you can insert them into existing logging rule sets or create new rule sets for them. These must be nested themselves in top-level rule sets known as log handlers.

You can also use the Default log handler for inserting new logging rule sets.

Task

1. Select Policy → Rule Sets.
2. From the Rule Sets menu, select Log Handler.
3. Click Add above the log handler tree, and from the drop-down list that appears, select Log Handler.

   The Add New Log Handler window opens.
4. In the Name field, type a name for the log handler.
5. Make sure Enable is selected.
6. [Optional] In the Comment field, type a plain-text comment on the log handler.
7. [Optional] Click the Permissions tab and configure who is allowed to access the log handler.
8. Click OK to close the Add New Log Handler window.

   The new log handler appears on the log handler tree.
9. Click Save Changes.

Elements of a logging rule

A logging rule handles the writing of log file entries into a particular log. Its elements are of the same types as with other rules.

Name				
Write Found Viruses Log				
Criteria		Action		Events
Antimalware.Infected equals true	–>	Continue	–	Set User-Defined.LogLine =
				+ DateTime.ToWebReporterString
				+ “ ””
				+ Authentication.Username
				+ “ ”

	+ String.ReplaceIf Equals (IP.ToString(Client.IP), “”, “-”)
	+ “” “”
	+ List.OfString.ToString (Antimalware.VirusNames)
	+ “” “”
	+ URL
	+ “”
	FileSystemLogging.WriteLogEntry (User-Defined.logLine)<Found Viruses Log>

The elements of this rule have the following meanings:

- **Criteria** — *Antimalware.Infected equals true*

  The criteria of the rule uses the *Antimalware.Infected* property. It is matched when the value of this property is *true*. This means that the rule applies when a filtered object is infected.
- **Action** — Continue

  When the rule applies, it executes the the Continue action. This action lets processing continue with the next rule after the events of the current rule have also been executed.
- **Events** — When the rule applies, it also executes two events:
  - **Set User-Defined.logLine = ...** — Sets the parameter values that are logged.

    Theses values are as follows:
  - **FileSystemLogging.WriteLogEntry ...** — Executes the write event

    The entry that is to be written and the log file it is written into are specified with the event:
    - (User-Defined.logLine) — Event parameter specifying the entry

      This is a log file line with the parameter values that have been set by the other event of the rule.
    - <Found Viruses Log> — Event settings specifying the log file

Best practices - Adding a log file field

Adding a log file field to the entries that are written in the log files of a log allows you to record additional information about activities that are performed on Web Gateway.

When you add a log file field, you might also want to adapt the log header and configure an entry for the new log file field. This way you ensure that the header, which is written into every log file, also includes information on this field.

Add a log file field

To add a log file field to an entry for a log file, append an appropriate element to the configuration for writing log file entries.

In this sample procedure, the destination IP address of a client request that is received on Web Gateway is added to the rule for writing log file entries into the default access log.

Task

1. Select Policy → Rule Sets.
2. Select Log Handler, expand the Default rule set on the log handler tree, and select Access Log.
3. Add an element for writing log file entries.
   a. Select the Write access log rule and click Edit immediately above.
   b. Select Events, then select the event Set User-Defined.logLine and click Edit.
   c. Under To concatenation of these strings, click Add.
   d. Click Parameter property, select IP.ToString from the properties list, and click Parameters next to the property name.

      Note: To search for the property, you can type a suitable combination of characters in the filter field above the list, for example, `ip.tos`.

      The Parameters For Property window opens.
   e. Click Parameter property and select URL.Destination.IP.
   f. Click OK in the Parameters For Property window, then in the Enter a String window.

      The new element appears in the Edit Set Property window, behind the last of the old elements, as shown here:

      ```
      + Number.ToString(Block.ID) + "" "" + Application.ToString(Application.Name) + """ +
      IP.ToString(URL.Destination.IP)
      ```
4. Insert a delimiter to let the new log file field be separated from the preceding.
   a. Select the line with the three double quotes and click Edit.
   b. Enter a blank next to the double quote that appears in the window, then click OK.

      The Enter a String window closes. In the Edit Set Property window, the line between the two elements should now look like this:

      ```
      + Application.ToString(Application.Name) + "" " + IP.ToString(URL.Destination.IP)
      ```
   c. Click OK in the Enter a string and Edit Set Property windows, then click Finish in the Edit Rule window.
5. Click Finish in the Edit Rule window, then click Save Changes.

Adapt the log header

Adapt the access log header by adding a header entry for the new element that you appended to the elements for log file writing.

Task

1. Select Policy → Settings.
2. On the settings tree, expand File System Logging and select the Access Log Configuration settings.
3. Under File System Logging Settings, make sure Enable header writing is selected, and at the end of the text string in the Log header field leave a blank after the last element and type `server_ip`.

   Note: Header field names, such as `server_ip`, must not include blanks inside them, so always use underscores.
4. Click Save Changes.

Best practices - Creating a log

Creating a log allows you to log particular activities that are performed on Web Gateway.

For example, you want to log all requests that were sent from a particular client and were either invalid or blocked.

The logging is performed by a logging rule. This rule applies when a request of the kind that you want to log is received on Web Gateway. It does the following:

- Record information about the request in a log file entry (also known as *log line*)
  This entry can include the time when a request was sent, the name of the user who sent it, and other information.
- Write the entry into a log file
  Log files are stored in a log. To avoid excessive memory consumption, log files can be rotated and deleted after some time.

Like any other rule on Web Gateway, the logging rule must be contained in a rule set, which is termed a log rule set.

So creating a log that logs what you want to record includes the following activities:

- Creating a log rule set
- Creating a logging rule
  To create the logging rule you need to configure:
    - The rule criteria
    - An event that builds log file entries
    - An event that writes log file entries into a log file

When the write event is configured, the log that stores the log files is also specified.

Configuring a log file entry

When creating a logging rule, you need to configure an event within this rule that builds the log file entries.

A log file entry provides information about an activity that is performed on Web Gateway, for example, receiving and filtering a request.

Properties are used to store this information, for example, the Authentication.Username property is used to store the name of a user who sent a request.

To configure an event that builds log file entries, you need to specify all the properties you want to use for storing information. The event combines the values of these properties to the value of a single property, which is named *User-Defined.logLine*.

The value of the User-Defined-logLine property is then written by a write event into a log file.

When specifying the properties for the event that builds the log file entries, you need to consider some requirements.

String format

A log file entry is a chain of data in string format. This means that the properties that are used for building an entry must have this format. Otherwise they need to be converted.

For example, to log a client IP address, the Client.IP property is used. The values of this property have a special IP address format, which can be converted using the *IP.ToString* property.

IP.ToString is specified as an element of the log file entry with Client.IP (in parentheses) as its parameter: *IP.ToString(Client.iP)*. When this term is processed, it delivers a value that is an IP address in string format.

Empty elements

When a logging rule is processed, not all properties might be filled with values for a particular request. Accordingly, a log file entry can contain empty elements.

For example, the *Authentication.Username* property has no value if no authentication was performed for a user who sent a request. You can insert a placeholder, such as a dash, in this case.

The *String.ReplaceIfEqual* property is available for this purpose. It takes three parameters:

- The value that is actually recorded for a property
- A value to compare value 1 with
- A value to replace value 2 if 1 matches 2

For example, Authentication.Username, a blank, and a dash as parameters of the String.ReplaceIfEqual property, result in a dash for the user-name element of a log file entry if no user name is recorded.

Delimiters

To improve the readability of log file entries, configure delimiters between the elements of a log file entry. Delimiters can be blanks, quotes, and other characters.

Delimiters are specified in the same way as the main elements of an entry and are also interpreted as strings.

Create a log rule set with a logging rule

To create a logging rule that lets entries be written into log files and stored in a log, create a log rule set for it and configure rule criteria and events.

The purpose of the sample rule described here is to log all requests that are received from a client with a particular ID if the following also applies:

- A request is not valid under the HTTP protocol (response 403) *or*
- A request is blocked by the web security rules on Web Gateway (block ID is not 0)

Task

1. Select Policy → Rule Sets.
2. Select Log Handler and on the log handler tree, expand the Default log handler rule set.
3. Create a rule set for the logging rule.

   This rule set will be nested in the Default rule set.

   a. Click Add, then Rule Set.

   b. In the Name field, type a suitable name for the new log rule set, for example, `Troubleshooting Log`.

   c. Click OK.

      The window closes and the new log rule set appears on the log handler tree.
4. Add a logging rule and name it.

   a. On the settings pane, click Add Rule.

   b. In the Name field, type a suitable name for the logging rule, for example, `Log requests that caused issues`.
5. Configure the rule criteria.

   a. Select Rule Criteria. then click Add and select Advanced Criteria.

   b. Select Client.IP as the property, equals as the operator, and in the field below Compare with, type an IP address as the operand, for example, `10.149.33.8`.

   c. Click OK.

      The Add Criteria window closes and the configured rule criteria appears in the Add Rule window.

   d. Click Add again for the second part of the criteria.

   e. In the same way as for the first part, configure Response.StatusCode, equals, and `403`, then click OK.

   f. In the same way configure Block.ID, does not equal, and `0`, then click OK.
6. Configure the event that builds the log file entry.

   a. Select Events and click Add, then select Set Property Value.

   b. From the properties list, select User-Defined.logLine and below To concatenation of these strings click Add.

   c. Click Parameter property and from the properties list, select DateTime.ToWebReporterString, as the first element of the log file entry. Then click OK.

      The Enter a String window closes and the configured element appears in the Add Set Property window.

   d. Click Add, make sure Parameter value is selected, and type a blank followed by a double quote. Then click OK.

      The Enter a String window closes and the configured delimiters appear in the Add Set Property window.

   e. Add the next element of the log file entry.

      - Click Add, then Parameter property.
      - Select String.ReplaceIfEquals and click Parameters next to the property name.

        The Parameters For Property window opens.
      - For the first parameter, click Parameter property, and select Authentication.Username.
      -

Select the second parameter and do not enter anything in the input field on the right.

This creates a blank as the parameter value.

- Select the third parameter and type a dash in the input field.
- Click OK in the Parameters For Property, then in the Enter a String window.

  The element appears in the Add Set Property window.

f. Click Add and type a double quote, a blank, and a double quote. Then click OK.

The Enter a String window closes and the configured delimiters appear in the Add Set Property window.

g. Add the remaining elements by selecting the properties listed in the following. Insert delimiters between the elements, as shown in substep f.

- Client.IP

  For this element, configure the IP.ToString property with the Client.IP property as its parameter.

  Add the parameter as shown in substep e.
- Request.Header.First.Line
- Header.Request.Get

  Configure a parameter for this element by typing `user-agent` as the parameter value.
- Rules.CurrentRuleSet.Name
- Rules.CurrentRule.Name
- Block.ID

  For this element, configure the Number.ToString property with the Block.ID property as its parameter.
- Block.Reason

  After this last element, only use one double quote as the delimiter.

h. Click OK to close the Add Set Property window.

The event that creates the log file entry appears in the Add Rule window.

It should look like this:

```
Set User-Defined.logLine = DateTime.ToWebReporterString + " "" + String.ReplaceIfEquals
(Authentication.UserName, "", "-") + "" "" + IP.ToString(Client.IP) + "" "" + Request.Header.First.Line +
"" "" + Header.Request.Get("user-agent") + "" "" + Rules.CurrentRuleSet.Name + "" "" +
Rules.CurrentRule.Name + "" "" + Number.ToString(Block.ID) + "" "" + Block.Reason + """
```

If you need to make changes, select the event, click Edit, and work with the Edit Set Property window.

Note:

In the representation of the event, the + symbols are added by the program.

Double quotes before and after a string value are also added. So, for example, the " "" for the first delimiter represents a blank and a double quote.

7. Configure the event that writes the log file entries into a log file.

a. Under Events in the Add Rule window (or Edit Rule window if you made changes), click Add and select Event.

b. From the events list, select FileSystemLogging.WriteLogEntry and click Parameters.

c. Click Parameter property and from the list below, select User-Defined.logLine, then click OK.

The Parameters for Execute Action window closes.

d. In the Add Event window, click Add below the Settings field.

e. Create new settings.

These settings will be used by the File System Logging module (or *engine*) to handle the new log you are creating.

- In the Name field, type a suitable name for the new settings, for example, `Troubleshooting Log Settings`.
- Under Name of the log, type `troubleshooting.log`.
- Select Enable header writing and in the Log header field, type the elements of the log header.

The log header will appear at the beginning of every log file in the new log. It should represent the elements of the log file entry that you configured in step 6.

So the log header might look like this:

```
time_stamp "auth_user" "src_ip" "req_line" "user_agent" "rule_set" "rule" "block_page_res" "block_res"
```

- Under Settings for Rotation, Pushing, and Deletion configure the following.
  - Select Enable specific settings for user-defined log.
  - Under Auto-Rotation, select GZIP log files after rotation to save memory space.
  - Configure the remaining settings under Auto-Rotation and those under Auto-Deletion as needed.

    Note: Do not enable and configure the Auto-Pushing settings, as the log files for this new log are not pushed anywhere.

f. Click OK in the Add Settings window, then in the Add Event window.

The windows close and the event that writes the log file entries appears in the Add Rule (or Edit Rule) window.

g. Click Finish.

The window closes and the new logging rule appears as a rule of the Troubleshooting Log rule set on the Rule Sets tab.

Results

You have now created a logging rule for recording a particular kind of requests.

The log files can be viewed in the log with the configured name. The log is accessible from the Troubleshooting top-level menu on the appliance you have configured the rule on.

The path to the log is Log files → user-defined logs.

Error handling

When errors and incidents occur on an appliance, appropriate measures can be taken. Some of these measures are controlled by rules.

Error handling using error IDs

Errors that occur on an appliance are identified by error IDs. These can be used by rules to trigger particular methods of error handling.

To enable the use of error IDs in rules, the *Error.ID* property is available. A rule can trigger an action or event when this property has a particular value, for example, 14000, which indicates a failure to load the Anti-Malware module.

The action or event that is triggered uses a particular method of error handling, such as blocking access to a web object or creating an entry in a log file.

A rule that uses an error ID to trigger an error handling measure could, for example, look at follows

Name		
Block if Anti-Malware engine cannot be loaded		
Criteria		Action
Error.ID equals 14000	–>	Block<Cannot Load Anti-Malware Engine>

Error handling using incident information

There is a group of activities and situations on an appliance that is termed *incidents*. Incident information can be used by rules to trigger particular methods of error handling.

Incidents can be related to the appliance system, as well as to its subsystems and modules. For example, a failure of the Log File Manager to push log files is recorded as an incident.

Incidents can be used by rules to trigger a particular method of error handling, such as sending a notification message or creating an entry in the system log. To enable the use of incidents in rules, key incident parameters, including the ID, severity, origin, and others, are made available as properties.

For example, there is the *Incident.ID* property. A rule can use this property to trigger an event that creates a syslog entry if the value of the property is a particular number.

Rules using incidents

The Default rule set for error handling contains a nested rule set providing rules that trigger a notification message and other error handling events when incidents concerning the Log File Manager occur. The name of this nested rule set is Log File Manager Incidents. Other nested rule sets handle incidents related to updates and licensing.

You can also create rules and rules sets of your own that use incidents for error handling.

Incident parameters and properties

Incidents are recorded on an appliance with their IDs and other parameters. For each parameter, there is a property, which can be used in an appropriate rule.

- **Incident ID** — Each incident is identified by a number. For example, the incident with ID 501 is a failure of the Log File Manager to push log files. The *Incident.ID* property can be used in a rule to check the ID of an incident.
- **Description** — An incident can be explained by a description in plain text. The name of the relevant property is *Incident.Description*.
- **Origin** — Each incident is assigned to the appliance component that is its origin. Origins are specified by numbers. For example, origin number 5 specifies the Log File Handler. The name of the relevant property is *Incident.Origin.*
- **OriginName** — The origin of an incident is further specified by the name of the appliance component that is involved in the incident. The name of the relevant property is *Incident.OriginName*.

  The origin name can specify a subcomponent that is a part of the component specified by the origin number. For example, origin number 2 (Core) can be further specified by the origin name as:

    - Core
    - Proxy
    - URL Filter
    - and other names of core subcomponents
- **Severity** — Each incident is classified according to its severity. Severity levels range from 0 to 7, with 0 indicating the highest level.

  These levels are the same as those used for entries in a syslog file.

  The name of the relevant property is *Incident.Severity*.
- **Affected host** — If there is an external system that is involved in an incident, for example, a server that the appliance cannot connect to, the IP address of this system is also recorded. The name of the relevant property is *Incident.AffectedHost*.

Configure error handling

You can configure error handling to adapt this process to the requirements of your network.

Complete the following high-level steps.

Task

1. Review the rules in the nested rule sets of the default rule set for error handling.
   The name of this rule set is *Default*.

2. Modify these rules as needed.
   You can, for example, do the following:
   - Enable rules that take additional measures for error handling when a particular error or incident has occurred.
   - Create new rules and rule sets for handling additional errors and incidents.
3. Save your changes.

View the error handling rule sets

You can view the rule sets that are implemented for error handling on a rule set tree that is provided on the user interface in addition to the normal rule set tree for web filtering rules.

Task

1. Select Policy → Rule Sets.
2. Below the rule sets tree, select Error Handler.
3. Expand the Default top-level rule set.
   The nested rule sets for error handling appear.
4. Select a nested rule set.
   The rules of the nested rule set appear on the settings pane.

Best practices - Working with the Error Handler

Working with the rules in the Error Handler rule sets gives you control over what happens when errors occur with processing web traffic on Web Gateway.

There are two main strategies of responding to errors:

- **Fail-closed** — When an error occurs, the request that a user sent to the web and that is being processed on Web Gateway is not allowed to proceed. A block message is shown to the user.
  This strategy is the default for error handling on Web Gateway.
- **Fail-open** — When an error occurs, the request that a user sent to the web and that is being processed on Web Gateway is allowed to proceed.
  In addition to this, logging activities and notifications can be triggered.
  This strategy is widely used within the web security policies of enterprise organizations.

The following are benefits of adopting a fail-open strategy for your network:

- Prevents business interruptions, as unimpeded web access is one of the most critical aspects for many jobs today.
- Avoids unnecessary calls to help desks, as you might consider it sufficient if the Web Gateway administrator is aware and can fix the problem. There is no need then to alert users.

A fail-open strategy can also be appropriate if failed components are compensated within your network while internal alerts are triggered and action is taken.

The flexibility of the Error Handler allows you to create rules to implement the main strategies in various ways, for example, as follows:

- Strict fail-closed strategy on all errors
- Broad fail-open strategy to prevent any user impact
- Notifications to the Web Gateway administrator as part of a fail-closed or fail-open strategy
- Exceptions for requests from particular users and clients
  For example, a fail-open strategy is configured for executives and a fail-closed strategy for other users.

The default rule set for error handling includes the *Block on All Errors* rule set. This nested rule set is placed at the end of the default rule set. It blocks requests in all error situations that are not covered by the other nested rule sets.

When you configure a fail-open rule, make sure that this rule set is disabled or the rule set with the fail-open rule is placed before it.

Configure a general fail-open strategy

Configure a general fail-open strategy to let processing continue after any processing error that occurs.

Task

1. Select Policy → Rule Sets.
2. Select Error Handler and expand the Default error handling rule set.
3. For all rules in the nested rule sets:
   a. Select a rule and click Edit for this rule.
   b. In the Edit Rule window, select Action, then select Continue as the rule action.
   c. Click Finish.
4. Click Save Changes.

Results

Processing of requests that users send to the web now continues on Web Gateway even when errors occur.

Configure a fail-open strategy with a notification

Configure a fail-open strategy with a notification to notify the administrator or another recipient when a particular error has occurred.

For the notification, you add an event to a rule that handles a particular error.

Task

1. Locate an existing rule:
   a. Select Policy → Rule Sets.
   b. Select Error Handler and expand the Default error handling rule set.
   c. Select a nested rule set, for example, Block on Anti-Malware Engine Errors. Then select one of its rules, for example, Block if anti-malware engine is overloaded, and click Edit for this rule.
2. In the Edit Rule window, select Action, then select Stop Rule Set as the rule action instead of Block.
3. Configure an event for notifying someone:
   a. Select Events and click Add.
   b. Select Event, then select Email.Send and click Parameters
   c. Type values for the three string parameters, for example, as follows:
      - Recipient (an email address): `anyrecipient@samplecompany.com`

        To configure more recipients, add their email addresses, separated by semicolons.
      - Subject (message name): `Anti-Malware Overload`
      - Body (message text): `The anti-malware engines are overloaded, please inspect the mwg-antimalware-errors-log for more information.`

   d. Click OK twice, then click Finish.
4. Click Save Changes.

Results

When the error that is handled by this rule occurs, a notification is sent to the configured recipient. You can also configure multiple notification events for different recipients with varying message texts.

Note:

Make sure that the *Block on All Errors* set is disabled or the rule set with the fail-open rule is placed before it.

Configure a fail-open strategy for user groups

Configure a fail-open strategy with a notification that is only sent for errors with processing requests from users belonging to a particular user group.

Task

1. Locate the rule that you configured a fail-open strategy with a notification for:
   a. Select Policy → Rule Sets.
   b. Select Error Handler and expand the Default error handling rule set.
   c. Select the Block on Anti-Malware Engine Errors nested rule set, then select the Block if anti-malware engine is overloaded rule and click Edit for this rule.
2. Configure an additional part for the rule criteria:
   a. In the Edit Rule window, select Rule Criteria, then select the criteria of the rule and click Add.
   b. Select User/Group criteria, then select:
      - Authentication.UserGroups as the property
      - at least one in list as the operator

   c. At the bottom of the right column, click Add List of String to add a list of user groups, and in the Add List window:
      - Name the list `Groups to bypass on anti-malware overloads`, then click OK.
      - Click Edit List and under List content, add the following string to the list (without quotes): `Executives`, then click OK twice.

   d. In the Edit Rule window, select AND as the Boolean operator for this additional criteria part, then click Finish.
3. Click Save Changes.

Results

When the error that is handled by this rule occurs, processing continues and a notification is sent to the configured recipient. It is only sent, however, if a user from the configured user group submitted the request that was processed when the error occurred.

Note:

Make sure that the *Block on All Errors* set is disabled or the rule set with the fail-open rule is placed before it.

Performance measurement

Processing time for several appliance functions is measured and shown as performance information on the dashboard. You can record this information in log files and also measure and record processing time for individual rule sets.

Performance is measured on an appliance, for example, with regard to the average time it takes to resolve host names by looking up names on a DNS server. You can view this and other performance information on the dashboard. Additionally, you can measure the time needed for processing individual rule sets.

You can also log all performance information, the one shown on the dashboard and the one you have measured yourself.

The following elements are involved when you measure and log performance information:

- Properties for logging performance information
- Logging rules that use these properties to log performance information
- Events that measure processing time for individual rule sets
- Rule sets that include rules with events to have their processing time measured

Logging properties

Several properties are available that correspond to performance information shown on the dashboard and can be used in logging rules.

For example, the property*Timer.ResolveHostNameViaDNS* corresponds to the dashboard information on the average time for looking up host name names on a DNS server.

Two properties are available for logging the time that has been measured for processing an individual rule set. The *Stopwatch.GetMilliSeconds* property records this time in milliseconds, the *Stopwatch.GetMicroSeconds* records it in microseconds.

Logging rules

The default logging rules on an appliance use one event to create log lines and another to write these lines into a log file.

If you add properties for logging performance information to the elements of the log lines, they are written into the log file together with the other elements of the log line.

You can use default rules for logging performance information or create rules of your own.

Events for measuring processing time

Two events are available for measuring the time consumed for processing individual rule sets. The *Stopwatch.Start* event starts the internal stopwatch that measures this time. The *Stopwatch.Stop* event stops the watch, so the time that has elapsed can be recorded.

Measured rule sets

To measure the time consumed for processing a particular rule set, you need to create a rule with the event for starting the internal stopwatch at the beginning of the rule set and another at the end with the stopping event.

You need to insert the stopping event also into existing rules of the rule set if they have actions that stop processing of the rule set. Otherwise, the watch would not be stopped because the rule with the stopping event at the end of the rule set is skipped.

View performance information

You can view performance information about several appliance functions on the dashboard.

Task

1. Select Dashboard → Charts and Tables.
2. Select Performance Information.
   Performance information appears on the tab.

Configure performance measurement

You can configure performance measurement to measure and log the performance of functions on an appliance.

Complete the following high-level steps.

Task

1. View the performance information shown on the dashboard and decide what kind of information you want to record in a log file.
   For example, you might want to record the average time consumed for looking up host names on a DNS server. This information is shown by the *DNS Lookup* feature of the dashboard.
2. Use the properties that are available for logging performance information in existing logging rules or new logging rules that you create.
   For example, insert the *Timer.ResolveHostNameviaDNS* property into the event that creates a log line in the *Write access.log* rule of the default *Access Log* rule set.
3. Measure the time consumed for processing particular rule sets.
   a. Insert a rule with an event that starts the internal stopwatch at the beginning of a rule set.
   b. To stop the watch and measure the time consumed:
      - Insert a rule with an event that performs these activities at the end of the rule set.

    - Insert an event that performs these activities into each of the existing rules that is capable of stopping the rule set before all its rules are processed.

   For example, to measure the processing time consumed by a URL filtering rule set:

   - insert a rule with the *Stopwatch.Start (URL Filtering)* event at the beginning of the rule set.
   - Insert a rule with the *Stopwatch.Stop (URL Filtering)* event at the end.
   - Insert the *Stopwatch.Stop (URL Filtering)* event into each of the whitelisting and blocking rules of the rule set because they all can stop the processing of further rules.

4. Use the properties that are available for logging the measured processing time in existing logging rules or new logging rules that you create.
   For example, insert the *Stopwatch.GetMilliSeconds (URL Filtering)* property into the event that creates a log line in the *Write access.log* rule of the default *Access Log* rule set.

Using properties in rules to log performance information

You can insert performance logging properties into logging rules to let performance information be logged.

For each type of performance information that is shown on the dashboard, a logging property is available.

For example, the dashboard shows the average time it takes to resolve host names by looking up names on a DNS server. The property *Timer.ResolveHostNameViaDNS* corresponds to this information. The value of the property is the time consumed for looking up a host name in a request that was processed on an appliance. The time is measured in milliseconds.

Other performance logging properties are*Timer.HandleConnect ToServer* for measuring the time needed to connect to external servers or *Timer.TimeConsumedByRule Engine* or the time the rule engine consumes for processing when a request is received on an appliance.

All properties that make dashboard performance information available for logging have the element *Timer* at the beginning of their names.

Measuring processing time for a transaction

The time that is measured and made available by a property for logging performance information shown on the dashboard is the time needed for a particular activity, for example, connecting to external servers, as long as processing for an individual request is continued throughout the relevant processing cycles.

Processing one individual request throughout the relevant cycles is considered one *transaction*.

It is not required for a transaction to include all three cycles (request, response, and embedded objects).

For example, if a user sends a request for a web page that falls into a blocked category, a block message is returned to this user, the request is not forwarded to the web server in question, and processing does not enter the response cycle.

Then the transaction includes only the request cycle, the response cycle is not relevant in this case.

Rule for logging performance information

An Access Log exists by default on an appliance with log files into which a log entry is written whenever a transaction has been completed for a request. This log is an appropriate device for recording performance information.

Writing log entries into the log files of the Access Log is performed by a logging rule. This rule uses one event to create a log file entry and another to write this entry into a log file.

Name		
Write access.log		
Criteria	Action	Events

Always	–>	Continue	–	Set User-Defined.logLine = DateTime.ToWebReporterString
				+ “””
				+ ...
				FileSystemLogging.WriteLogEntry (User-Defined.logLine)<Access Log Configuration>

A log entry is composed of several elements, each of which adds a particular piece of information, for example, the date and time when a request was received on the appliance. By adding an element providing performance information to the entry you can let this information be logged.

To log performance information, for example, on the processing time consumed by DNS lookups, you need to add the following two elements:

- `+ Number.ToString (Timer.ResolveHostNameViaDNS)`
- + “””

Since the log entry is a string, the numerical value for the processing time must be converted to string format before it can be logged.

This is done by the *Number.ToString* property, which takes the *Timer.ResolveHostNameViaDNS* property as a parameter.

Using events in rules to measure rule set processing time

You can measure the time it takes to process an individual rule set by inserting rules with measuring events into it.

The reason for measuring processing time could be that you want to know whether performance is improved or reduced after you have applied changes to a rule set.

The events for measuring rule set processing time control an internal stopwatch on an appliance. The following events are available:

- Stopwatch.Start — Starts the internal stopwatch
- Stopwatch.Stop — Stops the watch
- Stopwatch.Reset — Resets the watch

Each of these events takes a string parameter to indicate which rule set it measures. For example, an event that starts the internal watch to measure the processing time of the URL Filtering rule set, would appear in a rule as follows: *Stopwatch.Start ("URLFiltering")*.

Rules for measuring processing time

A rule that uses, for example, the *Stopwatch.Start* event to start measuring processing time for the URL Filtering rule set could look as follows:

Name				
Start stopwatch for rule set				
Criteria		Action		Event
Always	–>	Continue	–	Stopwatch.Start (“URLFiltering”)

To measure the time consumed for processing the rule set, you need to place a rule with the starting event at the beginning of the rule set and another one that contains the stopping event at the end.

However, if you have rules in a rule set that can execute a Stop Rule Set, Stop Cycle, or Block action, you also need to insert a stopping event into each of these rules.

A URL filtering rule with an event to stop the internal watch inserted would look as follows:

Name				
Allow URLs in URL Whitelist				
Criteria		Action		Event
URL matches in URL Whitelist	–>	Continue	–	Stopwatch.Stop ("URLFiltering")

When this rule is applied, it stops processing the URL Filtering rule set because the URL that a user requested access for has been found to be on the list of allowed URLs.The stopping event must therefore be inserted into this rule.

This is required because the rule with the stopping event at the end of the rule set is then not processed as the whitelisting rule stops processing of the rule set before this rule is reached.

Logging measured processing time

You can log the time that has been measured for rule set processing. Two properties are available for this purpose, which you can use in logging rules.

- Stopwatch.GetMilliSeconds — Time measured for rule set processing in milliseconds
- Stopwatch.GetMicroSeconds — Time measured for rule set processing in microseconds

Both properties have a string parameter, which indicates the rule set that processing time was measured for.

For example, a property for logging the processing time of the URL Filtering rule set in milliseconds would appear in a logging rule as follows: *Stopwatch.GetMilliSeconds ("URL Filtering")*.

Event monitoring with SNMP

Events that occur on the appliance system can be monitored using SNMP.

When monitoring is performed under SNMP (Simple Network Management Protocol), an SNMP agent that runs on a host system sends messages about events that occur on this system to other host systems that are its clients.

The messages are known as *traps* under SNMP, while the host system that the SNMP agent runs on is known as *management station*. The host systems that receive messages from the agent are also management stations, in addition to this, they are known as *trap sinks*.

Particular users or user communities are given permission to view the information sent with the traps. System information is also provided in the Management Information Base (MIB), which uses a tree structure to present the information.

Configure event monitoring with SNMP

To enable the use of SNMP for monitoring system events on an appliance, configure an SNMP protocol version, a user or community that is allowed to view monitored information, and other settings.

Task

1. Select Configuration → Appliances.

2. On the appliances tree, select the appliance where you want to configure SNMP event monitoring, then click SNMP.
3. Under SNMP Port Settings and SNMP System information add and modify information about listener ports and management stations as needed ,
4. To work with SNMPv1 or SNMPv2c, complete these options. Otherwise, continue with step 4.
   a. Under SNMP Protocol Options, make sure the respective version is selected.
   b. Above the list of communities that are allowed to view monitored information, click the Add icon, then create an entry for a community in the window that opens.
      - Under Community string, type the name of a community, for example, `public`.
      - Under Allowed root OID, type a root Object ID to identify the item on the MIB (Management Information Base) tree where the information begins that is allowed for viewing.

        For example, type this root Object ID to allow all information that is related to McAfee for viewing:

        ```
        .1.3.6.1.4.1.1230
        ```

        Information related to Web Gateway is a part of this information. So, type the following to allow only this information for viewing:

        ```
        .1.3.6.1.4.1.1230.2
        ```

        If you type an * (asterisk) here, all information is allowed for viewing.
      - Under Allowed from, specify the host system where viewing the information is allowed.

        If you specify no host system here, viewing is allowed from any system.
   c. Under SNMP Trap Sinks,.click the Add icon above the list and configure trap sinks as needed.
5. To work with SNMPv3, complete these options..
   a. Under SNMP Protocol Options, make sure this version is selected.
   b. Above the list of users who are allowed to view monitored information, click the Add icon, then create an entry for a user in the window that opens.
      - Under User name, type the name of a user.
      - Next to Password, click Set, then set a password in the window that opens.
      - Under Allowed root OID, type a root Object ID to identify the item on the MIB (Management Information Base) tree where the information that is allowed for viewing begins.

        If you type an * (asterisk) here, all information is allowed for viewing.
      - Under Authentication, select a method for calculating a hash value that is used to verify and control authentication data.

        Available methods: MD5, SHA-1, SHA-224, SHA-256, SHA-384, SHA-512

        Some authentication methods cannot be used together with some encryption methods that are used with regard to SNMP traffic. You cannot configure the following:
        - MD5 or SHA-1 with encryption method AES-192 or AES-256
        - SHA-224 with encryption method AES-256
      - If you want to have the SNMP traffic encrypted, select a method for this encryption under Encryption.

        Available methods: DES, AES, AES-128, AES-192, AES-256, or None (no encryption)

        Some encryption methods cannot be used together with some authentication methods that are also used for creating SNMP user information. See above for combinations that will not work.
   c. Under SNMP Trap Sinks,.click the Add icon above the list for SNMPv3 and configure trap sinks as needed.
6. Click Save Changes.

SNMP settings

The SNMP settings are settings for configuring the monitoring of system events under SNMP.

SNMP Port Settings

Settings for the ports of the SNMP agent on an appliance that listen to client requests

SNMP Port Settings

Option	Definition
Listener address list	Provides a list for entering the ports that listen to client requests.

The following table describes an entry in the listener address list.

Listener address – List entry

Option	Definition
Protocol	Specifies the protocol used for the communication between a port and the clients that it listens to. • UDP — When selected, UDP is used for this communication • TCP — When selected, TCP is used for this communication
Listener address	Specifies the IP address and port number of a listener port.
Comment	Provides a plain-text comment on a listener port.

The following two listener ports are available on an appliance and entered in this list by default.

- UDP — 0.0.0.0:161
- UDP — 0.0.0.0:9161

SNMP System Information

Settings for the appliance that is the monitored system

SNMP System Information

Option	Definition
Description	Identifies the monitored system.
Object ID	Specifies the ID of the object in the Management Information Base (MIB) where information on the monitored system begins. For example: .1.3.6.1.4.1.1230.2.7.1.1
Contact person	Specifies the name of the person who administers the SNMP functions of the monitored system.
Physical location	Specifies the location of the monitored system.

SNMP Protocol Options

Settings for SNMP protocol versions and user access to SNMP information

SNMP Protocol Options

Option	Definition
SNMP v1	When selected, system events are monitored under version 1 of SNMP

Option	Definition
SNMP v2c	When selected, system events are monitored under version 2c of SNMP.
Communities for SNMPv1 and SNMPv2c access	Provides a list for entering the user communities who are allowed access to SNMP information under versions 1 and 2c of SNMP.
SNMP v3c	When selected, system events are monitored under version 3 of SNMP.
SNMP v3 users	Provides a list for entering the users who are allowed access to SNMP information under version 3 of SNMP
SNMP v3 info	Provides information related to version 3 of SNMP • SNMPD Engine ID — ID of the host system for the SNMP agent This ID is also contained in a configuration file. The path to this file is /var/lib/net-snmp/snmpd.conf.

The following tables describe the entries in the list of user communities and the list of SNMP v3 users.

User communities – List entry

Option	Definition
Community string	Provides a string used for authenticating a user community to let it access SNMP information, for example, *public*.
Allowed root OID	Identifies the item on the MIB tree that is the beginning of the information with allowed access. If * or no value is specified here, access to all information is allowed.

Allowed from	Specifies the host name or IP address of a host system that access to SNMP information is allowed from.
	A range of IP addresses in an IP subnet can also be specified here to allow access from them.
	To specify this range, you must specify the IP address of the subnet, which is also known as the network prefix, and its bit-length, separated by a slash:
	`<network prefix/bit-length>`
	Example: `192.168.1.184/29`
	The IP address or prefix of the subnet is the IP address immediately preceding the first IP address that serves to identify a host system within the subnet.
	For example, if you have a subnet with the following IP addresses:
	`192.168.1.185`
	`192.168.1.186`
	`192.168.1.187`
	then `192.168.1.184` is the IP address or prefix of this subnet.

Read-only access	When selected, only reading access to SNMP information is allowed.
Comment	Provides a plain-text comment on a user community.

SNMP v3 users – List entry

Option	Definition
User name	Specifies the name of a user who is allowed access to SNMP information.
Allowed root OID	Identifies the item on the MIB tree that is the beginning of the information with allowed access. If * or no value is specified here, access to all information is allowed.
Authentication	Sets the authentication method used when SNMP information is accessed by a user. Available methods: MD5, SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 Some authentication methods cannot be used together with some encryption methods that are also used for creating SNMP user information. You cannot configure the following: • MD5 or SHA-1 with encryption method AES-192 or AES-256 • SHA-224 with encryption method AES-256
Encryption	Sets the encryption method used to protect SNMP traffic or leaves this traffic unencrypted Available methods: DES, AES, AES-128, AES-192, AES-256, or None (no encryption) Some encryption methods cannot be used together with some authentication methods that are also used for creating SNMP user information. See above for combinations that will not work.
Read-only access	When selected, only reading access to SNMP information is allowed.
Comment	Provides a plain-text comment on a user.

SNMP Trap Sinks

Settings for the host systems that receive SNMP messages

SNMP Trap Sinks

Option	Definition
Trap sinks	Provides a list for entering the host systems, known as *trap sinks*, that receive messages about system events from the SNMP agent on an appliance.

The following table describes an entry in the list of trap sinks for SNMPv1 and v2.

Trap sinks for SNMPv1 and v2 – List entry

Option	Definition
Host name or IP address	Specifies the host name or IP address of a host system that receives SNMP messages, which are known as *traps*.

Option	Definition
Port	Specifies the port on a host system that listens to SNMP messages.
Community string	Specifies the string used for authenticating a user community to let it access SNMP information, for example, *public*.
Send SNMP v2c traps	When selected, messages can be sent under version v2c of the SNMP protocol.
Comment	Provides a plain-text comment on a host system that receives SNMP messages.

The following table describes an entry in the list of trap sinks for SNMPv3.

Trap sinks for SNMPv3 – List entry

Option	Definition
Host name or IP address	Specifies the host name or IP address of a host system that receives SNMP messages, which are known as *traps*.
Port	Specifies the port on a host system that listens to SNMP messages.
Send INFORM	When selected, an SNMPv3 INFORM message is sent instead of a trap.
Identifying user	Specifies a user name for authenticating a user to be allowed access to SNMP information..
Comment	Provides a plain-text comment on a host system that receives SNMP messages.

SNMP MIB Files

Files in txt format providing additional information about SNMP monitoring on an appliance

SNMP MIB Files

Option	Definition
MCAFEE-SMI.txt	Provides Structure of Management Information (SMI) on McAfee, including contact information for the McAfee customer service.
MCAFEE-MWG-MIB.txt	Provides descriptions of the items in the Management Information Base (MIB) that you can do SNMP monitoring for on an appliance

Transferring data for McAfee ePO monitoring

Transferring data from an appliance to the McAfee ePolicy Orchestrator® (McAfee ePO™) console allows you to monitor the appliance from the console.

The McAfee ePolicy Orchestrator console is a device for performing security management on different McAfee products, including the McAfee Web Gateway appliance.

If you configure the McAfee ePO console and an appliance accordingly, you can log on to the appliance from the console and have monitoring data transferred from the appliance to the server that the console is running on. This server is also referred to as the McAfee ePO server.

The McAfee ePO server sends SSL-secured requests to retrieve the monitoring data that has been collected on the appliance in regular intervals. Then you need to allow the CONNECT request that the SSL-secured communication begins with to bypass the normal processing of web security rules, so it does not get blocked on the appliance.

For example, if you have authentication rules implemented, this would lead to blocking because the server does not support the authentication method used by these rules.

You can import an appropriate rule set from the library to enable the bypassing or create a rule set of your own.

Configure the ePolicy Orchestrator settings

You can configure the ePolicy Orchestrator settings to enable the transfer of monitoring data from an appliance to a McAfee ePO server.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to transfer monitoring data from and click ePolicy Orchestrator.
3. Configure the ePolicy Orchestrator settings as needed.
4. Click Save Changes.

Best practice: Monitoring file system usage

It is important to monitor file system usage in the /opt partition on Web Gateway, as this partition is used for storing system files while the appliance software is also installed there. This means that a full opt partition impacts the performance of the appliance.

The /opt partition can be monitored based on the following:

- **Incident ID** — An incident with ID 22 is generated on Web Gateway when the /opt partition is filled up to a level of 90%. This incident triggers an alert on the dashboard.

  The utilization level that leads to generating the incident is fixed and cannot be configured.
- **Statistical counter** — A statistical counter called FileSystemUsage is available on Web Gateway to record utilization of the opt partition.

  Using this statistical counter in a suitable rule, you can configure your own utilization threshold to trigger various kinds of alerts and log entries.

Working with a statistical counter

The statistical counter that you work with to monitor the /opt partition is configured as the criteria of a rule set. For example, if the statistical counter records an 85% utilization of the /opt partition, the rules in the rule set are processed.

The rule set is filled with the following:

- A rule that creates a notification message, for example, "/opt partition usage is at 85 %".
- Several rules that send or log this message

Place the rule set as an embedded rule set in the Monitoring rule set, which is by default provided among the rule sets of the Error Handler log on Web Gateway.

The Monitoring rule set and its embedded rule sets are processed every minute by the error handler on Web Gateway, due to the use of incident ID 5 in the criteria of the embedding rule set.

In accordance with the names of the embedded monitoring rule sets that are by default available, your rule set for monitoring the /opt partition might be named Check Opt Partition.

Depending on the threshold that you choose, the criteria for this rule set reads as follows:

StatisticCounter.GetCurrent ("FileSystemUsage") greater than or equals 85

Rules in a rule set for monitoring the /opt partition

A rule set for monitoring the /opt partition can be filled with the rules shown in the following.

Note: The structure of these rules is the same as that of the rules in the monitoring rule sets that are by default embedded in the Monitoring rule set. For example, the rules in the embedded Check Cache Partition rule set also have this structure.

This rule creates the notification message that is sent or logged by the other rules in the rule set.

Name			
Create notification message			
Criteria		Action	Events
Always	–>	Continue	Set User-Defined.notificationMessage = "/opt partition usage at:" + Number.ToString (Statistics.Counter.GetCurrent ("FileSystemUsage")<Default>) + "%"

These rules use the notification message to perform the following activities:

- Send an SNMP trap
- Create a syslog entry
- Send an email notification
- Write a log file entry

Name and criteria		Action	Events
Send SNMP trap			
Always	–>	Continue	Set SNMP.Trap.Additional = User-Defined.notificationMessage SNMP.Trap.Send.User (12, "High /opt partition utilization detected."
Create syslog entry			
Always	–>	Continue	Syslog (3, User-Defined.notificationMessage)
Send email for notification			
Always		Continue	Email.Send ("Enter valid email", "Message from McAfee Web Gateway, User-Defined.notificationMessage) <Monitoring>
Write /opt partition into log			
Always	–>	Continue	Set User.Defined.monitorLogMessage = "High /opt partition utilization detected."

	+ User-Defined.notificationMessage) <Monitoring> FileSystemLogging.WriteLogEntry (User.Defined.monitorLogMessage <Monitoring Incident Log>

Troubleshooting issues with file system usage

You can identify the reasons for issues with file system usage and take measures to prevent these issues.

Identify issues with file system usage

To identify issues with file system usage, you can search the /opt partition for large directories.

Task

1. Log on to the appliance that you want to identify issues on from a local system console, or remotely using SSH.
2. Run the following commands:

   ```
   df -h
   ```

   ```
   du /opt -Dm | sort -n|tail
   ```

   A search is started for the largest folders in the /opt partition. The results are output in ascending order of size, for example, as follows.

   ```
   1158 /opt/mwg/plugin/data 1320 /opt/mwg/plugin 5010 /opt/mwg/storage/default 6011 /opt/mwg/storage
   7937 /opt/mwg/log/debug/cores
   ```

3. If you cannot determine the source of an issue, create a listing for support, using the following commands:

   ```
   df -h
   ```

   ```
   du /opt -Dm | sort -n > opt_directory_listing.txt
   ```

   The command outputs a .txt file with a listing of the directories in the /opt partition.

Preventing excessive utilization of the /opt partition

You can prevent excessive utilization of the /opt partition by taking appropriate measures regarding the files that are stored in this partition.

The following file types often cause the /opt partition to run out of space due to their sizes.

User-defined logs

User-defined logs are, for example, the access.log and access_denied.log. You can find these logs under /opt/mwg/log in the user-defined-logs directory and its subdirectories.

To prevent utilization issues due to log files, set up appropriate rotation, deletion, and push schedules, using the File System Logging settings.

Make sure that you also compress the log files after rotation using the GZIP log files after rotation setting.

Trace files

Trace files are, for example, created for connection or rule engine tracing. You can find these files under /opt/mwg/log/ in the debug directory and its subdirectories.

This directory also contains trace files for authentication, quota, PD storage, the log manager, and the Coordinator subsystem of the Web Gateway appliance.

To prevent utilization issues due to trace files, create them only just before your testing or troubleshooting activities and stop tracing immediately after completing these activities. Many tracing options also allow you to restrict the tracing to individual IP addresses.

Tip: Best practice: Connection tracing files are by default removed when the appliance is restarted. Do not wait for a restart and manually remove these files as soon as possible.

Temporary files

Temporary files are found under /opt/mwg/ in the temp directory.

Web Gateway usually creates temporary mwg-core files in the temp directory while downloading and scanning files. These files are managed by the mwg-core process and are removed after the download and scanning has completed.

Tip: Best practice: Streaming Media files that are scanned for anti-malware filtering can cause very large temp files, as streams have no known end. Let these files bypass scanning.

Support files

Support might request you to provide additional troubleshooting files for reviewing, for example, files that you find under /opt/mwg/log in the debug directory. These files include tcpdump, feedback, and core files, as well as some others.

After support has confirmed the receipt of these files, you can remove them from the /opt partition.

Best practices - Sending access log data to a syslog server

You can configure Web Gateway to send data that is recorded in the access log to a syslog server.

Data about requests for web access that Web Gateway receives from its clients is recorded in the access log. The recording is performed by a rule in a rule set for log handling, which is enabled by default. By adding another rule this data can be made available to a daemon, which sends it to a particular syslog server.

The recorded data includes date and time of a request, the user name of the user who sent the request, the requested URL, and other information. You can modify the configuration to record more or different information about web access.

The data can be sent under different protocols and in different formats. You can also configure a severity level to send, for example, only data about emergencies.

To send the data, you must complete the following:

- Add a rule that makes access log data available to the syslog daemon.
- Adapt the *rsylog.conf* system file to let the daemon send data to a syslog server.

These activities must be completed on every Web Gateway appliance that access log data are to be sent from. In a similar way, you can also configure the sending of other log data.

Protocols for sending data

Data can be sent to a syslog server under the UDP or TCP protocol. Some syslog servers have no TCP listener ports, however. The most common UDP listener port is 514, whereas under TCP the port varies from application to application.

Data formats

Data is sent to syslog servers in different formats, depending on the server type. If in doubt, ask the administrator who is responsible for the syslog server.

- **Default format** — The default log handling rule uses this format to record access log data.

  The format and modified versions of it are also accepted by McAfee Content Security Reporter, version 2.0.
- **SIEM (Nitro) format** — This format is required if the syslog server is provided by McAfee® Enterprise Security Manager (McAfee ESM) (SIEM, formerly known as *Nitro*).

  You can import the *SIEM Nitro Integration* rule set from the online rule set library. This rule set contains a rule that uses the SIEM (Nitro) format to record access log data.
- **CEF format** — This format is required if the syslog server is provided by an ArcSight security manager or a similar program.

  You can import the *CEF Syslog* rule set from the online rule set library. This rule set contains a rule that uses the CEF format to record access log data.

Severity levels

Data with differing severity can be sent to a syslog server. The severity levels are listed in the following. Severity level 6 is recommended.

- 0: Emergency (emerg) — System unusable
- 1: Alert (alert) — Action to be taken immediately
- 2: Critical (critical) — Critical condition
- 3: Error (error) — Error condition
- 4: Warning (warning) — Warning condition
- 5: Notice (notice) — Normal, but significant condition
- 6: Information (info) — Informational message
- 7: Debug (debug) — Message for debugging

Add a rule for sending access log data

To send access log data from Web Gateway to a syslog server, add a rule to the rule for recording data in the Access Log rule set..

Task

1. Select Policy → Rule Sets.
2. Click Log Handler, expand the Default rule set, and select the nested Access Log rule set.
   The content of the nested rule set appears on the configuration pane. By default the rule set contains a rule that writes data about web access to a log line.
3. Add the following rule to make access log data available to the daemon that sends it to the syslog server.

Name			
Make access log data available to syslog daemon			
Criteria		Action	Event
Always	–>	Continue	Syslog (6, User-Defined.logLine)

   The rule uses an event to make the access data that has been written to a user-defined log line before to the syslog daemon. The syslog daemon sends it to the syslog server. The daemon is configured in the *rsyslog.conf* system file.

   The first event parameter specifies the severity level of the access log data.
4. Click Save Changes.

Results

The rule is for making available data that the preceding rule records in default format. If the syslog server requires a different format, replace the preceding rule with a rule that uses the required format.

You can import rule sets with rules that write data in SIEM or CEF format from the online rule set library.

Adapt the rsyslog.conf system file for sending access log data

Adapt the rsyslog.conf system file to ensure that access log data is successfully sent to a syslog server.

Note: Work with the File Editor on the user interface of Web Gateway to adapt the system file. If you use commands from a system console, your changes will be overwritten by future updates.

Task

1. Select Configuration → File Editor.
2. On the files tree, select rsyslog.conf.

   The file content appears on the configuration pane.
3. Edit the file to adapt it for sending access log data.

   a. Look for the following line:

   ```
   *.info;mail.none;authpriv.none;cron.none /var/log/messages
   ```

   The line is part of a section on rules.

   ```
   # Include config files in /etc/rsyslog.d $IncludeConfig /etc/rsyslog.d/*.conf ####RULES#### # Log all
   kernel messages to the console. # Logging much else clutters up the screen. #kern.* /dev/console # Log
   anything (except mail) of level info or higher. # Don't log private authentication messages!
   *.info;mail.none;authpriv.none;cron.none /var/log/messages
   ```

   b. Replace `mail` with `daemon` in this line and insert a – (dash) before the path information.

   ```
   *.info;daemon.none;authpriv.none;cron.none -/var/log/messages
   ```

   This modification prevents the syslog daemon from sending data to the var/log/messages partition on the disk of the Web Gateway appliance system.

   Note:

   The `info` before `daemon` specifies the severity level of the data.

   You can now direct the data to the intended destination.

   c. To send data to a syslog server under the UDP protocol, insert:

   ```
   daemon.info @x.x.x.x:514
   ```

   For `x.x.x.x`, substitute the IP address of the syslog server.

   To send data to a syslog server under TCP, insert:

   ```
   daemon.info @@x.x.x.x:<port number>
   ```

   You can send data to more than one syslog server. For every server, insert a line as shown in this substep.

   When you send data, the messages that carry the data are entered in a default queue when the target server is not available and processed when it is up again. If you send data to more than one server, we *strongly* recommend setting up a queue for each of them.

   Data messages are processed sequentially in a syslog queue. So, if sending data to a syslog server takes more time than usual, data messages to other servers following in the queue would be delayed if there was only one queue.
4. Set up a queue for each server that you send data to.

   A queue is set up by creating a rule that forwards data to the queue. The rsyslogconf system file includes a default forwarding rule in a code block at its end.

   To create a forwarding rule, copy and modify the code block, then append it to the end of the file.

   ○

   In the code block, activate this line and replace the spool file prefix, for example, with `fwdRule2`.

   ```
   #$ActionQueueFileName fwdRule1
   ```

   ○

   In this line, type the name or IP address and port of the syslog server that the data should be sent to.

   ```
   *.* @@remote-host:514
   ```

   Activate other lines of the code block as needed.

   ```
   # ### begin forwarding rule ### # The statements between begin ... and ... end define a SINGLE forwarding #
   rule. They belong together, do NOT split them. If you create multiple # forwarding rules, duplicate the whole
   block! # Remote logging (we use TCP for reliable delivery) # # An on-disk queue is created for this action. If
   the remote host is # down, messages are spooled to disk and sent when it is up again. #$ActionQueueFileName
   fwdRule1 # unique name prefix for spool files #$ActionQueueMaxDiskSpace 1g # 1gb space limit (use as much as
   possible) #$ActionQueueSaveOnShutdown on # save messages to disk on shutdown $ActionQueueType LinkedList # run
   asynchronously #$ActionResumeRetryCount -1 # infinite retries if host is down # remote host is: name/ip:port,
   e.g. 192.168.0.1:514, port optional *.* @@remote-host:514 # ### end of the forwarding rule ###
   ```
5. Click Save Changes.

Resolving issues with sending access log data

Several measures can be taken to resolve issues with sending access log data from Web Gateway to a syslog server.

- If access log data is not received on the syslog server, it might still be written to the var/log/messages partition on the disk of the Web Gateway appliance system.

  Run the following command from a system console to verify that data is not written to disk:

  ```
  tail -f /var/log/messages
  ```
- If access log data is not received on the syslog server, it might be due to restrictions that are, for example, imposed by a firewall. You can perform a tcpdump to see whether Web Gateway sends data packets to the syslog server at all.

  Run the following command from a system console to see the data packets, for example, when they are sent to the syslog server under the UDP protocol:

  ```
  tcpdump port 514
  ```

  You should also review the rsyslog.conf system file to make sure that sending data to the syslog server is configured correctly.
- Web Gateway truncates a data packet that is sent to the syslog server by default if it has more than 2000 characters.

  Add the following line to the rsyslog.conf system file to adjust the packet length:

  ```
  $MaxMessageSize <maximum number of characters>
  ```

Best practice: Implementing TLS-secured usage of syslog data

You can implement use of the TLS protocol that is provided by an rsyslog package for TLS-secured sending of messages with syslog data.

The *rsyslog-gnutls* package and several related packages are installed by default on the Web Gateway appliance system. The rsyslog-gnutls package provides the TLS protocol, which allows you implement TLS encryption for secure sending of log messages from a syslog client to a remote syslog (rsyslog) server.

TLS-secured sending of syslog messages requires the use of SSL certificates for the server and its clients, as well as for a root certificate authority (root CA) that signs these certificates.

The packages that are involved in implementing TLS encryption include:

- rsyslog-gnutls-5.8.10
- rsyslog-5.8.10
- gnutls-2.8.5

For more information about these packages, see the documentation of the vendor who provides them (RSYSLOG).

High-level steps for implementing TLS-secured usage of syslog data

To implement TLS-secured usage of syslog data with the rsyslog-gnutls package, complete the following high-level steps.

Task

1. Make sure the system time and date is the same on the appliance that you plan to configure as the syslog server and its clients.

   The system that you use for creating certificates on must also have the same time and date.
2. Create certificates for a root CA, as well as for the syslog server and its clients.
3. Configure a syslog server.
4. Configure one or more syslog clients.
5. Send test messages from the syslog clients to the syslog server.

Prepare the use of TLS-secured syslog data

Make sure that system time and date is the same on all appliances that you want to prepare the use of TLS-secured syslog data on and create certificates for the TLS encryption.

Task

1. Log on to a Web Gateway appliance that you want to prepare the use of TLS-secured data on from a local system console or remotely using SSH.
2. [Optional] If a version of the rsyslog-gnutls package is already installed on an appliance, you can run the following command to identify this version.

   ```
   rpm -qa rsyslog-gnutls
   ```

3. Set system time and date on this appliance and on all other appliances that you want to prepare for sending and receiving TLS-secured syslog messages. Set time and date also on the system that you use to create certificates.

   On Linux systems, you can run the following command.

   `date` <mm for the month><dd for the day><hh for the hour, using the 24-hours system><mm for the minute><yy for the year>

   For example, to set system time and date to November, 20th, 2016, 9:45 p. m., run:

   ```
   date 1120214516
   ```

   On Linux systems, you can also synchronize the time and date on the mainboard of the hardware platform for the appliance with that of the appliance software. For this synchronization run:

   ```
   hwclock -systohc
   ```

4. Create and store certificates for the root certificate authority (CA) and the appliances that send and receive TLS-secured syslog messages.
   a. Use a certificate creation tool, for example, OpenSSL or Certtool, to create the certificates.
      For more information, see the documentation of the vendor who provides the rsyslog package (RSYSLOG).
   b. Log on to the appliance that you want to store the certificates on from a local system console or remotely with SSH.
   c. Run the following command to create a directory for storing the certificates.

      ```
      mkdir -pv /etc/rsyslog.d/cert
      ```

   d. Copy the certificates to the directory. Run, for example:

      ```
      cp ca.pem syslogserver.cert.pem syslogserver.key.pem syslogclient1.cert.pem syslogclient1.key.pem
      syslogclient2.cert.pem syslogclient2.key.pem /etc/rsyslog.d/cert
      ```

   e. [Optional] Check the content of the certificates. Run, for example:

      ```
      openssl x509 -in syslogclient1cert.pem -text noout|less

      openssl x509 -in syslogclient1cert.der -inform der -text noout
      ```

Configure a syslog server to receive TLS-secured data

Work with a rsyslog system file on a Web Gateway appliance to configure a syslog server that receives TLS-secured data.

Task

1. On the user interface, select Configuration → File Editor.
2. On the appliances tree, select the appliance that you want to configure a syslog server on, then select rsyslog.conf.
   The content of the system file appears in the configuration frame.
3. Add the following lines to the file content.

   ```
   $ModLoad imtcp.so #Specifies the TCP listener that listens to requests sent from the clients.
   $DefaultNetstreamDriver gtls #Requires use of the netstream driver. $DefaultNetstreamDriverCAFile /etc/
   rsyslog.d/cert/ca.pem #Specifies the root CA. $DefaultNetstreamDriverCertFile /etc/rsyslog.d/cert/
   server.cert.pem #Specifies the certificate for the server. $DefaultNetstreamDriverKeyFile /etc/rsyslog.d/cert/
   server.key.pem #Specifies the certificate key for the server. $InputTCPServerStreamDriverAuthMode x509/name
   $InputTCPServerStreamDriverPermittedPeer <client IP address> #Specifies the client through its IP address.
   $InputTCPServerStreamDriverMode 1 #Requires the server to run in TLS mode only. $InputTCPServerRun 10514
   #Specifies the listener port that the syslog communication starts at.
   ```

4. Log on to the appliance from a local system console or remotely using SSH.
5. Run the following command to restart the rsyslog function on the appliance.

   ```
   /etc/init.d/rsyslog restart
   ```

   After restarting rsyslog, a TLS-secured connection is set up, using the settings in the configuration file and the certificates.
6. Verify that the TLS-secured connection has been set up successfully.

   ```
   cat /var/log/messages
   ```

   After running the verification command, you should see messages like the following displayed.

   ```
   Nov 15 11:23:37 testdev kernel: Kernel logging (proc) stopped. Nov 15 11:23:37 testdev rsyslogd: [origin
   software="rsyslogd" swVersion="5.8.10" x-pid="37290" x-info="http://www.rsyslog.com"] exiting on signal 15.
   Nov 15 11:23:37 testdev kernel: imklog 5.8.10, log source = /prog/kmsd started. Nov 15 11:23:37 testdev
   rsyslogd: [origin software="rsyslogd" swVersion="5.8.10" x-pid="41552" x-info="http://www.rsyslog.com"] start
   [root@testdev tls]
   ```

Configure a syslog client to send TLS-secured data

Work with a rsyslog system file on a Web Gateway appliance to configure a syslog client that sends TLS-secured data.

Task

1. On the user interface, select Configuration → File Editor.
2. On the appliances tree, select the appliance that you want to configure a syslog client on, then select rsyslog.conf.

   The content of the system file appears in the configuration frame.
3. Add the following lines to the file content.

   ```
   $template TEST-MESSAGE, "%HOSTNAME" ClientText: %syslogtag%%msg%" $WorkDirectory /var/spool/rsyslog #Specifies
   where to store spool files. $ActionQueueFileName fwdRule1 #Provides a unique name prefix for spool files.
   $ActionQueueMaxDiskSpace 1g #Sets a space limit of 1 GB. $ActionQueueSaveOnShutdown on #Saves messages to disk
   upon shutdown. $ActionQueueType LinkedList #Lets the process run asynchronously. $ActionResumeRetryCount -1
   #Triggers an unlimited number of retries if the server is down $DefaultNetstreamDriver gtls #Requires use of
   the netstream driver. $DefaultNetstreamDriverCAFile /etc/rsyslog.d/cert/ca.pem #Specifies the root CA.
   $DefaultNetstreamDriverCertFile /etc/rsyslog.d/cert/client.cert.pem #Specifies the certificate for the client.
   $DefaultNetstreamDriverKeyFile /etc/rsyslog.d/cert/client.key.pem #Specifies the certificate key for the
   client. $InputTCPServerStreamDriverAuthMode x509/name $InputTCPServerStreamDriverPermittedPeer <server host
   name or IP address> #Specifies the server through its host name or IP address. $InputTCPServerStreamDriverMode
   1 #Requires the server to run in TLS mode only. $InputTCPServerRun 10514 #Specifies the listener port that the
   syslog communication starts at.
   ```

4. Log on to the appliance from a local system console or remotely using SSH.
5. Run the following command to restart the rsyslog function on the appliance.

   ```
   /etc/init.d/rsyslog restart
   ```

   After restarting rsyslog, a TLS-secured connection is set up, using the settings in the configuration file and the certificates.
6. Verify that the TLS-secured connection has been set up successfully.

   ```
   cat /var/log/messages
   ```

   After running the verification command, you should see messages like the following displayed.

   ```
   Nov 15 11:23:37 HyperVMlos2AD kernel: Kernel logging (proc) stopped. Nov 15 11:23:37 HyperVMlos2AD rsyslogd:
   [origin software="rsyslogd" swVersion="5.8.10" x-pid="53727" x-info="http://www.rsyslog.com"] exiting on
   signal 15. Nov 15 11:23:37 HyperVMlos2AD kernel: imklog 5.8.10, log source = /prog/kmsd started. Nov 15
   11:23:37 HyperVMlos2AD rsyslogd: [origin software="rsyslogd" swVersion="5.8.10" x-pid="57261" x-info="http://
   www.rsyslog.com"] start [root@HyperVMlos2AD rsyslog.d]
   ```

Results

You can now send TLS-secured syslog messages. Run the following command to send a test message:

```
logger "TLS-secured test message"
```

Sending syslog data to McAfee Enterprise Security Manager

Data that is logged on Web Gateway in syslog log files can be sent to McAfee® Enterprise Security Manager (McAfee ESM).

The data transfer is controlled by a rule in a rule set that is available in the online rule set library for Web Gateway. The component of McAfee ESM that the data is sent to is the McAfee SIEM Receiver.

To enable the transfer, you adapt a system file for remote use of syslog data on Web Gateway. The name of this system file is *rsyslog* (the *r* in the file name stands for *remote*). You must also configure the McAfee SIEM Receiver to let Web Gateway be included as a data source in the McAfee ESM environment.

Version 9.3.2 or a later version of McAfee ESM is required for the data transfer to work.

Configure the sending of syslog data

To send syslog data that is collected on Web Gateway to McAfee ESM, complete the following high-level steps.

Task

1. Import the *McAfee SIEM* rule set from the online rule set library for Web Gateway. Place it as a nested rule set in the default Log Handler rule set.
   In the online rule set library, this rule set is available under *SIEM (Nitro) Integration*.
2. In the imported rule set, enable the *Send to syslog* rule and disable the *Send to nitro.log* rule.
3. Use the File Editor to adapt the *rsyslog* system file for the data transfer.
   If you are running multiple Web Gateway appliances in a Central Management cluster, adapt the system file on every appliance within the cluster.
4. On McAfee ESM, configure the McAfee SIEM Receiver to let Web Gateway be added as a data source.
   For more information, see the documentation for McAfee ESM and the *Data Source Configuration Guide*. The guide is provided in the online rule set library under *SIEM (Nitro) Integration*.

Adapt the rsyslog system file for the data transfer

Adapt the *rsyslog* system file on Web Gateway to ensure that syslog data is successfully sent to McAfee ESM.

Task

1. Select Configuration → File Editor.
2. On the files tree, select rsyslog.conf.
   The file content appears on the configuration pane.
3. Edit the file to adapt it for the data transfer.

   The edited file should look as shown in the following. The modified lines are in the paragraph that begins with: `The below will direct all daemon.info messages to the remote syslog server ...`

   Note: The information that you provide here includes the IP address of the McAfee SIEM Receiver.

   ```
   # default parameters $DirCreateMode 0755 $FileCreateMode 0640 $FileGroup adm $umask 0026 # Include config
   files in /etc/rsyslog.d $IncludeConfig /etc/rsyslog.d/*.conf # Log all kernel messages to the console. #
   Logging much else clutters up the screen. #kern.* /dev/console # Log anything (except mail) of level info or
   higher. # Don't log private authentication messages! # The following directs all daemon.info messages to the #
   remote syslog server at [IP_OF_MCAFEE_EVENT_RECEIVER] # add @@ for TCP syslog for example #daemon.info
   @192.168.1.1 *.info;daemon.!=info;mail.none;authpriv.none;cron.none -/var/log/messages # The authpriv file has
   restricted access. authpriv.* /var/log/secure # Log all the mail messages in one place. mail.* /var/log/
   maillog # Log cron stuff cron.* /var/log/cron # Everybody gets emergency messages *.emerg # Save news errors
   of level crit and higher in a special file. uucp,news.crit /var/log/spooler # Save boot messages also to
   boot.log local7.* /var/log/boot.log
   ```

4. Click Save Changes.

Fine-tuning the collection and evaluation of syslog data

Several fine-tuning activities can be performed to ensure that relevant syslog data is collected on Web Gateway and efficiently evaluated on McAfee ESM.

The amount of syslog data that is collected can be throttled by excluding less relevant data and restricting the process to logging only important events. Relevant data can also be added, however, to the syslog data by implementing additional logging activities.

On McAfee ESM, data aggregation can be disabled to ensure that no relevant data is overlooked.

Throttling the amount of syslog data

The amount of syslog data that Web Gateway sends to McAfee ESM can be throttled by taking, for example, the following measures.

- **Excluding Authentication Required (status code 407) responses** — These are standard responses that do not require much attention regarding web security.

  To exclude these responses from the syslog data that is transferred, add a rule in the rule set that you imported.

  The rule must be placed, together with other throttling rules that you might implement, at the top of the rule set. It should look as follows:

Name		
Exclude 407 responses		
Criteria		Action
Response.StatusCode equals 407	–>	Stop Rule Set

- **Sending only logged Block actions** — Block actions are crucial in maintaining web security, but usually account for only a small proportion of web traffic.

  To restrict the syslog data that is transferred to log files for these actions, add a rule in the rule set that you imported.

  The rule must be placed, together with other throttling rules that you might implement, at the top of the rule set. It should look as follows:

Name		
Send only logged Block actions		
Criteria		Action
Block.ID equals 0	–>	Stop Rule Set

Adding hashes of infected files to the syslog data

To the syslog data can be added the hash values of files that were processed on Web Gateway and found to be infected. File hashes can be useful for tracking infections and possible outbreaks.

Note: As hashing consumes a large amount of resources, we recommend using it only for important issues. If in doubt, consult McAfee support.

To enable the calculation and logging of file hashes, add an event to the rule that detects and blocks infected files. By default, this rule is *Block if virus was found* in the *Gateway Anti-Malware* rule set.

The event should look as follows:

```
Header.Block.Add('X-Hash-MD5, Body.Hash("md5"))
```

The *Header.Block.Add* event is a preconfigured event that you can select from the list of available events. It adds an entry to the syslog log when the rule that it is inserted in applies.

The event takes two parameters, which you must configure:

- `X-Hash-MD5` — Name of the log entry
- `Body.Hash("md5")` — Value of the log entry

This parameter is a property for calculating the hash value of a file. Here it calculates the hash value of the infected file that was sent to Web Gateway as the body of a request or response.

The property takes itself a parameter, which determines the method for calculating the hash.

Note: If you are working with the key elements view for rule sets, you must switch to the complete rules view to add the event.

After adding the event, the blocking rule should look as follows.

Name		
Block if virus was found		
Criteria	Action	Events
Antimalware.Infected<Gateway –> Anti-Malware>equals true	Block<Virus Found>	Statistics.Counter.Increment ("BlockedByAntiMalware", 1)<Default> Header.Block.Add ('X-Hash-MD5, Body.Hash("md5"))

Disabling the aggregation of syslog data

When the McAfee SIEM Receiver receives syslog data from Web Gateway, this data is by default aggregated into a single record. While aggregation can be useful for many data sources, it could be undesirable for Web Gateway, as critical information might get lost during aggregation.

You can disable aggregation for Web Gateway data on McAfee ESM.

For more information, see the documentation for McAfee ESM and the *Data Source Configuration Guide*. The guide is provided in the online rule set library under *SIEM (Nitro) Integration*.

Resolving issues with the transfer of syslog data

To resolve issues with sending syslog data from Web Gateway to McAfee ESM, several measures can be taken.

- Review the configuration on Web Gateway and make sure that the following applies:
    - The *Send syslog* rule is enabled.
    - The IP address of the McAfee SIEM Receiver is correctly specified in the *rsyslog* system file.
- Review the configuration on McAfee ESM.

  For more information on this step and on others that are performed on McAfee ESM, see the documentation on McAfee ESM and the *Data Source Configuration Guide*. The guide is provided in the online rule set library under *SIEM (Nitro) Integration*.
- Verify that syslog data is generated on Web Gateway, for example, by running the following command from a system console:

  ```
  tcpdump -s 0 –I any port 514
  ```
- Verify that syslog data is received on the McAfee SIEM Receiver.
- Verify that the syslog log is generated on Web Gateway in proper format.

  Entries in the syslog log usually look as follows:

  ```
  McAfeeWG|time_stamp=[30/Mar/2014:05:18:16 +0000]| auth_user=|src_ip=172.18.19.225|server_ip=69.20.171.162|
  host=www.nitroguard.com| url_port=80|status_code=200|bytes_from_client=187|bytes_to_client=272| categories=|
  rep_level=|method=GET|url=http://www.nitroguard.com/ngdb.dll?NG:StartIt:0| media_type=|application_name=|
  user_agent=Mozilla/4.0 (compatible; Synapse)| block_res=0|block_reason=|virus_name=|hash=| McAfeeWG|
  time_stamp=[30/Mar/2014:05:18:20 +0000]| auth_user=|src_ip=172.18.19.225|server_ip=69.20.171.162|
  host=www.nitroguard.com| url_port=80|status_code=200|bytes_from_client=376|bytes_to_client=200| categories=|
  rep_level=|method=GET| url=http://www.nitroguard.com/ngdb.dll?NG:DoIt:
  0:Info=D8BC0B7C97D2C352AFE4643FEA44AE4D4C70F79271
  D4620B64294729E046CB607B5458AC24BA31B061A12313E016EB7F62ED267DC6FE9A02A552681347EF796303514934
  EE08EF0DA76B27F5EEA225B0DB274367AF4FEA574EA6137728| media_type=|application_name=|user_agent=Mozilla/4.0
  (compatible; Synapse)| block_res=0|block_reason=|virus_name=|hash=|
  ```

Troubleshooting

Several methods and tools are available for troubleshooting problems on an appliance.

Troubleshooting methods

When problems arise on a Web Gateway appliance, you can use several methods to solve them.

Rule tracing

You can create and review rule traces on the Web Gateway interface. These traces record how rules were processed to deal with requests received from clients, as well as with the responses to these requests that were received from the web.

Reviewing these traces, you can find out which rules were processed and what actions were executed, for example, actions for blocking requests.

Tracing information is shown for:

- **Cycles** — Cycles in which rules were processed, such as the request, response, or embedded objects cycle
- **Rules** — Rules that were processed in these cycles
- **Rule sets** — Rule sets where these rules were contained
- **Rule Criteria** — Criteria that matched, so actions were executed
- **Properties** — Properties and their values at the time when the rule criteria matched
- **Actions** — Actions that were executed when the rule criteria matched
- **Events** — Events that were triggered when the rule criteria matched

Recording and inspecting data in files

You can record data about appliance behavior in files and inspect them. These file types can be created:

- **Log files** — Log events and functions, such as access to the web or file updates
- **Rule tracing files** — Record the processing of rules
- **Feedback files** — Trace processes that went on before a function failed
- **Core files** — Record memory content after a function failed and caused an appliance to finish operation
- **Connection tracing files** — Record activities on connections between an appliance and other network components
- **Packet tracing files** — Record network activities of an appliance

Using network tools

You can test whether connections from an appliance to other network components still work. Tools for this purpose include *ping*, *nslookup*, *ipneigh*, and others.

Using system tools

You can use system tools to perform a service restart on an appliance and to display the anti-malware filtering threads that are currently running.

Restoring a configuration

When other troubleshooting methods do not work, it might be necessary to replace a faulty appliance configuration with a backup.

Having a backup available can also help in other situations, for example, when you want to discard changes applied to an existing configuration.

Options are provided for creating backups and using them to restore configurations.

Resetting the appliance password

You can reset the appliance password. This password is the root password that is required when accessing an appliance from a system console using the command line. It is also known as root or console password.

Resetting this password may be required, for example, if you cannot remember it.

Rule tracing

To debug issues with rule processing, you can use rule tracing functions on the user interface.

Rule traces can be created, which record the activities that were completed to process the implemented rules when users of your network sent requests for web access from particular clients.

You can filter these traces according to the date of creation, the URL that was sent with a request, or the rule action, such as Block, Redirect, or Continue, and others, that was executed when a rule was processed.

Tracing covers all activities in the different processing cycles that were performed for a request, including the request, response, and embedded object cycles. Tracing results can be viewed separately for different cycles.

Properties in the criteria of the rules that were involved in the processing can also be viewed separately, together with the values they were set to when the rules were processed.

Three panes are provided on the rule tracing page of the user interface to let you complete rule tracing activities.

- **Traces pane** — Allows you to create traces, filter, and remove them

  You can also export and store traces and import them again for viewing later on or import traces that have been created on other Web Gateway appliances.
- **Rules pane** — Allows you to select a processing cycle and view the rule sets and individual rules that were processed in this cycle
- **Details pane** — Allows you to view the rule criteria of individual rules with their properties and the values the properties have been set to

Cycles in rule tracing

Processing starts when a request for web access has been received from a client of Web Gateway. It is performed in different cycles, beginning with the request cycle, in which rules are processed that are related to the elements of the request itself, for example, to a URL that was sent with a request.

If none of the rules in this cycle forbids a forwarding of the request to the web, for example, due to a negative categorization of a URL, the request is forwarded. Processing then waits for a response from the web.

When the response arrives, the rules of the response cycle are processed. For example, when a file that was requested for downloading is sent in response, it is scanned for virus and other malware infections according to a particular rule and eventually passed on or not to the client that requested the download.

Other processing cycles are performed for embedded objects sent with requests or responses. Processing activities can also be logged according to the configured logging rules.

All processing that is performed in the different cycles for an initial request from a client of Web Gateway can be viewed as an entity, which is termed a transaction.

To debug an issue with rule processing, you can analyze the complete rule trace of a transaction or focus on a particular cycle that seems interesting with regard to problem solving.

Properties in rule tracing

Whether a rule applies and executes a particular action, for example, a Block action that blocks a request for web access, depends on the rule criteria, which contains properties that are set to particular values during the processing.

For example, the *Antimalware.Infected* property, which is contained in the rule criteria of a default anti-malware rule, is set to *true* when a scanned web object has been found to be infected by viruses or other malware. Then the criteria of this rule matches, and a Block action is executed.

When analyzing a rule trace, it can be useful to look at the properties that were involved in rule processing and the values they were set to. Therefore, properties and their values can also be viewed separately.

Deleting and restoring rule traces

Rule traces can be removed from the panes of the rule tracing page, but not deleted on that page.

To delete rule traces, you need to access the Rule tracing files section, which is provided for every individual appliance under the Troubleshooting top-level menu.

In this section, you can also restore traces to the rule tracing panes that you have previously removed.

Note:

For each client IP address that is traced, up to 5000 traces can be stored on an appliance. When this number is exceeded, the oldest 100 traces are deleted.

The deletion is not reflected on the rule tracing panes, so you might see entries for traces that you cannot access because the traces have already been deleted.

Debug rule processing issues using rule tracing

Use the options of the rule tracing panes to create rule traces and review them to debug issues with rule processing.

Task

1. Select Troubleshooting.
2. On the troubleshooting tree, select Rule Tracing Central.
   The rule tracing panes appear.
3. Work with the rule tracing panes to debug rule processing issues.

Use rule tracing to find out why a request was blocked

When a request for web access that a user sent from a client of Web Gateway has been blocked, you can use rule tracing to find the rule that blocked the request and the reason why it was done.

This is a sample procedure that describes one of several ways to use rule tracing for recording and analyzing rule processing on Web Gateway.

Task

1. Select Troubleshooting and on the appliances tree, select Rule Tracing Central.
   The rule tracing panes appear.
2. Create rule traces.
   a. In the traces pane, leave the name of the current appliance, which appears in the appliances names field.
      In this sample procedure, you will perform rule tracing for requests that were processed on this appliance.
   b. In the client IP address field, enter the IP address of the client that sent the request you want to do rule tracing for.
   c. Click Go.
      Rule traces for the latest requests received from the client are created. When trace creation is completed, entries for the traces appear in the traces field.
3. Filter the rule traces.
   a. In the time and URL filtering field, enter the URL that was sent with the blocked request.

      The rule traces are filtered to show only entries for traces that were performed for requests to access a web object with this URL.

      Let us assume that a request with this URL was only submitted once by the client in question. This would mean only one entry is shown as the filtering result.
   b. Select the entry.
      Detailed information from the trace that recorded rule processing for the request with this URL is shown in the rules and details panes.
4. Review a rule trace.
   a. Review the tracing information in the rules pane.

      The rules that were processed to deal with the request are shown with their rule sets.

      The rule that blocked the request is selected and marked by a red arrow. If the arrow points to the right, the rule blocked the request in the request cycle. If the arrow points to the left, it was in the response cycle.

b. Review the tracing information in the details pane.

   - The cycle in which the rule blocked the request, the name of the rule, its criteria, action, and event are shown.

     The criteria is marked with a grey hook, which means it has matched.
   - Under Evaluated in the field below the criteria with the hook, the criteria is repeated.

     Under Value in the same field the value is shown that the property had at the time when the criteria matched and the rule blocked the request.

   Let us assume that, for example, the details pane shows the following details for the rule that blocked the request.

   - Cycle — Response
   - Rule name — Block if virus was found
   - Criteria — Antimalware.Infected<Gateway Anti.Malware> equals true
   - Evaluated — Antimalware.Infected equals true, Value — true
   - Action — Block<Virus found>
   - Event — Statistics.Counter.Increment<Default>("BlockedByAntiMalware", 1>

Results

This means that rule tracing showed the request was blocked because the requested object had been found to be infected by a virus or other malware.

The blocking action was performed by a virus and malware filtering rule, which was processed in the response cycle when the object was received from a particular web server in response to the request.

The criteria of this rule included the Antimalware.Infected property. To find out what this property must be set to, the Anti-Malware engine on Web Gateway was called. It scanned the requested web object and detected an infection, so the property could be set to *true* and the rule criteria matched.

Best practices - Find out why a web page displays no images

Use rule tracing to find out why a requested web page appears on a client system, but with text only and without displaying any images.

Imagine a sample issue, where a user requests access to the CNN channel homepage from a browser on a client of Web Gateway. The page appears, but displays only text.

You can use rule tracing to see whether a CNN server that provides the images on the homepage might have been blocked and why this happened.

Task

1. On the user interface of Web Gateway, select Troubleshooting.
   If you are using several Web Gateway appliances in a Central Management configuration, make sure you are logged on to the appliance that the client in question is connected to.
2. On the troubleshooting tree, select Rule Tracing Central.
3. Create a trace.
   a. In the input field on the top left, type the IP address of the client system that had a request blocked, then click Go on the toggle button next to the input field.

      Requests for web access sent from the client are now traced and entries for trace files are displayed in the output field on the lower left.

      The Go on the toggle button turns into a cross to let you stop the process when no more tracing is needed.

   b. On the client system, refresh the browser or click or enter *ccn.com* again to reproduce the issue.

      Trace file entries appear in the output field on Web Gateway.

      Note: Depending on the amount of data that is being transferred, it can take a while until the trace file entries appear.

   c. When you have reproduced the issue and the trace file entries have appeared, click the toggle button again to stop the tracing.

4. Review the trace file entries.

   For every request that has been traced, a time stamp and the requested URL are shown.

   At the beginning of an entry, a symbol for the most impacting action that was executed when processing the request is also shown. The most impacting of all actions is the Block action.

   When reviewing the trace file entries, you will see several entries with the blocking symbol and a URL beginning with *cdn.turner.com/ccn*. These are probably trace files for requests to access the CCN server that provides the images.
5. Select a trace file entry with *cdn.turner.com/ccn*.

   Information on this trace appears in the rules and details panes on the right.
6. Review the rules pane.

   The pane shows the rules that were processed for the request that was traced. The view stops at the last rule that applied before rule processing stopped. The rule is highlighted.

   This way you can see that *Block URLs whose category is in Category Block List* is the last rule that applied.
7. Review the details pane.

   On the two tabs of the details pane, more tracing information is shown.

   On the Top Properties tab, you will see, among other information, that the *URL.Categories* property had the value *Business* when the rule mentioned above was processed.

Results

This completes your rule tracing activities for this issue. Images from the CNN server were not displayed because the URLs that were submitted for accessing this server have fallen into the *Business* category and this category is on a blocklist.

If you want to see the images displayed, you need to reconfigure the web security policy for your network and put, for example, *cdn.turner.com/ccn/** on a URL whitelist.

Restore removed rule traces to the rule tracing panes

To restore rule traces that you have removed from the rule tracing panes, supply them from the rule traces directory of an appliance or import them in a source file.

How removed rule traces can be restored to the rule tracing panes depends on whether they were created on the appliance you are currently logged on to or were imported to this appliance.

Accordingly. you can supply them from the rule traces directory of the appliance or repeat the import of the source file.

Restore removed rule traces from an appliance directory

When rule traces that you have removed from the rule tracing panes had been created on the current appliance, you can restore them from the directory of rule tracing files on that appliance.

Task

1. Select Troubleshooting.
2. On the troubleshooting tree, expand the appliance you want to restore rule traces on..
3. Select Rule tracing files.

   The directory of the rule tracing files appears on the right side of the troubleshooting page.
4. Under Trace files, select the rule tracing files you want to restore.
5. Click Analyze.

Results

The rule traces are accessible again in the rule tracing panes.

Restore removed rule traces by importing a source file

When rule traces that you have removed from the rule tracing panes had previously been imported, you can restore them by importing the source file once again.

Task

1. Select Troubleshooting.
2. On the troubleshooting tree, select Rule Tracing Central.
3. Click Traces and then Import.
   The local file manager opens.
4. Browse to the location where you stored the zipped file that is the source for the rule traces you want to restore, select the file, and import it.

Results

The rule traces are accessible again on the rule tracing panes.

Delete rule traces

To delete rule traces, access the directory of rule tracing files on an appliance and use the delete option that is provided.

Task

1. Select Troubleshooting.
2. On the troubleshooting tree, select the appliance you want to delete rule traces on, then click Rule tracing files.
   The directory of rule tracing files appears on the right side of the troubleshooting page.
3. Under Trace files, select the rule tracing files you want to delete and click Delete.
4. In the window that opens, confirm the deletion.

Using a diagnostic tool to evaluate Regex terms

To troubleshoot performance issues, you can use the *regex-diagnosis* tool, which lets you know about how much time was needed to process individual Regex (Regular expression) terms.

The tool is stored in a folder with system files on Web Gateway. The path to it is /opt/mwg/bin. You can run the tool from a system console.

To evaluate Regex terms, you specify two files as parameters of the tool command. One that includes these terms and another that they are compared with to detect matches.

regex-diagnosis tool - Usage

You can use the *regex-diagnosis* tool to evaluate the processing of Regex (Regular expression) terms regarding the time that is consumed by this processing.

The tool resides in the /opt/mwg/bin system files folder on a Web Gateway appliance.

Usage

```
/opt/mwg/bin/regex-diagnosis <parameters>
```

Unless otherwise described, parameters are optional.

One of the files specified here can have — (dash) as its filename. The file is then retrieved from stdin.

Parameters

Parameter	Description
`-m (mwg|boost|stl)`	Mode of execution • `mwg` — MWG-internal implementation — Merged scanners • `boost` — boost::regex - single line • `stl` — c++11 std:Regex — single line Either this parameter or `-f` must be specified. Default: `mwg`
`-f <filename>`	com.scur.type.regex.* file from /opt/mwg/storage/ area Either this parameter or `-m` must be specified.
`-r <filename>`	Raw Regex input file
`-d <filename>`	Text file with string data to match the data in the file specified under- `-f <filename>` or `-r <filename>`
`-v`	Verbose logging

Evaluate Regex terms with the regex-diagnosis tool

Use the *regex-diagnosis* tool to evaluate the processing of Regex (Regular expression) terms regarding the time consumed..

In the following sample evaluation, two files with lists of Regex terms are compared with a list in a default file for matches. The time needed for detecting any matches is provided by the tool. The time is also provided if no match could be detected.

Task

1. From a system console, connect to the Web Gateway appliances where the files with the Regex terms that you want to evaluate reside.
2. Run the following command to start the tool.
   ```
   /opt/mwg/bin/regex-diagnosis -m mwg -f /opt/mwg/storage/subscribed_lists/update_server/com.scur.type.regex.
   3216.xml -d -
   ```
3. Review the tool output. It might look as follows.
   ```
   Regex mode : mwg Regex filename: /opt/mwg/storage/subscribed_lists/update_server/com.scur.type.regex.3216.xml
   Data filename : - successfully opened '/opt/mwg/storage/subscribed_lists/update_server/com.scur.type.regex.
   3216.xml' with mode: 'r' https://some.url.not.in.the.list.com https://some.url.in.the.list.office365.com #2
   data elements in file - data = https://some.url.not.in.the.list.com, match = false, matching Regex = '', CPU
   time used: 0.144074 ms data = https://some.url.in.the.list.office365.com NRE: First match='.office365.com'
   NRE: Found match='.office365.com' NRE: Matched regex/glob='*.office365.com' , match = true, matching Regex =
   '*.office365.com', CPU time used: 0.193654 ms Summary: #0 compilation errors, #1 matches, #1 mismatches, #0
   exceptions
   ```

Results

As a result from evaluating the Regex terms in these files, the following was found:

- The *some.url.not.in.the.list* list produced no matches.

  CPU time needed to find this out: 0.144074 ms
- The *some.url.not.in.the.list.office365* list produced one match:`*.office365.com`.

  CPU time needed to find this out: 0.193654 ms.

Create a feedback file

You can create a feedback file to backtrace processes after the failure of a function.

Task

1. Select the Troubleshooting top-level menu.
2. On the appliances tree, select the appliance you want to backtrace processes on and click Feedback.
3. Select or deselect Pause running McAfee Web Gateway to create a backtrace as needed.
   Note: We recommend that you select the checkbox.
4. Click Create Feedback File.
   A feedback file is created and appears with its name, size, and date in the list under Feedback file.

Results

Using the items on the toolbar, you can perform several file-related activities, such as view or download a file.

Enable the creation of core files

You can enable the creation of core files to record memory content after the failure of a function has caused an appliance to terminate operation.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to record memory content on and click Troubleshooting.
3. In the Troubleshooting section, select Enable core file creation.
4. Click Save Changes.
   Core files are now created whenever the appliance terminates due to the failure of a particular function.

Results

You can view the core files, after selecting the appliance under the Troubleshooting top-level menu and clicking Core Files. The files are then displayed in a list.

Using the items on the toolbar, you can perform several file-related activities, such as view or download a file.

Enable the creation of connection tracing files

You can enable the creation of trace files to record activities occurring on connections between an appliance and other network components.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to record connection activities on and click Troubleshooting.
3. In the Troubleshooting section, select Enable connection tracing.
4. [Optional] To trace only activities on a connection to a particular client of the appliance, select Restrict tracing to only one IP and type the IP address of the client in the Client IP field.
5. Click Save Changes.
   Connection tracing files are now created.

Results

You can view the connection tracing files, after selecting the appliance under the Troubleshooting top-level menu and clicking Connection Tracing. The files are then displayed in a list.

Using the items on the toolbar, you can perform several file-related activities, such as view or download a file.

Create a packet tracing file

You can create a packet tracing file to record the network activities of an appliance.

Task

1. Select the Troubleshooting top-level menu.
2. On the appliances tree, select the appliance you want to record network activities on and click Packet tracing.
3. In the Command line parameters field, type parameters for the packet tracing as needed.
4. Click tcpdump start.

   A packet tracing file is generated and appears with its name, size, and date in the list under Results (dump).

   To stop the ongoing generation of a packet tracing file, click tcpdump stop.

Results

Using the items on the toolbar of the list, you can perform several file-related activities, such as view or download a file.

Work with system and network tools

You can work with several system and network tools to troubleshoot problems on an appliance.

Task

1. Select the Troubleshooting top-level menu.
2. On the appliances tree, select the appliance you want to use a tool on and click System Tools or Network Tools.

   The available system tools are:

   - service restart
   - AV threads

   The available network tools are:

   - ping and ping6
   - nslookup
   - traceroute and traceroute6
   - ipneigh
   - ntp
   - hastats
3. In the Command line parameters field, type the parameters for a command that can be executed by a particular tool.

   For example, type the name of a host you want to connect to using the *ping* network tool.
4. Click the button for the tool that you want to use.

   The corresponding command is executed and the resulting output displayed under Results.

   The following could, for example, be displayed:

   ```
   Ping: Unknown host testhost
   ```

   Note: To export the results to a location within your file system, click Export and specify the location in the window that opens.

Restart a service of the operating system

Using the restart tool, you can stop a service of the operating system that is currently running and restart it again. If a service is not running at the time when the tool is applied to it, the service is started.

Note: You cannot restart the main *mwg* service and the *sysconfd* service, which is a daemon for implementing manual configuration changes, using this system tool.

Task

1. Select the Troubleshooting top-level menu.
2. On the appliances tree, select the appliance you want to start a service on and click System Tools.
3. In the Command line parameters field, type the service name and parameters as required.
4. Click service restart.

   The service is restarted and the executed service activities are displayed in the Results field.

   The following could, for example, be displayed after restarting the *ip6tables* service.

   ```
   Flushing firewall rules: [OK] Unloading ip6tables modules: [OK] Applying ip6tables firewall rules: [OK]
   ```

   Note: To export the results to a location within your file system, click Export and specify the location in the window that opens.

Display running AV threads

You can display the threads that are currently running to perform anti-malware scanning activities. Seeing many threads lets you know that scanning a particular request or response is consuming a high amount of resources.

The list of threads that is shown includes the threads that actually perform scanning activities, as well as the threads that deliver requests and responses to the scanning modules. Both kinds of threads are referred to as anti-malware working threads or simply as *AV threads*.

Task

1. Select the Troubleshooting top-level menu.
2. On the troubleshooting tree, select System Tools, then click AV threads.

   A list of the AV threads appears under Results.

   For each thread, an ID number is shown, the time when the thread was started, its current status, and other information.

   Note: To export the thread list to a location within your file system, click Export and specify the location in the window that opens.

Back up and restore an appliance configuration

You can store an appliance configuration in a backup file and use this file to restore the appliance configuration.

When backing up the appliance configuration, you have the option of including the SSO credentials in the credential store in the backup file. Likewise, when restoring, you have the option of restoring the SSO credentials from the backup file to the credential store.

When restoring a backup, you have the option of restoring all configurations and accounts or only the data configured under the Policy top-level menu, which includes data on rules, lists, and settings.

Note: Make sure the UTF-8 character format is used on your administration system for Web Gateway if you want to insert special characters, for example, German umlaut (*ä, ö, ü*), in the passwords that are required for encrypting and decrypting a backup.

Task

1. On the dashboard, select Troubleshooting.
2. On the appliances tree, select the appliance whose configuration you want to back up or restore, then click Backup/Restore.
3. To back up the appliance configuration:
   a. To include SSO credentials in the backup, select the SSO Credentials check box.
   b. Click Back up to file.
   c. In your local file manager, create or select the backup file.

4. To restore the appliance configuration:
   a. To include all configurations and accounts in the restore, select the Configurations and Accounts check box.
   b. To include SSO credentials in the restore, select the SSO Credentials check box.
   c. Click Restore from file.
   d. Confirm the message stating that you will be logged off during the restore.
   e. In your local file manager, select the backup file to use for restoring the appliance configuration.

Reset the appliance password

You can reset the appliance password, which is the root password that is required for working with the command line interface on a system console to access Web Gateway.

Task

1. On the user interface of Web Gateway, select Troubleshooting.
2. On the appliances tree. select the appliance you want to reset the root password on, then click Reset appliance password.
3. Type the new password in the input field that is provided, repeat it in the next field, then click Change Password.

   The password must contain between 8 and 100 characters. It must at least include one upper-case and one lower-case alphabetical, as well as one non-alphabetical character.

   The password is reset.

   The reset is confirmed by a message under Results. If the password could not be reset or other issues occurred, this is also stated in this field.

Results

To verify that the reset was actually performed, connect to the appliance from a system console using SSH. Use root as the user name and the password that you reset when you log on.

Note: You are not allowed to reset the root password in this way if you are running Web Gateway in FIPS mode.

Hybrid solution

A Web Protection license provides all components needed to set up the hybrid solution. When the hybrid solution is configured and enabled, the Web Gateway policy is pushed to McAfee WGCS at the synchronization interval you specify.

Using the hybrid solution, you can:

- Apply one policy across your organization.
- Ensure that the policy is updated in the cloud when it is changed on premise through configured synchronization intervals or manual synchronization.

On-premise and remote web access

The hybrid solution provides web protection for your organization whether users are working inside or outside your local network.

- **On-premise access** — Users who access the web from systems that are physically installed on your local network are working *on premise*. This term extends to users who are connected to your network by a virtual private network (VPN) interface.

  Web Gateway protects your organization when users make web requests while working on premise.
- **Remote access** — Users who access the web from systems that are not connected to your local network, for example, while traveling or working in a public location, are working *remotely*.

  McAfee WGCS protects your organization when users make web requests while working remotely.

Components of the hybrid solution

The hybrid solution integrates McAfee components installed on your network with McAfee cloud services.

Note: If you require a hardware platform to run Web Gateway, the hardware is a separate purchase.

Components of the hybrid solution include:

- Web Gateway — This hardware-based or virtual appliance is installed locally on your organization's network. The on-premise appliance protects your network from threats that might arise when users access the web from inside the network. The appliance has its own interface, where administrators manage the product.
- McAfee WGCS — This cloud service protects your network from threats that might arise when users access the web from inside or outside the network. The service is managed with McAfee ePO Cloud.
- Client Proxy — This software, when installed on the managed endpoints, is aware of the user's location and allows or redirects network traffic, accordingly:
  - Inside the network or connected to the network by VPN — Client Proxy allows network traffic to continue to Web Gateway for filtering.
  - Outside the network — Client Proxy redirects network traffic to McAfee WGCS for filtering.

  Client Proxy can be managed with McAfee ePO, McAfee ePO Cloud, or both depending on the setup.
- Content Security Reporter — This extension, which is managed with McAfee ePO, allows you to view web traffic and usage trends consolidated from Web Gateway and McAfee WGCS logs.
- McAfee ePO — This management platform, which is installed on your network, allows you to manage Client Proxy and Content Security Reporter.
- McAfee ePO Cloud — This cloud-based management platform allows you to manage McAfee WGCS and Client Proxy.

Managed endpoints are the client or user computers in your organization that are managed with McAfee ePO or McAfee ePO Cloud.

How the hybrid components are managed

This table summarizes how the hybrid components are managed with McAfee ePO, McAfee ePO Cloud, or both platforms.

Hybrid component	Managed with McAfee ePO	Managed with McAfee ePO Cloud
Web Gateway	no	no
McAfee WGCS	no	yes
Client Proxy	yes	yes
Content Security Reporter	yes	no

How it works

The on-premise and cloud components of the hybrid solution are set up to protect your organization from threats that might arise when users access the web from inside or outside the network.

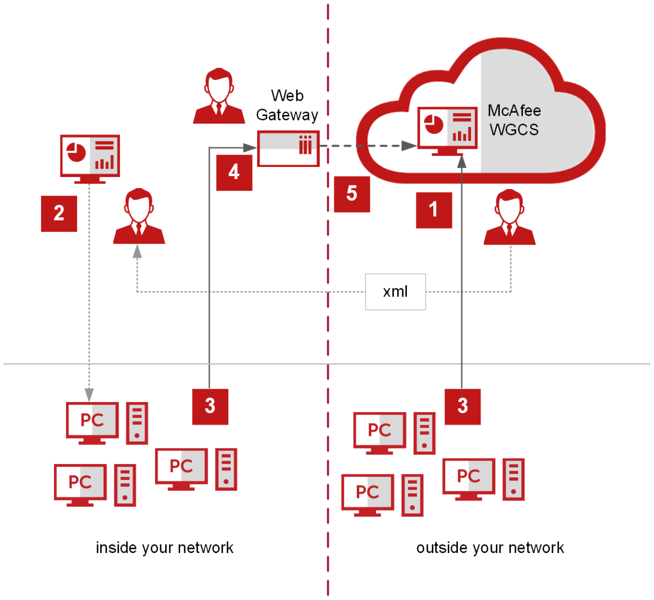
The diagram shows how the key hybrid components are set up and connected. It assumes that Client Proxy is managed with McAfee ePO and that the software is already installed on the McAfee ePO server and the endpoints.

Client Proxy credentials are configured with McAfee ePO Cloud, then exported and shared with McAfee ePO through an .xml file. These steps ensure that the Client Proxy policy is synchronized on premise and in the cloud.

1. From McAfee ePO Cloud, the administrator:
   - McAfee WGCS interface — Configures authentication
   - Client Proxy interface — Configures the shared password and exports the credentials to an .xml file
2. From McAfee ePO, the administrator:
   a. Imports the Client Proxy credentials from the .xml file
   b. Creates a Client Proxy policy for use with the hybrid solution
   c. Assigns the policy to all managed endpoints in the organization

   Note: The administrator can configure multiple policies and assign each one to a different group of managed endpoints.
3. Managed endpoints can be located inside your organization's network, connected to your network by VPN, or located outside your network. A typical Client Proxy policy:
   - **Users working inside your network or connected by VPN** — Allows web requests to continue to a Web Gateway appliance installed on your network
   - **Users working outside your network** — Redirects web requests to McAfee WGCS
4. From the Web Gateway interface, the administrator:
   a. Reviews the web protection policy and enables the rule sets to be pushed to the cloud
   b. Configures and enables the hybrid solution
5. When the deployment is enabled:
   - The Web Gateway policy is pushed to the cloud at the specified synchronization interval.
   - The Policy Browser, where the McAfee WGCS policy is configured in the McAfee ePO Cloud UI, is disabled. Instead, a Policy Unavailable message displays information about the hybrid synchronization, such as the date and time of the last sync.

Note: The hybrid solution doesn't change how the Client Proxy policy is applied. The Client Proxy software installed on the endpoints continues redirecting web requests as before.



Authentication considerations for the hybrid solution

McAfee WGCS authenticates users when they are working outside the network. Authentication settings configured on premise and in the cloud must be compatible.

Group name format

Web Gateway and McAfee WGCS use different formats for the names of user groups. We recommend updating the format of the group names used on premise to match the format used in the cloud. Otherwise, the policy rules configured on premise might apply differently to users when they are working outside the network.

Product	Group name formats used
Web Gateway	DomainName\GroupName GroupName
McAfee WGCS	DomainName\GroupName (recommended)

To make sure that the group names used on premise include the domain name, review the rules and rule sets that are enabled in the cloud. If you are using Client Proxy as the authentication method, configure the Authentication with McAfee Client Proxy rule and select Keep domain name in group name.

User name and user group properties

McAfee WGCS authenticates users when they are working outside the network and assigns values to the user name and user group properties according to the authentication method used. These properties correspond to the *Authentication.UserName* and *Authentication.UserGroups* properties in the Web Gateway interface.

Authentication methods used by McAfee WGCS

Authentication method	The user is authenticated when...	Policy decisions are based on...
Client Proxy	Client Proxy is installed on the managed endpoints, a policy is deployed to the endpoints, and the user sends a web request from an endpoint.	Group memberships returned by Client Proxy

Authentication method	The user is authenticated when…	Policy decisions are based on…
IP range	One or more IP address ranges are configured and the user sends a web request from one of the configured ranges.	Configured IP address ranges in McAfee WGCS
SAML	SAML authentication is configured and the user sends a web request to the SAML service port: 8084.	Group ID attribute value in the SAML assertion
IPsec site-to-site	Web requests are received through the IPsec VPN tunnel configured between your network and McAfee WGCS.	Membership in your organization

Using your own SSL certificate with SAML authentication

In a hybrid deployment, you can use your own SSL certificate with SAML authentication.

When Web Gateway is synchronized with McAfee WGCS in a hybrid deployment, the on-premise policy is pushed to the cloud and the cloud service performs authentication. When the authentication method is SAML, the cloud service encrypts SSL traffic using the McAfee root certificate authority.

To use your own SSL certificate, configure SAML authentication in a Web Gateway policy instead of in McAfee WGCS. Configuration is required in these components:

- Your Identity Provider service
- Web Gateway
- McAfee WGCS

Configuring SAML authentication in your Identity Provider service

When configuring SAML authentication in your Identity Provider service (the external Identity Provider), meet these requirements:

- **Entity ID** — The value you configure in the Identity Provider must exactly match the value you configure in Web Gateway. This setting allows the Identity Provider to uniquely identify Web Gateway as the Service Provider issuing the SAML request.
- **SAML response** — The values configured for the SAML response settings in the Identity Provider must match the values configured in Web Gateway.
- **Assertion Consumer Service (ACS) URL** — Specify this value for the ACS URL:

  ```
  https://saml.saasprotection.com/saml
  ```

  The authentication server performs the Assertion Consumer Service by consuming the SAML assertions that the Identity Provider produces and sends in a SAML response.

Configuring SAML authentication in Web Gateway

To set up a hybrid deployment with SAML authentication configured in a Web Gateway policy, perform these high-level tasks:

- **Enable SSL scanning** — Import your own SSL certificate, configure SSL scanning exceptions, and add the authentication server URL to a whitelist so that incoming SAML traffic is allowed to skip content inspection. The authentication server URL is https://saml.saasprotection.com.
- **Configure SAML authentication for a hybrid deployment** — Import the Cookie authentication with SAML back end and fixed ACS URL (hybrid) rule set from the online Rule Set Library and configure the rules.

Configuring authentication in McAfee WGCS

For a hybrid deployment with SAML authentication configured in a Web Gateway policy, verify that the following authentication settings are configured in the McAfee ePO Cloud UI:

- **IP range or IPsec site-to-site authentication** — Configured and enabled. These methods allow the McAfee WGCS software to associate the source IP address with a customer ID.

- **SAML authentication** — Disabled.

Import SAML rule set for a hybrid deployment

Import the Cookie Authentication with SAML backend and fixed ACS URL (hybrid) rule set from the online library, so that you can configure the rules.

Task

1. Select Policy → Rule Sets.
2. From the Add drop-down list, select Rule Set from Library, then click Online Rule Set Library.
3. Find the rule set Cookie Authentication with SAML backend and fixed ACS URL (hybrid), then download and unzip the .tar.gz file locally on your computer.
   The unzipped file contains: documentation.pdf and ruleset.xml.
4. In the Rule Set Library, click Import from file, browse for the .xml file you downloaded, and click Open.
5. Select Auto-Solve Conflicts → Solve by referring to existing objects, then click OK.
6. Click Save Changes.

Results

The Cookie Authentication with SAML backend and fixed ACS URL (hybrid) rule set is added to the Rule Sets tab.

Whitelist Identity Provider URL for a hybrid deployment

To ensure that the authentication server recognizes the authentication response sent by the external Identity Provider in a hybrid deployment, add the Identity Provider URL to the SAML IdP Whitelist.

Before you begin

Verify that the online rule set Cookie Authentication with SAML backend and fixed ACS URL (hybrid) is added to the Rule Sets tab.

SAML authentication allows you to use your own Identity Provider. The Identity Provider passes the authentication result and identity information to the authentication server in a SAML response that contains SAML assertions.

Task

1. Select Policy → Rule Sets.
2. Expand the nested rule sets Cookie Authentication with SAML backend and fixed ACS URL (hybrid) → Cookie Authentication at HTTP(S) Proxy, then select Authenticate Clients with Authentication Server.
3. Click Show details, then click SAML IdP Whitelist.
4. In the Edit List (Wildcard Expression) dialog box, click the add icon.
   The Add Wildcard Expression dialog box opens.
5. In the Wildcard Expression field, specify an expression that matches the URL of the external Identity Provider.
6. Click OK.
7. Click Save Changes.

Results

The matching expression is added to the SAML IdP Whitelist.

Configure authentication server settings for a hybrid deployment

Configure the authentication server settings for SAML authentication using an external Identity Provider in a hybrid deployment.

Before you begin

Verify that the online rule set Cookie Authentication with SAML backend and fixed ACS URL (hybrid) is added to the Rule Sets tab.

Task

1. Select Policy → Rule Sets.
2. Expand the nested rule sets Cookie Authentication with SAML backend and fixed ACS URL (hybrid) → Cookie Authentication at HTTP(S) Proxy, then select the rule set Authenticate Clients with Authentication Server.
3. Click Show details, then click Hybrid cookie authentication Server to open the Edit Settings dialog box.
4. Verify these settings:
   - Authentication server is selected from the Authentication method drop-down list.
   - The Authentication server URL has this value:

   ```
   https://saml.saasprotection.com
   ```
5. Review and change these settings as needed:
   - Session TTL (IP/cookie) for the authentication server — Specify the amount of time allowed per authentication session.
   - Cookie prefix — Specify a name prefix for the cookie that is set when the user is authenticated.
6. Provide a string value for this setting:
   Password for cookie signature — Specify a password that secures the cookie, which contains the user's identity and authentication information.
7. Click OK.
8. Click Save Changes.

Results

The authentication server is configured for SAML authentication using an external Identity Provider in a hybrid deployment.

Configure the SAML request for a hybrid deployment

Configure the settings that Web Gateway uses when sending SAML requests to an external Identity Provider in a hybrid deployment.

Before you begin

Verify that the online rule set Cookie Authentication with SAML backend and fixed ACS URL (hybrid) is added to the Rule Sets tab.

Task

1. Select Policy → Rule Sets.
2. Expand the nested rule sets Cookie Authentication with SAML backend and fixed ACS URL (hybrid) → Cookie Authentication at Authentication Server, then select the rule set Authentication Server Request.
3. Click Show details, then click the SAML Request instance in the Events column to open the Edit Settings dialog box.
4. Configure the Authn Request settings:
   - EntityID — Specify a name that uniquely identifies Web Gateway as the Service Provider issuing the SAML request. The Identity Provider uses the entity ID to identify SAML requests sent by Web Gateway.
   - IdP URL — Specify the URL of the SAML service provided by your Identity Provider. Web Gateway redirects SAML requests to this URL, which is available from your Identity Provider.
5. Click OK.
6. Click Save Changes.

Results

The SAML request configuration is saved for SAML authentication using an external Identity Provider in a hybrid deployment.

Configure the SAML response for a hybrid deployment

Configure the settings that Web Gateway uses when receiving SAML responses from an external Identity Provider in a hybrid deployment.

Before you begin

Verify that the online rule set Cookie Authentication with SAML backend and fixed ACS URL (hybrid) is added to the Rule Sets tab.

Task

1. Select Policy → Rule Sets.
2. Expand the nested rule sets Cookie Authentication with SAML backend and fixed ACS URL (hybrid) → Cookie Authentication at Authentication Server, then select the rule set Authentication Server Request.
3. Click Show details, then click the SAML Response instance in the Events column to open the Edit Settings dialog box.
4. Configure the Authn Response settings:
   - Response must be signed — If the Identity Provider signs the SAML response, select this checkbox. When selected, Web Gateway verifies the signed SAML response.
   - Assertion must be signed — If the Identity Provider signs the SAML assertion in the SAML response, select this checkbox. When selected, Web Gateway verifies the signed SAML assertion.
   - Import — If the SAML response or assertion is signed, click this button to import the X.509 certificate file provided by the Identity Provider. Web Gateway uses the certificate to verify the signatures of SAML responses and assertions signed by the Identity Provider.
   - EntityID — Specify the name that uniquely identifies the Identity Provider issuing the SAML response. Configure this setting if your Identity Provider uses an entity ID. When configured, Web Gateway uses this value to identify the SAML responses sent by your Identity Provider.
5. Click OK.
6. Click Save Changes.

Results

The SAML response configuration is saved for SAML authentication using an external Identity Provider in a hybrid deployment.

Configure user name and user groups for a hybrid deployment

Configure the names of the attributes, returned by the Identity Provider, that you want mapped to the Authentication.UserName and Authentication.UserGroups properties in Web Gateway.

Before you begin

Verify that the online rule set Cookie Authentication with SAML backend and fixed ACS URL (hybrid) is added to the Rule Sets tab.

If you are using Microsoft ADFS as your Identity Provider, you do not need to configure these attribute names. By default, Microsoft ADFS attribute names *userID* and *userGroup* are mapped to the Authentication.UserName and Authentication.UserGroups properties, respectively.

Task

1. Select Policy → Rule Sets.
2. Expand the nested rule sets Cookie Authentication with SAML backend and fixed ACS URL (hybrid) → Cookie Authentication at Authentication Server, then select the rule set Authentication Server Request.
3. Locate the Set user name and groups rule and double-click the rule in the Events column.
   The Edit Rule dialog box opens at the Events step.
4. Select the Authentication.UserName property, then click Edit. The Edit Set Property dialog box opens.
   a. Select the string on the right, then click Edit.
      The Enter a String dialog box opens.
   b. Click Parameters.
      The Parameters for Property "Map.GetStringValue" dialog box opens.

   c. Select 2. Key (String) on the left, then replace the existing string on the right with the name of the attribute that you want mapped to the Authentication.UserName property.
   d. Click OK → OK → OK.
5. Select the Authentication.UserGroups property, then click Edit. The Edit Set Property dialog box opens.
   a. Click Parameters on the right.
      The Parameters for Property "String.ToStringList" dialog box opens.
   b. Click Parameters on the right again.
      The Parameters for Property "Map.GetStringValue" dialog box opens.
   c. Select 2. Key (String) on the left, then replace the existing string on the right with the name of the attribute that you want mapped to the Authentication.UserGroups property.
   d. Click OK → OK → OK.
6. Click Finish.
7. Click Save Changes.

Results

The attribute names that you configured are mapped to the Authentication.UserName and Authentication.UserGroups properties.

Why some rules are restricted

Not all Web Gateway rules are compatible with McAfee WGCS.

Rule properties and events

Some rule properties and events cannot be used with McAfee WGCS. For example, events that increase counters, send emails to an administrator, or write entries to log files are not compatible with the cloud service.

In general, policy elements cannot be used in a hybrid solution if they:

- **Rely on particular network components or external services** — For example, mail services require a mail server, such as an SMTP server, SNMP properties require a trap sink, and next-hop proxy properties require next-hop proxy servers.
- **Relate to functions that McAfee WGCS does not support** — Examples include quota management and PDStorage functions and functions that require the exchange of runtime data, such as, between multiple Web Gateway appliances in a Central Management configuration.

In the Web Gateway interface, warning messages flag rules that are restricted.

- **Properties** — Pay attention to warnings about properties when they are set to their default values. These properties are not compatible with the hybrid solution and sometimes result in unexpected behavior.
  Tip: Best practice: Verify that rules having a property that is flagged with a warning work as expected when synchronized with McAfee WGCS.
- **Events** — You can ignore warning messages about events, because they are not executed and have no effect on the hybrid solution.

Sample warnings

The following sample warnings show rules that are restricted, because they contain a property or event that cannot be used with McAfee WGCS.

- **A property is incompatible with McAfee WGCS** — A warning is displayed, which identifies the rule and the property. It also contains a sentence like the following:

  ```
  Property PDStorage.GetAllData must not be used in SaaS.
  ```
- **An event is incompatible with McAfee WGCS** — A warning is displayed, which identifies the rule and the event. It also contains a sentence like the following:

  ```
  Event SNMP.Trap.Send.User(Number, String) must not be used in SaaS.
  ```

Identifying rule sets not supported in the cloud

Not all Web Gateway rule sets are compatible with the cloud. Incompatible rule sets can't be enabled in the cloud and synchronized with McAfee WGCS.

To identify which rule sets aren't supported in the cloud:

- View the rule sets in the Web Gateway interface — Select Policy → Rule Sets, then select an individual rule set. If the Enable in Cloud checkbox in the configuration pane is grayed out, the rule set isn't supported in the cloud.
- See the list of properties under *Configuration lists* in the *McAfee Web Gateway Interface Reference Guide* — Any rule sets that use properties identified as *not SaaS-compatible* are not supported in the cloud.

Enable rule sets for hybrid synchronization

You must enable the Web Gateway rule sets that you want synchronized with McAfee WGCS.

Before you begin

Review the rule sets and decide which ones to enable in the cloud. The default rule sets provide all rules needed for the hybrid solution.

When you enable a rule set for hybrid synchronization, the rule set view determines whether nested rule sets are also enabled in the cloud.

- **Key elements view** — Nested rule sets are enabled.
- **Complete rules view** — When the rule set is enabled from the context menu, nested rule sets are enabled too. When the rule set is enabled in the configuration pane, nested rule sets are not enabled and must be enabled individually.

Task

1. In the Web Gateway interface, select Policy → Rule Sets.
2. For each rule set that you want synchronized with McAfee WGCS, select it, then select Enable in Cloud.
3. Click Save Changes.

Results

The selected rule sets are enabled for synchronization with the cloud.

Configure and enable the hybrid solution

Configure the connection with McAfee WGCS and the synchronization interval.

Before you begin

The hybrid components are set up.

The Web Gateway rule sets that you want synchronized with McAfee WGCS are enabled in the cloud.

You have your McAfee ePO Cloud credentials and your McAfee WGCS customer ID.

Caution:

After hybrid synchronization is enabled in the Web Gateway interface, the web protection policy in the cloud is overwritten. To disable synchronization and restore the default McAfee WGCS policy, you must contact McAfee Technical Support.

Task

1. In the Web Gateway interface, select Configuration → Appliances.
2. On the Cluster branch of the appliances tree, click Web Hybrid.

   The hybrid settings open in the configuration pane.
3. Configure the settings as needed.
4. Click Save Changes.

Results

Hybrid synchronization is enabled, and the Web Gateway policy is pushed to McAfee WGCS at the specified synchronization interval or manually.

Verify that policy synchronization succeeded

To verify that the hybrid solution is correctly configured and that policy synchronization succeeded, you can perform the synchronization manually.

Task

1. In the Web Gateway interface, select Troubleshooting, then under the name of the appliance, select Synchronization to Cloud.
2. In the expanded list, select Synchronization to Cloud.
3. In the Synchronization to Cloud pane, click Synchronize.

Results

This message is displayed: *Policy synchronization successfully performed!*

Add hybrid information to a block page

You can add information to a block page that shows whether the on-premise appliance or cloud service blocked the user's request.

In a hybrid deployment, McAfee WGCS shares the block pages that are configured in the Web Gateway interface. Using the property *InTheCloud*, you can add information to a block page to show whether the blocked request was filtered on premise or in the cloud. This property returns a true value when McAfee WGCS filters and blocks the request.

In this task, you edit the default block page template named URL Blocked. The default template includes the title Blocked by URL Filter Database and the following standard text. Actual values for the properties in the template are written to the block page when it is generated.

> Your requested URL has been blocked by the URL Filter database module of McAfee Web Gateway. The URL is listed in categories that are not allowed by your administrator at this time.
>
> **URL:**
> **URL Categories:**
> **Reputation:**
> **Media Type:**

In the following steps, you add a line to the block page after Media Type using the suggested text or custom text that you specify.

Task

1. In the Web Gateway interface, select Policy → Templates.
2. Expand the template folders Default Schema → URL Blocked → en.
3. Click html.

   The HTML Editor opens and displays the contents of the URLBlocked.html file.
4. In the file, locate the line that begins: `<b>Media Type: </b>`.
5. Immediately following this line, add these lines:

```
<b>Filter: </b> <script type="text/javascript"> if ($InTheCloud$) { writeToDocument("McAfee Web Gateway Cloud
Service"); } else { writeToDocument("McAfee Web Gateway"); } </script>
```

6. Click Save Changes.
7. To view the output, click Preview.

The preview opens in a new tab. A line labeled **Filter** is added after the line labeled **Media Type**.

Tip: To view the text in the preview that is written to the block page when it is generated, you can replace the property `($InTheCloud$)` with the value `(true)` or `(false)`. Click Save Changes, then click Preview. Depending on the value of *InTheCloud*, one of these lines is displayed.

Filter: McAfee Web Gateway Cloud Service

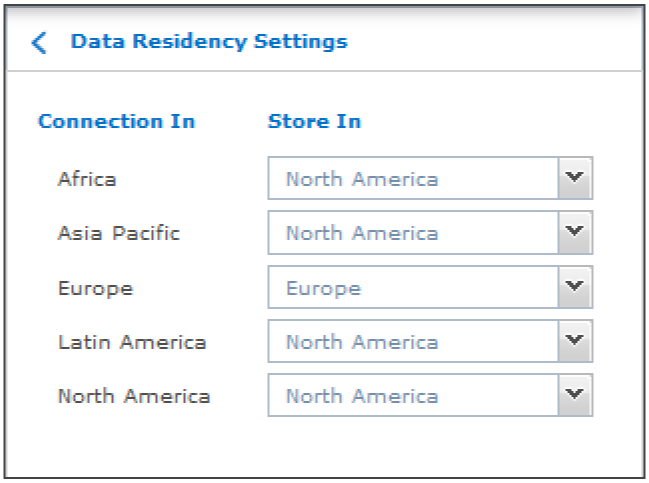
Filter: McAfee Web Gateway

Hybrid data residency settings

You can override McAfee WGCS data residency settings by adding a data residency event to Web Gateway policy rules. The event takes effect only when the hybrid solution is enabled.

McAfee WGCS data residency settings

This McAfee WGCS screenshot shows the connection regions, where your organization connects to McAfee WGCS, and the available storage regions, North America and Europe. In the default configuration, all web access data is stored in North America except when Europe is the connection region.



Data Residency Settings

Connection In	Store In
Africa	North America
Asia Pacific	North America
Europe	Europe
Latin America	North America
North America	North America

Hybrid data residency setup

You might want some web access data filtered in Europe to be stored in North America. For example, you might want to store data in North America when web requests are sent from users in specified user groups or from specified client IP addresses. To set this up in the Web Gateway interface:

1. Add settings to the Cloud Access Log Data Residency module — For example, you could add settings named `NA` and `EU` for the regions, North America and Europe, respectively.
2. Add the event CloudLogging.SetStorageRegion with NA selected for the storage region to the rule that tests for the condition you want met — In the complete rules view, the event is displayed as CloudLogging.SetStorageRegion<NA>.

When the condition is met, the rule applies, the event is executed, and the web access data for that request is stored in North America.

While the Web Gateway data residency event overrides the McAfee WGCS settings when hybrid is enabled, the McAfee WGCS settings still apply when no hybrid data residency event is executed.

Hybrid data residency example: group membership

When the user belongs to at least one user group in the specified list, the web access data for that request is stored in North America.

Rule element	Definition
Criteria	Authentication.UserGroups at least one in list Specified_User_Groups
Action	Continue
Event	CloudLogging.SetStorageRegion<NA>

Hybrid data residency example: client IP address

When the client IP address is included in the list of specified IP address ranges, the web access data for that request is stored in North America.

Rule element	Definition
Criteria	Client.IP is in range list Specified_Client_IPs
Action	Continue
Event	CloudLogging.SetStorageRegion<NA>

Cloud single sign-on

Cloud single sign-on (SSO) is the Web Gateway service that allows users in your organization to access cloud services and applications after providing credentials one time. The SSO service is implemented by the Single Sign On module.

In the context of cloud single sign-on, unless otherwise noted, the following terms are used as described here:

- Service Provider — The organization that provides the cloud service or application
- Users — Members of your organization who seek access to cloud services and applications

  Using the launchpad provided by Web Gateway, users submit credentials, open applications, and manage their accounts in the applications.
- User interface — The Web Gateway interface where administrators configure the SSO service
- Cloud connector — The configuration that allows Web Gateway to connect to and provide identity and SSO services for an application or service in the cloud
- Predefined connector — Any cloud connector that comes fully configured with Web Gateway
- Custom connector — Any cloud connector configured from a template

  Web Gateway provides a range of connector templates. Some templates come with most, but not all, configuration built in. Other templates allow you to build cloud connectors from scratch.

Note: The terms cloud service and cloud application are used interchangeably.

How cloud single sign-on is configured

At a high level, you configure cloud single-sign by adding predefined and custom cloud connectors to SSO Connector lists. You can then associate users with these lists through Web Gateway policies.

SSO tasks require the Single Sign On rule set, which you import from the Rule Set Library. All SSO tasks can be performed using the default rules, settings, and lists visible in the key elements view of this rule set. To view the rules making up the rule set and create rules, settings, and lists of your own, unlock the key elements view.

Note: Configuring single sign-on to some cloud services and applications requires configuration on the Service Provider side. Create an account in the Service Provider interface and complete the configuration steps there.

Task

1. Configure a method for authenticating users.
2. Import the Single Sign On rule set from the rule sets library and configure the rules.

   Note: The Single Sign On rule set is located in the Cloud Services rule set group. You can configure the rules in the key elements view or click Unlock View to view configuration details and configure rules of your own.
3. Configure the Single Sign On settings for the Single Sign On module, which retrieves values and parameters for SSO properties and events in the Single Sign On rule set. This module comes with default settings named Default. In the SSO rules, properties and events that require these settings reference them using the notation <Default>. You can change the default settings or create new settings.

   Note: To locate these settings, select Policy → Settings → Engines → Single Sign On → Default.
4. For single sign-on to SAML and IceToken cloud services, configure an X.509 certificate and private key pair.

   Note: To locate these settings, select Policy → Settings → Engines, then select SSO Certificates or SSO Private Keys, respectively.
5. Using SSO lists, you can configure custom cloud connectors from templates and lists of connectors to cloud services that users are allowed to access.

   - SSO Host to Service ID mapping — (Optional) Lets you map a name that is easy to remember (host name) to the Service ID of a configured custom connector.

     Note: To locate this list, select Policy → Lists → Custom Lists → MapType.
   - SSO Connector — Lets you configure lists of connectors to services that users are allowed to access. You can add connectors to the default lists that come with the SSO service or create and configure lists of your own.

     Note: To locate the SSO Services lists, select Policy → Lists → Custom Lists → SSO Connector.

- SSO Catalog — Lets you view the predefined connectors and the custom connectors configured from templates. You can configure new connectors from templates, then view them in the Custom connectors list.
  Note: To locate the catalog, select Policy → Lists → System Lists.

6. We recommend that you secure all launchpad communication with the HTTPS protocol. To do so, configure the Launchpad certificate settings used by the SSL Client Context without CA module, which handles certificates for SSL-secured communication.
   Note: To locate the Launchpad certificate settings: In the key elements view, locate SSL Scanner settings, then click Edit.
7. To secure communication between Web Gateway and all cloud services with the HTTPS protocol, configure the SSL Scanner module settings. This step is required for proxy mode.
8. To require OTP authentication for SSO access to cloud services, enable OTP authentication, configure the OTP server settings, select an OTP delivery method, and configure the list of connectors to services that require OTP authentication.
   Note: To locate these settings: In the key elements view, see the OTP Usage (One Time Passwords) section.
9. To log SSO requests to the SSO access log instead of the general access log, enable SSO logging. To enable detailed logging for debugging purposes, enable SSO trace logging.
   Note: To access these settings, select Policy → Rule Sets → Log Handler, then import the SSO Log rule set from the Logging rule set group in the Rule Set Library.
10. Save the changes.

Single Sign On rule set summary

You configure and manage single sign-on through the Single Sign On rule set as well as related lists and settings.

The Single Sign On rule set comes with a default configuration that you can use and modify. When you first import and select the rule set, the default configuration opens in the simpler, locked view. You can configure and manage single sign-on using the locked view alone.

To access the more advanced view of the rule set, you unlock the view. If you unlock the view and find that you prefer the simpler, locked view, you cannot undo this action. To go back to the simpler, locked view, you must delete the rule set and import it again.

In the unlocked view of the default configuration, the nested rule sets are arranged and processed in the following order. Unless noted, all rule sets are enabled by default.

1. Select Services — Rules in this rule set add services to an internal map that determines whether the current user has access to the requested cloud service. The services are added from default lists that you configure.
2. SSO Management — This rule set contains the nested rule sets that manage single sign-on.
3. Perform SSO — This rule set contains the rule that processes the logon form.

The SSO Management rule set contains the following nested rule sets. They are arranged and processed in the order shown.

1. HTTPS Handling — Rules in this rule set secure all launchpad communication using the HTTPS protocol.
2. Launchpad — Rules in this rule set generate the application launchpad and logon page using the Single Sign On module settings.
3. OTP Authentication — Rules in this rule set enforce OTP authentication as a secondary authentication method. This rule set is disabled by default.
4. Get Login Action — This rule set retrieves information about the connector to the service that the user is requesting. For HTTP services, rule set processing stops. For other services, the rule set checks whether the user has the right to access the requested service.
5. Process Common Tasks — This rule set processes common SSO tasks using the Single Sign On module settings. It also contains the rule that blocks access to SSO resources that do not exist.

The Get Login Action rule set contains the following nested rule sets. They are arranged and processed in the order shown.

1. Get Attributes on Premise — Rules in this rule set fetch user information from an external LDAP data source for SAML single sign-on. The rule set only applies when Web Gateway is installed and running on premise.

2. Get Attributes in the Cloud — This rule set constructs the data needed for SAML single sign-on from the authenticated user name. It only applies when Web Gateway is installed and running in the cloud.
3. Perform SAML SSO — This rule set generates a response that contains the user information needed for completing single sign-on to the requested SAML service.
4. Perform IceToken SSO — This rule set generates a response that contains the user information needed for completing single sign-on to the requested service using the custom IceToken Web Gateway provides.

Considerations when exporting and importing the SSO rule set

The SSO rule set export and import does not include the SSO credentials required for accessing HTTP cloud applications or the Service IDs of custom connectors.

SSO credentials (HTTP applications)

The SSO rule set is stored in the policy database. Importing the rule set updates the SSO policy. When you export or import the SSO rule set, the following information is included:

- All configured cloud connectors
- All configured connector lists
- All configured X.509 certificates and private key pairs

SSO credentials, which are required for accessing HTTP cloud applications and services, are stored in a separate database and are not part of the SSO policy. These credentials are not included in the export or import and must be re-created after the SSO rule set is imported.

When you back up the appliance configuration, you can include the SSO credentials in the backup. In this case, restoring the backup also restores the credentials.

Service IDs (Custom connectors)

The Single Sign On module assigns numeric Service IDs to custom connectors at the time they are created from templates. These Service IDs are not included in the export of the SSO rule set. When the rule set is imported later, new Service IDs are assigned to the custom connectors.

After importing the SSO rule set, you must update any Service IDs that are used to reference custom connectors, as follows:

- In the SSO Host to Service ID Mapping list, update the Key values to match the new Service IDs.
- Some Service Providers, such as Gmail, include the Service ID in the SSO configuration. For these Service Providers, log on to your account and update the Service ID.

Caution: Failure to update the Service IDs after importing the SSO rule set can break custom connectors and links to cloud services and applications.

SSO process in proxy and non-proxy modes

The steps in the SSO process depend on whether the user's credentials are submitted to the cloud application directly (non-proxy mode) or through Web Gateway (proxy or inline mode).

In proxy and non-proxy modes, Web Gateway authenticates the user, then presents the launchpad. The launchpad displays icons corresponding to the cloud applications the user is allowed to access. The SSO process appears the same to the user in both modes:

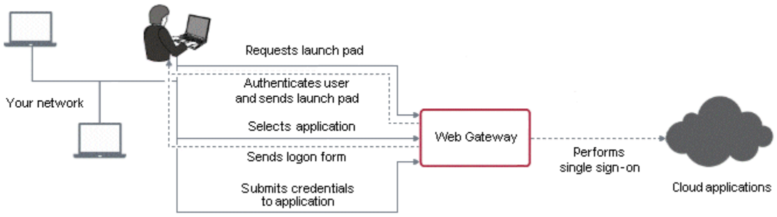
1. From a web browser on a client of Web Gateway, the user requests a launchpad.
2. After authenticating the user, Web Gateway sends a launchpad.
3. To open an application, the user clicks the icon corresponding to the application on the launchpad.
4. Web Gateway sends a logon form to the user.

5. If requesting access for the first time, the user is prompted for credentials, which the user provides and submits to Web Gateway. If requesting access for a second or later time, the logon form is automatically filled with the user's credentials and submitted to Web Gateway.
6. If the credentials are valid, the user is allowed SSO access to the cloud application.

Proxy mode

In proxy mode, Web Gateway forwards the user's credentials to the cloud application.

Single sign-on in proxy mode



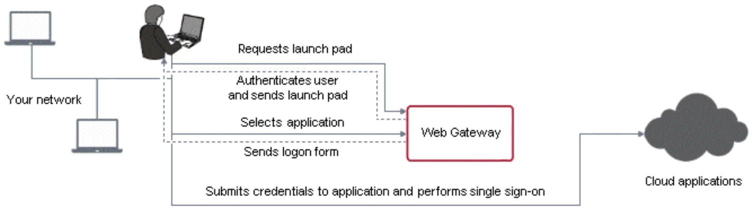
When single sign-on takes place in proxy mode, Web Gateway can provide additional functionality that is not available in non-proxy mode:

- **Dynamic cloud applications** — Web Gateway can support HTTP cloud applications that provide logon page information dynamically, such as DropBox, by adding Javascript to the logon page. The Javascript completes the fields on the page with information.
- **Encrypted password** — The password is encrypted and hidden from the client computer.

Non-proxy mode

In non-proxy mode, the user's browser forwards the credentials to the cloud application.

Single sign-on in non-proxy mode


Your network
Requests launch pad
Authenticates user and sends launch pad
Selects application
Sends logon form
Submits credentials to application and performs single sign-on
Web Gateway
Cloud applications

Note: When single sign-on takes place in non-proxy mode, Web Gateway functions as a web server. When configuring your Domain Name Service and all SSO settings, you must use the IP address of the Web Gateway appliance in place of a host name.

Supported authentication methods

Generally, each cloud service or application uses one authentication method to log on users.

Web Gateway provides SSO services for many cloud applications that use HTTP or SAML 2.0 authentication through individual cloud connectors. Web Gateway also provides SSO services for cloud applications using a proprietary authentication method through a custom token named IceToken.

SSO data sources

The data source from which Web Gateway obtains the user's credentials or information depends on whether single sign-on is to an HTTP or SAML service.

- **HTTP services** — Web Gateway uses an integrated credential store: a secure database that stores credentials like the user names and passwords required by HTTP services. Users who seek access to an HTTP service must first authenticate against the database.
- **SAML services** — Web Gateway retrieves identity information from an external data source and produces a SAML assertion attesting to the user's identity.

Viewing the SSO Catalog

The user interface provides the most complete and up-to-date view of the SSO Catalog.

The SSO Catalog consists of the cloud applications and services supported by Web Gateway with cloud connectors. It includes predefined connectors, connector templates, and custom connectors configured from the templates.

The catalog is implemented as a system list. Like other system lists, it is updated and released between major Web Gateway releases. Changes are delivered by update servers and can be viewed in the user interface. New connectors are added, and when possible, broken connectors are fixed. Connectors that are no longer supported are highlighted and the change is noted.

The SSO Catalog system list consists of these connector lists:

- Predefined connectors — These connectors come fully configured with Web Gateway and only need selecting from the catalog.
- Custom connectors — These connectors are configured from templates that come with some, but not all, configuration built in. Custom connectors require configuration before they can be added to the catalog and selected.

In the user interface, predefined connectors and connector templates are organized by the names of the cloud applications and services they support. Custom connectors configured from connector templates are organized by the names that you specify.

Each connector configuration is saved in a file that includes information like the following:

- Information about the cloud service, such as name and category
- URLs needed for the SSO process
- Pages containing logon forms
- Data for generating the launchpad

SSO Catalog in the user interface

Predefined and custom connectors are listed in table format. While the tables include the same information, the details differ for each type of list.

Note: Predefined connector values are provided by the Single Sign On module and cannot be changed. Custom connector values, which administrators configure, can be changed.

Column heading	Description
Icon	Displays the logo that represents the cloud application or service. When configuring a custom connector, you can specify a custom image.
Name	Uniquely identifies the predefined connector or custom connector instance. • Predefined connectors — Displays the name of the cloud application or service and can include spaces. Example: Air Canada • Custom connectors — Displays the name that you configure for each connector instance. **Note:** From one connector template, you can configure multiple connector instances. For example, you can

Column heading	Description
	configure one connector instance for each user group and assign the instances different names, as follows: ◦ Google Calendar - IT ◦ Google Calendar - Sales
Description	(Custom connectors) Allows you to provide a description for each connector instance.
Categories	Specifies the type of service provided by the cloud application or service. When configuring a custom connector, you can change the default category or create a new one. Examples: Collaboration, Marketing, Social
Service ID	• Predefined connectors — Specifies a name that uniquely identifies the predefined connector in the SSO Catalog. Typically, the service ID is the same as the name with the spaces removed. Example: AirCanada • Custom connectors — Specifies a number that uniquely identifies the custom connector in the SSO Catalog. This number is set by the Single Sign On module and cannot be changed.
Types	Specifies the method that each cloud application or service uses to authenticate users. Sometimes, applications and services are referred to by type, such as an HTTP application or a SAML service. This value is set by the Single Sign On module.

SSO Catalog as a service

The SSO Catalog is a cloud service. As a cloud service, it is updated between Web Gateway releases.

Note: The SSO Catalog is also known as the Connector Catalog as a Service (CCaaS).

Occasionally, a Service Provider changes the configuration details required for connecting to a cloud service or stops providing a cloud service altogether. These changes, which can break the connector to a service temporarily or permanently, require changes to the SSO Catalog. Changes to the SSO Catalog, including new and fixed connectors, are delivered by update servers.

When the catalog is updated, an update message is displayed in the Web Gateway user interface. Broken connectors for which no resolution is planned are no longer supported and are flagged in the user interface as follows:

- **SSO Catalog**
  - **Connectors** — Predefined and custom connectors that are no longer supported are available for selection in the catalog. But they are flagged with a yellow triangle and No longer supported message.
  - **Templates** — Templates for custom connectors that are no longer supported are available for selection in the catalog and can be configured. But they are flagged with a yellow triangle and No longer supported message.
- **SSO Connector lists** — SSO Connector lists are custom lists of connectors to cloud services that users are allowed to access. Connector lists containing connectors that are no longer supported are highlighted in yellow. Connectors in connector lists that are no longer supported are highlighted in yellow and flagged with a No longer supported message.

Finding information about the latest release of the SSO Catalog

To find information about the latest release of the SSO Catalog, see the following articles in the McAfee Knowledge Center.

Knowledge Base article	Description
KB82351	Lists the new connectors, renamed connectors, and connectors that are no longer supported in the latest release of the SSO Catalog. This article also lists the connectors which are not supported when accessed using the specified versions of Internet Explorer.
KB82379	Lists the connectors having known issues in the latest release of the SSO Catalog.

Generic vs. individual connector templates

Generic cloud connector templates support any cloud application that uses the specified authentication method. Because generic templates are more flexible than individual connector templates, they require more configuration.

Individual connector templates

Individual cloud connector templates provide the basis for configuring a connection to a specific cloud application. For example, the Salesforce connector template allows you to configure a custom connection to the Salesforce application in the cloud.

Because templates are configurable, you can create multiple custom connectors to a single cloud application such as Salesforce. To identify custom connectors, you assign them unique names.

Generic connector templates

Generic cloud connector templates allow you to configure a connection to any cloud application that uses the specified authentication method. For example, using the Generic HTTP Connector template, you can configure a connection to any cloud application that uses HTTP authentication to log on users. Generic templates allow you to configure connectors to cloud applications not found in the SSO Catalog.

Web Gateway provides generic cloud connector templates for the following authentication methods.

- **Generic HTTP connector** — Select this template when you want to configure a connector to an HTTP service that Web Gateway does not support with an individual connector.
- **Generic SAML2 connector** — Select this template when you want to configure a connector to a SAML 2.0 service that Web Gateway does not support with an individual connector.
- **Generic IceToken connector** — Select this template when you want to configure a connector to a service that uses an authentication method which Web Gateway does not support.

Configure a custom cloud connector using a template

After you configure a connector to a cloud service from a template, your users can access the service after authenticating one time.

Task

1. Select Policy → Lists.
2. In the Lists tree, expand System Lists → SSO Catalog, then click Custom connectors.
3. Click the Add icon.

   The Add Connector dialog box opens.
4. Provide values for the following fields and settings:
   - Name — Specifies a name that uniquely identifies the cloud connector instance.
   - Description — (Optional) Describes the cloud connector instance.
   - Template — Allows you to select the template corresponding to the cloud service where you want to configure SSO access.

Template-specific settings open.

- Categories — Specifies the type of service provided by the cloud service or application. When you select the template, a default value is loaded automatically. You can change this value by clicking Choose.
- Browse — Allows you to add or change the logo that represents the cloud connector you are creating.

5. Configure the template-specific settings.
6. Click OK.

Results

The newly configured cloud connector is added to the SSO Catalog. To view the connector in the catalog, select Custom connectors.

Delete a custom cloud connector

You can remove a custom cloud connector from the SSO Catalog if it is not included in any SSO Connector list. Custom cloud connectors are connectors configured from templates.

Caution: Removing a custom cloud connector from the SSO Catalog removes all user credentials entered for that connector. Re-creating the connector with the same settings does not restore the credentials that were lost when the connector was removed.

Task

1. Select Policy → Lists.
2. In the Lists tree, expand System Lists → SSO Catalog, then click Custom connectors.
3. Select the custom cloud connector you want to delete, then click the Delete icon.
   The Confirm deletion dialog box opens.
4. To confirm the deletion, click Yes.
   The custom cloud connector is removed from the SSO Catalog.

Managing cloud access through SSO Connector lists

Access to cloud services and applications is managed through lists of cloud connectors, each connector corresponding to a supported service in the SSO Catalog.

Some SSO Connector lists enable access. Other lists might require OTP authentication before access is permitted. Users are associated with lists through Web Gateway policies.

Managing access to cloud services through SSO Connector lists involves these high-level steps:

1. (Custom connectors) Configure cloud connectors to the cloud services you want users to access and add them to the SSO Catalog.
2. From the SSO Connector list that you are configuring, select the predefined and custom cloud connectors you want added to the list.

Add a cloud connector to an SSO Connector list

To control access to a cloud service, locate the corresponding cloud connector in the SSO Catalog and add it to an SSO Connector list.

Task

1. Select Policy → Lists.
2. In the Lists tree, expand Custom Lists → SSO Connector, then click the list you want to modify.
3. Click the Edit symbol.

A dialog box opens showing folders, each folder holding connectors in the specified category.

Example: Travel & Transportation

4. To add connectors to the list, select them individually or by category, then click OK.

   Note: If the connector you want does not exist, you can create it by clicking Create new.
5. Click Save Changes.

Providing SSO services for HTTP cloud applications

Web Gateway supports many cloud services and applications that use HTTP authentication to log on users with predefined cloud connectors or individual cloud connector templates.

A cloud connector is the configuration that allows Web Gateway to connect to and provide identity and SSO services for an application in the cloud. Web Gateway also provides a generic HTTP connector template, which can be configured for any cloud application that uses HTTP, but is not included in the SSO Catalog.

Before configuring a connector to an HTTP application, look up the application in the SSO Catalog. Predefined HTTP connectors come fully configured and only need selecting from the catalog. If the connector you want does not exist in the Predefined connectors or Custom connectors lists, you can create it from a template.

Most templates are partially configured connectors to specific cloud applications. If no template exists for your HTTP application, select the Generic HTTP Connector template. The generic HTTP template lets you configure connectors to HTTP applications that Web Gateway does not support with predefined connectors or connector templates.

Web Gateway supports single sign-on to dynamic HTTP applications that provide logon page information dynamically, such as Dropbox, by adding JavaScript to the logon page. Before the logon page can be changed, the SSO process must be running in proxy mode. In proxy mode, Web Gateway hides the real password from the client computer by replacing it with a token.

Note: Single sign-on to HTTP applications that are not dynamic can be implemented in proxy or non-proxy mode.

The SSO credential model for HTTP cloud applications

The SSO credential model for HTTP cloud services and applications supports individual users who have more than one account in a cloud service or application. It also supports shared accounts, where multiple users can access one or more cloud services or applications using the same credentials.

The following credential information is passed to most SSO properties and events:

- **Realm** — Specifies the name of the domain in which the current user is authenticated. The authentication domain can be an identity store, such as LDAP or Active Directory, or an authentication service.
- **User ID** — Identifies the current user. By default, the User ID has the same value as the Authentication.UserName property. You can change the default value by mapping a different authentication attribute to the User ID.
- **Service ID** — Identifies a cloud service or application.
- **Account ID** — Identifies an individual or shared account in the cloud service or application.

Individual users are organized under realms or authentication domains. Users in an authentication domain are associated with one or more lists of cloud services or applications that they are allowed to access. For each cloud service or application, each user can have one or more accounts. The accounts can be individual accounts or shared.

Configure an HTTP cloud connector

Configure a connector to an HTTP service or application using a template.

Task

1. Select Policy → Lists.

2. In the Lists tree, expand System Lists → SSO Catalog, then click Custom connectors.
3. Click the Add icon.
   The Add Connector dialog box opens.
4. Provide values for the fields and settings common to all cloud connectors.
5. From the Template drop-down list, select the template corresponding to the HTTP service.
6. In the Application Domain Name field, specify the domain name of your instance of the HTTP service or application.
   Example: If your service URL is https://myorg.cloudapp.com, *myorg* is the name of your application domain.
7. Click OK.

Results

The newly configured HTTP connector is added to the SSO Catalog → Custom connectors list.

Configure a generic HTTP cloud connector

Configure a generic HTTP cloud connector when you want to connect to an HTTP service that Web Gateway does not support with an individual connector.

Task

1. Select Policy → Lists.
2. In the Lists tree, expand System Lists → SSO Catalog, then click Custom connectors.
3. Click the Add icon.
   The Add Connector dialog box opens.
4. Provide values for the fields and settings common to all connectors.
5. From the Template drop-down list, select Generic HTTP Connector.
6. To configure a connector to a dynamic HTTP cloud service, select Dynamic service.
7. From the drop-down list, select the HTTP method that specifies how the form is sent.
8. In the https:// field, specify where to send the form in URL format.
9. For each attribute sent in the form, configure one form field.
10. For each form field whose source is the credential store, configure one launchpad field.
11. (Optional) Configure one or more logon pages.
    Note: Dynamic HTTP cloud services require one logon page. Some cloud services require more than one logon page.
12. (Optional) Configure the fields on the logon page.
    Note: You only need configure the logon fields when they are different from the form fields.
13. To configure another generic HTTP connector, click New Sign On Request.
14. To save the HTTP connector configuration, click OK.

Results

The newly configured generic HTTP connector is added to the SSO Catalog → Custom connectors list.

Providing SSO services for SAML 2.0 cloud applications

Web Gateway supports cloud services and applications that use SAML 2.0 authentication to log on users by providing cloud connector templates.

A cloud connector is the configuration that allows Web Gateway to connect to and provide identity and SSO services for an application or service in the cloud. Web Gateway provides connector templates for many individual cloud services and applications. Web Gateway also provides a generic SAML2 connector template. The generic template can be configured for any cloud service or application that uses SAML 2.0, but is not included in the SSO Catalog.

Note: Configuring single sign-on for a SAML 2.0 cloud application requires configuration in your SAML 2.0 application administrator account and in the Web Gateway user interface.

How SAML single sign-on is initiated

The SAML SSO process is initiated by the Identity Provider (IdP) or the Service Provider (SP).

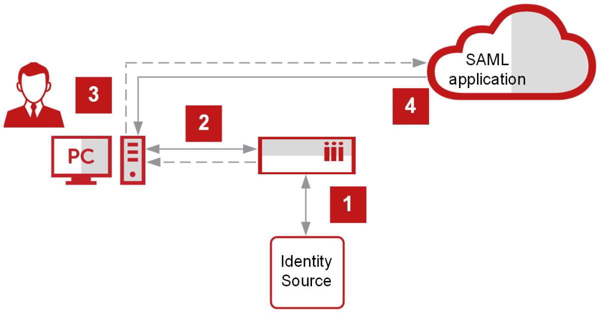
The Identity Provider is the service that authenticates the user. The Service Provider is the SAML cloud service or application that the user wants to access. In the following examples of SAML single sign-on, Web Gateway performs the Identity Provider role.

The SSO process begins when the user requests access to a SAML application in the cloud. Web Gateway authenticates the user, then redirects the authentication result to the SAML application through the user's browser. The redirected messages are automatic and take place quickly, so that the user is not aware of the authentication process running in the background.

IdP-initiated SAML single sign-on

Web Gateway initiates the SSO process, which consists of the following overall steps:

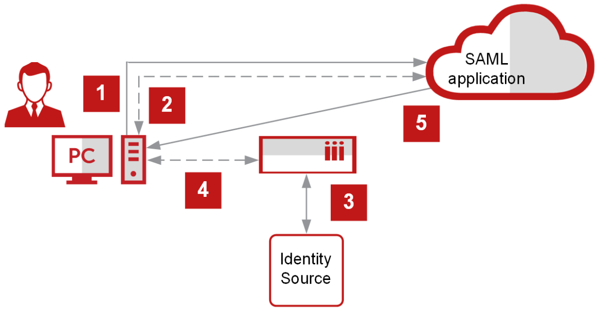
1. Web Gateway authenticates the user.
2. Web Gateway presents the user with a launchpad that includes icons for all SAML applications the user is allowed to access. The user requests access to a SAML application (the Service Provider) through Web Gateway (the Identity Provider) by selecting an icon on the launchpad.
3. Web Gateway redirects the authentication result to the SAML application through the user’s browser.
4. The SAML application grants access to the user.



SP-initiated SAML single sign-on

The SAML application in the cloud initiates the SSO process, which consists of the following overall steps:

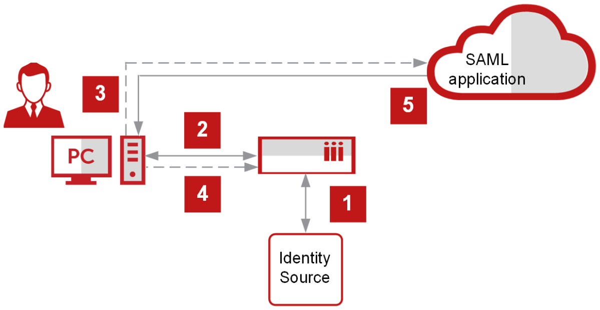
1. The user requests access to a SAML application (the Service Provider) directly.
2. The SAML application redirects the user’s request to Web Gateway (the Identity Provider) through the user’s browser.
3. Web Gateway authenticates the user.
4. Web Gateway redirects the authentication result to the SAML application through the user’s browser.
5. The SAML application grants access to the user.



Pure SP-initiated SAML single sign-on

Not all SAML applications support IdP-initiated and SP-initiated single sign-on. Some SAML applications support only one. SAML applications that support only SP-initiated single sign-on present a special use case called *pure* SP-initiated single sign-on.

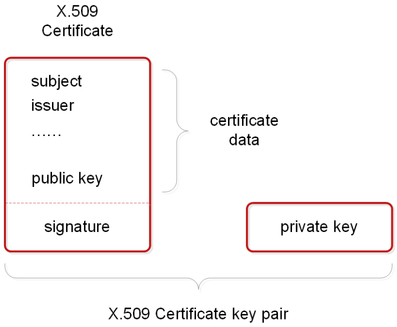
1. Web Gateway authenticates the user.
2. Web Gateway presents the user with a launchpad that includes icons for all SAML applications the user is allowed to access. The user requests access to a SAML application (the Service Provider) through Web Gateway (the Identity Provider) by selecting an icon on the launchpad.
3. Because the SAML application only supports SP-initiated single sign-on, Web Gateway redirects the user's request to the application through the user's browser. Because the user is not authenticated, the SAML application redirects the user to Web Gateway with an authentication request.
4. Web Gateway redirects the authentication result to the SAML application through the user’s browser.
5. The SAML application grants access to the user.



Certificate management for SAML single sign-on

SAML single sign-on requires an X.509 certificate and private key.

Together, the X.509 certificate and private key are known as the X.509 certificate key pair. The X.509 certificate contains the public key that makes up the key pair and a signature. The certificate can be self-signed or signed by a certificate authority.



The private key is used for signing outgoing SAML assertions and requests, and the X.509 certificate is used for verifying incoming signatures. The SSO party signing SAML assertions or requests with the private key provides the X.509 certificate to the SSO party verifying the signatures.

SAML services and applications have different certificate requirements. The following scenarios are common.

Certificate management

SSO process	Certificate management steps	SSO steps
IdP-initiated and SP-initiated SSO	1. In the Web Gateway interface, the administrator generates or imports a private key and certificate pair and exports the certificate for use by the Service Provider. 2. In the Service Provider interface, the administrator uploads the certificate corresponding to the private key.	1. Web Gateway uses the private key to create signed SAML assertions attesting to the user's identity. 2. The Service Provider uses the certificate to verify the signatures.
SP-initiated SSO	1. In the Service Provider interface, the administrator downloads the certificate corresponding to the private key. 2. In the Web Gateway interface, the administrator imports the Service Provider certificate.	1. The Service Provider uses the private key to create signed SAML SSO requests. 2. Web Gateway uses the certificate to verify the signatures.

Configure a SAML2 cloud connector

Configure a connector to a SAML 2.0 service or application using a template.

Task

1. Select Policy → Lists.
2. In the Lists tree, expand System Lists → SSO Catalog, then click Custom connectors.
3. Click the Add icon.

   The Add Connector dialog box opens.
4. Provide values for the fields and settings common to all connectors.

5. From the Template drop-down list, select the template corresponding to the SAML 2.0 service.
6. Provide values for the SAML settings.
7. Click OK.

Results

The newly configured SAML2 connector is added to the SSO Catalog → Custom connectors list.

Configure a generic SAML2 cloud connector

Configure a generic SAML2 cloud connector when you want to connect to a SAML 2.0 service that Web Gateway does not support with an individual connector template.

Task

1. Select Policy → Lists.
2. In the Lists tree, expand System Lists → SSO Catalog, then click Custom connectors.
3. Click the Add icon.
   The Add Connector dialog box opens.
4. Provide values for the fields and settings common to all connectors.
5. From the Template drop-down list, select Generic SAML2 Connector.
6. Provide values for the generic SAML2 settings.
7. Click OK.

Results

The newly configured SAML2 connector is added to the SSO Catalog → Custom connectors list.

Configuring external data sources for SAML single sign-on

While credentials for single sign-on to HTTP services are stored in the credential store that comes integrated with Web Gateway, SAML credentials come from external data sources, such as one or more LDAP servers, a database, or a web service. Several external data sources are configured using the external lists feature.

Identity information is fetched from external data sources as user attribute name-value pairs. The names must match the attribute names configured when the cloud connector was created.

SAML authentication using an external Identity Provider

To support organizations that want users to authenticate using a trusted, external Identity Provider, Web Gateway performs the SAML Service Provider role.

Note: SAML authentication refers to how identity information is shared between the Identity Provider and the Service Provider.

In this SAML scenario, the external Identity Provider is a database or authentication service that the organization trusts, but is outside the Web Gateway system. Web Gateway sends a SAML authentication request to the external Identity Provider. The Identity Provider authenticates the user using any authentication method and returns the identity information in a SAML assertion in the SAML authentication response. Web Gateway extracts the identity information from the SAML assertion and sets a cookie, which the user can use to authenticate to a cloud application.

Internally, Web Gateway implements SAML authentication using an external Identity Provider through the authentication server and the proxy, which provides the SAML functionality that the authentication server is missing.

Note: The application in the cloud provides a service and is also known as a Service Provider. However, in this scenario, the Identity Provider and Service Provider roles are assigned to the players in the SAML authentication process itself, not to the service provided in the cloud.

SAML authentication process using an external Identity Provider

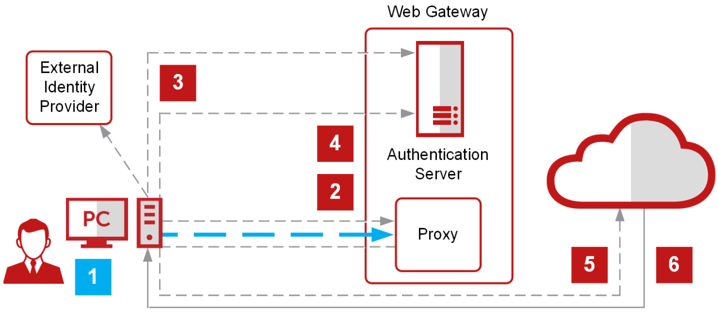
The authentication server consumes the SAML assertion in the response sent by the external Identity Provider and sets a cookie for the authenticated user.

The SAML authentication process begins when the user requests access to an application in the cloud through Web Gateway. The process consists of HTTP Redirect (GET) and POST messages that are sent through the user's browser (dashed lines). It also consists of messages that are sent and received by the user (solid lines). The messages that are sent through the user's browser to another SAML party take place automatically and quickly. The user is not aware of the authentication process running in the background.

Note: Web Gateway sends the SAML authentication request to and receives the SAML authentication response from the Identity Provider using the HTTP POST method.

1. The user requests access to an application in the cloud through Web Gateway.
2. If Web Gateway does not recognize the user, the proxy redirects the request to the authentication server through the user's browser.
3. The authentication server sends a SAML authentication request to the Identity Provider through the user's browser. The Identity Provider authenticates the user and sends a SAML authentication response back to the authentication server, also through the user's browser.

   Note: If the authentication server URL is static, the proxy intercepts the authentication response, constructs a dynamic URL, and redirects the response to the authentication server.
4. The authentication server consumes the SAML assertion in the response, sets a cookie, and redirects the authenticated user with the cookie to the application in the cloud through the proxy.
5. The proxy redirects the user with the cookie to the application in the cloud through the user's browser.
6. The application grants access to the user.



How Web Gateway supports static ACS URLs

Web Gateway supports Identity Providers that do not support dynamic URLs by saving the dynamic ACS URL in the RelayState parameter.

RelayState parameter

The URL of the authentication server, which provides the Assertion Consumer Service, is dynamic. Not all Identity Providers support dynamic URLs, which contain parameters. Web Gateway supports these Identity Providers by saving the value of the dynamic ACS URL at the time the authentication request is created in the RelayState parameter.

Note: The RelayState parameter is configured automatically. No configuration is required on your part.

The authentication server sends the RelayState parameter and the authentication request to the Identity Provider in a POST form. When the Identity Provider returns the RelayState parameter and the authentication response, also in a POST form, the value of the RelayState parameter is unchanged.

If the ACS URL in the response is static, the proxy intercepts the response and restores the dynamic ACS URL from the static ACS URL and the RelayState value. Using the restored ACS URL, the proxy can redirect the SAML authentication response to the authentication server.

Configuring a static ACS URL

If the external Identity Provider supports dynamic URLs, the authentication server automatically sends the dynamic value to the Identity Provider and validates the ACS URL that it receives in return. No configuration is required in the Web Gateway interface.

If the external Identity Provider does not support dynamic URLs, the static ACS URL must be configured in two locations in the Web Gateway interface. The configured values must match.

- Static ACS URL value sent to the Identity Provider in the *SAML request* — This value is configured in the Prepare fixed ACS URL rule.
- Static ACS URL value expected in the *SAML response* from the Identity Provider — This value is configured in the SAML Response settings.

  Note: The ACS URL value that you are expecting from the Identity Provider must also be configured at the Identity Provider.

High-level configuration tasks

Configuring SAML authentication with Web Gateway in the Service Provider role involves the following high-level tasks.

1. Import the Cookie authentication with SAML back end and fixed ACS URL rule set from the Rule Sets Library.

   Note: This rule set is located in the Authentication rule set group.
2. Configure a static ACS URL.

   Note: This task is required when the external Identity Provider does not support dynamic URLs.
3. Configure the SAML authentication request.

   Note: Web Gateway does not sign the SAML authentication request nor provide an X.509 certificate.
4. Configure the SAML authentication response. This task includes importing the X.509 certificate that the Identity Provider uses to sign the SAML authentication response and assertion.
5. To ensure that the authentication server recognizes the authentication response sent by the external Identity Provider, add the Identity Provider's service URL to the SAML IdP Whitelist.
6. Configure the SAML attributes that you want mapped to the Authentication.UserName and Authentication.UserGroups properties.
7. Manually configure the external Identity Provider to produce a SAML authentication response that meets the requirements you configure in the Web Gateway interface.

Configure a static ACS URL

Configure a static ACS URL when the external Identity Provider does not support dynamic URLs.

Before you begin

Make sure that the Cookie Authentication with SAML backend and fixed ACS URL rule set is imported from the Rule Set Library.

Task

1. Select Policy → Rule Sets.

2. Expand Cookie Authentication with SAML backend and fixed ACS URL → Cookie Authentication at Authentication Server, then select Authentication Server Request.
3. Select the rule Prepare Fixed ACS URL, then click Edit.
   The Edit Rule dialog box opens.
4. Select the step Events, select the event, then click Edit.
   The Edit Set Property dialog box opens with the User-Defined.SAMLUrlRewrite property selected.
5. Select "- enter your URL here -", then click Edit.
6. In the Enter a String dialog box, type the static URL, then click OK.
7. To close the Edit Rule dialog box, click Finish.

Results

The static ACS URL is updated in the **Authentication Server Request** rule set view.

Configure a SAML authentication request

Configure the SAML authentication request that the authentication server sends to the external Identity Provider.

Before you begin

Make sure that the Cookie Authentication with SAML backend and fixed ACS URL rule set is imported from the Rule Set Library.

Task

1. Select Policy → Settings.
2. Expand Engines → SAML Request, then select SAML Request.
   The Authn Request window opens for configuration.
3. Provide values for the Authn Request settings.
4. Click Save Changes.

Configure a SAML authentication response

Configure the SAML authentication response that the authentication server expects to receive from the external Identity Provider. To determine whether the SAML authentication response is valid, the authentication server compares the actual values in the response to the configured values.

Before you begin

Make sure that the Cookie Authentication with SAML backend and fixed ACS URL rule set is imported from the Rule Set Library.

Task

1. Select Policy → Settings.
2. Expand Engines → SAML Response, then select SAML Response.
   The Authn Response window opens for configuration.
3. Provide values for the Authn Response settings.
4. Click Save Changes.

Add external IdP URL to SAML IdP Whitelist

To ensure that the authentication server recognizes the authentication response sent by the external Identity Provider, add the Identity Provider URL to the SAML IdP Whitelist.

Before you begin

Make sure that the Cookie Authentication with SAML backend and fixed ACS URL rule set is imported from the Rule Set Library.

Task

1. Select Policy → Lists.
2. Expand Custom Lists → Wildcard Expression, then select SAML IdP Whitelist.
3. Click Add.
   The Add Wildcard Expression dialog box opens.
4. In the Wildcard Expression field, specify an expression that matches the URL of the external Identity Provider.
5. Click OK.
   The matching expression is added to the SAML IdP Whitelist.

Configure SAML attribute mapping

You configure the names of the SAML attributes that you want mapped to the Authentication.UserName and Authentication.UserGroups properties.

Before you begin

Make sure that the Cookie Authentication with SAML backend and fixed ACS URL rule set is imported from the Rule Set Library.

Task

1. Select Policy → Rule Sets.
2. Expand Cookie Authentication with SAML backend and fixed ACS URL → Cookie Authentication at Authentication Server, then select Authentication Server Request.
3. Select the rule Set user name and groups, then click Edit.
   The Edit Rule dialog box opens.
4. Select the step Events, select an event, then click Edit.
   The Edit Set Property dialog box opens with the Authentication.UserName or Authentication.UserGroups property selected.
5. Navigate to the Parameters for Property "Map.GetStringValue" dialog box.
6. Select an option:
   - Authentication.UserName — Select 2. Key (String) Value: "userId". Replace userID with the name of the SAML attribute that you want mapped to the user name property.
   - Authentication.UserGroups — Select 2. Key (String) Value: "userGroup". Replace userGroup with the name of the SAML attribute you want mapped to the user groups property.
7. To save your changes, click OK.
8. To close the Edit Rule dialog box, click Finish.

Results

The names of the SAML attributes that you want mapped are updated in the Authentication Server Request rule set view.

Validating the SAML authentication response

To validate the SAML authentication response sent by the external Identity Provider, the authentication server compares the values in the response to the values configured in the Web Gateway interface.

Important: You must manually configure the external Identity Provider to produce a SAML authentication response that meets the requirements configured in the Web Gateway interface.

To be valid, the SAML authentication response must meet the following requirements:

- The response must be a valid XML string.
- The response must include at least one SAML assertion.

- The `<saml2p:StatusCode>` element in the response must have the value `Success`.
- If configured, the response signature and assertion signature must be valid.
  - Response must be signed — If this setting is selected in the SAML authentication response configuration, the authentication server checks that the response is signed and that the signature is valid.
  - Assertion must be signed — If this setting is selected in the SAML authentication response configuration, the authentication server checks that the assertion is signed and that the signature is valid.
- The value of the `<saml2:Issuer>` element in the response must match the EntityID setting in the SAML authentication response configuration.
- The `<saml2:Conditions>` element in the response must include the attributes `notBefore` and `notAfter`.
- The current local time must fall within the specified time range, as follows.
  - Response must be already valid — When selected, this setting specifies that the current local time must be greater than or equal to the `notBefore` value.
  - Negative clock skew — The current local time must be greater than or equal to the `notBefore` time minus the negative clock skew value specified in the SAML authentication response configuration.
  - Positive clock skew —The current local time must be less than or equal to the `notAfter` time plus the positive clock skew value specified in the SAML authentication response configuration.
- If configured, the `audience` element must be included in the response and set to a predefined value, as follows.
  - Audience must be set in the response — If selected in the SAML authentication response configuration, the response must include the element `<saml2:Audience>`.
  - Audience must match the predefined value — If selected in the SAML authentication response configuration, the value of the `<saml2:Audience>` element must match the value specified in the Audience URI or ACS URL field in the configuration.
- The value of the `Destination` attribute in the `<saml2p:Response>` element in the response must match the ACS URL setting specified in the SAML authentication response configuration.

Providing SSO services for .NET and Java web applications

Using the Single Sign On rule set and the generic IceToken cloud connector template, you can configure single sign-on to any .NET or Java web application. Use this option when Web Gateway does not support the web application with a predefined connector or connector template.

Web Gateway implements single sign-on using the IceToken authentication method in the same way that it implements single sign-on using SAML authentication. Single sign-on using the two authentication methods has the following differences:

- In both cases, the Identity Provider sends the user information to the Service Provider in an assertion. The format of the user information in the assertion differs depending on the authentication method used.
- Single sign-on using the IceToken authentication method is simpler and easier to configure than single sign-on using SAML authentication.

Configure a generic IceToken cloud connector

To configure single sign-on to a .NET or Java web application, use the generic IceToken cloud connector template.

Task

1. Select Policy → Lists.
2. In the Lists tree, expand System Lists → SSO Catalog, then click Custom connectors.
3. Click the Add icon.
   The Add Connector dialog box opens.
4. Provide values for the fields and settings common to all connectors.
5. From the Template drop-down list, select Generic IceToken Connector.
6. Provide values for the generic IceToken settings.
7. Click OK.

Results

The newly configured IceToken connector is added to the SSO Catalog → Custom connectors list.

How users work with the application launchpad

Using the application launchpad, users can open applications and select and manage application accounts.

Overall workflow

From the user's point of view, the launchpad workflow appears as follows:

1. Using the launchpad URL provided by an administrator, the user opens the launchpad.
2. The launchpad opens displaying the logon form, where the user enters credentials.
3. If authentication is successful, the launchpad presents icons representing the applications the user is allowed to access. To request an application, the user clicks the corresponding icon.

Note: The launchpad filters the applications it displays. For example, it does not display dynamic HTTP applications in non-proxy mode, applications that are not supported, or applications that the user is not allowed to access.

Opening the launchpad

Using the launchpad URL provided by an administrator, the user opens the launchpad from a web browser on a client of Web Gateway.

Note: JavaScript must be enabled in the web browser.

The launchpad URL must contain the name of the management host configured in the Single Sign On settings. For example, if the host name is *sso.mwginternal.com*, the launchpad URL has one of the following values:

- https://sso.mwginternal.com
- https://sso.mwginternal.com/launchpad

Selecting a cloud application

Icons representing cloud applications are displayed in the left pane. When the user clicks an icon, the account information for that application is displayed in the right pane. If no account information is displayed, a couple of explanations are possible:

- **HTTP applications** — The first time an HTTP application is accessed, the user must provide credentials by clicking Add Account. The user also has the option of editing or deleting added accounts.
- **SAML applications** — Account information is not displayed or required, because SAML user information is retrieved from external sources.

You can initiate single sign-on to SAML applications and HTTP applications already configured with an account by clicking the application link in the left or right pane.

If the user has more than one account in an application, the icon representing that application is displayed multiple times in the left pane. Each icon is labeled with the user name corresponding to one account. To open a particular application account, the user can double-click the icon corresponding to the application-account pair.

Launchpad options

The launchpad provides options for displaying and selecting cloud applications and for working with application accounts. The following table describes these options.

Launchpad options

Option	Definition
Application logos	Allow the user to select cloud applications.
Find application	Allows the user to type a string that filters the cloud applications displayed by name.

Option	Definition
Display mode	From the drop-down list, the user selects a display mode: • Icons — Displays the application icons in rows. • List — Displays the application icons in list format and includes the categories to which the applications belong.
Sort applications	From the drop-down list, the user selects a method for sorting the cloud applications: • By name — Displays cloud applications sorted by name. • By category — Displays cloud applications sorted by category and name.
Name	Displays the icon and name of the cloud application selected by the user.
Note: The following account information is only available for HTTP applications. SAML user information is fetched from external sources.	
Account	Displays the email address of the cloud application account selected by the user.
Edit Account	Allows the user to edit the selected cloud application account.
Add Account	Allows the user to add a cloud application account.
Web Gateway user	Displays the name of the user selecting the cloud application.
Category	Displays the category of the cloud application selected by the user.
Description	Displays the description of the cloud application account selected by the user.
Delete Account	Allows the user to delete the selected cloud application account.

Customizing the application launchpad

In the Web Gateway interface, you can specify a name and description for your organization, customize the look of the text, and import images of your organization and product logos. You can also customize the header, footer, and sidebar that frame the launchpad.

Opening the template editor

To customize the launchpad, you edit a collection of files and templates named Single Sign On Schema. To open the collection in a template editor, navigate to the Single Sign On default settings and select Single Sign On Schema from the Collection drop-down list.

Note: Alternatively, you can access the templates directly by selecting the Templates tab.

Files in the /dat folder vs. /files folder

When generating the launchpad, the Single Sign On module uses files located in the following folders on the server where the appliance is installed and the SSO process is running:

- **/dat** — Files in this folder are system files maintained by the appliance. They are overwritten each time there is an update from the update server.

- **/files** — Files in this folder and subfolders, including /img, can be customized, because they are not overwritten by the update server.

Edit the Launchpad.html file

In the Launchpad.html file, you can specify a name and description for your organization and the names of the style sheet and the image files containing logos. You can also customize the header, footer, and sidebar that frame the launchpad.

For example, you can add a message of the day or links to your IT organization to the sidebar.

Task

1. Select Policy → Settings.
2. Expand Engines → Single Sign On, then select Default.
   The Single Sign On settings open.
3. From the Collection drop-down list, select Single Sign On Schema, then click Edit.
   The Template Editor opens with the Single Sign On Schema folder selected.
4. Expand Single Sign On Schema → Launchpad → en, then select html.
   The HTML Editor opens.
5. In the editor:
   a. Replace `Your Company Name` with the name of your organization.
   b. Replace `Your Company Description` with a description of your organization.
   c. (Optional) Replace `customLaunchpad.css` with the name of your custom .css file.
   d. Replace `sample_logo.png` with the name of the image file containing the logo that represents your organization.
   e. Replace `productCompLogo.png` with the name of the image file containing the logo that represents your product.
6. To customize the header, add content to the <div id="header"></div> element.
   Note: Do not remove the "header" element even if it is empty.
7. To customize the footer, add content to the <div id="footer"></div> element.
   Note: Do not remove the "footer" element even if it is empty.
8. To customize the sidebar:
   a. Add the <div id="aside"></div> element to the launchpad.html file, as follows.
   ```
   <div id="main"> <div id="aside"> : : </div> $SSO.GetDatFile(“launchpadMain.html”)$ </div>
   ```
   b. Add content to the <div id="aside"></div> element.
   Example:
   ```
   <div id="aside"> <img src="/files/img/your_logo.png"> <hr> $first line of SSO.GetDatFile(“version.txt”)$
   <hr> MWG: $MWG.Version$<br>$MWG.BuildNumber </div>
   ```
   This example displays the logo you provide, the version number of the latest update from the update server, and the version and build numbers of the appliance in the sidebar.
9. To close the Template Editor, click OK.
10. Click Save Changes.

Edit the default launchpad style sheet

In the default launchpad style sheet that comes with the appliance, you can customize the look of your organization's name and description. You can also customize the header, footer, and sidebar that frame the launchpad.

For example, you can specify the background image that frames the launchpad. The image file can be located in the /files/img or /dat folder.

Note: Alternatively, you can import a custom style sheet.

Task

1. Select Policy → Settings.

2. Expand Engines → Single Sign On, then select Default.
   The Single Sign On settings open.
3. From the Collection drop-down list, select Single Sign On Schema, then click Edit.
   The Template Editor opens with the Single Sign On Schema folder selected.
4. In the File System area, expand singleSignOn, then select customLaunchpad.css.
   The Editor opens.
5. In the editor, specify the font color, font-size, and font-family properties of your organization's name and description as you want them to look on the launchpad.
   Example:
   ```
   /* Organization Name */ #mainDesc { color:RGB(51,51,51); font-size: 12pt; font-family:verdana; } /*
   Organization Description */ #subDesc { color:RGB(102,102,102); font-size: 9pt; font-family:verdana; }
   ```
6. In the editor, specify the background image that frames the launchpad.
   In the following example, the background image can be a logo that is repeated until it fills the frame around the launchpad.
   ```
   body { width: 100%; } #main { // In one of the following lines, replace <image_file> with the filename // of
   the background image and remove the comment tag from that line: // background: url("/files/img/<image_file>")
   repeat; // background: url("/dat/<image_file>") repeat; padding: 0px; } #aside { display: inline-block; width:
   100px; align-self: flex-start; }
   ```
7. To close the Template Editor, click OK.
8. Click Save Changes.

Import a custom launchpad style sheet

In the user interface, you can import a launchpad style sheet. It can be one that you exported and edited or one of your own.

Note: When using your own style sheet, remember to place the .css file in the /files directory and to update the name of the .css file in Launchpad.html.

Task

1. Select Policy → Settings.
2. Expand Engines → Single Sign On, then select Default.
   The Single Sign On settings open.
3. From the Collection drop-down list, select Single Sign On Schema, then click Edit.
   The Template Editor opens with the Single Sign On Schema folder selected.
4. In the File System area, select singleSignOn.
5. From the Add drop-down list, select Existing File or Directory, browse for your style sheet file, then click Open.
   Your style sheet file is added to the File System under singleSignOn.
6. To close the Template Editor, click OK.
7. Click Save Changes.

Provide a custom logo for the launchpad

To provide a logo that represents your organization or product on the application launchpad, import a custom image file.

Note: Remember to place the image file in the /files/img or /dat folder and to update the name (and optionally the location) of the image file in Launchpad.html.

Task

1. Select Policy → Settings.
2. Expand Engines → Single Sign On, then select Default.
   The Single Sign On settings open.
3. From the Collection drop-down list, select Single Sign On Schema, then click Edit.
   The Template Editor opens with the Single Sign On Schema folder selected.

4. In the File System area, expand singleSignOn, then select img.
5. From the Add drop-down list, select Existing File or Directory, browse for the file containing your logo, then click Open.
   The image file is added to the File System under img.
6. To close the Template Editor, click OK.
7. Click Save Changes.

Creating bookmarks to cloud services for your organization

You can create bookmarks to cloud services or applications for users across your organization.

To create a bookmark, format the link as follows:

```
https://sso.mwginternal.com/login?service=<S>
```

where `<S>` specifies the Service ID in the SSO Catalog.

When users click the link to the service, the SSO module delivers the HTML template for the logon page. The JavaScript in the HTML template retrieves the user's account information for the specified service. Depending on the number of accounts the user has, one of the following actions takes place:

- **The user has no account in the service** — The user is redirected to the launchpad, which presents the option of creating an account. After creating an account, the user can log on to the service following the SSO process.
- **The user has one account in the service** — The user can log on to the service following the SSO process.
- **The user has more than one account in the service** — The user is redirected to the launchpad, which presents the option of selecting an account. After selecting an account, the user can log on to the service following the SSO process.

Monitoring logons to cloud services on the dashboard

On the dashboard in the user interface, you can view statistics about the number of logons to all cloud applications and services.

Select Dashboard → Charts and Tables → Single Sign On Statistics to view the following information:

- All Logins — Shows the number of logons to all cloud applications and services over the specified time period.
- Logins per service — Shows the number of logons by cloud application or service over the specified time period.
- Logins per service — Lists the specified number of cloud applications and services from most to least often accessed, including how many times each service was accessed.
- Number of forbidden logins — Shows the number of logons to all cloud applications and services over the specified time period that were denied because of invalid tokens.

Locate information about the latest SSO updates

When working with the cloud single sign-on feature, you might want to know which version of the software and the catalog you are using. In the user interface, you can view the version number and date and time of the latest updates to the SSO feature or engine.

- McAfee Single Sign On — Updates include changes to the SSO software, for example, a change to an SSO rule.
- McAfee SSO Connector Catalog — Updates include changes to the list of cloud applications and services that Web Gateway supports with connectors.

Note: If you are not receiving SSO updates, confirm that you have a valid license.

Task

1. Select Dashboard → Charts and Tables → System Summary.
2. To view the version numbers of the latest SSO updates: In the Update Status table, in the Feature column, locate the following rows:

- McAfee Single Sign On
- McAfee SSO Connector Catalog

3. To view the date and time of the latest SSO updates: In the Last Update table, in the Engine column, locate the following rows:
   - McAfee Single Sign On
   - McAfee SSO Connector Catalog
4. To refresh the view with the latest data, click the refresh icon in the upper right corner of the Update Status and Last Update tables.

SSO logging overview

The SSO Log rule set generates the SSO access log, and optionally the SSO trace log, from information about SSO requests that the proxy stores in the SSO.LogAttributes property.

The SSO proxy stores information about internal and external SSO requests in the SSO.LogAttributes property. When SSO logging is enabled:

- Internal requests are logged to the SSO access log instead of the general access log.
- External requests, which come from outside Web Gateway, are logged to the general access log.

To enable SSO logging, import the SSO Log rule set from the Logging rule set group in the Rule Set Library. The SSO Log rule set consists of the following nested rule sets:

- Access Log — Logs error and info messages to the SSO access log file.
- Trace Log — Logs all messages to the SSO trace log file.
- Stop Logging — Stops the SSO Log rule set cycle.

Note: The trace log is more detailed than the access log and is intended for debugging the SSO feature.

Enabling SSO logging involves these overall steps:

1. Add the SSO Log rule set to the Log Handler rule set tree.
2. Move the SSO Log rule set above the Default logging rule sets in the Log Handler tree. This step ensures that SSO requests are logged to the SSO access log before the general access log and that the logging cycle is then stopped.
3. (Optional) Enable SSO trace logging.

Enable SSO logging

When you enable SSO logging, SSO requests are logged to the SSO access log instead of the general access log. You can also enable SSO trace logging.

Note: If you enable trace logging, we recommend that you set the log level to Full. To locate the log level setting, select Policy → Settings → Engines → Single Sign On → Default → Advanced Settings.

Task

1. Select Policy → Rule Sets → Log Handler → Default.
2. From the Add drop-down list, select Rule Set from Library.

   The Add from Rule Set Library dialog box opens.
3. Expand Logging, then select SSO Log.
4. If importing the rule set creates conflicts, click Auto-Solve Conflicts, click one of the following strategies, then click OK.
   - Solve by referring to existing objects
   - Solve by copying and renaming to suggested

   The SSO Log rule set is added to the Log Handler tree.
5. In the Log Handler tree, move the SSO Log rule set above the Default rule sets.
6. (Optional) To enable detailed logging:
   a. In the Log Handler tree, expand SSO Log, then select the Trace Log rule set.

b. In the configuration window, select the Enable checkbox.

Resolving SSO issues

See the following table for SSO issues and ways to resolve them.

Resolving SSO issues

Issue	Resolution
The credential store fails to return credentials when requested.	Check the error log for credential store errors (34050–34090).
The user cannot log on to the selected cloud service.	The connector to the service might be broken. Contact the SSO Catalog support team.
The user cannot update credentials for a cloud service.	Check the order of the rules in the Single Sign On rule set. The Select Services rule set, which adds services to SSO Connector lists, must be located before the Manage Form Credentials rule set.
SAML single sign-on fails.	Possible reasons for SAML SSO failure are: • **Not all user information is provided** — Some cloud applications require specific user attributes. To view the missing user attributes, check the error log for SSO errors (34000–34999). • **Single sign-on is not configured correctly** — Verify that single sign-on is configured correctly in the Web Gateway user interface and in the SAML application administrator account.
When automatic downloading of SAML metadata is configured and the download fails, an error is returned stating that the requested service does not exist.	Possible reasons for this error are: • The metadata is downloaded from an HTTPS URL without a trusted certificate. • The signature in the SAML metadata file is incorrect. • The SAML metadata file is missing the signature. **Note:** For more information about this error, see the file: /opt/mcfc/log/mcfc.log.
After importing the SSO rule set, one or more custom connectors or links to cloud services and applications are broken.	When the rule set is imported, new Service IDs are assigned to the custom connectors. Update any Service IDs that are used to reference custom connectors.

Cloud storage encryption

When the users of your network work with data that is stored in the cloud using a cloud storage service, Web Gateway provides a function for encrypting this data, together with the corresponding function for decryption.

- **Cloud storage encryption** — When a user uploads data to a cloud storage service, the data can be encrypted.
- **Cloud storage decryption** — When a user downloads encrypted data from a cloud storage service, the data is decrypted to enable the user to work with it.

A suitable rule set is available from the rule set library on Web Gateway for configuring cloud storage encryption and decryption.

Encrypting and decrypting cloud storage data

To enhance security when users of your network complete in-the-cloud activities, you can configure the encryption of data that a user uploads to a cloud storage service. When the data is downloaded, it is decrypted to allow the user to work with it.

A module of Web Gateway, known as the Cloud Storage Encryption module (also referred to as Cloud Storage Encryption filter or engine), handles both encryption and decryption of data, including the metadata. Encryption and decryption remain transparent to the user.

Encryption and decryption is performed for "top-level" data that is processed in the request and response cycles. Data that is embedded in a request or a response and, accordingly, processed in the embedded objects cycle, cannot be encrypted or decrypted.

To encrypt and decrypt data, the module uses a standard algorithm, which can be one of the following:

- AES-128
- AES-192
- AES-256

The algorithm is also known as *cipher*.

A password is also required as a parameter of the encryption or decryption process.

For performing this process, the module relies on service description files, which exist for each of the various cloud storage services.

The files provide information on how to handle different data formats, the methods that can be used in an upload or download request, for example, PUT or POST, and the URLs that are sent with requests to identify the locations where the data should be uploaded to or downloaded from.

Note: Service description files are updated when a new version of Web Gateway is installed. It is not possible to download new versions of these files from an update server.

Encryption and decryption of data can be performed for the following cloud storage services:

- Box
- Dropbox
- Google Drive
- Microsoft SkyDrive

For the Box cloud storage service, encryption and decryption is supported when a web browser or a native Box client is used to upload and download data. For Dropbox, Google Drive, and Sky Drive, it is supported when upload or download is performed from a web browser.

Configuring encryption and decryption

To configure the encryption and decryption process, you need to implement suitable rules on Web Gateway. They are provided in the Cloud Storage Encryption rule set, which you can import from the library.

The rules in the library rule set control the Cloud Storage Encryption module and provide a default password for the encryption or decryption process. The rule set also contains an optional rule for logging the process.

The rule that controls the module for encryption applies if it is found that a request that was received on Web Gateway is a request for uploading data to one of the configured cloud storage services. Similarly, the rule that calls the module for decryption applies if a request for downloading data from one of these services has been received.

If one of the two rules applies, it triggers an event that lets the module perform the encryption or decryption process.

But whereas decryption is executed as soon as the rule processing module (rule engine) has actually found the relevant rule to apply, encryption is not executed before all the following rules have also been processed, including rules configured for processing in the embedded objects cycle.

This ensures the data that is sent with a request for uploading or downloading can be processed in unencrypted format by the other rules.

Module settings are implemented with the import of the library rule set, which you need to configure to specify the following:

- Algorithm (cipher) used for encryption and decryption
- Supported cloud storage services

Data trickling and decryption

If data trickling is implemented as a mode of transferring data, decryption of an encrypted file that is downloaded from a cloud storage service might fail. You should therefore configure these functions as follows:

- On the rule set tree, the Cloud Storage Encryption rule set should be placed immediately before or after the rule set that you use to implement data trickling.

  This prevents rules of other rule sets from being processed between the decryption and the data trickling rules, which can cause the decryption to fail.

  A rule for enabling data trickling is contained in the Progress Indication rule set, which is an embedded rule set of the Common Rules rule set of the default rule set system.

To be completely sure that data trickling does not lead to a failed decryption, you can additionally do the following:

- Replace the *Always* in the criteria of the data trickling rule by *CloudEncryption.IsDecryptionSupported equals false* .

  This prevents data trickling from being started when downloaded data is decrypted. However, configuring the criteria like this will have an impact on the performance of the data trickling process.

Note: A conflict between decryption and data trickling can also be the reason why a file that was downloaded from a cloud storage service is corrupted and cannot be opened, although no decryption errors were reported.

Multiple encryption of data

When a request for uploading data to a cloud storage service is received, the data can be encrypted more than once, performing encryption differently each time.

For each encryption, you need to configure a rule. You can, for example, specify a password for a user group in one rule and let encryption be performed under a particular algorithm, which you also specify in that rule, and then specify a password for an individual user in the next rule and let encryption be performed under a different algorithm.

So when it comes to downloading the data, it can only be decrypted if both passwords are known.

To decrypt what has been encrypted in multiple rules, the same number of rules for decryption is needed. Algorithms and passwords must be the same as in their encryption counterparts and the order of these rules must be the reverse of the order in which you placed the encryption rules.

SSL-secured upload and download requests

To cover also requests for uploading data to a cloud storage service or for downloading data when they are sent using an SSL-secured connection, you need to make sure the SSL Scanner rule set is enabled.

This rule set is implemented in disabled state with the default system of rule sets for Web Gateway.

The certificates that are needed for communication over SSL-secure connections must be installed on the web browsers that users work with to send upload and download requests.

Manual decryption of data

When Web Gateway is temporarily unavailable in your network or when a password conflict arises, it could be required that you decrypt cloud storage data manually.

This can be done if you know the algorithm and password that were used when the data was encrypted. You can download the data directly from the cloud storage service to your system and run a command for manual decryption on this system, which includes algorithm and password parameters.

Monitoring encryption and decryption on the dashboard

Statistics about activities performed for encrypting and decrypting cloud storage data can be monitored on the dashboard of the user interface.

The following parameters are shown:

- Number of encryption and decryption operations and errors (over time)
- Volume of encrypted and decrypted data (over time)
- Number of encryption and decryption operations and errors per cloud storage service
- Volume of encrypted and decrypted data per cloud storage service

Configure encryption and decryption of cloud storage data

To configure the encryption and decryption of data that is uploaded to a cloud storage service or downloaded, complete the following high-level steps.

Task

1. Import the *Cloud Storage Encryption* rule set from the rule set library.
   The rule set is located in the *Cloud Services* rule set group.
2. Configure the settings for encrypting and decrypting cloud storage data.
3. Ensure communication for encrypting and decrypting data can go on in SSL-secured mode.
   a. Enable the *SSL Scanner* rule set in the default rule set system of Web Gateway.
   b. For the browsers on the clients of Web Gateway that upload and download cloud storage data, make sure the certificates needed for SSL-secured communication are installed.
4. Save your settings.

Configure the settings for encrypting and decrypting data

To configure the settings for encrypting and decrypting cloud storage data, work with two different module (engine) settings.

Task

1. Select Policy → Settings.
2. On the Engines branch of the settings tree, expand Cloud Storage Encryption and select the particular settings for the Cloud Storage Encryption module you want to configure, for example, the Default settings.
   The settings appear in the settings pane.
3. Configure these settings as needed.
4. Expand Cloud Storage Encryption Support and select the particular settings for the Cloud Storage Encryption Support module you want to configure, for example, the Default settings.
   The settings appear in the settings pane.
5. Configure these settings as needed.
6. Click Save Changes.

Decrypt cloud storage data manually

When you cannot use Web Gateway to decrypt cloud storage data, you can perform the decryption manually by running a suitable command if you know the algorithm and password that were used for the encryption.

Task

1. Download the data in encrypted format from the cloud storage service that stored the data to your system.
2. Run the following command to decrypt the data:

   `openssl enc` -<cipher> `-d -in` <encrypted file> `-out` <decrypted file> `-k` <password> `-md sha256`

   The variable parameters have the following meanings:

<cipher>	Algorithm used to encrypt the data
<encrypted file>	Path and file name for the file that contains the encrypted data
<decrypted file>	Path and file name for the file the decrypted data should be written to
<password>	Password used when the data was encrypted

   The data is decrypted and written to the specified file.

Cloud Storage Encryption rule set

The Cloud Storage Encryption rule set is a library rule set for encrypting and decrypting data that is uploaded to and downloaded from cloud storage services.

Library rule set – Cloud Storage Encryption
Criteria – *Always*
Cycles – Requests (and IM), Responses

The rule set contains the following rules.

Set encryption password
Always –> Continue – Set User-Defined.Encryption Password = "webgateway"
The rule uses an event to set the default password for Web Gateway as the password that is used when data is encrypted.

Enable encryption
CloudEncryption.IsEncryptionSupported<Default> equals true –> Continue – CloudEncryption.Encrypt(User-Defined.Encryption Password)<Default>
The rule uses the CloudEncryption.IsEncryptionSupported property to check whether encryption of data can be performed. If this is the case, an event is used to perform the encryption.

Enable decryption
CloudEncryption.IsDecryptionSupported<Default> equals true –> Continue – CloudEncryption.Decrypt(User-Defined.Encryption Password)<Default>

The rule uses the CloudEncryption.IsDecryptionSupported property to check whether decryption of data can be performed. If this is the case, an event is used to perform the decryption.

Fix content type after decryption
CloudEncryption.IsDecryptionSupported<Default> equals true –> Continue – MediaType.Header.FixContentType
The rule uses the CloudEncryption.IsDecryptionSupported property to check whether a decryption of cloud storage data was performed.
If this is the case, an event is used to modify the Content-Type field in the header of the response that was sent to deliver the data to Web Gateway. Cloud storage services set this field by default to application/octet-stream, as they are not able to recognize real media types when data is encrypted. The *MediaType.Header.FixContentType* event sets the field to a value for a real media type.set to the value
This rule fixes the issue that cloud storage services set this field by default to *application/octet-stream*, as they cannot recognize different media types when data is encrypted. The *MediaType.Header.FixContentType* event sets the field to a value for the real media type.
The rule is not enabled by default.

Log encryption password
CloudEncryption.IsEncryptionSupported<Default> equals true –> Continue – Set User-Defined.encrypt-log.= DateTime.ToGMTString + ", User: " + Authentication.UserName + ", IP: " + IP.ToString (Client.IP) + ", Service: " + CloudEncryption.ServiceName + ", Cipher: " + CloudEncryption.CipherName<Default> + ", Password: " + User-Defined.EncryptionPassword FileSystemLogging.WriteLogEntry (User-Defined.encrypt-log)<Encryption Log>
The rule uses an event to create a log entry for an encryption.
A second event is used to write this entry into the log called Encryption Log, which is specified by the event settings. Since data is written into the log in encrypted format, you need a password to access it (default password: `webgateway`).
The rule is not enabled by default.

System settings

System settings are used to configure the appliance system.

Anti-Malware system settings

The Anti-Malware system settings are used for configuring the anti-malware queue.

Global Anti-Malware Settings

Settings for the anti-malware queue

Global Anti-Malware Settings

Option	Definition
Number of threads for AV scanning	Sets the number of anti-malware working threads that are available on an appliance. The number you specify here applies to both the threads that forward requests and responses to threads of the scanning modules and the scanning module threads themselves. For example, if you specify 25, there will be 25 threads for forwarding and 25 for scanning.
Use at least as many AV threads as the number of CPU cores available	When selected, the number of AV threads use for scanning activities is at least the same as the number of available CPU cores.
Maximum number of jobs in the queue	Limits the number of requests or responses that can be moved to the anti-malware queue as jobs for the scanning modules.
Number of seconds a scanning job stays in the queue before being removed	Limits the time (in seconds) that elapses before a request or response is removed from the anti-malware queue if it has not been forwarded for scanning.

Central Management settings

The Central Management settings are used for configuring appliances that you administer as nodes in a common configuration.

Central Management Settings

Settings for basic communication parameters of a node in a Central Management configuration

Central Management Settings

Option	Definition
IP addresses and ports of this node for Central Management communication	Provides a list for entering the IP addresses and port numbers that a node uses to communicate with other nodes in a Central Management configuration.

Option	Definition
Timeout for distributing messages to other nodes	Limits the time (in seconds) that is allowed for another node to respond to a message from the current node to the specified value. The time can range from 10 to 600 seconds. It is set on a slider scale.

The following table describes the elements of an entry in the IP addresses and ports list.

IP addresses and ports – List entry

Option	Definition
String	Specifies the IP address and port number for a node.
Comment	Provides a plain-text comment on an IP address and a port number.

Advanced Management Settings

Settings for advanced administration of a Central Management configuration

Advanced Management Settings

Option	Definition
Multiplier for timeout when distributing over multiple nodes	Sets a factor for increasing the time interval that has been configured under Timeout for distributing messages to other nodes in the *Central Management Settings* section. Increasing the time interval gives messages more time to proceed from one node to another, from there to the next node, and so on. The interval can be increased by a value between 1 and 2. The value is set on a slider scale.
Use and serve persistent connections	When selected, the nodes in a cluster use and serve persistent connections for their communication This option is selected by default on nodes that have been newly installed (clean install). It is not selected on existing nodes that have been updated. We recommend that you keep using persistent connections, which means that to ensure their use, you must also select the option on the updated nodes.
Node priority	Sets the priority that a node takes within a node group The highest priority is 1. If the configuration data on a node is no longer synchronized with that of other nodes, for example, because the node has been down for some time, the node receives the most recent configuration data from the node with the highest priority. If this is not your intention, make sure that all nodes have the same priority, which is also the recommended setting. The priority of a node can range from 1 to 100. It is set on a slider scale.
Allow a GUI server to attach to this node	When selected, a server providing an additional user interface for the appliance is allowed to connect to the node.

Option	Definition
Allow to attach a GUI server from non-local host	When selected, a server with an additional user interface that is not running on the current node is allowed to connect to the node.
GUI control address	Specifies the IP address and port number the additional user interface uses for connecting to the current node.
GUI request address	Specifies the IP address and port number of this server used when sending requests to it.
Use unencrypted communication	When selected, messages sent from this node to other nodes in the configuration are not encrypted. However, authentication using certificates is still performed. This option is not selected by default. **Note:** Make sure that all nodes in a Central Management configuration are configured in the same way with regard to this option Otherwise communication between the nodes will fail due to the differences in encryption handling.
Enable IP checking for other nodes	When selected, the IP address can be verified when messages are sent from this node to other nodes in the configuration. This function is intended to increase web security, but can lead to problems for some network setups, for example, NAT setups.
Allowed time difference	Limits the time difference (in seconds) allowed for accepting configuration changes to the specified value. The number of seconds can range from 10 to 600. It is set on a slider scale.
Enable version checking for other nodes	When selected, the version of the appliance software is checked before configuration changes are distributed between nodes. Configuration changes are not distributed to a node if the version of the appliance software on this node does not match the version on the node that distributes the changes. • Level of version check – Sets a level of thoroughness when verifying the version of the appliance software. The level is set on a slider scale. It can take the following values: • 1 – Only major version number (7 in 7.3.0) must match. • 2 – Minor version number (3 in 7.3.0) must also match. • 3 – Feature version number (0 in 7.3.0) must also match. • 4 – Maintenance version number (if any, for example, 1 in 7.3.0.1.2) must also match. • 5 – Hotfix version number (if any, for example, 2 in 7.3.0.1.2) must also match. • 6 – Build number (for example, 14379) must also match.

This Node is a Member of the Following Groups

Settings for including a node in a group of nodes

This Node is a Member of the Following Groups

Option	Definition
Group runtime	Determines the group of a node, in which runtime data can be shared with all nodes in the group, for example, time quotas.
Group update	Determines the group of a node, in which updates can be shared with all nodes in the group
Group network	Determines the group of a node, in which the node can immediately connect to all other nodes in the group A node can be a member of more than one network group. In this case, the nodes of a group that a node is a member of can connect through this node to nodes of another group that this node is also a member of. All groups that a node is a member of are listed in the Group network list.

The following table describes the elements of a list entry in the group network list.

Group network **– List entry**

Option	Definition
String	Specifies the name of a network node group.
Comment	Provides a plain-text comment on a network node group.

Automatic Engine Updates

Settings for scheduling automatic updates of database information for modules used in the filtering process

Automatic Engine Updates

Option	Definition
Enable automatic updates	When selected, database information is automatically updated.
Allow to download updates from the internet	When selected, database updates are downloaded from the internet.
Allow to download updates from other nodes	When selected, database updates are downloaded from other nodes in a Central Management configuration.
Update interval	Limits the time (in minutes) that elapses before database information is again updated to the specified value. The time is set on a slider scale. Allowed values range from 15 to 360.
CRL update interval	Limits the time (in hours) that elapses before certificate revocation lists used in filtering SSL-secured web traffic are updated to the specified value. This update uses a method that differs from those of other updates and must therefore be configured separately. The time is set on a slider scale

Option	Definition
	Allowed values range from 3 to 168.
Enable update proxies	When selected, proxies are used for performing updates. The proxies are configured in the Update proxies (fail over) list. These proxies are also used when the MLOS operating system of a Web Gateway appliance is updated.
Update proxies (fail over)	Provides a list for entering the proxies that are used for performing updates. The proxies are used in failover mode. The first proxy on the list is tried first and only if the configured timeout has elapsed is the next proxy tried.

The following table describes the elements of an entry in the Update proxies list.

Update proxies – List entry

Option	Definition
Host	Specifies the host name or IP address of a proxy for performing updates.
Port	Specifies the port on a proxy that listens for update requests.
User	Specifies the name of a user who is authorized to access a proxy for performing updates.
Password	Sets a password for this user.
Comment	Provides a plain-text comment on a proxy.

Advanced Update Settings

Settings for advanced update functions

Advanced Update Settings

Option	Definition
Allow to upload updates to other nodes	When selected, updated database information can be uploaded from the appliance (as a a node in a Central Management configuration) to other nodes.
The first time an update starts, it should wait an appropriate time before starting	Limits the time (in seconds) that elapses before an update is started to the specified value. Allowed values range from 5 to 1200.
The first time an automatic update starts, it uses the startup interval to update	Limits the time (in seconds) that elapses between attempts to start an automatic update for the first time to the specified value. During an update, the coordinator subsystem, which stores updated information on the appliance, tries to connect to the appliance core, where the modules reside that use this information. A low value for this interval can therefore speed up updates because it reduces the time the coordinator might have to wait until the core is ready to receive data.

Option	Definition
	Allowed values range from 5 to 600.
Try to update with start interval	Limits the number of attempts (1 to 9) the appliance makes when trying to start an update to the specified value.
Use alternative URL	Specified the URL of an update server that is used instead of the default server.
Verify SSL tunnel	When selected, a certificate sent to a node by an update server in SSL-secured communication is verified.
Enter a special custom parameter sequence for an update server	Updates of URL filtering information are taken from the URL filter database server that is specified by the URL entered here.
No updates should be made in defined time window	Provides a list for entering daily time slots during which no updates of database information should be made.

The following table describes the elements of an entry in the time slot list.

Time slot – **List entry**

Option	Definition
Start of time slot (hour)	Sets the hour when a daily time slot begins.
Start of time slot (minute)	Sets the minute in an hour when a daily time slot begins.
Start of time slot (second)	Sets the second in a minute when a daily time slot begins.
End of time slot (hour)	Sets the hour when a daily time slot ends.
End of time slot (minute)	Sets the minute in an hour when a daily time slot ends.
End of time slot (second)	Sets the second in a minute when a daily time slot ends
Comment	Provides a plain-text comment on a time slot.

Advanced Subscribed Lists Settings

Settings for advanced subscribed lists functions

Advanced Subscribed Lists Settings

Option	Definition
Allow to download customer subscribed lists	When selected, customer subscribed lists can be downloaded from the current appliance. If the appliance is a node in a Central Management configuration and this option is also selected on other nodes, one of the nodes will download the lists. If you want a particular node to download the lists, you need to make sure the option is deselected on every other node. When a node is restarted and one or more subscribed lists are configured on this node, list content is downloaded to ensure a valid configuration. **Note:** The download is performed regardless of whether this download option is selected or not.

Option	Definition
	When a node is added to a configuration with other nodes that have subscribed lists configured, list content is downloaded for these lists onto the new node. To reduce internal traffic, the download is performed without prior communication with other nodes. **Note:** The download is performed regardless of whether this download option is selected or not.

Manual Engine Updates

Setting for performing manual updates of database information for modules used in the filtering process

Manual Engine Updates

Option	Definition
Manual Engine Update	Updates database information for modules used in the filtering process immediately. Database information is only updated for the modules on the appliance you are currently working on.

Handle Stored Configuration Files

Settings for storing configuration file folders on disk

Handle Stored Configuration Files

Option	Definition
Keep saved configuration folders for a minimal time	Limits the time (in days) that configuration file folders are at least stored on disk to the specified value. The number of days can range from 1 to 100.
Keep minimal number of configuration folders	Limits the number of configuration file folders that are at least stored on disk at any time to the specified value. The number can range from 1 to 100.
Keep minimal number of packed folders	Limits the number of packed configuration file folders that are at least stored on disk at any time to the specified value. Configuration folders are packed when the minimal time configured for storing them on disk has elapsed and the minimal number of folders stored on disk at any time would be exceeded if they were stored unpacked any longer. The number of folders can range from 1 to 100.

Advanced Scheduled Jobs

Settings for scheduled jobs

Advanced Scheduled Jobs

Option	Definition
Job list	Provides a list of scheduled jobs.

The following table describes the elements of a list entry.

Job list – List entry

Option	Definition
Start job	Specifies the time setting for starting a scheduled job, for example, *hourly*, *daily*, *once*.
Start job immediately if it was not started at its original schedule	Lets a scheduled job start immediately if this has not happened according to the originally configured schedule.
Job	Specifies the type of job, for example, *Backup Configuration*.
Unique job ID	Identifies a scheduled job.
When this job has finished run job with ID	Provides the ID of a job that is run immediately after this job.
Comment	Provides a plain-text comment on a scheduled job.

Add Scheduled Job window

Provides settings for adding a scheduled job

- **Time Settings** — Settings for the time when a scheduled job is started
- **Job Settings** — Settings for the type and ID of a scheduled job
- **Parameter Settings** — Settings for additional parameters of a scheduled job

  These settings differ for each job type as follows:

  - (Backup configuration settings) — Settings for a scheduled job that creates a backup of an appliance configuration
  - (Restore backup settings) — Settings for a scheduled job that restores a backup of an appliance configuration
  - (Upload file settings) — Settings for a scheduled job that uploads a file to an external server using the HTTP or HTTPS protocol
  - (Download file settings) — Settings for a scheduled job that downloads a file to the appliance using the HTTP or HTTPS protocol

  For a scheduled job that performs a yum update, there are no additional parameter settings.

Time Settings

Option	Definition
Start job	Lets you select a time setting. • Hourly — Starts a scheduled job every hour • Daily — Starts a scheduled job once on a day • Weekly — Starts a scheduled job once in a week • Monthly — Starts a scheduled job once in a month • Once — Starts a scheduled job only once • Activated by other job — Starts a scheduled job after another job has been completed
(Time parameter settings)	Settings specifying the parameters for a time setting, for example, the minute in an hour when a job scheduled for hourly execution should be started Which time parameter settings are shown depends on the selected time setting. For example, if you have selected *Hourly*, you can configure the minute in an hour, but not the day in a month. • Minute — Sets a minute in an hour • Hour — Sets an hour on a day • Day of month — Sets a day in a month • Enter day of week — Provides a list for setting a day in a week

Option	Definition
	• Month — Sets a month in a year (specified by a number from 1 to 12) • Year — Sets a year (four digits)
Start job immediately if it was not started at its original schedule	When selected, a scheduled job is started immediately if this has not happened according to the originally configured schedule. This can be the case, for example, when an appliance is temporarily shut down due to overload and a job was scheduled to run during this downtime. The job is then executed as soon as the appliance is up again.

Job Settings

Option	Definition
Job	Lets you select the type of a scheduled job. • Backup configuration — Creates a backup of an appliance configuration • Restore backup — Restores a backup of an appliance configuration • Upload file — Uploads a file to an external server using the HTTP or HTTPS protocol • Download file — Downloads a file onto the appliance using the HTTP or HTTPS protocol • Yum update — Performs a yum update on an appliance configuration **Note:** This scheduled job type is not available when an appliance runs in a FIPS-compliant mode
Unique job ID	Identifies a scheduled job. The characters specified in this string are case-sensitive
Job description	Provides an optional description of a scheduled job in plain-text format.
When this job has finished run job with ID	Provides the ID of a scheduled job that is to run immediately after the job configured here has finished. For this job, you must have configured the Activated by other job time setting.
Execute job on remote node	Provides a list for selecting other nodes of the configuration to execute a scheduled job. The list displays the host names for the other nodes. The scheduled job that you configure on this appliance is executed with its time and parameter settings on the selected node or nodes. A message is sent to the other node or nodes to inform them about the scheduled job.

Parameter Settings – Backup configuration

Option	Definition
Use most recent configuration	When selected, the scheduled job creates a backup from the most recent appliance configuration Format: |*<path name>/<file name with extension>*
Backup configuration path	Specifies the name of the path to the folder where the configuration is stored that should be used for the backup. Format: */opt/mwg/storage/default/configfolder* This setting is only available when Use most recent configuration is deselected.
Save configuration to path	Specifies the path and file name for a backup configuration. Format: */<path name>/<file name with file name extension>* You must set user rights for the folder you want to store the backup configuration in, making the appliance the owner who is allowed to write data into the folder. On the command line provided, for example, by a serial console, run the appropriate commands to create a folder or change the rights for an existing folder.

Parameter Settings – Restore backup

Option	Definition
Restore backup from file	Specifies the path and file name for the file that should be used to restore a backup. Format: |*<path name>/<file name with extension>*
Only restore policy	When selected, a scheduled job backs up only settings related to the web security policy that was implemented on an appliance. Other settings, for example, settings needed for connecting an appliance to a network are not restored.
Lock storage during restore	When selected, no other files can be stored on the appliance until the scheduled job has completely restored the backup configuration.
Password	Sets a password that is submitted for basic authentication.
Set	Opens the New Password window for setting a password. When a password has been set, the Set button is replaced by a Change button, which opens the New Password window for changing a password. This setting is only available when Enable basic authentication is selected.

Parameter Settings – Upload file

Option	Definition
File to upload	Specifies the path and file name for a file that should be uploaded.

Option	Definition
	Format: |*<path name>/<file name with extension>*
Destination to upload file to	Specifies the name of the path to the server that a file should be uploaded to under the HTTP or HTTPS protocol and the file name for storing the file on the server. Format: *http|https: //<URL>/<file name with extension>*
Enable basic authentication	When selected, basic authentication is required for uploading a file.
User name	Specifies a user name that is submitted for basic authentication. This setting is only available when Enable basic authentication is selected.
Password	Sets a password that is submitted for basic authentication.
Set	Opens the New Password window for setting a password. When a password has been set, the Set button is replaced by a Change button, which opens the New Password window for changing a password. This setting is only available when Enable basic authentication is selected.

Parameter Settings – Download file

Option	Definition
URL to download	Specifies a URL for the location of a file that should be downloaded under the HTTP or HTTPS protocol and the name of the file. Format: *http|https: //<URL>/<file name with extension>*
Save downloaded file to	Specifies a path to the location where a downloaded file should be stored and the file name for storing the file. Format: |*<path name>/<file name with extension>*
Enable basic authentication	When selected, basic authentication is required for downloading a file
User name	Specifies a user name submitted for basic authentication. This setting is only available when Enable basic authentication is selected.
Password	Sets a password that is submitted for basic authentication.
Set	Opens the New Password window for setting a password. When a password has been set, the Set button is replaced by a Change button, which opens the New Password window for changing a password. This setting is only available when Enable basic authentication is also selected.

Special updates (for an OEM use case only)

Settings for updating TrustedSource data used in the URL filtering process on a Central Management cluster of appliances

You can configure these settings to integrate an externally created, customized URL filtering list file in this process.

Note: You need an OEM agreement with McAfee to obtain a license for the tools that are required to create the external URL filtering list file.

Operating a Central Management cluster in the usual way does not require this list file.

Special updates

Option	Definition
Enable separate download of TrustedSource Database	When selected, the TrustedSource Database can be downloaded separately according to what you configure for this process.
Manual Update	Triggers a manual update of the database.
Download URL	Specifies a URL for downloading the database.
User	Specifies a user name that is submitted for the download.
Password	Sets a password that is submitted for the download.
Set	Opens the New Password window for setting a password. When a password has been set, the Set button is replaced by a Change button, which opens the New Password window for changing a password.
Enable download by interval	When selected, the TrustedSource Database can be downloaded in intervals according to what you configure for this process.
Update interval	Sets the time (in minutes) that elapses before the database is downloaded again. The time is set on a slider scale. Time range: 15-1440 minutes
Enable update proxies	When selected, proxies are used for downloading the TrustedSource Database according what you configure for this process.
Update proxies	Lists the proxies that have been set up to enable a download of the database.

The following table describes the elements of an entry in the Update proxies list.

Update proxies – **List entry**

Option	Definition
Host	Specifies a host name for an update proxy.
Port	Specifies a port on the host for an update proxy.
User	Specifies a user name that is submitted for running an update proxy.
Password	Sets a password that is submitted for running an update proxy.

Option	Definition
Comment	Provides a plain-text comment on an update proxy

Coaching system settings

Coaching system settings are general settings for time intervals related to quota management.

Quota Intervals for Synchronisation and Saving in Minutes

Settings for time intervals related to quota management

Quota Intervals for Synchronisation and Saving in Minutes

Option	Definition
Enable synchronization of quota data	When selected, quota data is synchronized according to what you configure for this process
Save interval	Sets the time (in minutes) that elapses before current quota values are saved again on an appliance. Quota values to be saved are, for example, the byte volumes that have been consumed by users.
Interval for sending updated quota data	Sets the time (in minutes) that elapses before current quota values are distributed again from an appliance to all nodes in a Central Management configuration. The distributed data includes the changes in quota values that have occurred since the last time that data were distributed from the appliance.
Interval for base synchronisation	Sets the time (in minutes) that elapses before quota values are synchronized again on all nodes in a Central Management configuration. The synchronization takes a snapshot of the current quota values on all appliances. The values that are most recent with regard to individual users are distributed to all appliances. The values are also distributed to nodes that were temporarily inactive and did not receive updates sent during that time. They are, furthermore, distributed to nodes that have been newly added to the configuration, so they did not receive any previous updates.
Cleanup database after	Sets the time (in days) that elapses before data is deleted in the quota database. Before data is deleted, a check is performed to see whether the data is obsolete. Data is obsolete if the time interval that has been configured for a quota management function has elapsed. For example, if a particular amount of bytes has been configured as volume quota for a user to be consumed during a month, the amount that the user actually consumed during a month becomes obsolete when a new month begins. The cleanup then deletes this data if the time configured under the Cleanup database after option has also elapsed.

Option	Definition
	Stored data becomes obsolete after a month for time quotas. For other quota management functions, other time intervals are relevant. For example, for coaching and authorized overriding, the cleanup cannot be performed before the allowed session time has elapsed.

Date and Time settings

The Date and Time settings are used for configuring the time servers that synchronize date and time of the appliance system. They also allow you to set the system time manually.

Date and Time

Settings for date and time of the appliance system

Date and Time

Option	Definition
Enable time synchronization with NTP servers	When selected, the appliance uses time servers under the NTP (Network Time Protocol) for time synchronization. The system time of the appliance is then synchronized with the time on the NTP servers. This will fail, however, if the delta between both times is too big. **Tip: Best practice**: Restart the appliance after configuring time synchronization with NTP servers. When the appliance restarts, it sets system time to the time on the NTP servers.
NTP server list	Provides a list for entering the servers that are used for time synchronization under the NTP protocol. The list elements are as follows: • String — Specifies the name of an NTP server. • Comment — Provides a plain-text comment on an NTP server.
Select time zone	Provides a list for selecting a time zone. Time synchronization performed by the NTP servers or manually set time refer to the time zone that you select here

Set System Time Manually

Settings for configuring time and date on the appliance system manually

Set System Time Manually

Option	Definition
Current date and time	Provides items for setting date and time of the appliance system. • Date — Enables you to enter a date by typing it in the field or using a calendar. • Calendar icon — Opens a calendar for selecting a date. After selecting a date on the calendar and clicking OK, the date appears in the date field.

Option	Definition
	• Time — Lets you specify a time by typing it. The system time of an appliance is then synchronized with the time on the NTP servers. This will fail, however, if the delta between both times is too big. **Tip: Best practice**: Restart the appliance after configuring time synchronization with NTP servers. When the appliance restarts, it sets system time to the time on the NTP servers.
Set now	Sets the date and time you have entered into the corresponding fields.

DNS settings

The DNS settings are usedr for configuring the domain name servers an appliance connects to for retrieving IP addresses that match the host names submitted in user requests.

Domain Name Service Settings

Settings for the IP addresses of different domain name servers

Domain Name Service Settings

Option	Definition
Primary domain name server	Specifies the IP address of the first server.
Secondary domain name server	Specifies the IP address of the second server.
Tertiary domain name server	Specifies the IP address of the third server.

ePolicy Orchestrator settings

The ePolicy Orchestrator settings are used for configuring the transfer of monitoring and other data from a Web Gateway appliance to a McAfee ePO server.

ePolicy Orchestrator Settings

Settings for transferring monitoring data to a McAfee ePO server

ePolicy Orchestrator Settings

Option	Definition
ePO user account	Specifies a user name for the account that allows the retrieval of monitoring data from an appliance.
Password	Sets a password for a user. Clicking Change opens a window for setting a new password.
Enable data collection for ePO	When selected, monitoring data for the McAfee ePO server is collected on an appliance.

Option	Definition
Data collection interval in minutes	Limits the time (in minutes) that elapse between data collections. The time is set on a slider scale, ranging from 10 minutes to 6 hours.

ePO DXL Settings

Settings for configuring the credentials submitted by Web Gateway when connecting to a McAfee ePO server to enable DXL messaging

ePO DXL Settings

Option	Definition
ePO host name	Specifies the host name that Web Gateway uses when connecting to a McAfee ePO server.
ePO user account	Specifies a name for the user account that Web Gateway submits when connecting to a McAfee ePO server.
ePO user password	Sets the password that Web Gateway submits when connecting to a McAfee ePO server. Clicking Set opens a window for setting a new password.
ePO server port	Specifies the port on the McAfee ePO server that listens to requests sent from Web Gateway. Default port: 8443
Agent handler port	Specifies the agent handler port that is used for communication between the McAfee ePO server and Web Gateway. Default port: 443
Rejoining ePO for DXL communication	When clicked, rejoins communication with the McAfee ePO server to complete the setup. A message informs you of the result.

External Lists system settings

The External Lists system settings apply to all external lists that are processed on the appliance.

Global Configuration

Setting for the internal cache on the appliance that stores external list data

Global Configuration

Option	Definition
Flush External Lists Cache	Removes the data that is stored in the internal cache.
Time before retry after failure	Limits the time (in seconds) that the External Lists module remembers a failure to retrieve data from a particular external source to the specified value.

Option	Definition
	The module will not perform retries for a source as long as it remembers the failure. We recommend that you keep the default value or modify it according to the requirements of your network. This way you avoid adding load by constant retries to a web server that is already overloaded.

File Data Source Configuration

Setting for the local file system that external list data can be retrieved from

File Data Source Configuration

Option	Definition
File system allowed for file data access	Specifies the path that leads to the folder for storing external lists within your local file system. External lists that data is retrieved from must be stored in this folder. Otherwise an attempt to retrieve the data will lead to an access-denied error. **Note:** When external list data is retrieved from an SQLite database, the path specified here is the path to the folder within your local file system that contains the database.

Web Data Source Configuration

Setting for all web services that are the sources of external list data

Web Data Source Configuration

Option	Definition
Check SSL certificate identity	When selected, a certificate that a web server submits in SSL-secured communication under the HTTPS protocol is verified The verification is performed according to the SSL scanning rules that are implemented on the appliance. This can, for example, lead to an error if the web server uses a self-signed certificate.

File Server settings

The File Server settings are used for configuring dedicated file server ports on a Web Gateway appliance to enable, for example, file downloads by clients.

HTTP Connector Port

Settings for dedicated file server ports on an appliance

HTTP Connector Port

Option	Definition
Enable dedicated file server port over HTTP	When selected, the dedicated HTTP file server ports that are configured on an appliance are enabled.

Option	Definition

HTTP connector	Specifies a dedicated HTTP port for connecting to the file server.
	You can enter more than one port here, separating entries by commas. Ports can range from 1024 to 65335.
	To set up ports within the range from 1 to 1023, you can create a port forwarding rule.
	Together with a port, you can enter an IP address. This means connecting to a file server on an appliance over this port requires that you specify both the port and this IP address.
	For example, there are two interfaces for connecting on an appliance with these IP addresses:
	`eth0: 192.168.0.10, eth1: 10.149.110.10`
	You enter this under HTTP connector:
	`4711, 192.168.0.10:4722`
	Then connecting to a file server on the appliance over port 4711 is allowed using both IP addresses, whereas connecting over port 4722 requires that IP address 192.168.0.10 is used.
	Restricting connections in this way might be useful, for example, if you want to set up an intranet.

Enable dedicated file server port over HTTPS	When selected, the dedicated HTTPS file server ports that are configured on an appliance are enabled.
HTTPS connector	Specifies a dedicated HTTPS port for connecting to the file server. You can enter more than one port here, separating entries by commas. Ports can range from 1024 to 65335. To set up ports within the range from 1 to 1023, you can create a port forwarding rule. Entering a port together with an IP address can be done in the same way as under HTTP connector and has the same meaning. Using the following options, you can specify a protocol and a list of valid ciphers for the HTTPS communication. • SSL protocol version — Specifies the version of the SSL protocol that is used for communication with the file server. You can select one of these versions or any combination of them. ◦ TLS 1.2 ◦ TLS 1.1 ◦ TLS 1.0 • Server cipher list — Specifies a string of Open SSL symbols used for encrypting communication with the file server.
Enable protection against cross-site scripting	When selected, the communication with the file server is protected against cross-site scripting. When a cross-site scripting attack is launched, malicious JavaScript code is inserted into messages that are sent during the communication. Adding the following header to messages prevents the execution of this attack: Header name: X-XSS-Protection

Option	Definition
	Header value: 1
Enable protection against clickjacking	When selected, the communication with the file server is protected against clickjacking. When a clickjacking attack is launched, messages that are sent during the communication are embedded in iFrames, which can be used to steal data. Adding the following header to messages prevents the execution of this attack: Header name: X-Frame-Options Header value: DENY

Hybrid settings

When configured, the hybrid settings allow Web Gateway to connect to and communicate with McAfee WGCS.

Hybrid synchronization

The Web Gateway policy is synchronized with McAfee WGCS at the interval you specify in the hybrid settings. You can also perform synchronization manually. Manual synchronization doesn't affect the synchronization interval or schedule which continues as before.

Configuring the hybrid settings

The hybrid settings allow you to configure synchronization without a proxy server.

Web Hybrid Configuration

Option	Definition
Synchronize policy to Cloud	When selected, allows you to configure the Web Hybrid settings and enables the hybrid solution.
Appliance for Synchronization	From the drop-down list, select the Web Gateway appliance whose policy you want synchronized with McAfee WGCS. If you are running multiple appliances in a Central Management configuration, this setting ensures that the McAfee WGCS policy is always synchronized with the same appliance.
Cloud address	Specifies the address that Web Gateway uses to communicate with McAfee WGCS. **Value:** `https://msg.mcafeesaas.com:443`
Cloud administrator account name	Specifies your McAfee ePO Cloud user name.
Cloud administrator account password	Specifies your McAfee ePO Cloud password. To change the password, click Set, then enter the new password and click OK.
Customer ID	Specifies your McAfee WGCS customer ID.
Local policy changes will be uploaded within the same interval as defined below	Specifies the synchronization interval. **Default:** 15 minutes **Range:** 10–60 minutes

Configuring the advanced hybrid settings

The advanced hybrid settings allow you to add a proxy server to the configuration.

Advanced Synchronization Settings

Option	Definition
Verify server certificate on SSL connections	When selected, Web Gateway verifies the proxy server certificate for SSL connections.
Use a proxy for synchronization	When selected, allows you to configure the proxy server settings. When the settings are configured, the Web Gateway policy is pushed to McAfee WGCS through the proxy server.
Proxy host	Specifies the IP address or host name of the server which is used as a proxy.
Proxy port	Specifies the port number on the proxy server that listens for Web Gateway requests to transfer synchronization data. **Default:** 8080
Proxy user	Specifies the user name that Web Gateway sends to the proxy server when transferring synchronization data.
Proxy password	Specifies the password that Web Gateway sends to the proxy server when transferring synchronization data. To change the password, click Set, then enter the new password and click OK.

Kerberos Administration settings

The Kerberos Administration settings are specific settings for the Kerberos authentication method.

Kerberos Administration

Settings for the Kerberos authentication method

Kerberos Administration

Option	Definition
Key tab file	Specifies the file that contains the master key required to access the Kerberos server. You can type a file name or use the Browse button to browse to the file and enter its name in the field. When a ticket is issued for authentication according to the Kerberos method, the master key is read on the appliance and used to verify the ticket. If you are running a load balancer that directs web requests to the appliance, tickets are issued for the load balancer and verified on the appliance. It is then not checked whether a request is directed to the appliance.
Kerberos realm	Specifies an administrative domain configured for authentication purposes.

Option	Definition
	Within the boundaries of this domain the Kerberos server has the authority to authenticate a user who submits a request from a host or using a service. The realm name is case sensitive, however. normally only uppercase letters are used, and it is good practice to make the realm name the same as that of the relevant DNS domain.
Maximal time difference between appliance and client	Limits the time (in seconds) that the system clocks on the appliance and its clients are allowed to differ to the specified value. Configuring Kerberos as the authentication method can lead to problems when particular browsers are used for sending requests: • When the Microsoft Internet Explorer is used in a version lower than 7.0, Kerberos authentication might not be possible at all. • When this explorer runs on Windows XP, Kerberos authentication might not work as expected. • When Mozilla Firefox is used, Kerberos authentication must be configured in the browser settings to enable this authentication method.
Enable replay cache	When selected, a ticket that is issued for authentication cannot be used more than once. **Note:** Selecting this option reduces authentication performance

License settings

The License settings are used for importing a license to an appliance. Information about the license is shown together with these settings, and options for reviewing the agreements on license and data usage.

License Administration

Settings for importing a license

License Administration

Option	Definition
Import license	Provides the options that are required for importing a license.
I have read and accept the end user license agreement	Provides a link to the End User License Agreement and a checkbox to select after reading the document. To import a license, the checkbox must be selected, otherwise the import options remains grayed out.
License file	Shows the name and path of the license file that has been selected after browsing the local file system. When the name and path appear in this field, more license information is shown under License information. The license is activated by clicking Save Changes.

Option	Definition
Browse	Opens the local file system to let you browse for a license file.

License Information

Information about an imported license and an option for reviewing the Data Usage Statement

License Information

Option	Definition
Status	Shows the name of a license file.
Creation	Shows the date when a license file was created.
Expiration	Shows the date when a license file expires.
License ID	Shows the ID of a license.
Customer	Shows the name of the license owner.
Customer ID	Shows the ID of the license owner.
Seats	Shows the number of workplaces in the license owner's organization that the license is valid for,
Evaluation	Shows whether the license has been evaluated.
Features	Lists the features of Web Gateway that are covered by the license.
I have read and understood the data usage statement	Provides a link to the Data Usage Statement.

Mobile Cloud Security settings

The Mobile Cloud Security settings are used to provide certificates and user-related information for the McAfee Mobile Cloud Security (MMCS) solution.

CA Certificates to Identify Mobile Devices

Settings for providing CA certificates

Option	Definition
CA certificates	Provides a list with entries for every CA certificate that has been added on Web Gateway. Each CA certificate is issued for use with a particular mobile device. To make these certificates available for cloud use, cloud synchronization must be enabled. You can enable this synchronization as an option of the Web Hybrid settings.

CA certificates - List entry

Option	Definition
Certificate	Specifies the name of a CA certificate file.
User	Specifies the name of a mobile device user that the CA certificate has been issued for.
User group	Specifies the name of the user group that the user belongs to. Specifying user group information is optional.
Comment	Provides a comment on a CA certificate in plain text.

Device Certificates Test

Settings for performing a certificate test

Option	Definition
Test device certificates	Clicking this button opens a window where you can perform a test for a CA certificate.

Mobile Device Management Solution

Settings for managing mobile devices

Option	Definition
VPN gateway address information	Provides a VPN Gateway address. You must specify this address when configuring a mobile device using a particular management solution.

Network Interfaces settings

The Network Interfaces settings are used for configuring the network interfaces of an appliance.

Configuring network interfaces

When configuring network interfaces on Web Gateway, we recommend setting up at least two and dedicating them to different purposes to ensure more resilience and higher throughput in every field of activities.

As a minimum, we recommend that you configure the following:

- Proxy network interface for proxy traffic
- Management network interface for all management-related traffic, such as user-interface traffic, cluster-communication traffic, or logging traffic

For more complex networks, we recommend configuring more network interfaces for different purposes. You might, for example, configure one interface for each of these fields of activities:

- Inbound proxy traffic
- Outbound proxy traffic
- Access to the Web Gateway user interface

  In a cluster of Web Gateway appliances, you might also run one appliance as a dedicated "UI appliance" to prevent increased user-interface access from impacting proxy traffic filtering.
- Cluster communication
- Pushing and pulling log files

To improve performance even further, you can also configure network bonding, which means that two or more network interfaces are combined to run as a single interface.

Network Interface Settings

Settings for network interfaces

Network Interface Settings

Option	Definition
Host name / Fully qualified domain name	Specifies the host name of an appliance. The name must be specified as fully qualified domain name.
Default gateway (IPv4)	Specifies the default gateway for web traffic under IPv4.
Default gateway (IPv6)	Specifies the default gateway for web traffic under IPv6.
Enable these network interfaces	Provides a list of network interfaces that are available for being enabled or disabled. The *eth0* network interface is by default included in the list and enabled.
IPv4	Provides options for configuring network interfaces under IPv4. The options are provided on a separate tab.
IPv6	Provides options for configuring network interfaces under IPv6. The options are provided on a separate tab.
Advanced	Provides options for configuring additional media. The options are provided on a separate tab.
Add VLAN	Opens a window for adding a network interface for VLAN traffic. **Note:** You can use this option to run VLANs under IPv4 or IPv6. To add a network interface, you specify a number as its ID and click OK. The interface name is composed of two parts, separated by a dot. The first part is the name and number of the interface that is enabled in the list of available network interfaces. The second part is the number that you specify. For example, if the *eth0* interface is enabled and you specify 1, a network interface for VLAN traffic is added as *eth0.1*. It is initially not enabled. The range of numbers for VLAN network interfaces is 1–4094. **Note:** After adding one or more network interfaces for VLAN traffic, you must also add their IDs to the parameters of the port redirects for the network mode that you are using. The window for adding or editing port redirects provides the Optional 802.1Q VLANs field for entering VLAN IDs. Separate multiple entries by commas.

Option	Definition
Delete	Deletes a selected network interface for VLAN traffic.

The following tables describe the options on the IPv4, IPv6, and Advanced tabs.

IPv4

Tab for configuring network interfaces under IPv4

IPv4

Option	Definition
IP settings	Lets you select a method to configure an IP address for a network interface. • Obtain automatically (DHCP) — The IP address is automatically obtained, using the Dynamic Network Host Protocol (DHCP). • Configure manually — The IP address is configured manually. • Disable IPv4 — IPv4 is not used for this interface.
IP address	Specifies the IP address of a network interface (manually configured).
Subnet mask	Specifies the subnet mask of a network interface (manually configured).
Default route	Specifies the default route for web traffic using the network interface (manually configured).
IP aliases	Provides a list of aliases for the IP address. • Add alias — Opens the Input window for adding an alias. **Note:** To enable usage of an alias, you must restart Web Gateway. After entering an alias here, an alert reminds you of the restart. You can perform the restart by running the following command from the command line of a system console: `service mwg restart` • Delete — Deletes a selected alias.

IPv6

Tab for configuring network interfaces under IPv6

IPv6

Option	Definition
IP settings	Lets you select a method to configure an IP address for a network interface. • Obtain automatically (DHCP) — The IP address is automatically obtained, using the Dynamic Network Host Protocol (DHCP). • Solicit from router — The IP address is obtained from a router. • Configure manually — The IP address is configured manually. • Disable IPv6 — IPv6 is not used for this interface.

Option	Definition
IP address	Specifies the IP address of a network interface (manually configured).
Default route	Specifies a default route for web traffic using the network interface (manually configured).
IP aliases	Provides a list of aliases for the IP address. • Add alias — Opens a window for adding an alias. **Note:** To enable usage of an alias, you must restart Web Gateway. After entering an alias here, an alert reminds you of the restart. You can perform the restart by running the following command from the command line of a system console: `service mwg restart` • Delete — Deletes a selected alias.

Advanced

Tab for configuring advanced network interface functions.

Note: The tab provides different options when the currently selected network interface is a bonding interface. These options are described in a second table.

Advanced

Option	Definition
Media	Lets you select additional media for use with a network interface. • Automatically detect — Media for use with a network interface are automatically detected if available in the network environment of an appliance. • 1000BaseT-FD, 1000Base-HD, ... — The selected media item is used with a network interface.
Bond enabled	When selected, the currently selected network interface, for example, eth2, is configured as a bonded interface that is subordinated to a bonding interface. • Name — Specifies the name of the bonding interface.
MTU	Limits the number of bytes in a single transmission unit to the specified value. The default number is 1500. The minimum and maximum numbers depend on whether a network interface is configured under IPv4 or IPv6. • IPv4 — minimum: 576, maximum: 9216 • IPv6 — minimum: 1280, maximum: 9216 **Note:** If the configured number was set to less than either of these minimum values in an earlier product version, it is now set to 576 under IPv4 and 1280 under IPv6, respectively, by the configuration system on Web Gateway.

Option	Definition
	If it was set to more than the maximum value, it is now set to the default value of 1500. This option is not accessible if the following applies: • This network interface is configured as a bonded interface in a bonding configuration. In this case, Bond enabled is selected above.

The following table describes the options provided on the Advanced tab when a bonding interface is selected.

Advanced

Option	Definition
Bonding options	Provides options for a bonding interface. • Mode — Specifies the mode used to let the bonded network interfaces in the bonding configuration become active. ◦ Active/Passive — When selected, only one bonded interface is active at any time. A different bonded interface becomes active only if the active bonded interface fails. The MAC address of the bonding interface is only visible externally on one port, which avoids address confusion for a network switch. **Note:** This mode is referred to in some system messages as *mode 1*. The mode is selected by default. ◦ 802.3ad/LACP — When selected, all bonded interfaces in the bonding configuration are active. The bonded interface for outgoing traffic is selected according to the configured hash policy. **Note:** This mode is referred to in some system messages as *mode 4*. When this mode is selected, the LACP rate and Hash policy options become accessible. • Miimon — Sets the time interval (in milliseconds) for sending the polling messages of the MII monitoring program. The default interval is 100 milliseconds. • LACP rate — Sets the transmission rate for sending LACP-DU data packets in 802.3ad mode. ◦ Slow — When selected, data packets are sent every 30 seconds. This transmission rate is selected by default. ◦ Fast — When selected, data packets are sent every second. • Hash policy — Determines the way that a hash value is calculated for a bonding configuration. ◦ Layer2 — When selected, a combination of layer 2 values is used to calculate the hash. The values

Option	Definition
	that are included in this combination are hardware MAC addresses and packet type ID addresses. This hash policy is selected by default. ◦ Layer2+3 — When selected, a combination of layer 2 and layer 3 protocol information is used to calculate the hash.

Network Protection settings

The Network Protection settings are system settings that are used for configuring protective rules for traffic coming in to an appliance from your network.

We recommend configuring Network Protection settings in explicit proxy mode only. You can configure these settings also in the following modes, but you will not receive support when issues occur::

- Proxy HA
- Transparent Router

Network Protection Rules

Settings for configuring network protection rules

Network Protection Rules

Option	Definition
Enable network protection	When selected, the settings configured in the following for network protection are enabled.
Input policy	Lets you select the action taken on incoming traffic. Incoming traffic can either be dropped or accepted.
Allow Ping requests	When selected, the appliance accepts and answers Ping requests.
Exceptions from default policy	Provides a list for entering the network devices that send traffic to an appliance. Traffic from these devices is not handled according to the rules that are currently implemented. When these rules drop incoming traffic, traffic sent from the devices listed here is accepted and vice versa.

The following table describes an entry in the list of exceptions from the default policy.

Exceptions from default policy – List entry

Option	Definition
Device	Specifies the name of a network device that sends traffic to the appliance. Typing * or no input means all devices are covered.
Protocol	Specified the protocol used for sending traffic.

Option	Definition
Source	Specifies the IP address or address range of the network device or devices that send traffic to the appliance.
Destination port	Specifies the port on an appliance that is the destination of network traffic.
Comment	Provides a plain-text comment on an exception.

Persistent Data Storage settings

Persistent Data Storage settings are settings for time intervals related to storing data persistently.

Persistent Data Storage is shortly referred to as *PDStorage*.

It enables you to store data beyond any particular transaction that is completed on Web Gateway when an incoming request is processed through all filtering cycles that apply.

When a transaction is completed, values that were retrieved for properties during the transaction are not preserved, but overwritten during the next transaction.

Using PDStorage, you can persistently store data and continue to use it in any following transaction. Data is stored then in a key-value format. You can limit the time for storing the data.

For example, you can store the IP address of a client system that a user sends a request from. When the same user sends another request, you can have a rule that includes suitable PDStorage properties compare the client IP address coming in with this request to the one that is persistently stored.

If the two differ, the rule will, for example, block the request. This way you can restrict web usage for a user to using one particular client system only.

PDStorage Intervals for Synchronisation and Saving in Minutes

Settings for time intervals related to Persistent Data Storage

PDStorage Intervals for Synchronisation and Saving in Minutes

Option	Definition
Save interval	Sets the time (in minutes) that elapses before persistent data is saved again on an appliance.
Enable synchronization of PDStorage data	When selected, persistent data is synchronized according to what you configure for this process.
Interval for sending PDStorage data	Sets the time (in minutes) that elapses before persistent data is distributed again from an appliance to all nodes in a Central Management configuration.
Delay in seconds between the PDStorage messages to be sent	Sets the time (in seconds) that elapses until another PDStorage message follows the message that was sent before it.

PDStorage Memory Management

Setting for the memory size that is available to Persistent Data Storage

PDStorage Memory Management

Option	Definition
Maximum byte size for PDStorage	Limits the size (MiB) of the memory where persistent data is stored.

Port Forwarding settings

The Port Forwarding settings are used for configuring rules that let an appliance forward web traffic sent from a port on a particular host to another port.

Port Forwarding

Settings for configuring port forwarding rules

Port Forwarding

Option	Definition
Port forwarding rules	Provides a list of port forwarding rules.

The following table describes an entry in the list of port forwarding rules.

Port forwarding rules – List entry

Option	Definition
Source host	Specifies the IP address of a host that is the source of web traffic in a port forwarding rule.
Bind IP	Specifies the bind IP address.
Target port	Specifies the port that web traffic from the source host is forwarded to.
Destination host	Specifies the IP address of the host that is the destination of web traffic sent from the source host.
Destination port	Specifies the port on the destination host used for listening to web traffic coming in from the source host.
Comment	Provides a plain-text comment on a port forwarding rule.

The Port Forwarding settings continue as follows.

Port Forwarding (continued)

Option	Definition
Enable extended connection logging	When selected, all logs for port forwarding are stored on the appliance system under */var/log/mwg_fwd.log*. The logging options that you configure here apply to all port forwarding that performed under the configured port forwarding rules. The stored log files can also be viewed on the user interface under the Troubleshooting top-level menu.

Option	Definition
	Select the appliance that you want to view log files for, then select Log files and open the system folder.
Customize extended logging fields	When selected, the input fields for configuring the type of data that should be logged become accessible.
Log on success	Lets you enter the type of data to be logged when web traffic is successfully forwarded. You can enter one or more of the following data types by typing them in capital letters, separated by commas: PID, HOST, USERID, EXIT, DURATION, TRAFFIC.
Log on failure	Lets you enter the type of data to be logged when forwarding web traffic failed. You can enter one or more of the following data types by typing them in capital letters, separated by commas: HOST, USERID, ATTEMPT. HOST data is logged by default.

Proxies settings

The Proxies settings are used for configuring proxies on a Web Gateway appliance.

For more information, see the sections on proxies and their settings in the *McAfee Web Gateway Product Guide*.

Static Routes settings

The Static Routes settings are used for configuring routes that always use the same gateway and interface on this gateway when web traffic is routed from an appliance to a particular host.

Static Routes

Settings for static routes under IPv4 or IPv6

Static Routes

Option	Definition
Static routes list	Provides a list of static routes for transmitting web traffic under IPv4 or IPv6.

The following table describes an entry in the list of static routes.

Static routes list – List entry

Option	Definition
Destination	Specifies the IP address and (optionally) net mask of the host that is the destination of a static route.
Gateway	Specifies the IP address of the gateway for routing web traffic from the appliance to a host.

Option	Definition
Device	Specifies the interface used on a gateway for a static route.
Description	Provides a plain-text description of a static route.
Comment	Provides a plain-text comment on a static route.

Source-based routing

Settings for source-based routing under IPv4 or IPv6

Source-based routing

Option	Definition
Source-based routing for IPv4	When selected, source-based routing is performed under IPv4.
Source-based routing for IPv6	When selected, source-based routing is performed under IPv6.
Static source routing table number	Provides a list of entries for source routing tables that are used to route the traffic that is sent and received through the management user interface.
Source-based routing list for IPv4	Provides a list of routing entries for the traffic that is sent and received through the management user interface. These routing entries are for a network where IPv4 is followed.
Source-based routing list for IPv6	These routing entries are for a network where IPv6 is followed.

The following table describes an entry in the list for static source routing tables.

Static source routing table number – List entry

Option	Definition
Source information to look up routing table	Specifies the source IP address of the traffic that is routed according to the configured static source routing table.
Routing table number	Specifies the number of the routing table for routing the traffic that is sent and received through the management user interface.
Comment	Provides a plain-text comment on a static source routing table.

The following table describes an entry in the list for source-based routing under IPv4.

Source-based routing list for IPv4 – List entry

Option	Definition
Destination	Specifies the IP address range (in CIDR notation) for the destinations of the traffic that is sent through the management network interface.

Option	Definition
Routing table number	Specifies the number of the routing table for routing the traffic that is sent and received through the management user interface.
Gateway	Specifies the IP address of the gateway for the traffic that is sent and received through the management network interface.
Device	Specifies the name of the network interface that is configured as the management network interface.
Source IP	Specifies the IP address of the network interface that is configured as the management network interface. This address is the source IP address of the traffic that is routed according to the routing table.
Comment	Provides a plain-text comment on an entry for source-based routing.

The following table describes an entry in the list for source-based routing under IPv6.

Source-based routing list for IPv6 – **List entry**

Option	Definition
Destination	Specifies the IP address range (in CIDR notation) for the destinations of the traffic that is sent through the management network interface.
Routing table number	Specifies the number of the routing table for routing the traffic that is sent and received through the management user interface.
Gateway	Specifies the IP address of the gateway for the traffic that is sent and received through the management network interface.
Device	Specifies the name of the network interface that is configured as the management network interface.
Source IP	Specifies the IP address of the network interface that is configured as the management network interface. This address is the source IP address of the traffic that is routed according to the routing table.
Comment	Provides a plain-text comment on an entry for source-based routing.

Telemetry settings

The Telemetry settings are used for configuring the collection of feedback data about web objects that are potentially malicious, as well as about policy configuration.

Feedback Settings

Settings for collecting feedback data

Note: You can separately enable or disable each of the following options.

Feedback Settings

Option	Definition
Send feedback to McAfee about system information and suspicious URLs to improve its threat prediction and protection services	When selected, feedback data is collected and sent to special McAfee feedback servers. McAfee collects this data to analyze it and improve the threat prediction and protection features of Web Gateway. For more information, see the *Data Usage Statement*.
Send feedback to McAfee about potentially malicious websites	When selected, relevant data for virus and malware filtering is collected and sent to a special McAfee feedback server.
Send feedback to McAfee about dynamically classified websites	When selected, relevant data for classifying websites is collected and sent to a special McAfee feedback server.
Send feedback to McAfee about policy configuration to improve the product	When selected, relevant data for policy configuration is collected and sent to a special McAfee feedback server.

Further Information

Link to the Data Usage Statement

Further Information

Option	Definition
Data Usage Statement	Provides a link to the data usage statement, which explains: • What McAfee uses collected feedback data for • What data is collected • How data collection can be turned off for different types of data **Note:** The data usage statement has also been presented to you at the initial setup of the appliance.

Advanced Settings

Advanced settings for collecting feedback data

Advanced Settings

Option	Definition
Use upstream proxy	When selected, a proxy server is used to send feedback data to McAfee.
IP or name of the proxy	Specifies the IP address or host name of the proxy server.
Port of the proxy	Specifies the port number of the port on the proxy server that listens for requests to send feedback data. The port number can range from 1 to 65635. The default port number is 9090.
User name	Provides the user name that is required for logging on to the proxy server.

Option	Definition
Password	Provides the password that is required for logging on to the proxy server. Clicking Set opens a window for setting the password.
Choose feedback server	When selected, an IP address and port number can be configured for the server that feedback data is sent to.
IP of the server	Specifies the IP address of the feedback server.
Port of the server	Specifies the port number of the port on the feedback server that listens for requests to send data. The port number can range from 1 to 65635. The default port number is 443.
Port of the server	When selected, feedback-sending activities are logged.

User Interface settings

The User Interface settings are used for configuring the local user interface on a Web Gateway appliance. This includes the configuration of ports, the logon page, a certificate for communication under HTTPS, and other items.

UI Access

Settings for configuring access to the interface of an appliance

UI Access

Option	Definition

HTTP connector	Provides options for configuring access to the interface of an appliance under HTTP.
	• Enable local user interface over HTTP — When selected, the HTTP ports that are configured on an appliance for connecting to the interface are enabled.
	• HTTP connector — Specifies an HTTP port for connecting to the interface.
	You can enter more than one port here, separating entries by commas. Ports can range from 1024 to 65335.
	Together with a port, you can enter an IP address. This means connecting to the interface of an appliance over this port requires that you specify both the port and this IP address.
	For example, there are two interfaces for connecting on an appliance with these IP addresses:
	`eth0: 192.168.0.10, eth1: 10.149.110.10`
	You enter this under HTTP connector:
	`4711, 192.168.0.10:4722`
	Then connecting to a file server on the appliance over port 4711 is allowed using both IP addresses, whereas connecting over port 4722 requires that IP address 192.168.0.10 is used.
	Restricting connections in this way might be useful, for example, if you want to set up an intranet.

Option	Definition
	• Enable REST interface over HTTP — When selected, you can use the HTTP ports that are configured to connect to the REST interface.
HTTPS connector	Provides options for configuring access to the interface of an appliance under HTTPS. • Enable local user interface over HTTPS — When selected, the HTTP ports that are configured on an appliance for connecting to the interface are enabled. • HTTPS connector — Specifies an HTTPS port for connecting to the interface. You can enter more than one port here, separating entries by commas. Ports can range from 1024 to 65335. Entering a port together with an IP address can be done in the same way as under HTTP connector and has the same meaning. • Enable REST interface over HTTPS — When selected, you can use the HTTP ports that are configured to connect to the REST interface. Using the following options, you can specify a protocol and a list of valid ciphers for the HTTPS communication. • SSL protocol version — Specifies the version of the SSL protocol that is used for communication with the interface. ◦ TLS 1.2 ◦ TLS 1.1 ◦ TLS 1.0 • Server cipher list — Specifies a string of Open SSL symbols used for encrypting communication with the interface.
HTTPS client certificate connector	Provides options for configuring a client certificate connector. • Enable client certificate authentication — When selected, client certificate authentication can be performed. • HTTPS connector for client certificate authentication — Specifies a port for connecting to the interface when client certificate authentication is performed. You can enter more than one port here, separating entries by commas. Ports can range from 1024 to 65335. Entering a port together with an IP address can be done in the same way as under HTTP connector and has the same meaning. • Redirect target after authentication — When selected, a request is redirected after client certificate authentication has successfully been performed. • Redirection host and port — Specifies the host system and the port on the system that requests are redirected to.
Miscellaneous	Provides miscellaneous options for configuring access to the interface of an appliance. • Session timeout — Limits the time (in minutes) that elapses before a session on the interface is closed if no activities occur. The range for the session timeout is 1–99,999 minutes. The timeout is 30 minutes by default.

Login Page Options

Settings for the page that is used to log on to the interface of an appliance

Login Page Options

Option	Definition
Allow browser to save login credentials	When selected, credentials submitted by a user for logging on to the interface are saved by the browser.
Restrict browser session to IP address of user	When selected, a session for working with the interface is only valid as long as the IP address of the client that the user started this session from remains the same.
Let user decide to restrict session for IP address or not	When selected, it is up to the user who started a session for working with the interface whether it should be valid only for the IP address of the client that the session was started from.
Allow multiple logins per login name	When selected, more than one user can log on to the interface under the same user name and password.
Use HTTPOnly session cookies (applet loading may take longer)	When selected, HTTPOnly cookies are used for a session with the user interface.
Enable protection against cross-site scripting and clickjacking	When selected, the page used by the administrator for logging on to the interface of a Web Gateway appliance from a browser is protected against a common type of attack. The attack can be performed by combining two methods. Two HTTP headers are added when the page is sent to the browser to prevent these methods from being executed. • **Cross-site scripting** — Malicious JavaScript code is inserted in the page, which is executed when the administrator responds to a prompt on the page, for example, by entering a user name. Adding the following header to messages prevents the execution of this attack: Header name: X-XSS-Protection Header value: 1 • **Clickjacking** — The page is embedded in an iFrame, which can be used to steal the data that is entered on the page. Adding the following header to messages prevents the execution of this attack: Header name: X-Frame-Options Header value: DENY
Maximum number of active applet users	Limits the number of users that can be logged on to the interface at the same time. The maximum number of users is 20 by default.
Login message	Provides the following options for displaying an additional message on the page used for logging on to the interface. **Note:** You can work with these options if you want to display a message, for example, to comply with internal policies or external regulations. • Show on login page — When selected, the text that you type in the HTML message field, appears on the logon page.

Option	Definition
	• HTML message — The text that you type in this field appears on the logon page.

User Interface Certificate

Settings for a certificate that is used in SSL-secured communication over the HTTPS port for the interface of an appliance.

User Interface Certificate

Option	Definition
Subject, Issuer, Validity, Extensions	Provide information about the certificate that is currently in use.
Import	Opens the Import Certificate Authority window for importing a new certificate.
Certificate chain	Displays a certificate chain that is imported with a certificate.

Import Certificate Authority window

Settings for importing a certificate that is used in SSL-secured communication

Import Certificate Authority window

Option	Definition
Certificate	Specifies the name of a certificate file. The file name can be entered manually or by using the Browse button in the same line.
Browse	Opens the local file manager to let you browse for and select a certificate file.
Private key	Specifies the name of a private key file. The file name can be entered manually or by using the Browse button in the same line. Only keys that are AES-128-bit encrypted or unencrypted keys can be used here.
Browse	Opens the local file manager to let you browse for and select a private key file.
Password	Sets a password that allows the use of a private key.
Import	Opens the Import Certificate Authority window for importing a new certificate.
OK	Starts the import process for the specified certificate.
Certificate chain	Specifies the name of a certificate chain file. The file name can be entered manually or by using the Browse button in the same line.
Browse	Opens the local file manager to let you browse for and select a certificate chain file.

Option	Definition
	After importing a certificate with a certificate chain, the certificate chain is displayed in the Certificate chain field of the User Interface Certificate settings.

Memory Settings

Settings for the memory that is available when working with the interface of an appliance

Memory Settings

Option	Definition
Amount of maximum memory available for GUI applet	Limits the amount of memory (in MiB) that is available for the interface applet. The range for the available maximum is 100–999 MiB. The available maximum is 512 MiB by default.
Amount of maximum memory available for MWG UI backend	Limits the amount of memory (in MiB) that is available for the backedn of the interface. The range for the available maximum is 100–9999 MiB. If no value is specified here, the default maximum of 512 MiB is configured.

REST Settings

Settings for configuring use of the REST interface to work with an appliance

REST Settings

Option	Definition
Maximum size of a REST request	Limits the size (in MiB) of a request that is sent to the REST interface. **Note:** The maximum amount of memory that is available when working with the REST interface is 200 MiB. The maximum size of a request is 2 MiB by default.
Maximum memory per REST session	Limits the amount of memory (in MiB) that is available for a session when working with the REST interface. **Note:** The maximum amount of memory that is available when working with the REST interface is 200 MiB. The maximum amount of memory for a session is 10 MiB by default.
Maximum number of active REST users	Limits the number of users that can work with the REST interface at the same time. The maximum number of users is 20 by default.

Windows Domain Membership settings

The Windows Domain Membership settings are used for joining an appliance to a Windows domain.

Join Domain

Settings for joining an appliance to a Windows domain

Join Domain

Option	Definition
Windows domain name	Specifies the name of the domain.
McAfee Web Gateway account name	Specifies the name of an account for an appliance.
Overwrite existing account	When selected, an existing account is overwritten.
Use NTLM version 2	When selected, NTLM version 2 is used.
Timeout for requests to this NTLM domain	Limits the time (in seconds) that elapses before processing stops for a request sent from an appliance to a domain controller if no response is received to the specified value.
Wait time for reconnect to domain controller	Specifies the time (in seconds) that elapses before another attempt is made to connect to a domain controller after a previous attempt failed. The allowed range is from 5 to 300 seconds.
Configured domain controllers	Provides a list for entering the domain controllers that an appliance can connect to in order to retrieve authentication information. Entries must be separated by commas.
Number of active domain controllers	Maximum number of configured domain controllers that can be active at the same time The allowed range is from 1 to 10.
Administrator name	Specifies the logon name of an existing administrator account that has privileges to join an appliance to a domain by creating a machine account in Active Directory. Logon name and password are only used once to create the machine account. They are not stored.
Password	Specifies the password of the existing administrator account.

Module settings

Module settings are used to configure the behavior of modules on a Web Gateway appliance. These modules are also known as engines or filters.

For example, the Anti-Malware module calls the scanning engines, such as the Gateway Anti-Malware (GAM) engine, when the body of a response sent by a web server should be scanned for infections.

By configuring the settings for this module, you can modify the scanning process. Depending on what you configure, the module might not call the GAM engine, which is the default, but a different engine for scanning.

Other modules are the URL Filter module, the TIE Filter module, or the Authentication module.

Different settings for a module

A module can have one particular instance of settings or several. Different instances of module settings are distinguished by their names. Usually, they differ in how the values of the various settings options are configured.

For example, after the initial setup of Web Gateway, there is one instance of the settings for the Anti-Malware module available by default. The settings name for this instance is Gateway Anti-Malware settings.

When the module runs with these settings, it calls the GAM engine for scanning, as this behavior is configured for one of the settings options.

After importing the Advanced Threat Defense rule set, however, a second instance of settings for the module is available. Its name is Gateway ATD settings.

When the module uses these settings, a web object is passed on from Web Gateway to Advanced Threat Defense for scanning, as the value for the relevant option within the settings differs now from the value for the same option in the default settings.

You can also create and configure settings instances of your own for any of these modules to let them show the behavior that meets your requirements.

Anti-Malware settings

The Anti-Malware settings are the settings for the Anti-Malware module, which handles activities related to anti-malware filtering on a Web Gateway appliance.

For more information, see the sections on anti-malware filtering and Anti-Malware settings in the *McAfee Web Gateway Product Guide*.

Authentication settings

The Authentication settings are the settings for the Authentication module, which handles authentication of users and user groups.

Authentication Method

Settings for selecting an authentication method.

Authentication Method

Option	Definition
Authentication method	Provides a list for selecting an authentication method. You can select one of the following: • NTLM • NTLM-Agent • User Database

Option	Definition
	• LDAP **Note:** If you want to configure Secure LDAP, also known as *LDAPS*, you must work with LDAP version 3. This version can be selected under LDAP Specific Parameters. It is by default selected. • RADIUS • Kerberos • SSL Client Certificate • Authentication Server • One-Time Password • SWPS (McAfee® Client Proxy) After selecting a method, settings that are specific to it appear below the common settings

Authentication Test

Settings for testing whether a user with given credentials would be authenticated.

Authentication Test

Option	Definition
User	Specifies the user name that is tested.
Password	Specifies the tested password.
Authenticate User	Executes the test.
Test result	Displays the outcome of the test.

Common Authentication Parameters

Settings common to all authentication methods.

Note: There is also an advanced setting that is common to all authentication methods. It is described at the end of this main section after the last of the subsections for the specific authentication parameters.

Common Authentication Parameters

Option	Definition
Proxy Realm	Specifies the location of the proxy that receives requests from users who are asked to authenticate.
Authentication attempt timeout	Limits the time (in seconds) that elapses before the authentication process terminates if not completed successfully to the specified value.
Use authentication cache	When selected, authentication information is stored in a cache. Authentication is then based on this stored information, rather than on information retrieved from an authentication server or the internal user database.
Authentication cache TTL	Limits the time (in minutes) that authentication information is stored in the cache to the specified value.

NTLM Specific Parameters

Settings for the NTLM authentication method.

NTLM Specific Parameters

Option	Definition
Send domain and machine name to the client	When selected, the names of the appliance and its domain are sent to the client that a user who is to be authenticated sent a request from. An appliance can, however, be joined to more than one domain, so different domain names can be used when connecting to a client, which can lead to problems with user authentication. Sending a particular domain name to the client might result in an authentication failure because a particular user name is unknown in this domain. Web browsers do usually not require domain name information, but some third-party applications that Web Gateway works with might require it. So we recommend proceeding as follows: • If an appliance has been joined to only one domain: Select this option. • If an appliance has been joined to more than one domain: Leave this option deselected. There are, however, applications that require this option to be selected anyway. Otherwise they will close the connection to Web Gateway This applies, for example, to some .NET based applications as well as to some popular open-source products, such as the Cntlm proxy.
Default NTLM domain	Specifies the name of the default Windows domain used for looking up authentication information. This is one of the domains you have configured on the Appliances tab of the Configuration top-level menu.
Get global groups	When selected, information on global user groups is searched for on the Windows domain server.
Get local groups	When selected, information on local user groups is searched for on the Windows domain server.
Prefix group name with domain name (domain\group)	When selected, the name of the Windows domain appears before the name of the user group when authentication information on this group is sent from the domain server.
Enable basic authentication	When selected, the basic NTLM authentication method is applied to authenticate users. Information that a user submits for authentication is then sent in plain-text format (less secure) to the Windows domain server.
Enable integrated authentication	When selected, the integrated NTLM authentication method is applied to authenticate users.

Option	Definition
	Information that a user submits for authentication is then encrypted before it is sent to the Windows domain server.
Enable NTLM cache	When selected, NTLM authentication information is stored in this cache. Authentication is then based on this stored information, rather on information retrieved from the Windows domain server.
NTLM cache TTL	Limits the time (in seconds) that authentication information is stored in this cache to the specified value.
International text support	Specifies a set of characters used by default for a request sent from a client, for example, ISO-8859-1.

NTLM Agent Specific Parameters

Settings for the NTLM Agent authentication method.

NTLM Agent Specific Parameters

Option	Definition
Use secure agent connection	When selected, the connection used for communicating with the NTML Agent is SSL-secured
Authentication connection timeout in seconds	Limits the time (in seconds) that elapses before the connection to the NTLM Agent is closed if no activities occur on it to the specified value.
Agent Definition	Provides a list for entering the agents that are involved in performing NTLM authentication.
Default NTLM domain	Specifies the name of the default Windows domain used for looking up authentication information. This is one of the domains you have configured on the Appliances tab of the Configuration top-level menu.
Get global groups	When selected, information on global user groups is searched for on the Windows domain server.
Get local groups	When selected, information on local user groups is searched for on the Windows domain server.
Prefix group name with domain name (domain\group)	When selected, the name of the Windows domain appears before the name of the user group when authentication information on this group is sent from the domain server.
Enable basic authentication	When selected, the basic NTLM authentication method is applied to authenticate users. Information that a user submits for authentication is then sent in plain-text format (less secure) to the Windows domain server.
Enable integrated authentication	When selected, the integrated NTLM authentication method is applied to authenticate users.

Option	Definition
	Information that a user submits for authentication is then encrypted before it is sent to the Windows domain server.
Enable NTLM cache	When selected, NTLM authentication information is stored in this cache. Authentication is then based on this stored information, rather on information retrieved from the Windows domain server.
NTLM cache TTL	Limits the time (in seconds) that authentication information is stored in this cache to the specified value.
International text support	Specifies a set of characters used by default for a request sent from a client, for example, ISO-8859-1.

User Database Specific Parameters

Settings for the User Database authentication method.

User Database Specific Parameters

Option	Definition
Send domain and machine name to the client	When selected, the names of the appliance and the domain it has been assigned to are sent to the client that a user who is to be authenticated sent a request from.
Enable basic authentication	When selected, the basic NTLM authentication method is applied to authenticate users. Information that a user submits for authentication is then sent in plain-text format (less secure) to the Windows domain server.
Enable integrated authentication	When selected, the integrated NTLM authentication method is applied to authenticate users. Information that a user submits for authentication is then encrypted before it is sent to the Windows domain server.
Enable NTLM cache	When selected, NTLM authentication information is stored in this cache. Authentication is then based on this stored information, rather on information retrieved from the Windows domain server.
NTLM cache TTL	Limits the time (in seconds) that authentication information is stored in this cache to the specified value.
International text support	Specifies a set of characters used by default for a request sent from a client, for example, ISO-8859-1.

LDAP Specific Parameters

Settings for the LDAP authentication method.

LDAP Specific Parameters

Option	Definition
LDAP server(s) to connect to	Provides a list for entering the LDAP servers that authentication information is retrieved from.
List of certificate authorities	Provides a list for entering the certificate authorities that issue certificates when a Secure LDAP (S-LDAP) connection is used for communication with an LDAP server.
Credentials	Specifies the user name of an appliance for logging on to an LDAP server.
Password	Sets the password for a user name. The Set button opens a window for configuring a new password.
International text support	Specifies a set of characters used by default for a request sent from a client, for example, ISO-8859-1.
Enable LDAP version 3	When selected, version 3 of the LDAP protocol is used. **Note:** If you want to configure Secure LDAP authentication, also known as *LDAPS*, it is this LDAP version that you must use. This version is by default selected.
Allow LDAP library to follow referrals	When selected, the lookup of user information can be redirected from the LDAP server to other servers.
Connection live check	Limits the time (in minutes) that elapses between checks to see whether the connection to the LDAP server is still active to the specified value.
LDAP operation timeout	Limits the time (in seconds) that elapses before the connection to the LDAP server is closed if no communication occurs to the specified value.
Base distinguished name to user objects	Specifies the Distinguished name (DN) in the directory on an LDAP server where the lookup of user attributes should begin.
Map user name to DN	When selected, the name of the user who asks for authentication must map to a DN (Distinguished Name). This name identifies the user in the directory on the LDAP server
Filter expression to locate a user object	Specifies a filtering term for restricting the lookup of user attributes. To substitute the user name in the filtering term, u% is used as a variable.
Get user attributes	When selected, user attributes are looked up on the LDAP server to authenticate a user.
User attributes to retrieve	Provides a list for entering the user attributes that should be retrieved from an LDAP server.

Option	Definition
Attributes concatenation string	Specifies a string for separating user attributes found by a lookup, for example, / (slash).
Get groups attributes	When selected, user group attributes are also looked up on the LDAP server to authenticate a user.
Base distinguished name to group objects	Specifies the Distinguished name (DN) in the directory on the LDAP server where the lookup of group attributes should begin
Filter expression to locate a group object	Specifies a filtering term for restricting the lookup of group attributes. To substitute the user name in the filtering term, *u%* is used as a variable.
Group attributes to retrieve	Provides a list for entering the group attributes that should be retrieved from an LDAP server.

Digest Authentication

Settings for LDAP digest authentication.

Digest Authentication

Option	Definition
Enable digest authentication	When selected, digest authentication is performed as method for authenticating users under the LDAP authentication method.
User attribute with password hash	Specifies the attribute of a user entry on the LDAP server that stores the value for the authentication hash.
Nonce maximal use count	Sets a limit to repeated uses of the nonce (number only once) that is transmitted in the authentication process and required as a parameter for calculating the authentication hash. The maximum number of times that a nonce can be used by default is 100.
Nonce maximal TTL	Sets a limit to the time period (in minutes) that a nonce remains valid. The maximum time that a nonce can remain valid by default is 30 minutes.
Enable digest URI check	When selected, a check is performed to ensure that the URL that a client sends as a parameter for calculating the authentication hash is the same as the URL that this client sends in its request for accessing a particular destination in the web. If this check fails, the request is blocked. As this check might also fail due to problems with the different formats that the browsers on the clients use for sending URLs, it is optional. The check is enabled by default.

Option	Definition
Allow digest authentication only	When selected, digest authentication must always be performed if a user is to be authenticated under the LDAP authentication method.

Novell eDirectory Specific Parameters

Settings for the Novell eDirectory authentication method.

Novell eDirectory Specific Parameters

Option	Definition
LDAP server(s) to connect to	Provides a list for entering the eDirectory servers that take the role of LDAP servers in providing authentication information.
List of certificate authorities	Provides a list for entering the certificate authorities that issue certificates when a Secure LDAP (S-LDAP) connection is used for communication with an LDAP server.
Credentials	Specifies the user name of an appliance for logging on to an LDAP server.
Password	Sets a password for a user name. The Set button opens a window for configuring a new password.
International text support	Specifies a set of characters used by default for a request sent from a client, for example, ISO-8859-1.
Enable LDAP version 3	When selected, version 3 of the LDAP protocol is used.
Allow LDAP library to follow referrals	When selected, the lookup of user information can be redirected from an LDAP server to other servers.
Connection live check	Limits the time (in minutes) that elapses between checks to see whether the connection to an LDAP server is still active to the specified value.
LDAP operation timeout	Limits the time (in seconds) that elapses before the connection to an LDAP server is closed if no communication occurs to the specified value.
eDirectory network address attribute	Specifies the name of the attribute that provides the network addresses used for an eDirectory server
eDirectory network login time attribute	Specifies the name of the attribute that provides the logon time used on an eDirectory server.
eDirectory network minimal update interval	Specifies the time that elapses (in seconds) before information from an eDirectory server is updated.
Base distinguished name to user objects	Specifies the Distinguished name (DN) in the directory on an LDAP server where the lookup of user attributes should begin.

Option	Definition
Map user name to DN	When selected, the name of the user who asks for authentication must map to a DN (Distinguished Name). This name identifies the user in the directory on the LDAP server.
Filter expression to locate a user object	Specifies a filtering term for restricting the lookup of user attributes. To substitute the user name in the filtering term, u% is used as a variable.
Get user attributes	When selected, user attributes are looked up on the LDAP server to authenticate a user.
User attributes to retrieve	Provides a list for entering the user attributes that should be retrieved from an LDAP server.
Attributes concatenation string	Specifies a string for separating user attributes found by a lookup, for example, / (slash).
Get groups attributes	When selected, user group attributes are also looked up on the LDAP server to authenticate a user.
Base distinguished name to group objects	Specifies the Distinguished name (DN) in the directory on an LDAP server where the lookup of group attributes should begin.
Filter expression to locate a group object	Specifies a filtering term for restricting the lookup of group attributes. To substitute the user name in the filtering term, *u%* is used as a variable.
Group attributes to retrieve	Provides a list of group attributes that should be retrieved from an LDAP server.

RADIUS Specific Parameters

Settings for the RADIUS authentication method.

RADIUS Specific Parameters

Option	Definition
RADIUS server definition	Provides a list for entering the RADIUS servers that authentication information is retrieved from.
Default domain name	Specifies the name of the domain that information is retrieved from if no other domain is specified.
Shared secret	Sets the password used by an appliance to get access to a RADIUS server.
Radius connection timeout in seconds	Limits the time (in seconds) that elapses before the connection to the RADIUS server is closed if no traffic occurs to the specified value.
International text support	Specifies the set of characters used by default for a request sent from a client, for example, ISO-8859-1.

Option	Definition
Value of attribute with code	Sets the code value for the attribute retrieved with the user group information, according to RFC 2865. For example, 25 is the code for the “class” attribute.
Vendor specific attribute with vendor ID	Sets the Vendor ID that is required for retrieving vendor-related data in the search for user group information. According to RFC 2865, the vendor ID is a part of the vendor attribute, followed by a number of subattributes. Its code value is 26.
Vendor subattribute type	Sets a code value for the type of subattributes included in a vendor attribute. according to RFC 2865. Since not all vendors adhere to this structure, we recommend to specify 0 as value here. This allows the authentication module to retrieve all available vendor information.

Kerberos Specific Parameters

Settings for the Kerberos authentication method.

Note: More settings for this authentication method can be configured using the Kerberos Administration system settings, which can be accessed under the Configuration top-level menu.

Kerberos Specific Parameters

Option	Definition
Extract group membership IDs from the ticket	When selected, information to identify the groups that a user is a member of is retrieved from the ticket that is used in the process of authenticating users under the Kerberos authentication method. When this option is selected, the following option becomes accessible.
Look up group names via NTLM	When selected, the names of the groups that a user is a member of are retrieved using the NTLM authentication method.

Authentication Server Specific Parameters

Settings for the Authentication Server method.

Authentication Server Specific Parameters

Option	Definition
Authentication server URL	Specifies the URL of a server that is used under this method to look up authentication information.
Require client ID	When selected, the authentication server requires the ID of the client that a user sent a request from.
Store authentication result in a cookie	When selected, the information retrieved from the authentication server is stored in a cookie If cookie authentication is implemented, the cookie is added to the next request sent by the respective user, so that this user need not authenticate again.

Option	Definition
Allow persistent cookie for the server	When selected, a cookie can be used persistently for sending multiple requests to the authentication server
Cookie TTL for the authentication server in seconds	Limits the time (in seconds) that a cookie sent with a request to the server is stored to the specified value.
Cookie prefix	Specifies a prefix that is added on the appliance to a cookie, for example, *MWG_Auth* .

One-Time Password Specific Parameters

Settings for the One-Time Password authentication method.

One-Time Password Specific Parameters

Option	Definition
OTP server	Specifies the IP address and port number of the OTP server that Web Gateway connects to when authenticating a user under the One-Time Password authentication method.
Communicate with SSL and trust certificate below	When selected, communication with the OTP server is performed using an SSL-secured connection. When this option is selected, the information in the following four fields is no longer grayed out and the Import button below these fields becomes accessible. The fields provided detailed information about the certificate that is currently used in SSL-secured communication with the OTP server. • Subject — Provides general information about the certificate. ◦ Common Name (CN) — Specifies the common name of the certificate. By default, this name is *localhost*. ◦ Organization (O) — Specifies the organization of the certificate. By default, the organization is *OTP Server*. ◦ Organizational Unit (OU) — Specifies the organizational unit of the certificate. By default, the organizational unit is not set. • Issuer — Provides information about the issuer of the certificate. ◦ Common Name (CN) — Specifies the common name of the issuer. By default, this name is *localhost*. ◦ Organization (O) — Specifies the organization of the issuer. By default, the organization is *OTP Server*. ◦ Organizational Unit (OU) — Specifies the organizational unit of the server certificate.

Option	Definition
	By default, the organizational unit is not set. • Validity — Limits the time the certificate is valid. ◦ Not before — Shows the date and time when the validity of the certificate begins. ◦ Not after — Shows the date and time when the validity of the server certificate ends. • Extensions — Provides additional information on the certificate. ◦ Comment — Provides a plain-text comment on the certificate. By default no comment is provided. • Import — Opens a window for importing a certificate.
WS client name	Specifies the user name for Web Gateway in communication with the OTP server.
WS client password	Specifies the password for Web Gateway in communication with the OTP server.
OTP message	Specifies the prefix to messages that are sent from the OTP server to Web Gateway and the delimiters that include a message. By default a message looks like this: `OTP for MWG: $$`<OTP message>`$$`

McAfee Client Proxy

Settings for the SWPS (McAfee Client Proxy) authentication method.

McAfee Client Proxy

Option	Definition
Customer ID	Specifies an identifier for a customer.
Shared password	Sets a password for a customer. Clicking Set opens a window that allows you to perform the setting.
Keep domain in group name	When selected, domain information contained in the name of a user group is kept. This option is selected by default.
Remove custom headers used for authentication	When selected, headers contained in the information that is submitted for authentication are removed. This option is selected by default.
Export MCP credentials to XML file	Lets you export the credentials that are submitted when performing the SWPS (McAfee Client Proxy) authentication method. By default a message looks like this: `OTP for MWG: $$`<OTP message>`$$`

Advanced Parameters

Setting for configuring advanced authentication.

Note:

This is setting is the same for all authentication methods. Its description is therefore also provided at the beginning of this description of the authentication settings, after the description of the common settings.

Advanced Parameters

Option	Definition
Always evaluate property value	When selected, a new evaluation to assign a value to a property is performed each time a rule containing this property is processed. If a value has been stored for a property in the cache, it is not used. While it is normally recommended to let cache values be used to improve performance, there can be situations where the new evaluation of a property is required. In these situations, the same property is used more than once within the authentication rules and with the same settings of the Authentication module. A new evaluation ensures the most current value is assigned to the property each time.

Authorized Override settings

The Authorized Override settings are used for configuring the module that handles authorized overriding.

Hours and Minutes of Maximum Session Time

Settings for configuring the maximum time length of a session with authorized overriding.

Hours and Minutes of Maximum Session Time

Option	Definition
Days	Sets the days of an Authorized Override session.
Hours	Sets the hours of an Authorized Override session.
Minutes	Sets the minutes of an Authorized Override session.

Azure Directory settings

The Azure Directory settings are the settings for the Azure Directory module, which handles the retrieval of user group lists from an Azure Active Directory (Azure AD).

There is no default instance of the Azure Directory settings.

Application Settings

Settings for the application that is registered at a Microsoft Application Registration Portal to represent Web Gateway in communication with an Azure AD.

Application Settings

Option	Definition
Tenant ID	Identifies an Azure AD.
App ID	Identifies the application.
Password	Provides the password that the application submits when attempting to access the Azure AD.
Redirect URI	Identifies a location that a request for accessing the Azure AD is redirected to.

Search Parameters

Settings for the parameters used when searching for user group information in an Azure AD.

Search Parameters

Option	Definition
Map user name to UPN	When selected, a user name is mapped accordingly.
Filter expression to locate a user object	Specifies a term that serves as a filter when searching for a user name. Within this term use {user} to substitute the user name, for example: `mailnickname eq '{user}'`
UPN attribute	Specifies the UPN attribute that is searched for. Default: `id`
Group attribute	Specifies the group attribute that is searched for. Default: `memberOf`
Group name	Specifies the name of the group that is searched for. Default: `displayName`
Use cache	When selected, user group information that is searched for is stored and retrieved from a cache.
Cache entry TTL	Limits the time (in minutes) that an entry remains in the cache. Default: 30 minutes

Network Setup

Settings for the network setup that is configured to enable the retrieval of user group lists from an Azure AD.

Network Setup

Option	Definition
Use system proxy list to connect to MS Graph API	When selected, the proxies that have been configured for Web Gateway on an appliance system and entered in a list are used when setting up a connection for retrieving user group information from an Azure AD.

Option	Definition
TCP timeout	Limits the time (in seconds) that a TCP connection is kept open if no traffic occurs in the process of retrieving user group information. Default: 5 seconds
Search operation timeout	Limits the time (in seconds) that elapses before a search operation performed to retrieve user group information is terminated. Default: 10 seconds
Retry interval if token request fails	Specifies the time that must elapse after a failed token request before a new request is performed in the process of retrieving user group information. Default: 15 seconds
List of certificate authorities	Provides a list of certificate authorities that are used for securing the communication performed to retrieve user group information under HTTPS. Clicking Add or Edit opens windows for adding or editing the list.
Revocation checking method order	Allows you to choose the order in which to use the OCSP and CRL methods for checking whether a certificate has been revoked. • OCSP, CRL • CRL, OCSP
Treat OCSP response 'unknown' as revoked	When selected, a certificate is considered as revoked if the response to an OCSP query is that its revocation status is unknown. Default: 30 minutes

Cache settings

The Cache settings are module (engine) settings for configuring the behavior of the web cache on Web Gateway.

The following particular settings are provided for the Cache module after the initial setup.

- Cache HTTP — Default settings

Cloud Access Log Data Residency settings

The Cloud Access Log Data Residency settings are used for configuring geographic regions where web access data is stored when the hybrid solution is enabled.

Data Residency Settings

You can create and name multiple instances of this setting and reuse them in policy rules.

Data Residency Settings

Option	Definition
Store in	Select the geographic region where the web access data is stored: • North America • Europe

Cloud Storage Encryption settings

The Cloud Storage Encryption settings are used for configuring the encryption and decryption of cloud storage data.

Encryption Parameters

Settings for encrypting and decrypting cloud storage data

Encryption Parameters

Option	Definition
Cipher	Provides a list for selecting an algorithm to encrypt and decrypt cloud storage data. The following algorithms can be selected: • AES 128 • AES 192 • AES 256

Coaching settings

The Coaching settings are used for configuring the module that handles coaching.

Hours and Minutes of Session Time

Settings for configuring the length of a coaching session

Hours and Minutes of Session Time

Option	Definition
Days	Sets the days of a coaching session.
Hours	Sets the hours of a coaching session.
Minutes	Sets the minutes of a coaching session.

Composite Opener settings

The Composite Opener settings are module (engine) settings for configuring the Composite Opener, which extracts data from archives and similar files on Web Gateway.

Instances of the Composite Opener settings include:

- Default settings — Default settings

Default settings - Composite Opener

Default settings are available for the Composite Opener after the initial setup of Web Gateway. They are called Default.

Extraction Process Handling

Settings for the extraction process that is performed by the Composite Opener

Extraction Process Handling

Option	Definition
Set maximum level of nested objects	When selected, the number of nesting levels that the Composite Opener will handle when extracting objects is limited to the value that is configured here. When this limit is exceeded, no file opening is performed. • Maximum level of nested objects — Value that limits the number of nesting levels Default: 100
Set size limit for uncompressed data	When selected, the size of uncompressed data that the Composite Opener will provide when extracting objects is limited to the value that is configured here. When this limit is exceeded, no file opening is performed and the value of the Error.ID property is set to 10064. • Maximum size of extracted data — Value (in MB) that limits the data size Default: 4096 MB
Set limit for compression ratio	When selected, the size of the compression ratio that the Composite Opener will handle when extracting objects is limited to the value that is configured here. When this limit is exceeded, no file opening is performed and the value of the Error.ID property is set to 10065. • Maximum compression ratio — Value that limits the compression ratio For example, if you set this value to 1000, the maximum compression ratio is 1:1000 Default: 1000

Data Loss Prevention (Classifications) settings

The Data Loss Prevention (Classifications) settings are used for configuring entries in classification lists that specify sensitive or inappropriate content.

DLP Classifications Parameters

Settings for configuring the use of classification lists when searching for sensitive or inappropriate content

DLP Classifications Parameters

Option	Definition
Tracking policy	Sets the scope of the search for sensitive or inappropriate content in the body text of requests and responses. The search is carried out for all classifications that have been selected. You can, however, configure it in the following ways: • Minimum — The search stops when an instance of sensitive or inappropriate content has been found for a particular classification or if no instance could be found. It is then continued for the next classification. This goes on until all classifications have been processed. • Maximum — The search tries to find all instances of sensitive or inappropriate content for a particular classification. When the search is completed for a classification, it continues with the next. This goes on until all classifications have been processed.
DLP Classifications	Provides a list for selecting entries in classification lists from the system lists provided under DLP Classification on the lists tree.

The following table describes an entry in the DLP Classifications list

DLP Classifications Parameters – List entry

Option	Definition
DLP Classification	Provides information about detecting sensitive or inappropriate content.
Comment	Provides a plain-text comment on an entry.

Advanced Parameters

Settings for configuring advanced functions for data loss prevention

Advanced Parameters

Option	Definition
Reported context width	Limits the number of characters shown around a matching term in a list to the specified value. The matching term is the value of the *DLP.Classification.Matched.Terms* property.
Context list size	Limits the number of matching terms shown in a list to the specified value. The matching terms are the values of the *DLP.Classification.Matched.Terms* property.

Data Loss Prevention (Dictionaries) settings

The Data Loss Prevention (Dictionaries) settings are used for configuring text and wildcard expressions that specify sensitive or inappropriate content.

DLP Dictionary Parameters

Settings for configuring text and wildcard expressions specifying sensitive or inappropriate content

DLP Dictionaries Parameters

Option	Definition
Tracking policy	Sets the scope of the search for sensitive or inappropriate content in the body text of requests and responses. The search is carried out for all dictionary entries that have been created. It can, however, be configured in the following ways: • Minimum — The search stops when an instance of sensitive or inappropriate content has been found for a particular dictionary entry or if no instance could be found. It is then continued for the next entry. This goes on until all entries have been processed. • Maximum — The search tries to find all instances of sensitive or inappropriate content for a particular dictionary entry. When the search is completed for an entry, it continues with the next. This goes on until all entries have been processed.
Dictionary	Provides a list for entering text strings and wildcard expressions that are sensitive or inappropriate content or match with it.

The following table describes an entry in the Dictionary list.

Dictionary – List entry

Option	Definition
Text or wildcard expression	Specifies a text string or wildcard expression that is sensitive or inappropriate content or matches with it.
Comment	Provides a plain-text comment on a text string or wildcard expression.

Advanced Parameters

Settings for configuring advanced functions for data loss prevention

Advanced Parameters

Option	Definition
Reported context width	Limits the number of characters shown around a matching term in a list to the specified value. The matching term is the value of the *DLP.Dictionary.Matched.Terms* property-

Option	Definition
Context list size	Limits the number of matching terms shown in a list to the specified value. The matching terms are the values of the *DLP.Classification.Matched.Terms* property.

Data Trickling settings

The Data Trickling settings are used for configuring the data trickling process that is applied when a user has started the download of a web object.

Data Trickling Parameters

Settings for the portions of a web object that are forwarded in data trickling mode

Data Trickling Parameters

Option	Definition
Size of first chunk	Specifies the size (in bytes) of the first chunk of a web object that is forwarded using the data trickling method.
Forwarding rate	Specifies the portion of a web object that is forwarded every five seconds. The forwarding rate is the thousandth part of the entire volume that is to be forwarded multiplied by the value you configure here.

File System Logging settings

The File System Logging settings are used for configuring the rotation, deletion, and pushing of log files that are maintained by logging rules.

File System Logging Settings

Settings for the log that stores rule-maintained log files

File System Logging Settings

Option	Definition
Name of the log	Specifies the name of a log.
Enable log buffering	When selected, the log is buffered. The buffer interval is 30 seconds.
Enable header writing	When selected, the header below is added to all log files.
Log header	Specifies a header for all log files.
Encrypt the log file	When selected, log files are stored encrypted.
First password, Repeat password	Sets a password for access to encrypted log files.

Option	Definition
[Optional] Second password, Repeat password	Sets a second password for access to encrypted log files.

Settings for Rotation, Deletion, and Pushing

Settings for log file management

The settings for rotating, deleting, and pushing rule-maintained log files include the same options and are configured in the same way as the corresponding settings for module-maintained log files, which are configured as part of the Log File Manager settings.

Hardware Security Module settings

The Hardware Security Module settings are used to configure the handling of private keys on a Hardware Security Module.

HSM Server

Settings for implementing an HSM solution on the Web Gateway appliance that you are currently configuring

HSM Server

Option	Definition
Start local HSM server	When selected, an HSM solution for storing and loading keys is implemented on this appliance. Other Web Gateway appliances in your network can connect to this appliance as clients. The appliance then takes the role of as server towards these clients.
Crypto module	Provides a list for selecting an HSM solution. • Entrust nShield Solo/Connect — These solutions let the functions of a Hardware Security Module be provided on a module card (*nShield Solo*), which is installed on a Web Gateway appliance, or on an additional appliance (*nShield Connect*). **Note:** The module card and the appliance are provided by a McAfee partner (Entrust). • SafeNet Network HSM (formerly LUNA SA) — This solution lets the functions of a Hardware Security Module be provided on a remote server. **Note:** The remote server is provided by a McAfee partner (Thales). • OpenSSL — This solution is an emulation that runs on the appliance and uses OpenSSL to provide the functions of a Hardware Security Module.
Keys to be loaded	Provides a list of IDs for the private keys that are stored on a Hardware Security Module and can be loaded from there. For every key that you want to use, you must add the key ID in string format to this list. **Note:** The key IDs are configured when private keys are generated on the Hardware Security Module.
Allow local connections	When selected, connections are allowed for using the functions of a Hardware Security Module on the appliance that you are currently configuring.

Option	Definition
Allow remote connections	When selected, connections are allowed for letting other appliances that are configured as clients of this appliance use the functions of a Hardware Security Module.
HSM server port definition list	Provides a list of the ports on the appliance that takes the role of a server towards other appliances.
Permitted clients	Provides a list of other appliances in your network that run as clients of this appliance.

These tables describe the entries in the key list and the lists of HSM server ports and permitted clients.

Keys to be loaded – List entry

Option	Definition
String	Specifies the key ID for a private key that is stored on the Hardware Security Module.
Comment	Provides a plain-text comment on a key.

HSM server port definition list – List entry

Option	Definition
Listener address	Specifies the IP address and port number of a port on the appliance that takes the role of a server towards other appliances.
Comment	Provides a plain-text comment on a port.

Permitted clients – List entry

Option	Definition
Host	Specifies the host name or IP address of an appliance that is permitted to run as client of this appliance.
Certificate	Provides a certificate that a client submits when connecting to the server.
Comment	Provides a plain-text comment on a permitted client.

Server Identification

Settings for the certificate that an appliances submits when taking the role of a server towards other appliances that run as its clients

Note:

A certificate issued by the McAfee root CA is provided by default after the initial setup of a Web Gateway appliance.

We recommend that you replace this certificate by a certificate of your own.

Server Identification

Option	Definition
Subject, Issuer, Validity, Extensions, Private key	These fields provide information on the server certificate that is currently in use.
Server certificate	Provides buttons for performing various activities that are related to a server certificate: • Generating a certificate • Importing a certificate • Exporting a certificate • Exporting a certificate key

HSM Client

Settings for configuring an appliance as client of an appliance that has an HSM solution implemented

HSM Client

Option	Definition
Use remote HSM server	When selected, this appliance runs a client of another appliance that has an HSM solution implemented.
Remote server	Provides a list of appliances in your network that have an HSM solution implemented and that this appliance can connect to.

This table describes an entry in the list of remote servers.

Remote server– List entry

Option	Definition
Host	Specifies the host name or IP address of an appliance in your network that takes the role of a server towards this appliance.
Certificate	Specifies the certificate that an appliance submits when connecting to a client.
Comment	Provides a plain-text comment on a remote server.

Client Identification

Settings for the certificate that this appliance submits when connecting as a client to an HSM server

Note:

A certificate issued by the McAfee root CA is provided by default for this client after the initial setup of a Web Gateway appliance. We recommend that you replace this certificate by a certificate of your own.

Client Identification

Option	Definition
Subject, Issuer, Validity, Extensions, Private key	These fields provide information on the client certificate that is currently in use.
Client certificate	Provides buttons for performing various activities that are related to a client certificate:

Option	Definition
	• Generating a certificate • Importing a certificate • Exporting a certificate • Exporting a certificate key

Troubleshooting

Settings for troubleshooting the use of a Hardware Security Module

Troubleshooting

Option	Definition
Write connection traces	When selected, traffic on the connections set up for using the functions of a Hardware Security Module are traced.

ICAP Client settings

The ICAP Client settings are the settings for the ICAP Client module, which handles communication between an ICAP client on a Web Gateway appliance and ICAP servers.

Instances of the ICAP Client settings

There are no instances of the ICAP Client settings available by default.

After importing suitable rule sets, instances are available as follows.

- ReqMod — Available after importing the Data Loss Prevention (DLP) with ICAP rule set
- ReqMod for Cloud — Available after importing the Data Loss Prevention (DLP) with ICAP for Cloud rule set

ICAP Service

Settings for ICAP servers that the ICAP client on an appliance sends requests to.

ICAP Service

Option	Definition
List of ICAP Servers	Provides a list for selecting a list of servers that are used in ICAP communication. Requests coming in from ICAP clients are distributed to the servers on the selected list in round-robin mode.
Add	Opens the Add List window to let you add a list of ICAP servers.
Edit	Opens the Edit List window to let you edit a list of ICAP servers.
Select deployment type for these settings	Allows you to select the type of deployment for the Web Gateway appliance that you want to run an ICAP client on. You can select one of the following deployment types: • On premise — Web Gateway is deployed on premise. • Cloud only — Web Gateway is deployed in the cloud. • Hybrid — Web Gateway is deployed as a hybrid solution, which combines on-premise and cloud use.
Exclude below user-defined ICAP request header(s)	Drops authentication headers that are included by default when an ICAP client sends a request to an ICAP server.

Option	Definition
	Configuring this option is useful because some ICAP servers don't accept lengthy authentication headers in a request and respond with an error message. **Note:** This option can be configured for on-premise and cloud use. You can drop either or both of these headers: • X-Authenticated-User — When selected, requests to an ICAP server are forwarded without this header. • X-Authenticated-Groups — When selected, requests to an ICAP server are forwarded without this header.

The following table describes an entry for an ICAP server in the list.

List of ICAP servers — List entry

Option	Definition

URI	Specifies the URI for an ICAP server using the following format:
	`icap[s]://<IP address>|<fully qualified domain name>[:<port>][/<ICAP method>]`
	The list contains the following entry for an ICAP server by default:
	`icap://0.0.0.0:1344`

Respect max concurrent connections limit	When selected, the ICAP client on the appliance does not open more connections at the same time for sending requests than the ICAP server can handle.
Comment	Provides a plain-text comment on an ICAP server.

Secure ICAP (ICAPS) Certificate Verification

Settings for configuring certificate verification in Secure ICAP communication.

Secure ICAP (ICAPS) Certificate Verification

Option	Definition
Enable server certificate verification	When selected, certificate verification is performed in Secure ICAP (ICAPS) communication. **Note:** This option can be configured for on-premise and cloud use. This allows you to implement certificate verification, for example, in the communication between an ICAP client running in the cloud and a DLP server that runs on-premise on a Web Gateway appliance taking the role of an ICAP server. To perform this verification, the ICAP client checks whether the certificate sent by the DLP server (ICAP server) is included in a list of trusted server certificates.
Server certificate list	Provides a list of trusted server certificates for performing verification in Secure ICAP communication. There is no list available by default.

Option	Definition
Add	Opens the Add List window where you can add a list of server certificates. **Note:** The ICAP client does not accept any server certificate that has a private key with a format of less than 2048 bit.
Edit	Opens the Edit List window where you can edit a list of server certificates.

Next Hop Proxy settings

The Next Hop Proxy settings are used for configuring next-hop proxies to forward requests that have been received on the appliance to the web.

Next Hop Proxy Server

Settings for next-hop proxies

Next Hop Proxy Server

Option	Definition
List of next-hop proxy servers	Provides a list for selecting a next-hop proxy server list.
Round robin	When selected, the Next Hop Proxy module uses the next-hop proxy following the one in the list that has been used last. When the end of the list has been reached, the first next-hop proxy in the list is again selected.
Fail over	When selected, the Next Hop Proxy module tries the first next-hop proxy in the list first. If the first next-hop proxy fails to respond, it is retried until the configured retry maximum has been reached. Then the second next-hop proxy in the list is tried, and so on, until a server responds or all are found to be unavailable.
Sticky	When selected, the Next Hop Proxy module uses the same next-hop proxy over a time period that you can also configure.
Minimum time for stickiness	Sets the period of time (in seconds) that the same next-hop proxy is used for forwarding a request. The default time period is 300 seconds.
Proxy style requests	When selected, requests in proxy style are forwarded to the requested web servers using next-hop proxies. This options is selected by default.

Progress Page settings

The Progress Page settings are used for configuring the progress page that is shown to users when they are downloading web objects.

Progress Page Parameters

Settings for the progress page

Progress Page Parameters

Option	Definition
Templates	Provides settings for the templates that are used by the progress page.
Timeouts	Provides settings timeouts that are related to the progress page.

Templates

Settings for the templates used by the progress page

Templates

Option	Definition
Language	Provides settings for selecting the language of the progress page. • Auto (Browser) — When selected, the message is in the language of the browser that the blocked request was sent from. • Force to — When selected, the message is in the language chosen from the list that is provided here. • Value of 'Message.Language' property — When selected, the message is in the language that is the value of the Message.Language property This property can be used for creating a rule.
Collection	Provides a list for selecting a template collection. • Add — Opens the Add Template Collection window for adding a template collection. • Edit — Opens the Template Editor for editing a template collection.
Template name for progress bar page	Provides a list for selecting a template. • Add — Opens the Add Template window for adding a template. • Edit — Opens the Template Editor for editing a template,
Template name for download finished page	Provides a list for selecting a template. • Add — Opens the Add Template window for adding a template • Edit — Opens the Template Editor for editing a template
Template name for download canceled page	Provides a list for selecting a template. • Add — Opens the Add Template window for adding a template. • Edit — Opens the Template Editor for editing a template.

Timeouts

Settings for the timeouts that are related to the progress page

Templates

Option	Definition
Delay for redirects to progress page	Limits the time (in seconds) that elapses before the progress page appears to the specified value.
File availability time before download	Limits the time (in minutes) that elapses before a file is no longer available to a user before the download to the specified value.
File availability time after download	Limits the time (in minutes) that elapses before a file is no longer available to a user after the download to the specified value.

SSL Client Context with CA settings

The SSL Client Context with CA settings are used to configure the sending of certificates with information about the certificate authority to the clients of a Web Gateway appliance.

Define SSL Client Context (Certificate Authority)

Settings for sending a certificate to the clients with information about the certificate authority

Define SSL Client Context (Certificate Authority)

Option	Definition
(Current certificate and default root certificate authority)	Under Subject, Issuer, and other field names. information about the certificate is provided that is currently sent to the clients of an appliance in SSL-secured communication. Information is also provided about the root certificate authority (root CA) that signed this certificate. After the initial setup, the certificate is signed by the default root certificate authority. This certificate authority is McAfee. The certificate is therefore called a self-signed certificate, as McAfee signed a certificate for one of their own products. Self-signed certificates are not trusted by all partners in SSL-secured communication. For further administration of the SSL functions on Web Gateway, we recommend that you create your own root certificate authority. Use the Generate New option to create this certificate authority.
Certificate Authority	Provides several options for performing activities that are related to a certificate authority. • Generate New — Opens a window for generating a new certificate authority. • Import — Opens a window for importing a certificate authority. The window provides an option for importing a file with information about a certificate authority and the certificate that was signed by it.

Option	Definition
	Additionally, you can include a file with information about the chain of certificate authorities that were involved in the validation process. **Note:** The file with information about the certificate chain can be a file that you created and stored in the file system before. In this case, the file will contain information about the following: ◦ The certificate that an appliance sends as server to its clients ◦ The intermediate certificate authorities, one of which signed the certificate, while the others each validated another certificate authority ◦ The root certificate authority, which is the first instance that validated another certificate authority When importing a certificate chain file, you must make sure that it only contains information about the intermediate certificate authorities. All other information must be removed from the file. Otherwise the import will fail. • Export — Lets you browse to a location within your file system that you can export a certificate authority file to. • Export key — Lets you browse to a location within your file system that you can export the key file for a certificate authority to.
Send certificate chain	When selected, the appliance sends information on the chain of certificates and certificate authorities that were involved in the process of validating a certificate with this certificate to its clients. To retrieve this information, you must include the certificate chain when using the option for importing a certificate authority. The appliance sends the certificate that is configured here as a server to its clients. The certificate is therefore also referred to as the server certificate. The server certificate is considered to exist on level 0. When a certificate authority signs this certificate to validate it, it is done on level 1. When an additional certificate authority validates the first certificate authority, it is done on level 2. With each additional certificate authority that is involved, the level increases by one.
Certificate chain	Provides information on a certificate chain. After importing a certificate authority file with information about the certificate chain, the information appears in this field.
Use custom domain key	When selected, a key is sent with the certificate that you have configured on your own. This key is used for sending certificates throughout the domain of a Web Gateway appliance.

Option	Definition
Custom domain key	Provides the following options for handling a custom domain key. • Import Key — Lets you browse to a location within your file system that you can import a custom domain key file from. • Export Key — Lets you browse to a location within your file system that you can export a custom domain key file to.
Digest	Provides a list for selecting a digest mode.
RSA server key size	Limits the size of the key file for a certificate.
Certificates that are signed by the CA are valid for	Limits the time (in days) that a certificate signed by the certificate authority configured here is valid.
Client cipher list	Specifies a string of Open SSL symbols used for decrypting client data.
Include OCSP responder URL	When selected, a URL for sending responses to OCSP queries is included in the Authority Information Access (AIA) field of the certificate to enable the retrieval of information about revoked certificates.
Include CRL distribution point	When selected, a URL for accessing CRL lists is provided on the certificate to enable the retrieval of information about revoked certificates.
SSL session cache TTL	Limits the time (in seconds) that SSL session parameters are stored in the cache.
Perform insecure renegotations	When selected, Web Gateway renegotiates the parameters for the SSL-secured communication even if this is insecure to do.
Send empty plain-text fragment	When selected, an empty plain-text fragment is sent with the certificate to the clients.
Allow legacy signatures in the handshake	When selected, legacy signatures are allowed in the initial handshake.
SSL protocol version	Selects the version of the protocol that the SSL scanning module follows when dealing with handshakes. • TLS 1.3 — When selected, TLS (Transport Layer Security) version 1.3 is used. • TLS 1.2, TLS 1.1, or TLS 1.0 — The selected TLS version is used. • SSL 3.0 — When selected, SSL version 3.0 is used **Note:** Use the SSL option for compatibility reasons only.

SSL Client Context without CA settings

The SSL Client Context without CA settings are used to configure the sending of certificates with no information about the certificate authority to the clients of a Web Gateway appliance.

Define SSL Client Context (Without Certificate Authority)

Settings for sending a certificate to the clients with no information about the certificate authority

Define SSL Client Context (Without Certificate Authority)

Option	Definition
Select server certificate by host or IP	Provides a list of certificates that are sent to the clients and the host systems that they have been retrieved from. A host system is identified by a host name or an IP address. The certificates are sent from an appliance in its role as a server to the clients. The certificates are therefore referred to as server certificates.

Select server certificate by host or IP — **List entry**

Option	Definition
Host	Specifies the host name or IP address of the host system that a certificate is retrieved from.
Server Certificate	Provides information on the certificate that is currently sent from an appliance in its role as a server to its clients. When adding an entry for a new certificate to the list, you can generate or import the certificate. Options for performing these activities are provided in the window for adding a list entry under Server Certificate. • Generate — Opens a window for generating a new certificate. • Import — Opens a window for importing a certificate. The window provides an option for importing a file with information about a certificate. Additionally, you can include a file with information about the chain of certificate authorities that were involved in the validation process. **Note:** The file with information about the certificate chain can be a file that you created and stored in the file system before. In this case, the file will contain information about the following: ◦ The certificate that an appliance sends as server to its clients ◦ The intermediate certificate authorities, one of which signed the certificate, while the others each validated another certificate authority ◦ The root certificate authority, which is the first instance that validated another certificate authority When importing a certificate chain file, you must make sure that it only contains information about the intermediate certificate authorities. All other information must be removed from the file. Otherwise the import will fail. • Export — Lets you browse to a location within your file system that you can export a certificate authority file to.

Option	Definition
	• Export key — Lets you browse to a location within your file system that you can export the key file for a certificate authority to.
HSM	Provides information on a Hardware Security Module that is used to protect the certificate information.
Certificate chain	Provides information on the chain of certificates and certificate authorities that were involved in the validation of the certificate that is sent to the clients.
Comment	Provides a plain-text comment on a certificate.

Define SSL Client Context (Without Certificate Authority) — **Continued**

Option	Definition
SSL Scanner functionality applies only to client connection	When selected, traffic is only processed using the SSL scanning functions on the connection from an appliance to its clients.
Client cipher list	Specifies a string of Open SSL symbols used for decrypting client data.
SSL session cache TTL	Limits the time (in seconds) that SSL session parameters are stored in the cache.
Perform insecure renegotations	When selected, Web Gateway renegotiates the parameters for the SSL-secured communication even if this is insecure to do.
Send empty plain-text fragment	When selected, an empty plain-text fragment is sent with the certificate to the clients.
SSL protocol version	Selects the version of the protocol that the SSL Scanner module follows when dealing with handshakes. • TLS 1.3 — When selected, TLS (Transport Layer Security) version 1.3 is used. • TLS 1.2, TLS 1.1, or TLS 1.0 — The selected TLS version is used. • SSL 3.0 — When selected, SSL version 3.0 is used **Note:** Use the SSL option for compatibility reasons only.

SSL Scanner settings

The SSL Scanner settings are used for configuring the way certificates are verified and content inspection is enabled for SSL-secured web traffic, which is also known as HTTPS traffic.

They apply to traffic that is going on between Web Gateway and a web server when Web Gateway runs as a proxy that receives traffic from its clients, filters it according to the rules of your web security policy, and forwards it to web servers depending on the filtering results.

Enable SSL Scanner

Settings for configuring certificate verification or the enabling of content inspection

Enable SSL Scanner

Option	Definition
SSL scanner function	Selects the function that is performed by the SSL Scanner module. - Certificate verification — When selected, the module verifies certificates submitted in SSL-secured communication. - SSL inspection — When selected, the module inspects the content of web objects transmitted in SSL-secured communication. - Identify and bypass Skype for Business traffic — When selected, web traffic going on over the Skype for Business communication tool is exempted from any SSL scanning.
SSL protocol version	The module follows the selected protocol version when web objects are transmitted in SSL-secured communication. - TLS 1.3 — When selected, TLS (Transport Layer Security) version 1.3 is used. - TLS 1.2, TLS 1.1, or TLS 1.0 — The selected TLS version is used. - SSL 3.0 — When selected, SSL version 3.0 is used **Note:** Use the SSL option for compatibility reasons only.
Server cipher list	Provides a list with strings of Open SSL symbols that are known as ciphers and used to decrypt server data for you to select from. The HTTP Scanner module can use different types of ciphers for decryption when it performs default certificate verification or verifies certificates from web servers that do not support the EDH (Ephemeral Diffie-Hellman) method. Ciphers for use in decrypting client data are selected as part of the SSL Client Context with CA and SSL Client Context wihout CA settings. You can select different types of ciphers here as well. This means that you can configure the use of ciphers differently depending on whether they are used for traffic going on between Web Gateway and web servers or between Web Gateway and its clients. If a client only supports older types of ciphers that could not be used in communication with a web server that uses newer types to ensure stronger encryption, you can select these stronger ciphers here for traffic coming in from and going to the web server. For the client traffic, you can select weaker ciphers when configuring the client settings.
SSL session cache TTL	Limits the time (in seconds) for keeping the parameter values of a session in SSL-secured communication stored in the cache to the specified value.
Allow handshake and renegotiation with servers that do not implement RFC 5746	When selected, the SSL Scanner module performs these activities also in communication with web servers that fail to comply with the specified standard.
Send empty plaintext fragment	When selected, this fragment is sent in the communication.

Option	Definition
Allow legacy signatures in the handshake	When selected, legacy signatures are accepted in the communication.

Allow Alternative Handshakes

Settings for handshakes in SSL-secured communication that use alternative parameter values

Allow Alternative Handshakes

Option	Definition
Use alternative handshake settings after handshake failure	When selected, the SSL Scanner module uses alternative parameter values after the first attempt to perform a handshake in SSL-secured communication has failed.
SSL protocol version	Selects the version of the protocol the SSL Scanner module follows when it performs an alternative handshake. • TLS 1.3 — When selected, TLS (Transport Layer Security) version 1.3 is used. • TLS 1.2, TLS 1.1, or TLS 1.0 — The selected TLS version is used. • SSL 3.0 — When selected, SSL version 3.0 is used **Note:** Use the SSL option for compatibility reasons only.
Server cipher list	Specifies a string of Open SSL symbols used for decrypting server data. The SSL Scanner module uses different strings for default certificate verification and for verifying certificates from servers that do not support the EDH (Ephemeral Diffie-Hellman) method.
Send empty plaintext fragment	When selected, this fragment is sent in the communication.
Allow legacy signatures in the handshake	When selected, legacy signatures are accepted in the communication.
Include indication that previous handshake failed	When selected, a failure of the previous handshake is indicated.

TIE Filter settings

The TIE Filter settings are used for configuring the TIE Filter module, which is involved in the process of exchanging information between Web Gateway and a TIE server.

Stream Detector settings

The Stream Detector settings are used to configure the module that calculates the probability for web objects that they are streaming media.

Streaming Detector

Setting for the module that calculates streaming media probabilities

Streaming Detector

Option	Definition
Minimal probability	Sets the probability (in percent, specified by a number from 0 to 100) that is sufficient for a web object to be considered as streaming media.

Time Quota settings

The Time Quota settings are used for configuring the module that handles time quota management.

Time Quota per Day, Week, Month, and Session Time

Settings for time quotas

When a time unit or the session time is selected, the heading of the next section reads accordingly.

Time Quota per Day, Week, Month, and Session Time

Option	Definition
Time quota per day (week, month)	When selected, the quota that is configured in the next section applies to the selected time unit.
Session time	When selected, the quota that is configured in the next section applies to the session time.

Hours and Minutes for . . .

Settings for time quotas that apply to the selected time unit or the session time

The heading of this section varies according to what you selected in the preceding section.

For example, if you selected *Time quota per week*, the heading reads *Hours and Minutes for Time Quota per Week*.

Hours and Minutes for . . .

Option	Definition
Hours	Sets the allowed hours per day, week, month, or for the session time.
Minutes	Sets the allowed minutes per day, week, month, or for the session time.

Actual Configured Time Quota

Displays the configured time quotas.

Actual Configured Time Quota

Option	Definition
Time quota per day (week, month)	Shows the allowed time per day, week, or month.
Session time	Shows the allowed session time.

URL Filter settings

The URL Filter settings are used for configuring the URL Filter module, which handles activities related to URL filtering on a Web Gateway appliance.

Instances of the URL Filter settings include the following:

- Default settings — Default settings

  These settings are used when working with the default rule set for URL filtering. This rule set is named Default and nested within the URL Filtering rule set.
- Special URL Filtering Group settings — Settings used when working with the nested Special URL Filtering Group rule set

Extended List

Settings for extended lists

Extended List

Option	Definition
Use the extended list	Provides a list for selecting an extended list.
Add	Opens the Add List window for adding an extended list.
Edit	Opens the Edit List (Extended List) window for editing the selected extended list.

Rating Settings

Settings for retrieving rating information on URLs based on categories and reputation scores

Rating Settings

Option	Definition
Search the CGI parameters for rating	When selected, CGI parameters are included in the search for information. CGI (Common Gateway Interface) parameters in a URL trigger scripts or programs when the URL is accessed. Information on CGIs is considered when categorizing a URL.
Search for and rate embedded URLs	When selected, embedded URLs are included in the search for information and rated. Information on an embedded URL is considered when categorizing the embedding URL. **Note:** Searching for embedded URLs can impact performance.
Do a forward DNS lookup to rate URLs	When selected, a DNS lookup is performed for a URL that no relevant information has been found for. The IP address that was looked up is used for another search.
Do a backward DNS lookup for unrated IP-based URLs	When selected, a backward DNS lookup, based on its IP address, is performed for a URL that no relevant information has been found for. The host name that was looked up is used for another search.
Use the built-in keyword list	When selected, the built-in keyword list is included in the search.

Option	Definition
Disable local GTI database	When selected, no information about web reputation and categories is retrieved from the local Global Threat Intelligence database..
Use online GTI web reputation and categorization services if local rating yields no result	When selected, information on URL categories and reputation scores is only retrieved from the Global Threat Intelligence service if the search in the internal database yielded no results.
Use default server for online GTI web reputation and categorization services	When selected, the appliance connects to the default server for retrieving information on URL categories and reputation scores from the Global Threat Intelligence system. • IP of the server — Specifies the IP address of the server used to connect to the Global Threat Intelligence system when the default server is not used. Format: <domain name> or <IPv4 address> or <IPv4 address mapped to IPv6 address> Regular IPv6 addresses cannot be specified here. • Port of the server — Specifies the port number of the port on this server that listens to requests from the appliance. Allowed range: 1–65535
Enabke the Dynamic Content Classifier if GTI web categorization yields no result	When selected, the Dynamic Content Classifier is involved in the URL filtering process if a search performed by the Global Threat Intelligence service yielded no results.

Advanced Settings

Advanced settings for the URL Filter module

Advanced Settings

Option	Definition
Treat connection problems to the cloud as errors	When selected, problems arising on the connection from the appliance to the Global Threat Intelligence server are logged as errors. Properties for error handling are set and eventually rules from an Error Handler rule set are executed.
Do a backward DNS lookup also for private addresses	When selected, private IP addresses are included in the backward DNS lookup. Excluding these addresses from the lookup leads to an increase in performance for URL filtering. This option is disabled by default. The lookup includes the following types of addresses: • IPv4 ◦ Private addresses ◦ Zeroconf addresses • IPv6 ◦ Link local addresses ◦ Site local addresses ◦ Unique local addresses

Proxy Settings

Option	Definition
Use upstream proxy	When selected, the appliance uses a proxy for connecting to the Global Threat Intelligence server on which lookups for URL category information, also known as “in-the-cloud” lookups, can be performed.
IP or name of the proxy	Specifies the IP address or host name of the proxy.
Port of the proxy	Specifies the number of the port on the proxy that listens for lookup requests from the appliance.
User name	Specifies a user name for the appliance when logging on to the proxy.
Password	Sets a password for an appliance.
Set	Opens a window for setting a password.
Connect to GTI cloud via host name also when a proxy is configured	When selected, Web Gateway connects to a cloud service for performing GTI lookups using the host name of the server where the cloud service resides, regardless of whether a proxy is also configured..
Try to bypass the proxy if unreachable	When selected, Web Gateway tries to bypass a proxy that has been set up if this proxy cannot be reached.
Trust server certificate	When selected, a certificate sent under HTTPS by a cloud service for performing GTI lookups is trusted on Web Gateway. • Subject, Issuer, Validity, Extensions, Fingerprint, Private Key — Provide information about the certificate that is sent by the cloud service.. • Import — Opens a window for importing a server certificate..
Provide client certificate	When selected, Web Gateway provides a certificate when connecting as a client under HTTPS to a cloud service for performing GTI lookups. • Subject, Issuer, Validity, Extensions, Fingerprint, Private Key — Provide information about the certificate that Web Gateway sends to the cloud serviice. • Import, Export, Export Key — Open windows for importing a client certificate and for exporting a client certificate and key.

Logging

Option	Definition
Enable logging	When selected, URL filtering activities are logged on the appliance. If this option is not selected, the following logging options are grayed out.

Option	Definition
Log level	Provides a list for selecting the log level. Log levels are as follows: • 00 FATAL — Logs only fatal errors. • 01 ERRORS — Logs all errors. • 02 WARNING — Logs errors and warnings. • 03 INFO — Logs errors, warnings, and additional information. • 04 DEBUG1 ... 013 DEBUG9 — Log information required for debugging URL filtering activities. The amount of logged information increases from level DEBUG1 to DEBUG9. • 14 TRACE — Logs information required for tracing URL filtering activities. • 15 ALL — Logs all URL filtering activities
(Log area)	Provides a set of options for including different areas of URL filtering activities into the logging. • LOG_AREA_ALL — When selected, all URL filtering activities are logged. • LOG_AREA_NETWORK — When selected, activities regarding the network connections used for URL filtering are logged. • LOG_AREA_DATABASE_SEARCH — When selected, activities regarding the retrieval of data for URL filtering from the internal database are logged. • LOG_AREA_DNS — When selected, activities regarding a DNS lookup that is performed for URL filtering are logged. • LOG_AREA_URL — When selected, activities for handling URLs, such as parsing them, are logged. • LOG_AREA_CLOUD — When selected, activities regarding the retrieval of information from the Global Threat Intelligence system are logged.

Cloud Settings

Option	Definition
Connection count (maximum)	Limits the number of connections that can be active at the same time. Maximum number of connections by default: 4
Request timeout	Limits the time between retries of requests on a connection. Maximum time by default: 2000 ms
Request attempts	Limits the number of retries. Maximum number of retries: 3

Troubleshooting

Settings for troubleshooting issues with URL filtering

Air-Gap Mode Setting

Option	Definition
Automatic air-gap mode	An automatic air-gap mode can be enabled for connections from a Web Gateway appliance to a Global Threat Intelligence (GTI) server when issues impacting response time arise. Enabling this mode prevents increased response times on GTI server connections from creating overload issues elsewhere, for example, on the anti-malware or the proxy working queue. Traffic resulting from queries sent to and received from the GTI server is reduced in air-gap mode to the minimum that is required to monitor response times in order to recognize a return to normal. When a return to normal is recognized, the automatic air-gap mode is disabled. What is considered a normal response time here can be configured. While the automatic air-gap mode is enabled, information about URL categories and reputation scores can still be retrieved from the local database on Web Gateway. Monitoring functions can be enabled with or without the automatic air-gap mode. The following can be selected for the automatic air-gap mode: • Off — When selected, no monitoring is performed on GTI server connections and the automatic air-gap mode is never enabled. This option is selected by default. • Monitor only — When selected, GTI server connections are monitored, but the automatic air-gap mode is still never enabled. When these connections are monitored, issues impacting response time are logged like this: ◦ When the maximum average response time exceeds a configured threshold as long as or longer than a time interval that is also configured, a warning message is logged, as a possible trigger to taking appropriate measures. ◦ When response times return to normal again, falling below the threshold as long as or longer than configured, an information message is logged. Default values are configured for the threshold and the time intervals. You can modify these values to adapt them to your network conditions. • Active — When selected, GTI server connections are monitored and the automatic air-gap mode is enabled and disabled depending on how response times on these connections develop. The configured threshold and time intervals are then evaluated for both enabling the air-gap mode and logging warnings and information messages.
Maximum average delay threshold	Sets a threshold value that marks the acceptable maximum average response time (in ms) on connections to a GTI server.

Option	Definition
	Default: 250 ms
Retention time enable air gap	Sets the time interval (in seconds) over which the average response time on GTI server connections must exceed the configured threshold before a warning message is logged and the automatic air-gap mode is enabled if available and activated. Default: 10 seconds
Retention time disable air gap	Sets the time interval (in seconds) over which the average response time on GTI server connections must fall below the configured threshold before a back-to-normal message is logged and the automatic air-gap mode is disabled if previously enabled. Default: 120 seconds
Probing rate if enabled	Sets the percentage of requests for web access submitted by users for which queries are sent to a GTI server to a minimal value that applies when the automatic air-gap mode is enabled. Keeping a minimal amount of traffic on the connections to the GTI server is required to monitor this traffic in order to recognize when response times return to normal, so the automatic air-gap mode can be disabled. Default: 1 %

Volume Quota settings

The Volume Quota settings are used for configuring the module that handles volume quota management.

Volume Quota per Day, Week, and Month

Settings for volume quotas

When a time unit or the session time is selected, the heading of the next section reads accordingly.

Volume Quota per Day, Week, and Month

Option	Definition
Volume quota per day (week, month)	When selected, the quota that is configured in the next section applies to the selected time unit
Session time	When selected, the quota that is configured in the next section applies to the session time

Volume for . . .

Settings for volume quotas that apply to the selected time unit or the session time

The heading of this section varies according to what you selected in the preceding section.

For example, if you selected *Volume quota per week*, the heading reads *Volume for Volume Quota per Week*.

However, if you selected *Session Time*, the heading reads *Hours and Minutes*.

Volume for . . .

Option	Definition
GiB	Specifies the number of GiB that are allowed as volume.
MiB	Specifies the number of MiB that are allowed as volume.

Actual Configured Volume Quota

Displays the configured volume quotas.

Actual Configured Volume Quota

Option	Definition
Volume quota per day (week, month)	Shows the allowed volume per day, week, or month.
Session time	Shows the allowed session time.

Action settings

Action settings are used for configuring rule actions.

The following rule actions can be configured using action settings.

- Authenticate
- Block
- Redirect

Authenticate settings

The *Authenticate* settings are used for configuring the way the Authenticate action is executed when a filtering rule with that action applies.

Failed Login Message Template

Settings for configuring user messages and a block reason for logging purposes

Failed Login Message Template

Option	Definition
Language	Provides settings for selecting the language of a user message. • Auto (Browser) — When selected, the message is in the language of the browser that the blocked request was sent from. • Force to — When selected, the message is in the language chosen from the list that is provided here. • Value of Message.Language property — When selected, the message is in the language that is the value of the *Message.Language* property. This property can be used for creating a rule.
Template collection	Provides a list for selecting a template collection. • Add — Opens the Add Template Collection window for adding a template collection. • Edit — Opens the Template Editor for editing a template collection.
Template name	Provides a list for selecting a template. • Add — Opens the Add Template window for adding a template. • Edit — Opens the Template Editor for editing a template.
McAfee Web Reporter block reason ID	Provides a numerical value that identifies a block reason.
Block reason	States the block reason in plain text.

Block settings

The Block settings are used for configuring the way the Block action is executed when a filtering rule with that action applies.

Language and Template Settings

Settings for configuring user messages and a block reason for logging purposes

Language and Template Settings

Option	Definition
Language	Provides settings for selecting the language of a user message. • Auto (Browser) — When selected, the message is in the language of the browser that the blocked request was sent from. • Force to — When selected, the message is in the language chosen from the list that is provided here. • Value of Message.Language property — When selected, the message is in the language that is the value of the *Message.Language* property. This property can be used for creating a rule.
Template collection	Provides a list for selecting a template collection. • Add — Opens the Add Template Collection window for adding a template collection. • Edit — Opens the Template Editor for editing a template collection.
Template name	Provides a list for selecting a template. • Add — Opens the Add Template window for adding a template. • Edit — Opens the Template Editor for editing a template.
McAfee Web Reporter block reason ID	Provides a numerical value that identifies a block reason.
Block reason	States the block reason in plain text.

Redirect settings

The *Redirect* settings are used for configuring the way the Redirect action is executed when a filtering rule with that action applies.

Redirect Settings

Settings for configuring user messages and a block reason for logging purposes

Redirect Settings

Option	Definition
Redirect.URL	When selected, the value of the *Redirect.URL* property is the URL that is used for redirecting. This property can be used in a suitable rule.

Option	Definition
User-defined URL	When selected, the redirecting URL must be specified by you
Redirect URL	Specifies the URL for a redirecting URL.
Language	Provides settings for selecting the language of a user message. • Auto (Browser) — When selected, the message is in the language of the browser that the blocked request was sent from. • Force to — When selected, the message is in the language chosen from the list that is provided here. • Value of Message.Language property — When selected, the message is in the language that is the value of the *Message.Language* property. This property can be used for creating a rule.
Template collection	Provides a list for selecting a template collection. • Add — Opens the Add Template Collection window for adding a template collection. • Edit — Opens the Template Editor for editing a template collection.
Template name	Provides a list for selecting a template. • Add — Opens the Add Template window for adding a template. • Edit — Opens the Template Editor for editing a template.
McAfee Web Reporter block reason ID	Provides a numerical value that identifies a block reason.
Block reason	States the block reason in plain text.

Rule sets

Rule sets contain rules for a handling a particular field of web security. These fields include anti-malware filtering, URL filtering, media type filtering, and others.

Availability of rule sets

Rule sets are made available for your administration activities as follows:

- **Default rule sets** — After the initial setup of Web Gateway, default rule sets are provided for some important fields of web security.

  You can modify, rename, and delete these rule sets and the rules within them later on and also create new rule sets and rules.
- **Library rule sets** — A built-in rule set library is shipped with Web Gateway. You can import rule sets from this library to cover more fields of web security or extend the coverage of the default fields. All default rule sets are also contained in this library.

An online rule set library offers even more rule sets, which you can import with relevant documentation after accessing the built-in library.

Rule set views

When working with a default or library rule set, there are usually two views available:

- **Key elements view** — This view allows you to configure key elements of the rules in this rule set.

  Key elements are those parts of the rules that you will most likely want to work with when configuring your policy for a particular field of web security. They include, for example, lists of web objects or settings for modules.

  In some cases, you can also enable or disable a rule, but you cannot view any rule completely, nor perform any other activities where a complete rule would be involved, such as deleting or creating a rule.
- **Complete rules view** — This view allows you to view all rules in the rule set and to configure all their elements, including the key elements.

  You can also enable or disable, move, copy, delete, and create rules in this view.

When you create a rule set on your own, this can only be done using the complete rules view. This is also the only view that will be available for a rule set of this kind later on.

Access log rule set

The Access Log rule set is a nested rule set in the Default Log Handler rule set.

Nested default rule set – Access Log
Criteria – *Always*

The rule set contains the following rule.

Write access.log
Always –> Continue —
Set User-Defined.logLine = DateTime.ToWebReporterString + “ ”” ...
FileSystemLogging.WriteLogEntry (User-Defined.logLine)<Access Log Configuration>
The rule uses an event to fill a log file entry with parameter values relating to requests sent by users, such as user names or request headers.

It uses another event to write this entry into a log file.
The log file entry is specified as a parameter in both events. The log that stores the log file is specified by the settings of the write event.
Values for the following parameters are set and logged by the events of the rule (properties used by the event that sets the values are shown in italics): • Date and time — *DateTime.ToWebReporterString* • User name — *Authentication.UserName* • Client IP address — *String.ReplaceIfEquals (IP.ToString(Client.IP), “”, “-”)* • Response status — *String.ReplaceIfEquals (Number.ToString (Response.StatusCode), “”, “-”)* • Request header — *RequestHeader.FirstLine* • URL category — *List.OfCategory.ToString (URL.Categories)* • URL reputation — *String.ReplaceIfEquals (URL.ReputationString, “”, “-”) (URL.Reputation<Default>)* • Media type — *MediaType.ToString (MediaType.FromHeader)* • Body size — *String.ReplaceIfEquals (Number.ToString (Body.Size), “”, “-”)* • User agent — *Header.Request.Get(“User-Agent”)* • Virus and malware names — *List.OfString.ToString (Antimalware.VirusNames)* • Block action ID — *Number.ToString (Block.ID)*
The logging rule applies whenever a request for access to the web is received.
The two rule events for filling and writing a log entry are then executed.
Processing continues with the next rule or rule set.

Advanced Threat Defense rule set

The Advanced Threat Defense rule set is a library rule set. It enables Web Gateway to use Advanced Threat Defense for additional scanning of web objects in the anti-malware filtering process.

When working with this rule set, you can use different views:

- **Key elements view** — Allows you to configure key elements of the rules in this rule set.

  Key elements are those parts of the rules that you will most likely want to work with when configuring your policy for a particular field of web security. You can also enable or disable some rules in this view.
- **Complete rules view** — Allows you to view all rules in the rule set and to configure all their elements, including the key elements.

  You can also enable or disable, move, copy, or delete any of the existing rules, as well as create new rules in this view.

Key elements of the Advanced Threat Defense rule set

The key elements of the Advanced Threat Defense rule set deal with important elements of the process that performs additional scanning of web objects.

Enable Advanced Threat Defense for These Supported Media Types

Key element for selecting web objects that are eligible for additional scanning by Advanced Threat Defense.

Enable Advanced Threat Defense for These Supported Media Types

Option	Definition
Media types to insert	Clicking Edit opens a window to let you edit the Advanced Threat Defense Supported Media Types list that is used by a rule. Only web objects that belong to media types on this list will additionally be scanned by Advanced Threat Defense if also the other criteria are met. You can add, modify, and remove entries on the list.

Gateway Anti-Malware Settings

Key element for configuring the scanning by the Anti-Malware module before the additional scanning by Advanced Threat Defense.

Gateway Anti-Malware Settings

Option	Definition
Settings	Clicking Edit opens a window to let you edit the settings for the Anti-Malware module when it runs with the module components that are usually available on Web Gateway. This scanning is performed before any scanning by Advanced Threat Defense. Depending on the result of this scanning, additional scanning by Advanced Threat Defense is performed or not.

Gateway Advanced Threat Defense Settings

Key element for configuring additional scanning by Advanced Threat Defense.

Bypass scanning for these agents and hosts

Option	Definition
Settings	Clicking Edit opens a window to let you edit the settings for the Anti-Malware module on Web Gateway when the scanning is actually performed by Advanced Threat Defense.

Complete rules of the Advanced Threat Defense rule set

When working with the complete rules of the Advanced Threat Defense rule set, all rules and rule elements of this rule set can be viewed and configured.

Library rule set – Advanced Threat Defense
Criteria – *Antimalware.Proactive.Probability<Gateway Anti-Malware> greater than or equals 60 AND MediaType.EnsuredTypes at least one in list Advanced Threat Defense Supported Types*
Cycles – Responses, Embedded Objects

The rule set criteria specifies that the rule set applies if the following is true:

- As a result of previous scanning by the anti-malware engines on Web Gateway, the probability that a web object is malicious equals or exceeds 60 percent
- The media type of the object is on the list of supported types for scanning by Advanced Threat Defense.

The rule set contains the following rules.

Enable progress page
Always –> Continue – Enable Progress Page<Default>
The rule enables an event that lets a page be shown to indicate the progress made when a web object is downloaded to a client.

Upload file to ATD and wait for scanning result
Antimalware.Infected<Gateway ATD> –> Block<Virus Found> – Statistics.Counter.Increment("BlockedByMATD",1)<Default>
The rule uses the Antimalware.Infected property to check whether a web object, for example, a file, is infected by a virus or other malware. The scanning that is required for this check is performed under the Gateway ATD settings, which means it is carried out by Advanced Threat Defense.
If the object is found to be infected, the process of forwarding the object to the requesting client is blocked and a block message is shown to the user who requested access to the object.
The block action is recorded by the statistics counter.

Application Control rule set

The Application Control rule set is a library rule set for application filtering.

Library rule set – Application Control
Criteria – *Always*
Cycles – Requests (and IM), responses

The following rule sets are nested in this rule set:

- Block Applications in Request Cycle
- Block Applications in Response Cycle

Block Applications in Request Cycle

This nested rule set handles application filtering in the request cycle.

Nested library rule set – Block Applications in Request Cycle
Criteria – *Always*
Cycle – Requests (and IM)

The rule set contains the following rules:

Block instant messaging applications

Application.Name is in list Instant Messaging –> Block<Default>
The rule uses the *Application.Name* property to check whether the name of an application is contained in a specified list. If it is, it blocks a request for this application.
The action settings specify a message to the requesting user.

The rule is not enabled by default.

Block web applications with high risk
Application.HighRisk equals true AND Application.Name is in list Web Browsing and Web Conferencing –> Block<Default>
The rule uses the *Application.HighRisk* property to check the reputation score of an application and the *Application.Name* property to check whether the name of this application is contained in a specified list. If the reputation score reaches or exceeds the high-risk level and the application name is also on the list, it blocks a request for this application.
The action settings specify a message to the requesting user.

Block Facebook chat
Application.ToString (Application .Name) equals "Facebook.Chat" –> Block<Default>
The rule uses the *Application.To String* property to check whether the name of an application is equal to a specified string. For this purpose, the name of the application is converted into a string. If the converted application name equals the specified string, a request for the application is blocked.
The action settings specify a message to the requesting user.

The rule is not enabled by default.

Block Applications in Response Cycle

This nested rule set handles application filtering in the response cycle.

Nested library rule set – Block Applications in Response Cycle
Criteria – *Always*
Cycle – Responses

The rule set contains the following rule:

Applications to be looked for in response cycle
Application.Name is in list of Applications to Search for in Response Cycle –> Block<Default>
The rule uses the *Application.Name* property to check whether the name of an application is contained in a specified list. If it is, it blocks a request for this application.
The action settings specify a message to the requesting user.

The rule is not enabled by default.

Block web applications with high risk
Application.HighRisk equals true AND Application.Name is in list Web Browsing and Web Conferencing –> Block<Default>
The rule uses the *Application.HighRisk* property to check the reputation score of an application and the *Application.Name* property to check whether the name of this application is contained in a specified list. If the reputation score reaches or exceeds the high-risk level and the application name is also on the list, it blocks a request for this application.
The action settings specify a message to the requesting user.

Block Facebook chat
Application.ToString (Application .Name) equals "Facebook.Chat" –> Block<Default>
The rule uses the *Application.To String* property to check whether the name of an application is equal to a specified string. For this purpose, the name of the application is converted into a string. If the converted application name equals the specified string, a request for the application is blocked.
The action settings specify a message to the requesting user.

The rule is not enabled by default.

Complete rules of the Application Control rule set

When working with the complete rules of the Application Control rule set, all rules and rule elements of this rule set can be viewed and configured.

Library rule set – Application Control
Criteria – *Always*
Cycles – Requests (and IM), Responses

The following rule sets are nested in this rule set:

- Block Applications in Request Cycle
- Block Applications in Response Cycle

Block Applications in Request Cycle

This nested rule set handles application filtering in the request cycle.

Nested library rule set – Block Applications in Request Cycle
Criteria – *Always*
Cycle – Requests (and IM)

The rule set contains the following rules:

Block instant messaging applications

Application.Name is in list Instant Messaging –> Block<Default>
The rule uses the Application.Name property to check whether the name of an application is contained in a specified list. If it is, it blocks a request for this application.
The action settings specify a message to the requesting user.

The rule is not enabled by default.

Block web applications with high risk
Application.HighRisk equals true AND Application.Name is in list Web Browsing and Web Conferencing –> Block<Default>
The rule uses the Application.HighRisk property to check the reputation score of an application and the Application.Name property to check whether the name of this application is contained in a specified list. If the reputation score reaches or exceeds the high-risk level and the application name is also on the list, it blocks a request for this application.
The action settings specify a message to the requesting user.

Block Facebook chat
Application.ToString (Application .Name) equals "Facebook.Chat" –> Block<Default>
The rule uses the Application.To String property to check whether the name of an application is equal to a specified string. For this purpose, the name of the application is converted into a string. If the converted application name equals the specified string, a request for the application is blocked.
The action settings specify a message to the requesting user.

The rule is not enabled by default.

Block Applications in Response Cycle

This nested rule set handles application filtering in the response cycle.

Nested library rule set – Block Applications in Response Cycle
Criteria – *Always*
Cycle – Responses

The rule set contains the following rule:

Applications to be looked for in response cycle
Application.Name is in list of Applications to Search for in Response Cycle –> Block<Default>
The rule uses the Application.Name property to check whether the name of an application is contained in a specified list. If it is, it blocks a request for this application.
The action settings specify a message to the requesting user.

The rule is not enabled by default.

Block web applications with high risk
Application.HighRisk equals true AND Application.Name is in list Web Browsing and Web Conferencing –> Block<Default>
The rule uses the Application.HighRisk property to check the reputation score of an application and the Application.Name property to check whether the name of this application is contained in a specified list. If the reputation score reaches or exceeds the high-risk level and the application name is also on the list, it blocks a request for this application.
The action settings specify a message to the requesting user.

Block Facebook chat
Application.ToString (Application .Name) equals "Facebook.Chat" –> Block<Default>
The rule uses the Application.To String property to check whether the name of an application is equal to a specified string. For this purpose, the name of the application is converted into a string. If the converted application name equals the specified string, a request for the application is blocked.
The action settings specify a message to the requesting user.

The rule is not enabled by default.

ATD - Offline Scanning with Immediate File Availability rule set

The ATD – Offline Scanning with Immediate File Availability rule set is a library rule set for enabling Web Gateway to work with Advanced Threat Defense when filtering web objects.

When this rule set is implemented, a web object is forwarded to the user who requested it before it has been additionally scanned by Advanced Threat Defense, so the object is immediately available to the user.

If the scanning result is that the web object is infected, a message is sent to the administrator of the network that the user sent the request from.

This use of the scanning functions of Advanced Threat Defense is also known as *offline scanning* or *background scanning*.

After importing this rule set, the following two rule sets are implemented and appear on the rule sets tree:

- ATD - Init Offline Scan
- ATD - Handle Offline Scan

A rule set with the name *ATD - Offline Scanning with Immediate File Availability* is not implemented.

ATD - Init Offline Scan

This rule set initiates the additional scanning by Advanced Threat Defense.

Library rule set – ATD - Init Offline Scan
Criteria – *Antimalware.Proactive.Probability<Gateway Anti-Malware> greater than or equals 60 AND MediaType.EnsuredTypes at least one in list Advanced Threat Defense Supported Types AND Body.Size less than 30000000*
Cycles – Responses, Embedded Objects

The rule set criteria specifies that the rule set applies if the following is true:

- As a result of previous scanning by Web Gateway, the probability that a web object is malicious equals or exceeds 60 percent.
- The media type of the object is on the list of supported types for scanning by Advanced Threat Defense.
- The web object does not exceed a particular size.

The rule set contains the following rule.

Offline scanning with immediate file availability
Antimalware.MATD.InitBackgroundScan(5) equals false –> Block<ATD Communication Failed>
When this rule is processed, all data related to the request for web access that has been sent to Web Gateway is recorded, including the response that was received from the requested web server. The response usually includes in its body the requested web object, for example, a file. The body with the web object is stored on Web Gateway. An internal request is also created within Web Gateway to initiate the scanning by Advanced Threat Defense. Web Gateway then waits for an answer to this internal request to see whether the request is accepted and the scanning will be performed. The time that Web Gateway waits for this answer is measured in seconds and a parameter of the *Antimalware.MATD.InitBackgroundScan* property. By default, this time is 5 seconds. You can configure this time by editing the property parameter.
If no answer to the internal request is received within the configured time, the property is set to *false*, so this criteria matches and the rule applies. A message is then sent to inform the administrator that the additional scanning by Advanced Threat Defense could not be executed. If the answer is received within the configured time, the web object is forwarded to the user.
Further handling of the additional scanning is performed by the next rule set..

Library rule set – ATD - Handle Offline Scan
Criteria – *Antimalware.MATD.IsBackgroundScan equals true*
Cycles – Requests, Embedded Objects

The rule set criteria specifies that the rule set applies if the value of the *Antimalware.MATD.IsBackgroundScan* is *true*.

It is true if the additional scanning by Advanced Threat Defense has successfully been initiated by the rule in the preceding rule set . In this case, the data that was recorded and stored by this rule is used by Advanced Threat Defense to scan a requested web object.

The rule set contains the following rules.

Upload file to ATD and wait for scanning result
Antimalware.Infected<Gateway ATD> equals true –> Continue – Statistics.Counter.Increment("BlockedByMATD",1)<Default>
The rule uses the *Antimalware.Infected* property to check whether a web object, for example, a file, is infected by a virus or other malware. The scanning that is required for this check is performed under the Gateway ATD settings, which means it is carried out by Advanced Threat Defense. For this purpose, the previously stored web object is forwarded from Web Gateway to Advanced Threat Defense.
If the scanning result is that the web object is infected, this is recorded by a statistics counter.

Offline scanning with immediate file availability
Antimalware.Infected<Gateway ATD> equals true –> Block<Virus Found> – Set User-Defined.MessageText = "Client.IP: " + IP.ToString(Client.IP) + "Requested URL: "

+ URL + "Virus name: " + ListOfString.ToString (Antimalware.VirusNames<Gateway.ATD>, "," Email.Send ("Administrator@", "MATD offline scan detected a virus", User-Defined.MessageText)<Default>
When the rule is processed, it is checked whether the value of the *Antimalware.Infected* property is *true*. If it is, it means the scanning that was performed by Advanced Threat Defense has found a web object to be infected by a virus or other malware.
A warning message is then created and sent to the administrator for the network of the user who sent the request to access the web object. The message contains information on the request that was recorded by the rule of the preceding rule set.

Stop cycle
Always –> Stop Cycle
This rule stops the processing cycle. It is always executed after the preceding rules have been processed.

Complete rules of the ATD - Offline Scanning with Immediate File Availability rule set

When working with the complete rules of the ATD - Offline Scanning with Immediate File Availability rule set, all rules and rule elements of this rule set can be viewed and configured.

After importing this rule set, the following two rule sets are implemented and appear on the rule sets tree:

- ATD - Init Offline Scan
- ATD - Handle Offline Scan

A rule set with the name ATD - Offline Scanning with Immediate File Availability is not implemented.

ATD - Init Offline Scan

This rule set initiates the additional scanning by Advanced Threat Defense.

Library rule set – ATD - Init Offline Scan
Criteria – *Antimalware.Proactive.Probability<Gateway Anti-Malware> greater than or equals 60 AND MediaType.EnsuredTypes at least one in list Advanced Threat Defense Supported Types AND Body.Size less than 30000000*
Cycles – Responses, Embedded Objects

The rule set criteria specifies that the rule set applies if the following is true:

- As a result of previous scanning by Web Gateway, the probability that a web object is malicious equals or exceeds 60 percent.
- The media type of the object is on the list of supported types for scanning by Advanced Threat Defense.
- The web object does not exceed a particular size.

The rule set contains the following rule.

Offline scanning with immediate file availability
Antimalware.MATD.InitBackgroundScan(5) equals false –> Block<ATD Communication Failed>

When this rule is processed, all data related to the request for web access that has been sent to Web Gateway is recorded, including the response that was received from the requested web server. The response usually includes in its body the requested web object, for example, a file. The body with the web object is stored on Web Gateway. An internal request is also created within Web Gateway to initiate the scanning by Advanced Threat Defense. Web Gateway then waits for an answer to this internal request to see whether the request is accepted and the scanning will be performed. The time that Web Gateway waits for this answer is measured in seconds and a parameter of the Antimalware.MATD.InitBackgroundScan property. By default, this time is 5 seconds. You can configure this time by editing the property parameter.
If no answer to the internal request is received within the configured time, the property is set to *false*, so this criteria matches and the rule applies. A message is then sent to inform the administrator that the additional scanning by Advanced Threat Defense could not be executed. If the answer is received within the configured time, the web object is forwarded to the user.
Further handling of the additional scanning is performed by the next rule set..

Library rule set – ATD - Handle Offline Scan
Criteria – *Antimalware.MATD.IsBackgroundScan equals true*
Cycles – Requests, Embedded Objects

The rule set criteria specifies that the rule set applies if the value of the *Antimalware.MATD.IsBackgroundScan* is *true*.

It is true if the additional scanning by Advanced Threat Defense has successfully been initiated by the rule in the preceding rule set . In this case, the data that was recorded and stored by this rule is used by Advanced Threat Defense to scan a requested web object.

The rule set contains the following rules.

Upload file to ATD and wait for scanning result
Antimalware.Infected<Gateway ATD> equals true –> Continue – Statistics.Counter.Increment("BlockedByMATD",1)<Default>
The rule uses the *Antimalware.Infected* property to check whether a web object, for example, a file, is infected by a virus or other malware. The scanning that is required for this check is performed under the Gateway ATD settings, which means it is carried out by Advanced Threat Defense. For this purpose, the previously stored web object is forwarded from Web Gateway to Advanced Threat Defense.
If the scanning result is that the web object is infected, this is recorded by a statistics counter.

Offline scanning with immediate file availability
Antimalware.Infected<Gateway ATD> equals true –> Block<Virus Found> – Set User-Defined.MessageText = "Client.IP: " + IP.ToString(Client.IP) + "Requested URL: " + URL + "Virus name: " + ListOfString.ToString (Antimalware.VirusNames<Gateway.ATD>, "," Email.Send ("Administrator@", "MATD offline scan detected a virus", User-Defined.MessageText)<Default>
When the rule is processed, it is checked whether the value of the Antimalware.Infected property is *true*.

If it is, it means the scanning that was performed by Advanced Threat Defense has found a web object to be infected by a virus or other malware.
A warning message is then created and sent to the administrator for the network of the user who sent the request to access the web object. The message contains information on the request that was recorded by the rule of the preceding rule set.

Stop cycle
Always –> Stop Cycle
This rule stops the processing cycle. It is always executed after the preceding rules have been processed.

Authorized Override rule set

The Authorized Override rule set is a library rule set for imposing a time limit on web usage that can be passed by through the action of authorized user.

Library rule set – Authorized Override
Criteria – *SSL.Client.Context.IsApplied equals true OR Command.Name does not equal "CONNECT"*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies to SSL-secured communication and to any other communication, which does not use the CONNECT command at the beginning.

The following rule sets are nested in this rule set:

- Authorized Override With URL Configuration
- Authorized Override With IP Configuration

  This rule set is not enabled initially.
- Authorized Override With Authenticated User Configuration

  This rule set is not enabled initially.

Authorized Override With URL Configuration

This nested rule set handles authorized overriding related to URL categories.

Nested library rule set – Authorized Override With URL Configuration
Criteria – *URL.Categories<Default> at least one in list URL Categories Blocklist for Authorized Override*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies when a user sends a request for a URL that falls into a category on the blocking list for authorized overriding related to URL categories.

The rule set contains the following rules:

Redirect after authenticating for authorized override
Quota.AuthorizedOverride.IsActivationRequest<URL Category Configuration> equals true AND Authentication.Authenticate<User Database> equals true –> Redirect<Redirection After Authorized Session Activation>

The rule redirects a request to let a user again access a web object after session time has been exceeded and the credentials the user submitted to continue with a new session have been validated.
The action settings specify a message to the requesting user.

Check if authorized override session has been exceeded
Quota.AuthorizedOverride.SessionExceeded<URL Category Configuration> equals true –> Block<Action Authorized Override Blocked>
The rule uses the *Quota.AuthorizedOverride.SessionExceeded* property to check whether the configured session time has been exceeded for a user. If it has, the user's request for web access is blocked.
The *URL Category Configuration* settings, which are specified with the property, are the settings of the module that handles authorized overriding.
The action settings specify a message to the requesting user.

Authorized Override With IP Configuration

This nested rule set handles authorized overriding related to IP addresses.

Nested library rule set – Authorized Override With IP Configuration
Criteria – *Client.IP is in list IP Blocklist for Authorized Override*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies when a user sends a request from a client with an IP address that is on the blocking list for authorized overriding related to IP addresses.

The rules in this rule set are the same as in the Authorized Override with URL Configuration rule set, except for the module settings in the rule criteria, which are *IP Configuration*.

Authorized Override With Authenticated User Configuration

This nested rule set handles authorized overriding related to user names.

Nested library rule set – Authorized Override With Authenticated User Configuration
Criteria – *Authenticated.RawUserName is in list User Blocklist for Authorized Override*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies when a request is sent by a user whose user name is on the blocking list for authorized overriding related to user names.

The rules in this rule set are the same as in the Authorized Override with URL Configuration rule set, except for the module settings in the rule criteria, which are *Authenticated User Configuration*.

Blocking Sessions rule set

The Blocking Sessions rule set is a library rule set for blocking web sessions after an attempt to access a web object that is not allowed.

Library rule set – Blocking Sessions
Criteria – *SSL.Client.Context.IsApplied equals true OR Command.Name does not equal “CONNECT”*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies to SSL-secured communication and to any other communication, which does not use the CONNECT command at the beginning.

The following rule set is nested in this rule set: *Blocking Sessions With URL Configuration*

Blocking Sessions With URL Configuration

This nested rule set handles blocking sessions related to URL categories.

Nested library rule set – Blocking Sessions With URL Configuration
Criteria – *URL.Categories<Default> at least one in list URL Categories Blocklist for Blocking Sessions*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies when a user sends a request for a URL that falls into a category on the blocking list for blocking sessions related to URL categories.

The rule set contains the following rules:

Block user if blocking session is active
BlockingSession.IsBlocked<Blocking Session Configuration> equals true –> Block<Blocking Session Template>
The rule uses the *BlockingSession.IsBlocked* property to check whether a blocking session has been activated for a user who sends a request. If it has, the request is blocked.
The action settings specify a message to the requesting user.

Activate blocking session if category is in list Category List for Blocking Sessions
URL.Categories<Default> at least one in list Category List for Blocking Session –> Continue — BlockingSession.Activate<Blocking Session Configuration>
The rule uses the *URL.Categories* property to check whether a URL that a user requests access to falls into a category on the blocking list maintained especially for blocking sessions. If it falls into a category on the list, a blocking session is activated for the user.
The *BlockingSession.Activate* event is used to activate the blocking session. The event settings are specified with the event.

Bypass ePO Requests rule set

The Bypass ePO Requests rule set is a library rule set for allowing requests from a McAfee ePO server to bypass filtering rules on an appliance.

Library rule set – Bypass ePO Requests
Criteria – *Command.Name equals "CONNECT"*
Cycles – Requests (and IM)

The rule set criteria specifies that the rule set applies when the SSL-secured communication between an ePO server and an appliance begins with a request from the server to connect to the appliance.

The rule set contains the following rule.

Skip subsequent rules for ePO requests
URL.Host equals "127.0.0.1" OR URL.Host equals "[::1]" –> Stop Cycle – Enable SSL Client Context<Default CA> – Enable SSL Scanner <Certificate verification without edh>
The rule uses the URL.Host property to identify the host of a requested URL, based on the IP address of the host.
If this address is 127.0.0.1, the host of the requested URL is the appliance. When the ePO server sends a request to connect to the appliance, it uses this address.
So if 127.0.0.1 is the requested address, the rule applies and stops all further processing in the request cycle. This way the CONNECT request is allowed to pass through.
The next step in this process is sending and verifying certificates. The rule includes an event to enable the sending of a client certificate that is issued by the default certificate authority.
You can modify the event settings to have the certificate issued by another authority.
When certificate verification has been completed, the SSL-secured communication can go ahead.

Bypass Microsoft (Office 365) Services rule set

The Bypass Microsoft (Office 365) Services rule set is the default rule set for letting requests and responses in traffic to and from Office 365 and other Microsoft services bypass filtering on Web Gateway.

When working with this rule set, you can use different views:

- **Key elements view** — Allows you to configure key elements of the rules in this rule set.

  Key elements are those parts of the rules that you will most likely want to work with when configuring your policy for a particular field of web security. You can also enable or disable some rules in this view.
- **Complete rules view** — Allows you to view all rules in the rule set and to configure all their elements, including the key elements.

  You can also enable or disable, move, copy, or delete any of the existing rules, as well as create rules in this view.

Key elements for Microsoft services bypassing

The key elements of the rules that handle bypassing for Office 365 and other Microsoft services are related to the individual services that requests and responses are sent to and received from.

Bypassing for Microsoft services

Options for handling Microsoft services bypassing

Bypassing for Microsoft services

Option	Definition
Bypass Exchange Online, Bypass Microsoft Federation Gateway, and other options for handling Microsoft services bypassing	When selected, a request from a client of Web Gateway to access Exchange Online or another Microsoft service is forwarded to the service unfiltered. When a response is received from the service, it is also passed on to the client unfiltered. None of these options is enabled by default.

Bypass Microsoft (Office 365) Services rule set

The Bypass Microsoft (Office 365) Services rule set is the default rule set for letting requests and responses in traffic to and from Office 365 and other Microsoft services bypass filtering on Web Gateway.

Default rule set – Bypass Microsoft (Office 365) Services
Criteria – Always
Cycles – Requests (and IM), Responses

The rule set contains the following rules.

Shortcut Microsoft service in response
Cycle.Name equals "Response" AND User-Defined.Shortcut_Microsoft_Service equals true – Stop Cycle
The rule uses the Cycle.Name property to find out whether processing on Web Gateway is currently going on in the response cycle.
It also uses a user-defined property to check whether the response that is processed in this cycle was triggered by a client requesting access to Office 365 or any of several other Microsoft services.
If such a request is received on Web Gateway, a particular rule that is processed in the request cycle sets the user-defined property to *true*. The current rule checks whether the property is actually set this way in the response cycle, using the second part of its criteria.
If both criteria parts match, the rule applies and the response cycle is stopped. The response is then forwarded to the requesting client without filtering.
This rule is enabled by default.

Note:

All rules that follow the first rule in the rule set work in a similar way. They ensure that a request sent by a client of Web Gateway to a particular Microsoft service is forwarded to this service unfiltered.

Each of them also sets the property that is evaluated by the first rule to *true* after receiving such a request.

The first of these subsequent rules is explained here as an example in full detail. A summary is then given for all other rules.

Bypass Exchange Online

URL.Destination.IP is in range list Exchange Online IP Addresses OR URL.Destination.IP is in range list Exchange Online Protection P Addresses OR URL.Host matches in list Exchange Online URLs – Stop Cycle – Set User-Defined.Shortcut_Microsoft_Service = true
The rule uses the URL.Destination.IP and URL.Host properties to find out whether the IP address and URL that are sent with a request are on particular lists.
If they are, the request cycle is stopped and the request is forwarded to the requested destination, which is the Microsoft Exchange Online service.
The User-Defined.Shortcut_Microsoft_Service property is then set to *true* by an event. The property is evaluated in the response cycle by the first rule in the rule set.
This rule is not enabled by default.

Bypass Microsoft Federation Gateway, Bypass Microsoft Lync/Skype for Business Online, and other rules for Microsoft services bypassing
Similar to the Bypass Exchange Online rule, these rules use the URL.Destination.IP property or the URL.Host property or both (in one case also the URL property) to find out whether the IP addresses or URLs that are sent with requests are on particular lists. The lists vary with each rule depending on the respective service.
If the IP addresses or URLs are found on the lists, the request cycle is stopped and the request is forwarded to the requested destination, which is one of the Microsoft services.
The User-Defined.Shortcut_Microsoft_Service property is then set to *true* by an event. The property is evaluated in the response cycle by the first rule in the rule set.
None of these rules is enabled by default.

Cloud Storage Encryption rule set

The Cloud Storage Encryption rule set is a library rule set for encrypting and decrypting data that is uploaded to and downloaded from cloud storage services.

Library rule set – Cloud Storage Encryption
Criteria – *Always*
Cycles – Requests (and IM), Responses

The rule set contains the following rules.

Set encryption password
Always –> Continue – Set User-Defined.Encryption Password = "webgateway"
The rule uses an event to set the default password for Web Gateway as the password that is used when data is encrypted.

Enable encryption

CloudEncryption.IsEncryptionSupported<Default> equals true –> Continue – CloudEncryption.Encrypt(User-Defined.Encryption Password)<Default>
The rule uses the CloudEncryption.IsEncryptionSupported property to check whether encryption of data can be performed. If this is the case, an event is used to perform the encryption.

Enable decryption
CloudEncryption.IsDecryptionSupported<Default> equals true –> Continue – CloudEncryption.Decrypt(User-Defined.Encryption Password)<Default>
The rule uses the CloudEncryption.IsDecryptionSupported property to check whether decryption of data can be performed. If this is the case, an event is used to perform the decryption.

Fix content type after decryption
CloudEncryption.IsDecryptionSupported<Default> equals true –> Continue – MediaType.Header.FixContentType
The rule uses the CloudEncryption.IsDecryptionSupported property to check whether a decryption of cloud storage data was performed.
If this is the case, an event is used to modify the Content-Type field in the header of the response that was sent to deliver the data to Web Gateway. Cloud storage services set this field by default to application/octet-stream, as they are not able to recognize real media types when data is encrypted. The *MediaType.Header.FixContentType* event sets the field to a value for a real media type.set to the value
This rule fixes the issue that cloud storage services set this field by default to *application/octet-stream*, as they cannot recognize different media types when data is encrypted. The *MediaType.Header.FixContentType* event sets the field to a value for the real media type.
The rule is not enabled by default.

Log encryption password
CloudEncryption.IsEncryptionSupported<Default> equals true –> Continue – Set User-Defined.encrypt-log.= DateTime.ToGMTString + ", User: " + Authentication.UserName + ", IP: " + IP.ToString (Client.IP) + ", Service: " + CloudEncryption.ServiceName + ", Cipher: " + CloudEncryption.CipherName<Default> + ", Password: " + User-Defined.EncryptionPassword FileSystemLogging.WriteLogEntry (User-Defined.encrypt-log)<Encryption Log>
The rule uses an event to create a log entry for an encryption.

A second event is used to write this entry into the log called Encryption Log, which is specified by the event settings. Since data is written into the log in encrypted format, you need a password to access it (default password: `webgateway`).
The rule is not enabled by default.

Cookie authentication with SAML back end and fixed ACS URL — rule set

To support SAML authentication using an external Identity Provider, Web Gateway performs the Service Provider role. The rules in this rule set support this SAML scenario.

Library rule set – Cookie authentication with SAML back end and fixed ACS URL
Criteria – Always
Cycles – Requests (and IM)

This rule set contains the following nested rule sets:

- Cookie authentication with SAML back end and fixed ACS URL
    - Intercept SAML assertion if IdP uses a fixed ACS URL
    - Cookie authentication at HTTP(S) proxy
        - Set cookie for authenticated clients
        - Authenticate clients with authentication server
    - Cookie authentication at authentication server
        - Authentication server request

This rule set contains the following rule.

Set client context

Rule element	Definition
Criteria	Always
Action	Continue
Events	Enable SSL Client Context without CA <Default Without CA>

This rule secures all HTTP communication with the SSL protocol using the default certificate that comes with Web Gateway or one that you import. To configure the SSL certificate, click <Default Without CA>.

Intercept SAML assertion if IdP uses a fixed ACS URL

The proxy intercepts SAML authentication responses containing a static ACS URL. It processes the SAML response and redirects the SAML assertion to the authentication server, which provides the Assertion Consumer Service.

Nested library rule set – Intercept SAML assertion if IdP uses a fixed ACS URL
Criteria – Command.Name equals "POST" AND URL.Path is in list SAMLAuthResponseList
Cycles – Requests (and IM)

Note: To configure the list of fixed ACS URLs, click SAMLAuthResponseList.

This rule set contains the following rules.

Handle incoming SAML response

Rule element	Definition
Criteria	Always
Action	Continue
Events	Set Authentication.Token = Request.POSTForm.Get ("SAMLResponse") Set Authentication.SAML.RelayState = Request.POSTForm.Get ("RelayState")

The proxy retrieves the SAML response and RelayState parameter from the POST form sent by the external Identity Provider. It stores the response in the Authentication.Token property and the RelayState in the property Authentication.SAML.RelayState. When the Identity Provider does not support dynamic URLs, the proxy uses the URL returned in the RelayState to restore the dynamic authentication server URL.

Redirect SAML assertion to authentication server

Rule element	Definition
Criteria	Always
Action	Block <SAMLRedirectToAuth>
Events	HTTP.SetStatus (200)

After restoring the dynamic authentication server URL, the proxy redirects the SAML assertion (stored in the Authentication.Token property) to the authentication server and sets the HTTP status code to 200 (OK). To provide custom settings for logging purposes, click <SAMLRedirectToAuth>.

Cookie authentication at HTTP(S) proxy

In the context of SAML authentication using an external Identity Provider, the proxy redirects requests that do not contain a valid cookie to the authentication server. The authentication server consumes SAML assertions and stores the user's identity in a cookie.

Nested library rule set – Cookie authentication at HTTP(S) Proxy
Criteria – Authentication.IsServerRequest equals false AND (Connection.Protocol equals "HTTP" OR Connection.Protocol equals "HTTPS") AND Command.Name does not equal "CONNECT" AND Command.Name does not equal "CERTVERIFY"
Cycles – Requests (and IM)

This rule set contains the following nested rule sets:

- Set Cookie for Authenticated Clients
- Authenticate Clients with Authentication Server

Set cookie for authenticated clients

After the authentication server consumes the SAML assertion and stores the user's identity in a cookie, it redirects the user with the cookie through the proxy to the requested application.

Nested library rule set – Set cookie for authenticated clients
Criteria – Authentication.IsLandingOnServer equals true
Cycles – Requests (and IM)

This rule set contains the following rules.

P3P header to permit third party cookies in Internet Explorer

Rule element	Definition
Criteria	Always
Action	Continue
Events	Header.Block.Add ("P3P", "CP="NOI CUR OUR STP STA"")

The P3P string is required for the Platform for Privacy Preferences Project (P3P). The string must match the privacy settings in the user's browser. If the P3P string is not updated as shown in the table and the browser is Internet Explorer, processing fails.

Set cookie and redirect client to the requested URL

Rule element	Definition
Criteria	Always
Action	Redirect <Redirect Back From Authentication Server>
Events	None

The authentication server redirects the authenticated user with a cookie through the proxy to the requested application. To provide custom settings for logging purposes, click <Redirect Back From Authentication Server>.

Authenticate clients with authentication server

The proxy allows requests from external Identity Providers whose URLs are on the SAML IdP Whitelist and checks for a valid cookie in the requests. If none exists, the proxy redirects the requests to the authentication server.

Nested library rule set – Authenticate clients with authentication server
Criteria – Always
Cycles – Requests (and IM)

This rule set contains the following rules.

Allow IDP requests

Rule element	Definition
Criteria	URL.Domain matches in list SAML IdP Whitelist
Action	Stop Rule Set
Events	None

The proxy checks that the URL of the external Identity Provider making a request matches one of the URLs in the SAML IdP Whitelist.

Note: To add URLs to the whitelist, click SAML IdP Whitelist.

Redirect clients that do not have a valid cookie to the authentication server

Rule element	Definition
Criteria	Authentication.Authenticate <Local Cookie Authentication Server> equals false
Action	Authenticate <Default>
Events	None

If the request from the external Identity Provider does not include a valid cookie, the proxy redirects the request to the authentication server. To configure a different authentication method, click <Local Cookie Authentication Server>. To provide custom settings for logging purposes, click <Default>.

Cookie authentication at authentication server

This rule set is a container for the Authentication server request rule set.

Nested library rule set – Cookie authentication at authentication server
Criteria – Always
Cycles – Requests (and IM)

This rule set contains the following nested rule set: Authentication server request.

Authentication server request

The rules in this rule set apply to the authentication server when it manages SAML authentication using an external Identity Provider. The authentication server processes the SAML authentication response, but does not set the cookie in this rule set. Cookie authentication is handled by the rules in the Cookie authentication at HTTP(S) rule set instead.

Nested library rule set – Authentication server request
Criteria – Authentication.IsServerRequest equals true
Cycles – Requests (and IM)

This rule set contains the following rules.

Redirect clients that have a valid cookie

Rule element	Definition
Criteria	Authentication.Authenticate <Authentication Server - Cookie Check> equals true
Action	Redirect <Redirect Back From Authentication Server>
Events	None

The authentication server redirects users having a valid cookie to the proxy. To change the cookie checking settings used by the authentication server, click <Authentication Server - Cookie Check>. To provide custom settings for logging purposes, click <Redirect Back From Authentication Server>.

Prepare fixed ACS URL

Rule element	Definition
Criteria	Always
Action	Continue
Events	Set User-Defined.SAMLUrlRewrite = URL.Protocol + "://" + URL.Host + "- enter your URL here -"

You can configure a static ACS URL for external Identity Providers who do not support dynamic URLs in this rule. If set, this value must match the ACS URL value configured in the SAML Response settings.

POST SAML authentication request

Rule element	Definition
Criteria	Command.Name does not match POST
Action	Block <SAML request>
Events	Set Authentication.SAML.RelayState = URL Set Authentication.Token = Authentication.SAML.CreateAuthnRequest (User-Defined.SAMLUrlRewrite)<SAML Request> HTTP.SetStatus (200)

The authentication server sends the RelayState parameter and SAML authentication request in a POST form to the external Identity Provider. The RelayState parameter saves the value of the authentication server URL at the time the request is created. The request is created using values configured in the Web Gateway interface. The authentication server then sets the HTTP status code to 200 (OK). To change the SAML authentication request configuration, click <SAML Request> in this event.

Handle SAML authentication response

Rule element	Definition
Criteria	Command.Name equals "POST"
Action	Continue
Events	Set Authentication.Token = Request.POSTForm.Get ("SAMLResponse") Set Authentication.IsAuthenticated = Authentication.SAML.ParseAuthnResponse ("POST",

Rule element	Definition
	User-Defined.SAMLUrlRewrite, Authentication.Token) <SAML Response>

This rule retrieves the SAML response in the POST form sent by the external Identity Provider and stores it in the Authentication.Token property. It parses the response and returns a TRUE value if the response is valid and a FALSE value if it is not. To change the SAML authentication response configuration, click <SAML Response>.

Block invalid SAML response

Rule element	Definition
Criteria	Command.Name equals "POST" AND Authentication.IsAuthenticated equals false
Action	Block <Authorized Only>
Events	None

After the SAML response is parsed, this rule checks the value of the property Authentication.IsAuthenticated. If the property is false, the SAML response is invalid and processing of the response is blocked. To provide custom settings for logging purposes, click <Authorized Only>.

Set user name and groups

Rule element	Definition
Criteria	Always
Action	Continue
Events	Set Authentication.UserName = Map.GetStringValue (Authentication.SAML.Attributes, "userId") Set Authentication.UserGroups = String.ToStringList (Map.GetStringValue (Authentication.SAML.Attributes, "userGroup"), ", ", "")

This rule maps the SAML attributes "userId" and "userGroup" to the Authentication.UserName and Authentication.UserGroups properties, respectively. You can use the rule editor to change the names of the SAML attributes that are mapped to the authentication properties.

Block empty user name

Rule element	Definition
Criteria	Authentication.UserName equals ""
Action	Block <Authorized Only>
Events	None

If the user name property is empty, this rule blocks processing of the response. To provide custom settings for logging purposes, click <Authorized Only>.

P3P header to permit third party cookies in Internet Explorer

Rule element	Definition
Criteria	Always
Action	Continue
Events	Header.Block.Add ("P3P", "CP="NOI CUR OUR STP STA"")

The P3P string is required for the Platform for Privacy Preferences Project (P3P). The string must match the privacy settings in the user's browser. If the P3P string is not updated as shown in the table and the browser is Internet Explorer, processing fails.

Redirect authenticated client back to proxy

Rule element	Definition
Criteria	Always
Action	Redirect <Redirect Back From Authentication Server>
Events	None

According to the final rule in the rule set, the authentication server redirects the authenticated user back to the proxy. To provide custom settings for logging purposes, click <Redirect Back From Authentication Server>.

Data Loss Prevention rule set

The Data Loss Prevention (DLP) rule set is a library rule set for preventing sensitive content from leaving your network or inappropriate content from entering it.

Default rule set – Data Loss Prevention (DLP)
Criteria – *Always*
Cycles – Requests (and IM), responses, embedded objects

The following rule sets are nested in this rule set:

- DLP in Request Cycle
- DLP in Response Cycle

  This rule set is not enabled by default.

DLP in Request Cycle

This nested rule set blocks requests that are sent from clients of our network to web servers if it is detected that sensitive content is involved. For example, a request to upload a file to the web that has sensitive content is blocked.

Nested library rule set – DLP in Request Cycle
Criteria – *Cycle.TopName equals "Request"*
Cycles – Requests (and IM) and embedded objects

The rule set criteria specifies that the rule set applies when a request is processed on the appliance.

The rule set contains the following rules:

Block files with HIPAA information
DLP.Classification.BodyText.Matched <HIPAA> equals true –> Block<DLP.Classification.Block> – Statistics.Counter.Increment (“BlockedByDLPMatch”,1)<Default>
The rule uses the *DLP.Classification.BodyText.Matched* property to check whether the body of the request that is currently processed contains text that is considered to be sensitive content. This text could, for example, be in a file that uploading to the web is requested for.
Text is considered to be sensitive content according to the HIPAA health care regulations. Use of the relevant information is configured as part of the module settings, which are specified after the property name.
If there is sensitive content in the text of a request body, the request is blocked. The settings of the Block action specify a message to the requesting user.
The rule also uses an event to count blocking due to a data loss prevention match.

Block files with Payment Card Industry information
DLP.Classification.BodyText.Matched <Payment Card Industry> equals true –> Block<DLP.Classification.Block> – Statistics.Counter.Increment (“BlockedByDLPMatch”,1)<Default>
The rule uses the *DLP.Classification.BodyText.Matched* property to check whether the body of the request that is currently processed contains text that is considered to be sensitive content. This text could, for example, be in a file that uploading to the web is requested for.
Text is considered to be sensitive content according to the regulations that apply for payment cards. A credit card number would, for example, be content under these regulations. Whether there is sensitive content in a text, is detected using appropriate information in the same way as for the HIPAA-related rule.
If there is sensitive content in the text of a request body, the request is blocked. The settings of the Block action specify a message to the requesting user.
The rule also uses an event to count blocking due to a data loss prevention match.

Block files with SOX information
DLP.Classification.BodyText.Matched <SOX> equals true –> Block<DLP.Classification.Block> – Statistics.Counter.Increment (“BlockedByDLPMatch”,1)<Default>
The rule uses the *DLP.Classification.BodyText.Matched* property to check whether the body of the request that is currently processed contains text that is considered to be sensitive content. This text could, for example, be in a file that uploading to the web is requested for.
Text is considered to be sensitive content according to the regulations of the Sarbanes-Oxley (SOX) act on public company accountability. Board meeting minutes would, for example, be sensitive content under this act. Whether there is sensitive content in a text, is detected using appropriate information in the same way as for the HIPAA-related rule.
If there is sensitive content in the text of a request body, the request is blocked. The settings of the Block action specify a message to the requesting user.
The rule also uses an event to count blocking due to a data loss prevention match.

DLP Response Cycle

This nested rule set blocks responses that are received on the appliance from web servers if it is detected that they contain inappropriate content, for example, discriminatory or offensive language.

Nested library rule set – DLP Response Cycle
Criteria – *Cycle.TopName equals "Response"*
Cycles – Responses and embedded objects

The rule set criteria specifies that the rule set applies when a response is processed on the appliance.

The rule set contains the following rule:

Acceptable use
DLP.Classification.BodyText.Matched <Acceptable Use> equals true –> Block<DLP.Classification.Block> – Statistics.Counter.Increment ("BlockedByDLPMatch",1)<Default>
The rule uses the *DLP.Classification.BodyText.Matched* property to check whether the body of the response that is currently processed contains text that is considered to be sensitive content. This text could, for example, be in a file that is sent in response to a download request.
The module that Is called by the rule to find out whether there is inappropriate content in the response body uses appropriate information from classification lists. Use of these lists is configured as part of the module settings, which are specified after the property name.
If there is inappropriate content in the text of a response body, the response is blocked. The settings of the Block action specify a message to the user who the response should have forwarded to.
The rule also uses an event to count blocking due to a data loss prevention match.

Complete rules of the Data Loss Prevention (DLP) rule set

When working with the complete rules of the Data Loss Prevention (DLP) rule set, all rules and rule elements of this rule set can be viewed and configured.

Library rule set – Data Loss Prevention (DLP)
Criteria – *Always*
Cycles – Requests (and IM), Responses, Embedded objects

The following rule sets are nested in this rule set:

- DLP in Request Cycle
- DLP in Response Cycle

This rule set is not enabled by default.

DLP in Request Cycle

This nested rule set blocks requests that are sent from clients of our network to web servers if it is detected that sensitive content is involved. For example, a request to upload a file to the web that has sensitive content is blocked.

Nested library rule set – DLP in Request Cycle
Criteria – *Cycle.TopName equals "Request"*
Cycles – Requests (and IM), Embedded objects

The rule set criteria specifies that the rule set applies when a request is processed on the appliance.

The rule set contains the following rules:

Block files with HIPAA information
DLP.Classification.BodyText.Matched <HIPAA> equals true –> Block<DLP.Classification.Block> – Statistics.Counter.Increment (“BlockedByDLPMatch”,1)<Default>
The rule uses the DLP.Classification.BodyText.Matched property to check whether the body of the request that is currently processed contains text that is considered to be sensitive content. This text could, for example, be in a file that uploading to the web is requested for.
Text is considered to be sensitive content according to the HIPAA health care regulations. Use of the relevant information is configured as part of the module settings, which are specified after the property name.
If there is sensitive content in the text of a request body, the request is blocked. The settings of the Block action specify a message to the requesting user.
The rule also uses an event to count blocking due to a data loss prevention match.

Block files with Payment Card Industry information
DLP.Classification.BodyText.Matched <Payment Card Industry> equals true –> Block<DLP.Classification.Block> – Statistics.Counter.Increment (“BlockedByDLPMatch”,1)<Default>
The rule uses the DLP.Classification.BodyText.Matched property to check whether the body of the request that is currently processed contains text that is considered to be sensitive content. This text could, for example, be in a file that uploading to the web is requested for.
Text is considered to be sensitive content according to the regulations that apply for payment cards. A credit card number would, for example, be content under these regulations. Whether there is sensitive content in a text, is detected using appropriate information in the same way as for the HIPAA-related rule.
If there is sensitive content in the text of a request body, the request is blocked. The settings of the Block action specify a message to the requesting user.
The rule also uses an event to count blocking due to a data loss prevention match.

Block files with SOX information
DLP.Classification.BodyText.Matched <SOX> equals true –> Block<DLP.Classification.Block> – Statistics.Counter.Increment (“BlockedByDLPMatch”,1)<Default>
The rule uses the DLP.Classification.BodyText.Matched property to check whether the body of the request that is currently processed contains text that is considered to be sensitive content. This text could, for example, be in a file that uploading to the web is requested for.

Text is considered to be sensitive content according to the regulations of the Sarbanes-Oxley (SOX) act on public company accountability. Board meeting minutes would, for example, be sensitive content under this act. Whether there is sensitive content in a text, is detected using appropriate information in the same way as for the HIPAA-related rule.
If there is sensitive content in the text of a request body, the request is blocked. The settings of the Block action specify a message to the requesting user.
The rule also uses an event to count blocking due to a data loss prevention match.

DLP Response Cycle

This nested rule set blocks responses that are received on the appliance from web servers if it is detected that they contain inappropriate content, for example, discriminatory or offensive language.

Nested library rule set – DLP Response Cycle
Criteria – *Cycle.TopName equals "Response"*
Cycles – Responses and embedded objects

The rule set criteria specifies that the rule set applies when a response is processed on the appliance.

The rule set contains the following rule:

Acceptable use
DLP.Classification.BodyText.Matched <Acceptable Use> equals true –> Block<DLP.Classification.Block> – Statistics.Counter.Increment (“BlockedByDLPMatch”,1)<Default>
The rule uses the DLP.Classification.BodyText.Matched property to check whether the body of the response that is currently processed contains text that is considered to be sensitive content. This text could, for example, be in a file that is sent in response to a download request.
The module that Is called by the rule to find out whether there is inappropriate content in the response body uses appropriate information from classification lists. Use of these lists is configured as part of the module settings, which are specified after the property name.
If there is inappropriate content in the text of a response body, the response is blocked. The settings of the Block action specify a message to the user who the response should have forwarded to.
The rule also uses an event to count blocking due to a data loss prevention match.

Data Loss Prevention (DLP) with ICAP for Cloud rule set

The Data Loss Prevention (DLP) with ICAP for Cloud rule set is a library rule set. It is used for data loss prevention in the cloud.

When working with this rule set, you can use different views:

- **Key elements view** — Allows you to configure key elements of the rules in this rule set.

  Key elements are those parts of the rules that you will most likely want to work with when configuring your policy for a particular field of web security. You can also enable or disable some rules in this view.
- **Complete rules view** — Allows you to view all rules in the rule set and to configure all their elements, including the key elements.

  You can also enable or disable, move, copy, or delete any of the existing rules, as well as create new rules in this view.

Complete rules of the Data Loss Prevention (DLP) with ICAP for Cloud rule set

When working with the complete rules of the Data Loss Prevention (DLP) with ICAP for Cloud rule set, all rules and rule elements of this rule set can be viewed and configured.

Library rule set – Data Loss Prevention (DLP) with ICAP for Cloud
Criteria — URL.Host does not equal " " AND Cycle.TopName equals "Request" AND InTheCloud equals true
Cycles — Requests (and IM), Embedded Objects

The rule set criteria specifies that the rule set applies if *all* of these criteria match:

- A host name can be found for a URL that is sent in a request to the appliance.
- The processing cycle that is currently performed is the request cycle.
- The rule set is applicable for cloud use

The rule set contains the following rules.

Skip requests that do not carry information
Body.Size equals 0 AND ListOfString.IsEmpty(URL.Parameters) equals true –> Stop Rule Set
The rule uses the Body.Size property to check whether a request has a body that is empty. It also uses the ListOfString.IsEmpty property to check whether a request has URL parameters.
If one of the two parts of this criteria is matched, processing of the rule set stops and the request is not forwarded to the ICAP server.

Skip body that is greater than 50 MB
Body.Size greater than 52428800 –> Stop Rule Set
The rule uses the Body.Size property to check whether the body of a request does not exceed 50 MB. If it does, processing of the rule set stops and the request is not forwarded to the ICAP server.
In the rule set criteria, the size of a request body that must not be exceeded is specified in bytes.

Skip all GET requests
Command.Name equals GET –> Stop Rule Set
The rule uses the Command.Name property to check whether the command that is sent with a request is GET. If it is, processing of the rule set stops and the request is not forwarded to the ICAP server.
This rule is not enabled by default.

Store original authentication method
Always –> Continue – Set User-Defined.Original.Method = Authentication.Method

The rule event always sets the name of the currently used authentication method as the value of a user-defined property to store it, so it can be restored after this name has temporarily been replaced with "NTLM".

Set authentication method to "NTLM" (for ICAP compatibility)
Authentication.Method does not equal "NTLM" AND Authentication.Method does not equal "LDAP" AND Authentication.Method does not equal "Radius" –> Continue — Set Authentication.Method = "NTLM"
The rule uses the Authentication.Method property to check whether the authentication method that is currently in use is NTLM, LDAP or Radius. These methods are compatible with using ICAP in a DLP configuration.
If a different method is used, which would not be compatible, the rule event replaces this method with "NTLM" by setting the value of Authentication.Method accordingly.

Call ReqMod server
ICAP.ReqMod.Satisfaction<ReqMod> equals true –> Stop Cycle
When a request has passed filtering according to the first two rules of the rule set, it is forwarded to the ICAP server. If this has been done, the value of the ICAP.ReqMod.Satisfaction property is *true*.
The rule checks whether this is the case for a request and if it is, stops processing the current cycle, as no more processing of the rules in this cycle is required after forwarding a request to the ICAP server.

Restore original authentication method
Always –> Continue — Set Authentication.Method = User-Defined.Original.Method
The rule event always sets the name that was stored using the user-defined property to the value of the Authentication.Method property. The name of the authentication method is this way restored to its original value.
The rule is only processed if the proceeding rule, which stops processing the remaining rules in the cycle, has not applied.
This means no ICAP communication is performed and the original authentication method, which might not be ICAP-compatible, can be used again.

Default error handler rule set

The Default error handler rule set is the default rule set for error handling.

Default error handler rule set – Default
Criteria – *Always*

The following rule sets are nested in this rule set:

- Long Running Connections
- Monitoring
    - Check CPU Overload
    - Check Cache Partition

    - Check Request Overload

- Log File Manager Incidents
- Handle Update Incidents
- Handle License Incidents
- Block on Antimalware Engine Errors
- Block on URL Filter Errors
- Block on All Errors

Long Running Connections

This nested error handler rule set keeps connections alive when a proxy module error occurs.

Nested error handler rule set – Long Running Connections
Criteria – *Error.ID equals 20000*

The rule set criteria specifies that the rule set applies when the value of the Error.ID property is 20000, which indicates a malfunction of the proxy module.

The rule set contains the following rule.

Keep connection always alive
Always –> Stop Cycle
When the rule is executed, it stops the current processing cycle. The rule is always executed when the criteria of its rule set is matched. Stopping the processing cycle prevents the connection from being closed in the course of further rule processing.
The rule is not enabled by default.

Monitoring

This nested error handler rule set handles measures taken when an incident occurs that involves the appliance system.

Nested error handler rule set – Monitoring
Criteria – *Incident.ID equals 5*

The rule set criteria specifies that the rule set applies when the value of the Incident.ID property is 5, which indicates an incident that involves the appliance system.

The following rule sets are nested in this rule set:

- Check CPU Overload
- Check Cache Partition
- Check Request Overload

Check CPU Overload

This nested error handler rule set handles measures that are taken when the CPU load exceeds a configured value.

Nested error handler rule set – Check CPU Overload
Criteria – *Statistics.Counter.GetCurrent("CPULoad")<Default> greater than or equals 95*

The rule set criteria specifies that the rule set applies when the value of the Statistics.Counter. GetCurrent property for CPU load is 95 or higher. This value indicates the percentage of the maximum load that the CPU is currently running with.

The Statistics module, which provides the value, runs with default settings, as is specified after the CPU Load property parameter.

The rule set contains the following rules.

Create notification message
Always –> Continue – Set User-Defined.loadMessage = “CPU load at “ + Number.ToString (Statistics.Counter.GetCurrent(“CPULoad”)<Default>) + “%”
The rule is always executed when the criteria of its rule set is matched.
The rule then uses an event to set a user-defined property to a chain of values that make up a message text about the CPU overload.
The Continue action lets processing continue with the next rule.

Send SNMP trap and other rules
Always –> Continue – ...
The Send SNMP trap rule and other rules in the rule set are always executed when the rule set criteria is matched.
The rules then use different events for taking measures to make the administrator aware of the CPU overload.
These rules are not enabled by default.

Check Cache Partition

This nested error handler rule set handles measures that are taken when the web cache usage exceeds a configured value.

Nested error handler rule set – Check Cache Partition
Criteria – *Statistics.Counter.GetCurrent(“WebCacheDiskUsage”)<Default> greater than or equals 95*

The rule set criteria specifies that the rule set applies when the value of the Statistics.Counter. GetCurrent property for web cache usage is 95 or higher.This value indicates the percentage of the maximum allowed usage of the web cache that is currently in use.

The Statistics module, which provides the value, runs with default settings, as is specified after the WebCacheDiskUsage property parameter.

The rule set contains the following rules.

Create notification message
Always –> Continue – Set User-Defined.cacheMessage = “Cache partition usage at “ +Number.ToString (Statistics.Counter.GetCurrent(“WebCacheDiskUsage”)<Default>) + “%”
The rule is always executed when the criteria of its rule set is matched.
The rule then uses two events to set user-defined properties. One of these properties is set to the number of requests that are currently processed on the appliance per second. The other is set to a chain of values that make up a message text about the web cache usage..

The Continue action lets processing continue with the next rule.

Send SNMP trap and other rules
Always –> Continue – ...
The Send SNMP trap rule and other rules in the rule set are always executed when the rule set criteria is matched.
The rules then use different events for taking measures to make the administrator aware of the web cache usage.
These rules are not enabled by default.

Check Request Overload

This nested error handler rule set handles measures that are taken when the number of requests processed on an appliance per second exceeds a configured value.

Nested error handler rule set – Check Request Overload
Criteria – *Statistics.Counter.GetCurrent("HttpRequests")<Default> greater than or equals 480000*

The rule set criteria specifies that the rule set applies when the value of the Statistics.Counter. GetCurrent property for requests is 480,000 or higher. This value is the number of requests that are currently processed one an appliance per second.

The Statistics module, which provides the value, runs with default settings, as is specified after the HttpRequests property parameter.

The rule set contains the following rules.

Create notification message
Always –> Continue – Set User-Defined.requestsPerSecond = Statistics.Counter.GetCurrent("HttpRequests")<Default>) / 60 Set User-Defined.requestLoadMessage = "detected high load: " + Number.ToString (User-Defined.requestsPerSecond) + "requests per second"
The rule is always executed when the criteria of its rule set is matched.
The rule then uses two events to set user-defined properties. One of these properties is set to the number of requests that are currently processed on an appliance per second. The other is set to a chain of values that make up a message text about this number.
The Continue action lets processing continue with the next rule.

Send SNMP trap and other rules
Always –> Continue – ...
The Send SNMP trap rule and other rules in the rule set are always executed when the rule set criteria is matched.
The rules then use different events for taking measures to make the administrator aware of the request overload.

These rules are not enabled by default.

Log File Manager Incidents

This nested error handler rule set handles measures taken when an incident occurs that involves the Log File Manager.

Nested error handler rule set – Log File Manager Incidents
Criteria – *Incident.ID greater than or equals 501 AND Incident ID less than or equals 600*

The rule set criteria specifies that the rule set applies when the value of the Incident.ID property is within the range of incidents that involve the Log File Manager.

The rule set contains the following rules.

Create notification message
Incident.ID equals 501 –> Continue – Set User-Defined.notificationMessage = “License expires in ” + Number.ToString (License.RemainingDays) + “ days”
The rule is always executed when the criteria of its rule set is matched.
The rule then uses an event to set a user-defined property to a chain of values that make up a message text on the remaining number of days for your license.
The Continue action lets processing continue with the next rule.

Create syslog entry
Always –> Continue – ...
The Create syslog entry rule and other rules in the rule set check the value of the Incident.ID property in the same way as the Create notification message rule and use different events to take measures if this value is 501.
These rules are not enabled by default.

Handle Update Incidents

This nested error handler rule set handles measures taken when an incident occurs that involves the Log File Manager.

Nested error handler rule set – Handle Update Incidents
Criteria – *IIncident.OriginName equals “Updater” OR Incident.ID equals 850 OR Incident.ID equals 851 OR Incident.ID equals 940 OR Incident.ID equals 941 OR Incident.ID equals 1050 OR Incident.ID equals 1051 OR Incident.ID equals 1650 OR Incident.ID equals 1651*

The rule set criteria specifies that the rule set applies when the update module is specified by the value of the Incident.OriginName property or the value of the Incident.ID property is one of those hat involve the update module.

The rule set contains the following rules.

Create update incident message
Always –> Continue – Set User-Defined.eventMessage =

“Update Event triggered [“ + Number.ToString (Incident.ID) + “]:” + Incident.Description + “; origin:” + Incident.OriginNamey + “; severity:” + Number.ToString (Incident.Severity)
The rule is always executed when the criteria of its rule set is matched.
The rule then uses an event to set a user-defined property to a chain of values that make up a message text about the update incident. The message includes values for several incident properties.
The Continue action lets processing continue with the next rule.

Create syslog entry
Always –> Continue – ...
The Create syslog entry rule and other rules in the rule set use different events to take measures if the respective rule criteria is matched.
These rules are not enabled by default.

Handle License Incidents

This nested error handler rule set handles measures taken when an incident occurs that involves the expiration date of the license for your appliance.

Nested error handler rule set – Handle License Incidents
Criteria – *Incident.ID equals 200*

The rule set criteria specifies that the rule set applies when the value of the Incident.ID property is 200, which indicates that the remaining number of days for your licence has been checked.

The rule set contains the following rules.

Create license incident message
Always –> Continue – Set User-Defined.notificationMessage = “A log file cannot be pushed. Please have a look at the mwg-logfilemanager errors log (/opt/mwg/log/mwg-errors/mwg-logmanager.errors.log).”
The rule checks whether the value of the Incident.ID property is 501, which indicates that the Log File manager could not push a log file.
If this is the case, the rule uses an event to set a user-defined property for sending a notification message to a string value that is the text of this message.
The Continue action lets processing continue with the next rule.

Create syslog entry

Always –> Continue – ...
The Create syslog entry rule and other rules in the rule set use different events to take measures if the respective rule criteria is matched.
These rules are not enabled by default.

Block on Anti-Malware Errors

This nested error handler rule set blocks access to all web objects when the Anti-Malware module cannot be loaded or is overloaded.

Nested error handler rule set – Block on Anti-Malware Errors
Criteria – *Always*

The rule set contains the following rules.

Block if Anti-Malware engine cannot be loaded
Error.ID equals 14000 –> Block<Cannot Load Anti-Malware>
The rule blocks access to all web objects when the value of the Error.ID property is 14000, which indicates an error that prevents the Anti-Malware module (also known as *engine*) from loading.
The action settings specify a message to a requesting user.

Block if Anti-Malware engine is overloaded
Error.ID equals 14001 –> Block<Anti-Malware Engine Overloaded>
The rule blocks access to all web objects when the value of the Error.ID property is 14001, which indicates all connections to the Anti-Malware module (also known as *engine*) are currently in use and the module is overloaded.
The action settings specify a message to a requesting user.

Block on URL Filter Errors

This nested error handler rule set blocks access to all web objects when the URL Filter module cannot be loaded or another error regarding this module occurs.

Nested error handler rule set – Block on URL Filter Errors
Criteria – *Error.ID greater than or equals 15000 AND Error.ID less than or equals 15999*

The rule set criteria specifies that the rule set applies when the value of the Error.ID property lies within the specified range, which is the range for errors related to URL filtering.

The rule set contains the following rules.

Block if the URL Filter engine cannot be loaded
Error.ID equals 15000 OR Error.ID equals 15002 OR Error.ID equals 15004 OR Error.ID equals15005 –> Block<Cannot Load URL Filter>

The rule blocks all requests for web access when the value of the Error.ID property is one of those specified in the rule criteria. These values indicate errors that prevent the URL Filter module (also known as *engine*) from loading.
The action settings specify a message to a requesting user.

Block all other internal URL Filter errors
Always –> Block<Internal URL Filter Error>
The rule is always executed when its rule set applies and the rule preceding it in the rule set has not been executed. The rule then blocks all requests for web access.
The action settings specify a message to a requesting user.

Block on All Errors

This nested error handler rule set blocks access to all web objects when an internal error occurs on the appliance.

Nested error handler rule set – Block on All Errors
Criteria – *Always*

The rule set contains the following rule.

Always block
Always –> Block<Internal Error>
The rule blocks access to all web objects when an internal error occurs.
The action settings specify a message to a user who requested access.
The rule in this rule set is for handling internal errors on the appliance. It is executed at the time when an internal error occurs, which can, of course, not be predicted and can happen at any time during the filtering process or not at all. In this sense, processing the rule is not part of the normal process flow.
After executing the blocking, the rule stops all further processing of rules for the requests,responses, or embedded objects that were being filtered when the internal error occurred.
This way it is ensured that no malicious or inappropriate web objects enter your network or leave it while the appliance is not fully available.
The process flow continues when the next request is received if the internal error did not lead to a general interruption of the appliance functions.

Enable Opener rule set

The Enable Opener rule set is the default rule set for handling file opening on Web Gateway.

Key elements of the Enable Opener rule set

The key elements of the Enable Opener rule set include settings for file opening and several block options.

Key elements of the Enable Opener rule set

Option	Definition
Composite Opener settings	Clicking Edit makes the Composite Opener settings available for editing.
Block encrypted media types	When selected, a rule is enabled that blocks encrypted media types.
Block multipart media types	When selected, a rule is enabled that blocks multipart media types.
Block corrupted media types	When selected, a rule is enabled that blocks corrupted media types.

Complete rules of the Enable Opener rule set

The Enable Opener rule set includes the following rules.

Default rule set – Enable Opener
Criteria – *Always*
Cycles – Requests (and IM), Responses, Embedded Objects

Enable Composite Opener
Always –> Continue – Enable Composite Opener <Default>
The rule uses the *Enable Composite Opener* event to enable the Composite Opener on Web Gateway for file opening.
The opener is enabled with the *Default* settings.

Block encrypted media types
Body.IsEncryptedObject equals true –> Block<Not Supported Archive>
The rule uses the *Body.IsEncryptedObject* property to check whether a requested media type is encrypted.
If it is, the request is blocked and not passed on to the requested web server.
The event settings specify a message to the requesting user.
Block multipart media types
Body.IsMultiPartObject equals true –> Block<Multipart Archive>

The rule uses the *Body.IsMultiPartObject* property to check whether a requested media type is a multipart object.
If it is, the request is blocked and not passed on to the requested web server.
The event settings specify a message to the requesting user.
Block corrupt media types
Body.IsCorruptedObject equals true –> Block<Media Type (Common)>
The rule uses the *Body.IsMultiPartObject* property to check whether a requested media type is a multipart object.
If it is, the request is blocked and not passed on to the requested web server.
The event settings specify a message to the requesting user.

Gateway Anti-Malware rule set

The Gateway Anti-Malware rule set is the default rule set for anti-malware filtering.

Default rule set – Gateway Anti-Malware
Criteria – *Always*
Cycles – Requests (and IM), Responses, Embedded Objects

The rule set contains the following rules.

Allow if user agent matches User Agent Whitelist
Header.Request.Get ("User-Agent") matches in list User Agent WhiteList –> Stop Rule Set
The rule uses the *Header.Request.Get* property to check the user agent information that is sent with the header of a request.
If the user agent in question is on the specified whitelist, processing of the rule set stops, so the blocking rule at the end of the rule set is not processed.
A parameter of the property specifies that it is the user agent information that must be checked when the rule is processed.
This rule is not enabled by default.
Note: Using this rule alone for whitelisting will cause a security problem because usually a client can set whatever user agent it prefers.

Allow URL host that matches in list Anti-Malware URL Whitelist
URL.Host matches in list Anti-Malware URL Whitelist –> Stop Rule Set
The rule uses the *URL.Host* property to check whether a given URL matches one of the entries on the specified whitelist.
If it does, processing of the rule set stops and the blocking rule at the end of the rule set is not processed.

You can use this rule to exempt web traffic from filtering when the hosts of the URLs involved are well-known web servers for which it is safe to assume that they spread no viruses and other malware.

Whitelisting increases performance because it avoids the effort of scanning the respective web objects.

Remove partial content for HTTP requests

Cycle.TopName equals “Request” AND (Connection.Protocol equals “http” OR Connection.Protocol equals “https”) –> Continue – Header.RemoveAll (“Range”)

The rule uses the *Cycle.TopName* and *Connection.Protocol* properties to check whether the current processing cycle is the request cycle and whether a request is sent in HTTP or HTTPS mode.

If this is the case, the *Header.RemoveAll* event modifies the request by removing the specification that only partial content is requested. A request for complete content is then forwarded to the relevant web server and eventually received from there, so that the complete content of a web object can be processed on the appliance.

For example, a complete archive can be opened and scanned for viruses and other malware. Malicious content that is distributed over several parts of a file can be detected by scanning the complete file, while it could go unnoticed if only parts of the file were scanned.

The Continue action lets processing continue with the next rule.

Block partial content for FTP requests

Cycle.TopName equals “Request” AND Connection.Protocol equals “ftp” AND Command.Categories contains “Partial” –> Block<Partial Content Not Allowed>

The rule uses the *Cycle.TopName*, *Connection.Protocol*, and *Command.Categories* properties to check whether the current processing cycle is the request cycle, the request is sent in FTP mode, and the command category used for the FTP transfer contains *Partial* as a string.

This allows Web Gateway to detect an FTP request for partial content and block it.

Unlike with HTTP or HTTPS requests, an FTP request for partial content cannot be modified to make it a request for complete content. However, security problems would arise if partial content was accepted on the appliance, which are the same as the ones that were explained in the comment on the rule for blocking HTTP and HTTPS requests.

The action settings specify a message to the requesting user.

Start Media Stream Scanner on streaming media and skip anti-malware scanning

Cycle.Name equals "Response" AND StreamDetector.IsMediaStream<Default Streaming Detection> equals true –> Stop Rule Set – Enable Media Stream Scanner

The rule uses the *Cycle.Name* property to check whether processing is in the response cycle and the *StreamDetector.IsMediaStream* property to check whether the web object that is sent in response to Web Gateway is streaming media.

If both are the case, processing of the rule set stops, so the remaining rule is not processed, and an event is used to start the Media Stream Scanner.

Block if virus was found

Antimalware.Infected<Gateway Anti-Malware> equals true –> Block<Virus Found> – Statistics.Counter.Increment ("BlockedByAntiMalware",1)<Default>
The rule uses the *Antimalware.Infected* property to check whether a given web object is infected by a virus or other malware.
When the Anti-Malware module is called to scan the object, it runs with the Gateway Anti-Malware settings, as specified with the property. These settings let the module use all its three submodules and their methods to scan web objects.
If the module finds that a web object is infected, processing of all rules stops and the object is not passed on further. Access to it is blocked this way.
In a request cycle, the infected web object is not passed on to the web. In the response and embedded object cycles, it is not passed on to the user who requested it.
The action settings specify a message to this user.
The rule also uses an event to count blocking due to virus and malware infections.
The event parameters specify the counter that is incremented and the size of the increment. The event settings specify the settings of the Statistics module, which executes the counting.

Key elements for anti-malware filtering

The key elements of the rules for anti-malware filtering deal with important parts of this filtering process.

Bypass Scanning for These Agents and Hosts

Key elements for bypassing scanning by the Anti-Malware module

Bypass scanning for these agents and hosts

Option	Definition
User agent whitelist	Clicking Edit opens a window to let you edit the User Agent Whitelist that is used by a rule. You can add, modify, and remove entries on the list.
URL host whitelist	Clicking Edit opens a window to let you edit the URL Host Whitelist that is used by a rule. You can add, modify, and remove entries on the list.

Scanning Options

Key elements for the scanning activities of the Anti-Malware module

Scanning Options

Option	Definition
Remove partial content for HTTP requests	When selected, a rule is enabled that removes the specification in an HTTP or HTTPS request for accessing only a part of the content of a web object and lets the request ask for the complete content. If a web object, for example, a file, is delivered completely by the web server in question, it can also be scanned completely on Web Gateway. A complete scan can detect infections that

Option	Definition
	might not be noticed if only a part of the web object was scanned.
Block partial content for FTP requests	When selected, a rule is enabled that blocks FTP requests for access to only a part of the content of a web object. Under the FTP protocol. it is not possible to remove a specification in a request for access to only a part of the content of a web object. For this reason it might be advisable to block such requests.
Use the Media Stream Scanner	When selected, the Media Stream Scanner scans and delivers web objects that are streaming media chunk-by-chunk, to speed up the process. The proactive functions of the McAfee Gateway Anti-Malware engine are used for the scanning, but the other engines that are available for this purpose on Web Gateway are not involved.

Gateway Anti-Malware Settings

Key elements for configuring the settings of the Anti-Malware module

Gateway Anti-Malware Settings

Option	Definition
Enable Anti-Malware scanning	When selected, a rule is enabled that calls the Anti-Malware module, which scans web objects for infections by viruses and other malware.
Settings	Clicking Edit opens a window to let you edit the settings for the Anti-Malware module.

Complete rules of the Gateway Anti-Malware rule set

When working with the complete rules of the Gateway Anti-Malware rule set, all rules and rule elements of this rule set can be viewed and configured.

Default rule set – Gateway Anti-Malware
Criteria – *Always*
Cycles – Requests (and IM), Responses, Embedded Objects

The rule set contains the following rules.

Allow if user agent matches User Agent Whitelist
Header.Request.Get ("User-Agent") matches in list User Agent WhiteList –> Stop Rule Set
The rule uses the *Header.Request.Get* property to check the user agent information that is sent with the header of a request.

If the user agent in question is on the specified whitelist, processing of the rule set stops, so the blocking rule at the end of the rule set is not processed.
A parameter of the property specifies that it is the user agent information that must be checked when the rule is processed.
This rule is not enabled by default.
Note: Using this rule alone for whitelisting will cause a security problem because usually a client can set whatever user agent it prefers.

Allow URL host that matches in list Anti-Malware URL Whitelist
URL.Host matches in list Anti-Malware URL Whitelist –> Stop Rule Set
The rule uses the *URL.Host* property to check whether a given URL matches one of the entries on the specified whitelist.
If it does, processing of the rule set stops and the blocking rule at the end of the rule set is not processed.
You can use this rule to exempt web traffic from filtering when the hosts of the URLs involved are well-known web servers for which it is safe to assume that they spread no viruses and other malware.
Whitelisting increases performance because it avoids the effort of scanning the respective web objects.

Remove partial content for HTTP requests
Cycle.TopName equals "Request" AND (Connection.Protocol equals "http" OR Connection.Protocol equals "https") –> Continue – Header.RemoveAll ("Range")
The rule uses the *Cycle.TopName* and *Connection.Protocol* properties to check whether the current processing cycle is the request cycle and whether a request is sent in HTTP or HTTPS mode.
If this is the case, the *Header.RemoveAll* event modifies the request by removing the specification that only partial content is requested. A request for complete content is then forwarded to the relevant web server and eventually received from there, so that the complete content of a web object can be processed on the appliance.
For example, a complete archive can be opened and scanned for viruses and other malware. Malicious content that is distributed over several parts of a file can be detected by scanning the complete file, while it could go unnoticed if only parts of the file were scanned.
The Continue action lets processing continue with the next rule.

Block partial content for FTP requests
Cycle.TopName equals "Request" AND Connection.Protocol equals "ftp" AND Command.Categories contains "Partial" –> Block<Partial Content Not Allowed>
The rule uses the *Cycle.TopName*, *Connection.Protocol*, and *Command.Categories* properties to check whether the current processing cycle is the request cycle, the request is sent in FTP mode, and the command category used for the FTP transfer contains *Partial* as a string.
This allows Web Gateway to detect an FTP request for partial content and block it.

Unlike with HTTP or HTTPS requests, an FTP request for partial content cannot be modified to make it a request for complete content. However, security problems would arise if partial content was accepted on the appliance, which are the same as the ones that were explained in the comment on the rule for blocking HTTP and HTTPS requests.
The action settings specify a message to the requesting user.

Start Media Stream Scanner on streaming media and skip anti-malware scanning
Cycle.Name equals "Response" AND StreamDetector.IsMediaStream<Default Streaming Detection> equals true –> Stop Rule Set – Enable Media Stream Scanner
The rule uses the *Cycle.Name* property to check whether processing is in the response cycle and the *StreamDetector.IsMediaStream* property to check whether the web object that is sent in response to Web Gateway is streaming media.
If both are the case, processing of the rule set stops, so the remaining rule is not processed, and an event is used to start the Media Stream Scanner.

Block if virus was found
Antimalware.Infected<Gateway Anti-Malware> equals true –> Block<Virus Found> – Statistics.Counter.Increment (“BlockedByAntiMalware”,1)<Default>
The rule uses the *Antimalware.Infected* property to check whether a given web object is infected by a virus or other malware.
When the Anti-Malware module is called to scan the object, it runs with the Gateway Anti-Malware settings, as specified with the property. These settings let the module use all its three submodules and their methods to scan web objects.
If the module finds that a web object is infected, processing of all rules stops and the object is not passed on further. Access to it is blocked this way.
In a request cycle, the infected web object is not passed on to the web. In the response and embedded object cycles, it is not passed on to the user who requested it.
The action settings specify a message to this user.
The rule also uses an event to count blocking due to virus and malware infections.
The event parameters specify the counter that is incremented and the size of the increment. The event settings specify the settings of the Statistics module, which executes the counting.

Gateway Anti-Malware with TIE rule set

The Gateway Anti-Malware with TIE rule set is a library rule set for integrating anti-malware flitering on Web Gateway with information retrieved from a TIE server.

Library rule set – Gateway Anti-Malware with TIE
Criteria – Always
Cycles – Requests (and IM), Responses, Embedded Objects

The rule set contains the rules that are also contained in the default Gateway Anti-Malware rule set, as well the following rules, which are needed to enable the integrated filtering.

Note: This rule set is provided only in the complete rules view.

TIE - Trusted reputations
MediaType.EnsuredTypes at least one in list Executables AND TIE.Filereputation<TIE Reputations> greater than or equals 70 AND TIE.Filereputation<TIE Reputations> less than or equals 99 –> Stop Rule Set
The rule uses the MediaType.EnsuredTypes property to check whether a given web object is an executable file by looking it up in a list.
It also uses the TIE.Filereputation property to check whether the file reputation score for this object, which is retrieved from a TIE server, is between 70 and 99. This score means that the object is not considered malicious.
When the TIE.Filter module is called to retrieve the file reputation, it runs with the TIE.Reputation settings, as specified with the property.
If all parts of the criteria match, processing of the rule set stops and the rules that follow this rule in the rule set are skipped.
Skipping these rules means that the object is not scanned and filtered by the submodules of the Anti-Malware module on Web Gateway, which include the Gateway Anti-Malware (GAM) and Avira engines.

TIE - Unknown reputations
TIE.Filereputation<TIE Reputations> equals 50 AND TIE.Filereputation<TIE Reputations> greater than 0 –> Continue
The rule uses the TIE.Filereputation property to check whether the file reputation score for this object, which is retrieved from a TIE server, equals 50, which means the reputation is not known.
When the TIE.Filter module is called to retrieve the file reputation, it runs with the TIE.Reputation settings, as specified with the property.
If the criteria matches, processing continues, which means the rule does not take any particular action on objects with unknown reputations.
This rule is not enabled by default.

TIE - Malicious reputations
TIE.Filereputation<TIE Reputations> less than or equals 30 AND TIE.Filereputation<TIE Reputations> greater than 0 –> Block<TIE Reputation>
The rule uses the TIE.Filereputation property to check whether the file reputation score for this object, which is retrieved from a TIE server, is between 30 and 0, which means it is considered malicious.
When the TIE.Filter module is called to retrieve the file reputation, it runs with the TIE.Reputation settings, as specified with the property.
If both parts of the criteria match, processing of all rules stops and the object is not passed on further. Access to it is blocked this way.
In a request cycle, the infected web object is not passed on to the web. In the response and embedded object cycles, it is not passed on to the user who requested it. The action settings specify a message to this user.

This rule is not enabled by default.

Block if virus was found
MediaType.EnsuredTypes at least one in list Executables AND Antimalware.Infected<Gateway Anti-Malware with TIE> equals true AND Antimalware.Proactive.Probability<Gateway Anti-Malware with TIE> greater than or equals 60 AND Antimalware.Proactive.Probability<Gateway Anti-Malware with TIE> less than 80 –> Block<Virus Found> – Statistics.Counter.Increment (“BlockedByAntiMalware”,1)<Default> – TIE: Report File Reputation (30)
The rule uses the MediaType.EnsuredTypes property to check whether a given web object is an executable file by looking it up in a list.
It also uses the Antimalware.Infected and Antimalware.Proactive.Probability properties to find out whether this object is infected by a virus or other malware and whether the probability that it is infected is between 60 and 80, which means it is likely that it is malicious.
When the Anti-Malware module is called to scan the object and rate its malware probability, it runs with the Gateway Anti-Malware with TIE settings, as specified with the properties.
These settings let the module use both its submodules, the Gateway Anti-Malware (GAM) engine and the Avira engine, and their methods to scan web objects.
If all parts of the criteria match, processing of all rules stops and the object is not passed on further. Access to it is blocked this way.
In a request cycle, the infected web object is not passed on to the web. In the response and embedded object cycles, it is not passed on to the user who requested it. The action settings specify a message to this user.
The rule also uses an event to count blocking due to virus and malware infections. The event parameters specify the counter that is incremented and the size of the increment. The event settings specify the settings of the Statistics module, which executes the counting.
The rule uses another event to notify the TIE server that there is a high probability that the scanned object is malicious. Corresponding to this high probability grade, a low reputation score is sent to the TIE server.

Block if virus was found
MediaType.EnsuredTypes at least one in list Executables AND Antimalware.Infected<Gateway Anti-Malware with TIE> equals true AND Antimalware.Proactive.Probability<Gateway Anti-Malware with TIE> greater than or equals 80 AND Antimalware.Proactive.Probability<Gateway Anti-Malware with TIE> less than 90 –> Block<Virus Found> – Statistics.Counter.Increment (“BlockedByAntiMalware”,1)<Default> – TIE: Report File Reputation (15)
The rule uses the MediaType.EnsuredTypes property to check whether a given web object is an executable file by looking it up in a list.
It also uses the Antimalware.Infected and Antimalware.Proactive.Probability properties to find out whether this object is infected by a virus or other malware and whether the probability that is infected is between 80 and 90, which means it is very likely that it is malicious.
When the Anti-Malware module is called to scan the object and rate its malware probability, it runs with the Gateway Anti-Malware with TIE settings, as specified with the properties.
These settings let the module use both its submodules, the Gateway Anti-Malware (GAM) engine and the Avira engine, and their methods to scan web objects.

If all parts of the criteria match, processing of all rules stops and the object is not passed on further. Access to it is blocked this way.

In a request cycle, the infected web object is not passed on to the web. In the response and embedded object cycles, it is not passed on to the user who requested it. The action settings specify a message to this user.

The rule also uses an event to count blocking due to virus and malware infections. The event parameters specify the counter that is incremented and the size of the increment. The event settings specify the settings of the Statistics module, which executes the counting.

The rule uses another event to notify the TIE server that there is a very high probability that the scanned object is malicious. Corresponding to this very high probability grade, a very low reputation score is sent to the TIE server.

Block if virus was found

MediaType.EnsuredTypes at least one in list Executables AND Antimalware.Infected<Gateway Anti-Malware with TIE> equals true AND Antimalware.Proactive.Probability<Gateway Anti-Malware with TIE> greater than or equals 90 –> Block<Virus Found> – Statistics.Counter.Increment (“BlockedByAntiMalware”,1)<Default> – TIE: Report File Reputation (1)

The rule uses the MediaType.EnsuredTypes property to check whether a given web object is an executable file by looking it up in a list.

It also uses the Antimalware.Infected and Antimalware.Proactive.Probability properties to find out whether this object is infected by a virus or other malware and whether the probability that is infected is greater than or equals 90, which means it is almost sure that it is malicious.

When the Anti-Malware module is called to scan the object and rate its malware probability, it runs with the Gateway Anti-Malware with TIE settings, as specified with the properties.

These settings let the module use both its submodules, the Gateway Anti-Malware (GAM) engine and the Avira engine, and their methods to scan web objects.

If all parts of the criteria match, processing of all rules stops and the object is not passed on further. Access to it is blocked this way.

In a request cycle, the infected web object is not passed on to the web. In the response and embedded object cycles, it is not passed on to the user who requested it. The action settings specify a message to this user.

The rule also uses an event to count blocking due to virus and malware infections. The event parameters specify the counter that is incremented and the size of the increment. The event settings specify the settings of the Statistics module, which executes the counting.

The rule uses another event to notify the TIE server that it is almost sure that the scanned object is malicious. Corresponding to this extremely high probability grade, an extremely low reputation score is sent to the TIE server.

Block if virus was found

Antimalware.Infected<Gateway Anti-Malware with TIE> equals true –> Block<Virus Found> – Statistics.Counter.Increment (“BlockedByAntiMalware”, 1)<Default>

The rule uses the Antimalware.Infected property to check whether a given web object is infected by a virus or other malware.

When the Anti-Malware module is called to scan the object, it runs with the Gateway Anti-Malware with TIE settings, as specified with the property.

These settings let the module use both its submodules, the Gateway Anti-Malware (GAM) engine and the Avira engine, and their methods to scan web objects.
If the module finds that a web object is infected, processing of all rules stops and the object is not passed on further. Access to it is blocked this way.
In a request cycle, the infected web object is not passed on to the web. In the response and embedded object cycles, it is not passed on to the user who requested it. The action settings specify a message to this user.
The rule also uses an event to count blocking due to virus and malware infections. The event parameters specify the counter that is incremented and the size of the increment. The event settings specify the settings of the Statistics module, which executes the counting.
Note: The rule does not notify the TIE server of any scanning results.

Global Whitelist rule set

The Global Whitelist rule set is the default rule set for global whitelisting.

Default rule set – Global Whitelist
Criteria – *Always*
Cycles – Requests (and IM), responses, embedded objects

The rule set contains the following rules.

Client IP is in list Allowed Clients
Client.IP is in list Allowed Clients –> Stop Cycle
The rule uses the *Client.IP* property to check whether the IP address of a client that a request was sent from is on the specified whitelist.
If it is, the rule applies and stops the current processing cycle. The request is then forwarded to the appropriate web server.

URL.Host matches in list Global Whitelist
URL.Host matches in list Global Whitelist –> Stop Cycle
The rule uses the *URL.Host* property to check whether the host that a URL sent in a request provides access to is on the specified whitelist.
If it is, the rule applies and stops the current processing cycle. The request is then forwarded to the web server that is the requested host.

Complete rules of the Global Whitelist rule set

When working with the complete rules of the Global Whitelist rule set, all rules and rule elements of this rule set can be viewed and configured.

Default rule set – Global Whitelist
Criteria – *Always*
Cycles – Requests (and IM), Responses, Embedded Objects

The rule set contains the following rules.

Client IP is in list Allowed Clients
Client.IP is in list Allowed Clients –> Stop Cycle
The rule uses the Client.IP property to check whether the IP address of a client that a request was sent from is on the specified whitelist.
If it is, the rule applies and stops the current processing cycle. The request is then forwarded to the appropriate web server.

URL.Host matches in list Global Whitelist
URL.Host matches in list Global Whitelist –> Stop Cycle
The rule uses the URL.Host property to check whether the host that a URL sent in a request provides access to is on the specified whitelist.
If it is, the rule applies and stops the current processing cycle. The request is then forwarded to the web server that is the requested host.

Media Type Filtering rule set

The Media Type Filtering rule set is the default rule set for media type filtering.

Library rule set – Media Type Filtering
Criteria – *Always*
Cycles – Requests (and IM), responses, embedded objects

The following rule sets are nested in this rule set:
• Upload Media Type This rule set is not enabled by default.

• Download Media Type

Upload Media Type
This nested rule set blocks the upload of media belonging to particular media types. It is processed in request cycles when users request to upload media to the web, as well as in embedded object cycles when objects are embedded in media.

Nested library rule set – Upload Media Type
Criteria – *Always*
Cycles – Requests (and IM) and embedded objects

The rule set contains the following rule:

Block types from list Upload Media Type Blocklist
Media.TypeEnsuredTypes at least one in list Upload Media Type Blocklist –> Block<Media Type (Block List)> — Statistics.Counter.Increment ("BlockedByMediaFilter", 1)<Default>
The rule uses the *Media.TypeEnsuredTypes* property to check for media that have their type ensured if they are on the specified list. If they are, access to the media type is blocked and processing rules stops.
The rule uses an event to count blocking due to media type filtering. The event parameters specify the counter that is incremented and the size of the increment. The event settings specify the settings of the Statistics module, which executes the counting.
Processing continues with the next request that is received on the appliance.

Download Media Type
This nested rule set blocks the download of media belonging to particular media types. It is processed in response cycles when web servers send media in response to user requests for downloading them, as well as in embedded object cycles when objects are embedded in media.

Nested library rule set – Download Media Type
Criteria – *Always*
Cycles – Responses and embedded objects

The rule set contains the following rule.

Block types from list Download Media Type Blocklist

Media.TypeEnsuredTypes at least one in list Download Media Type Blocklist –> Block<Media Type (Block List)> — Statistics.Counter.Increment ("BlockedByMediaFilter", 1)<Default>
The rule uses the *Media.TypeEnsuredTypes* property to check for media that have their type ensured if they are on the specified list. If they are, access to the media type is blocked and processing rules stops.
The rule uses an event to count blocking due to media type filtering. The event parameters specify the counter that is incremented and the size of the increment. The event settings specify the settings of the Statistics module, which executes the counting.
Processing continues with the next request that is received on the appliance.

Key elements of the Media Type Filtering rule set

The key elements of the Media Type Filtering rule set deal with important parts of the media type filtering process.

Block Media Types in Uploads

Key elements for filtering media that are uploaded to the web

Block Media Types in Uploads

Option	Definition
Media types to block	Clicking Edit opens a window to let you edit the Upload Media Type Block List that is used by a rule. You can add, modify, and remove entries on the list.

Block Media Types in Downloads

Key elements for filtering media that are downloaded from the web

Block Media Types in Downloads

Option	Definition
Media types to block	Clicking Edit opens a window to let you edit the Download Media Type Block List that is used by a rule. You can add, modify, and remove entries on the list.
Block undetectable media types	When selected, a rule is enabled that blocks media if no type could be detected for them.
Block unsupported media types	When selected, a rule is enabled that blocks media if it belongs to a type that cannot be handled on Web Gateway.
Block multimedia	When selected, a rule is enabled that blocks media if it belongs to the multimedia type.
Block streaming media	When selected, a rule is enabled that blocks media if it is streaming media.

Complete rules of the Media Type Filtering rule set

When working with the complete rules of the Media Type Filtering rule set, all rules and rule elements of this rule set can be viewed and configured.

Library rule set – Media Type Filtering
Criteria – *Always*
Cycles – Requests (and IM), Responses, Embedded Objects

The following rule sets are nested in this rule set:
• Upload Media Type This rule set is not enabled by default. • Download Media Type

Upload Media Type
This nested rule set blocks the upload of media belonging to particular media types. It is processed in request cycles when users request to upload media to the web, as well as in embedded object cycles when objects are embedded in media.

Nested library rule set – Upload Media Type
Criteria – *Always*
Cycles – Requests (and IM), Embedded Objects

The rule set contains the following rule:

Block types from list Upload Media Type Blocklist
Media.TypeEnsuredTypes at least one in list Upload Media Type Blocklist –> Block<Media Type (Block List)> — Statistics.Counter.Increment (“BlockedByMediaFilter”, 1)<Default>
The rule uses the Media.TypeEnsuredTypes property to check for media that have their type ensured if they are on the specified list. If they are, access to the media type is blocked and processing rules stops.
The rule uses an event to count blocking due to media type filtering. The event parameters specify the counter that is incremented and the size of the increment. The event settings specify the settings of the Statistics module, which executes the counting.
Processing continues with the next request that is received on the appliance.

Download Media Type

This nested rule set blocks the download of media belonging to particular media types. It is processed in response cycles when web servers send media in response to user requests for downloading them, as well as in embedded object cycles when objects are embedded in media.

Nested library rule set – Download Media Type
Criteria – *Always*
Cycles – Responses, Embedded Objects

The rule set contains the following rule.

Block types from list Download Media Type Blocklist
Media.TypeEnsuredTypes at least one in list Download Media Type Blocklist –> Block<Media Type (Block List)> — Statistics.Counter.Increment ("BlockedByMediaFilter", 1)<Default>
The rule uses the *Media.TypeEnsuredTypes* property to check for media that have their type ensured if they are on the specified list. If they are, access to the media type is blocked and processing rules stops.
The rule uses an event to count blocking due to media type filtering. The event parameters specify the counter that is incremented and the size of the increment. The event settings specify the settings of the Statistics module, which executes the counting.
Processing continues with the next request that is received on the appliance.

Single Sign On rule set

Using the nested rule sets that come with the Single Sign On rule set, you can configure SSO access to cloud services and applications for users in your organization.

Library rule set – Single Sign On
Criteria – Always
Cycles – Requests (and IM), Responses

The Single Sign On rule set contains the following nested rule sets:

- Select Services
- SSO Management
  - HTTPS Handling
  - Launchpad
  - OTP Authentication
  - Get Login Action
    - Get Attributes on Premise
    - Get Attributes in the Cloud
    - Perform SAML SSO
    - Perform IceToken SSO

    - Process Common Tasks
- Perform SSO

The rule sets nested in the SSO Management rule set are executed when the SSO.IsManagementRequest property returns a true value. This property is set to true in response to internal and external SSO requests, as follows:

- **Internal SSO requests** — The SSO.Action property returns a string value corresponding to an internal SSO request action.
- **External SSO requests** — An external SSO request is sent to the Web Gateway SSO service URL.

The rule sets nested in the Get Login Action rule set fetch user information and perform single sign-on to SAML cloud services and applications.

Select Services rule set

The rules in this rule set retrieve the specified list of cloud services, which the authenticated user or users of a shared account are allowed to access. The list and other information that you configure using the rules in this rule set are then available to the module for other SSO operations.

Nested library rule set – Select Services
Criteria – Always
Cycles – Requests (and IM)

This rule set contains the following rules.

Add default SSO services (individual accounts)

Rule element	**Definition**
Criteria	Authentication.IsAuthenticated equals true AND String.IsEmpty(Authentication.UserName) equals false
Action	Continue
Events	SSO.AddServices ("defaultIDP", Authentication.UserName, Default SSO Services, { "label":"Individual", "permit-usage":"yes", "permit-management":"yes" })<Default>

If the user is authenticated, the Single Sign On module retrieves the specified list of cloud services, which the user is then allowed to access.

The Single Sign On module executes the event with the following properties and settings:

- "defaultIDP" — Specifies the domain in the credential store where user account information is stored.
- Authentication.UserName — Specifies the name of the authenticated user.
- Default SSO Services — Specifies a list of services that the authenticated user is allowed to access.
- The following options form one parameter in JSON format:
    - "label" — Specifies the type of account: individual or shared.
    - "permit-usage" — Allows you to permit, deny, or require OTP authentication for access to the services on the list by the authenticated user. To configure access, specify the following values respectively: "yes", "no", or "otp".

    - "permit-management" — Allows you to permit, deny, or require OTP authentication for access to account management functions by the authenticated user. To configure access, specify the following values respectively: "yes", "no", or "otp".

- <Default> — Specifies settings for connecting to the SSO service provided by Web Gateway.

Add OTP secured SSO services (individual accounts, use after OTP authentication)

Rule element	Definition
Criteria	Authentication.IsAuthenticated equals true AND String.IsEmpty(Authentication.UserName) equals false
Action	Continue
Events	SSO.AddServices ("defaultIDP", Authentication.UserName, OTP Secured SSO Services, { "label":"Individual", "permit-usage":"otp", "permit-management":"otp" })<Default>

If the user is authenticated, the Single Sign On module retrieves the specified list of cloud services. The user is allowed to access or manage these OTP-secured services after authenticating again with a one-time password entered on the launchpad.

The module executes the event with the following properties and settings:

- "defaultIDP" — Specifies the domain in the credential store where user account information is stored.
- Authentication.UserName — Specifies the name of the authenticated user.
- OTP Secured SSO Services — Specifies a list of services that the authenticated user is allowed to access after authenticating again with a one-time password.
- The following options form one parameter in JSON format:
    - "label" — Specifies the type of account: individual or shared.
    - "permit-usage" — Allows you to require OTP authentication for access to the services on the list by the authenticated user. Value: "otp"
    - "permit-management" — Allows you to require OTP authentication for access to account management functions by the authenticated user. Value: "otp"
- <Default> — Specifies settings for connecting to the SSO service provided by Web Gateway.

Add shared SSO services (shared accounts)

Rule element	Definition
Criteria	Always
Action	Continue
Events	SSO.AddServices ("defaultIDP", "sharedAccounts", Shared SSO Services, { "label":"Shared", "permit-usage":"yes", "permit-management":"yes" })<Default>

The Single Sign On module retrieves the specified list of cloud services, which authenticated users of the shared account are then allowed to access.

- "defaultIDP" — Specifies the domain in the credential store where user account information is stored.
- "sharedAccounts" — Specifies a shared account.
- Shared SSO Services — Specifies a list of services, which authenticated users of the shared account are allowed to access.
- The following options form one parameter in JSON format:
    - "label" — Specifies the type of account: individual or shared.
    - "permit-usage" — Allows you to permit, deny, or require OTP authentication for access to the services on the list by users of the shared account. To configure access, specify the following values respectively: "yes", "no", or "otp".
    - "permit-management" — Allows you to permit, deny, or require OTP authentication for access to account management functions by users of the shared account. To configure access, specify the following values respectively: "yes", "no", or "otp".
- <Default> — Specifies settings for connecting to the SSO service provided by Web Gateway.

Handle single sign on using memorable hosts

Rule element	Definition
Criteria	Map.HasKey (SSO Host to Service ID mapping, URL.Host) equals true
Action	Redirect
Events	Set Redirect.URL = "http://" + SSO.ManagementHost<Default> + "/login?service=" + Map.GetStringValue (SSO Host to Service ID mapping, URL.Host)

If the SSO Host to Service ID Mapping includes the host name configured for the requested cloud service, the request is redirected to the URL configured for that service.

The Single Sign On module constructs the redirect URL from the specified string values and the following properties and settings:

- SSO.ManagementHost — Specifies the host name of the SSO service provided by Web Gateway.
- <Default> — Specifies settings for connecting to the SSO service provided by Web Gateway.
- Map.GetStringValue (SSO Host to Service ID Mapping, URL.Host) — Looks up the host name of the requested service in the SSO Host to Service ID map and returns the Service ID of that service.

HTTPS Handling rule set

This rule set secures SSO communication between users and the launchpad with the HTTPS protocol.

Nested library rule set – HTTPS Handling
Criteria – Always
Cycles – Requests (and IM)

This rule set contains the following rules.

Enable SSL

Rule element	Definition
Criteria	Command.Name equals "CONNECT"
Action	Stop Cycle
Events	Enable SSL Client Context without CA <Launchpad certificate> Enable SSL Scanner <Enable Content Inspection>

If an SSO connection is required, this rule stops the request cycle. The Single Sign On module provides an SSL certificate and enables content inspection.

The module executes the events with the following settings:

- <Launchpad certificate> — Specifies the SSL certificate and settings. This certificate can be the default or one that you import.
- <Enable Content Inspection> — Specifies the settings that enable content inspection by the SSL Scanner module.

Enforce SSL

Rule element	Definition
Criteria	Connection.Protocol equals "HTTP"
Action	Redirect<Default>
Events	Set URL.Protocol = "https" Set Redirect.URL = URL

If the connection protocol is HTTP, the Single Sign On module sets the SSO protocol to "https" and the SSO request is redirected to the requested URL.

The rule executes the redirect action with the following settings:

<Default> — Specifies settings for connecting to the SSO service provided by Web Gateway.

Launchpad rule set

This rule set assembles all information needed for generating the launchpad or a logon page.

Nested library rule set – Launchpad
Criteria – Always
Cycles – Requests (and IM)

This rule set contains the following rules.

Create launchpad

Rule element	Definition
Criteria	URL.Path equals "/" OR URL.Path equals "/launchpad"
Action	Block<SSO Launchpad>
Events	HTTP.SetStatus (200)

If the requested URL specifies the SSO service or the launchpad, this rule generates the launchpad using the following settings:

<SSO Launchpad> — Specifies the language and template settings used to generate the launchpad.

Note: We recommend that you do not modify the launchpad settings.

The Single Sign On module sets the HTTP status code to 200 (OK).

Create automatic login page

Rule element	Definition
Criteria	URL.Path equals "/login"
Action	Block<SSO Login Page>
Events	HTTP.SetStatus (200)

If the requested URL specifies the SSO logon page, this rule generates the logon page, including the JavaScript, using the following settings:

<SSO Login Page> — Specifies the language and template settings used to generate the logon page.

Note: We recommend that you do not modify the logon page settings.

The Single Sign On module sets the HTTP status code to 200 (OK).

Create automatic login page (compatibility with some services)

Rule element	Definition
Criteria	URL.Path matches regex(/login-.+)
Action	Block<SSO Login Page>
Events	Set URL.Parameters = List.OfString.Append (URL.Parameters, String.Concat ("service=", String.SubString (URL.Path, 7, -1))) Set URL.Path = "/login" HTTP.SetStatus (200)

This rule applies when the requested URL specifies the SSO logon page using the format "/login-<Service ID>" instead of the default format that the SSO service is expecting: "/login?service=<Service ID>". This rule generates the logon page using the following settings:

<SSO Login Page> — Specifies the language and template settings used to generate the logon page.

Note: We recommend that you do not modify the logon page settings.

The Single Sign On module rebuilds the requested URL using the default format and sets the HTTP status code to 200 (OK).

Note: Some SAML services do not allow query parameters in the IdP URL when single sign-on is SP-initiated.

OTP Authentication rule set

Enabling this rule set allows you to enforce OTP authentication as a secondary authentication method for users who want to access cloud services and applications.

Nested library rule set – OTP Authentication
Criteria – SSO.OtpRequired<Default> equals true
Cycles – Requests (and IM)

The rules in this rule set are executed when the SSO action requires OTP authentication.

Prepare OTP context

Rule element	Definition
Criteria	URL.HasParameter ("requestOTP") equals true OR URL.HasParameter ("pledgeOTP") equals true
Action	Continue
Events	Authentication.SendOTP<OTP>

If there is a request for a one-time password from an authenticated user, the Single Sign On module sends the password to the user. The types of OTP requests are:

- "requestOTP" — The user requests the one-time password through the McAfee OTP server.
- "pledgeOTP" — The user requests the one-time password through Pledge, an OTP client running locally on a computer or mobile device.

The module executes the event with the following settings:

<OTP> — Specifies settings for OTP authentication.

Return OTP context

Rule element	Definition
Criteria	URL.HasParameter ("requestOTP") equals true
Action	Stop Cycle
Events	HTTP.GenerateResponse (JSON.ToString (JSON.StoreByName (JSON.CreateObject, "otp-context", JSON.FromString (Authentication.OTP.Context<OTP>)))) HTTP.SetStatus (403)

If there is a request for a one-time password from an authenticated user, this rule stops the request cycle. The Single Sign On module generates a response containing the OTP context in a JSON object. The OTP context is provided in a header field when the McAfee OTP Server responds with a one-time password.

The module executes this event with the following settings:

<OTP> — Specifies settings for OTP authentication.

The module sets the HTTP status code to 403 (Forbidden).

Verify delivered OTP

Rule element	Definition
Criteria	Authentication.Authenticate<OTP> equals false
Action	Stop Cycle
Events	HTTP.GenerateResponse ("{"authentication-required":"delivered-otp"}") HTTP.SetStatus (403)

If OTP authentication fails, this rule stops the request cycle. The Single Sign On module generates a response specifying the authentication result and method. The method, delivered OTP, specifies delivery of the one-time password by the McAfee OTP Server.

The module executes this event with the following settings:

<OTP> — Specifies settings for OTP authentication.

The module sets the HTTP status code to 403 (Forbidden).

Note: Enable this rule if one-time passwords are delivered by McAfee OTP Server.

Verify Pledge generated OTP

Rule element	Definition
Criteria	Authentication.Authenticate<OTP> equals false
Action	Stop Cycle
Events	HTTP.GenerateResponse ("{"authentication-required":"generated-otp"}") HTTP.SetStatus (403)

If OTP authentication fails, this rule stops the request cycle. The Single Sign On module generates a response specifying the authentication result and method. The method, generated OTP, specifies generation of the one-time password by the Pledge OTP client.

The module executes this event with the following settings:

<OTP> — Specifies settings for OTP authentication.

The module sets the HTTP status code to 403 (Forbidden).

Note: Enable this rule if one-time passwords are generated by the Pledge OTP client.

Get Login Action rule set

This rule set retrieves information about the connector to the requested cloud service or application. For HTTP cloud connectors, processing of the rule set then stops. For other cloud connectors, the rule set checks whether the user has the right to access the requested cloud service or application.

Nested library rule set – Get Login Action
Criteria – SSO.Action<Default> equals "GetLoginAction"
Cycles – Requests (and IM)

This rule set contains the following rules.

Get connector info

Rule element	Definition
Criteria	Always
Action	Continue
Events	Set User-Defined.sso-conn-info = SSO.GetConnectorInfo (String.ToSSOConnector (URL.GetParameter ("service")))

The Single Sign On module retrieves information about the connector to the service the user is requesting and stores it as a JSON object in a local variable named sso-conn-info. This information includes the following:

- **Name** (string) — Specifies a user-defined name for the cloud connector.

- **Service ID** (string) — Uniquely identifies the cloud service or application.
- **Type** (string) — Specifies the authentication method used by the cloud service.
  Values: HTTP, SAML2
- **Inline** (Boolean) — If true, the cloud connector supports a dynamic HTTP cloud service, which requires single sign-on in proxy or inline mode.
- **Deprecated** (Boolean) — If true, the cloud connector is no longer supported.

Stop rule set for form based logins

Rule element	Definition
Criteria	JSON.AsString (JSON.GetByName (User-Defined.sso-conn-info, "type")) equals "http"
Action	Stop Rule Set
Events	None

If the cloud connector type is HTTP, this rule stops the Get Login Action rule set.

Validate user's access permissions

Rule element	Definition
Criteria	SSO.UserHasAccessToService (URL.GetParameter ("realm"), URL.GetParameter ("user"), URL.GetParameter ("service"), "usage")<Default> equals false
Action	Block<SSO: User Has No Access To Service>
Events	None

This rule checks the "service" and "usage" parameters to verify that the user has the right to access the requested service or application. If the "service" parameter is empty or the "usage" parameter is set to "no", this rule blocks access to the requested service.

This rule is executed with the following settings:

- <Default> — Specifies settings for connecting to the SSO service provided by Web Gateway.
- <SSO: User Has No Access To Service> — Specifies the language and template settings used to generate the block message for the user.

Get Attributes on Premise rule set

The rules in this rule set fetch user information from an external LDAP data source for SAML single sign-on. This rule set is disabled by default and only applies when Web Gateway is installed and running on premise and the SSO type is SAML2.

Nested library rule set – Get Attributes on Premise
Criteria – InTheCloud equals false AND JSON.AsString (JSON.GetByName (User-Defined.sso-conn-info, "type")) does not equal "HTTP"
Cycles – Requests (and IM)

This rule set contains the following rules.

Get additional attributes from LDAP

Rule element	Definition
Criteria	Always
Action	Continue
Events	Set Authentication.RawUserName = Authentication.UserName Set User-Defined.sso-user-data = Authentication.GetUserGroupsJSON<LDAP Authentication>

The Single Sign On module fetches information about the user from an external LDAP data source through the Authentication filter. It then stores the information as a JSON object in a local variable named sso-user-data. The user information consists of the attribute name-value pairs expected by the SAML service or application.

This event is executed with the following settings:

<LDAP Authentication> — Specifies the Authentication module settings configured for the external LDAP data source.

Get additional attributes from LDAP using External Lists

Rule element	Definition
Criteria	Always
Action	Continue
Events	Set User-Defined.sso-user-data = ExtLists.JSON (Authentication.UserName, "", "")<LDAP Source>

The Single Sign On module fetches information about the user from an external LDAP data source through the External Lists module. It then stores the information as a JSON object in a local variable named sso-user-data. The user information consists of the attribute name-value pairs expected by the SAML service or application.

This event is executed with the following settings:

<LDAP Source> — Specifies the External Lists module settings configured for the external LDAP data source.

Get Attributes in the Cloud rule set

This rule set constructs the data needed for SAML single sign-on from the authenticated user name. It is disabled by default and only applies when Web Gateway is installed and running in the cloud and the SSO type is SAML2.

Nested library rule set – Get Attributes in the Cloud
Criteria – InTheCloud equals true AND JSON.AsString (JSON.GetByName (User-Defined.sso-conn-info, "type")) does not equal "HTTP"
Cycles – Requests (and IM)

This rule set contains the following rule.

Populate user's data from user name

Rule element	Definition
Criteria	Authentication.IsAuthenticated equals true AND Authentication.UserName matches *@* AND

Rule element	Definition
	JSON.Size (User-Defined.sso-user-data) equals 0
Action	Continue
Events	Set User-Defined.sso-user-data = JSON.StoreByName (User-Defined.sso-user-data, "mail", JSON.FromString (Authentication.UserName))

This rule only applies when the user is authenticated, the user name is an email address, and the sso-user-data variable is empty. The rule stores the attribute name-value pair formed by "mail" and the user's email address as a JSON object in the sso-user-data variable.

Perform SAML SSO rule set

This rule set generates a response that contains the user information needed for completing single sign-on to the requested SAML service or application.

Nested library rule set – Perform SAML SSO
Criteria – Always
Cycles – Requests (and IM), Responses, Embedded Objects

This rule set contains the following rule.

Get login action (SAML)

Rule element	Definition
Criteria	JSON.AsString (JSON.GetByName (User-Defined.sso-conn-info, "type")) matches saml*
Action	Stop Cycle
Events	HTTP.GenerateResponse (SSO.GetSAMLLoginAction (URL.GetParameter ("service"), User-Defined.sso-user-data)<Default>)

If the cloud connector type is SAML2, this rule stops the request cycle. The Single Sign On module generates a response containing the user information needed for completing single sign-on to the requested SAML service or application.

This event is executed with the following settings:

<Default> — Specifies settings for connecting to the SSO service provided by Web Gateway.

Perform IceToken SSO rule set

This rule set generates a response that contains the user information needed for completing single sign-on to the requested service or application.

Nested library rule set — Perform IceToken SSO
Criteria — Always
Cycles — Requests (and IM)

This rule set contains the following rule.

Get login action (IceToken)

Rule element	Definition
Criteria	JSON.AsString (JSON.GetByName (User-Defined.sso-conn-info, "type")) equals "icetoken"
Action	Stop Cycle
Events	HTTP.GenerateResponse (SSO.GetIceTokenLoginAction (URL.GetParameter ("service"), User-Defined.sso-user-data)<Default>)

If the cloud connector type is IceToken, this rule stops the request cycle. The Single Sign On module generates a response containing the user information needed for completing single sign-on to the requested service or application.

This event is executed with the following settings:

<Default> — Specifies settings for connecting to the SSO service provided by Web Gateway.

Process Common Tasks rule set

This rule set processes common SSO tasks and blocks access to SSO resources that do not exist.

Nested library rule set – Block Management Requests
Criteria – Always
Cycles – Requests (and IM)

This rule set contains the following rules.

Process common tasks

Rule element	Definition
Criteria	SSO.ProcessTask<Default> equals true
Action	Stop Cycle
Events	None

This rule processes common SSO tasks, such as credential management.

Block invalid or unhandled management requests

Rule element	Definition
Criteria	Always
Action	Block<File Not Found>
Events	HTTP.SetStatus (404)

This rule blocks access to a requested resource, when the resource does not exist, and is executed with the following settings:

<File Not Found> — Specifies the language and template settings used to generate the block message for the user.

The Single Sign On module sets the HTTP status code to 404 (Not Found).

Perform SSO rule set

This rule set allows the user to log on to an HTTP cloud service or application when single sign-on is implemented in proxy (inline) mode.

Nested library rule set – Perform SSO
Criteria – Always
Cycles – Requests (and IM), Responses

This rule set contains the following rule.

Process form login

Rule element	Definition
Criteria	Always
Action	Continue
Events	SSO.ProcessFormLogin<Default>

The Single Sign On module processes the logon form that users complete to access HTTP cloud services or applications in proxy (inline) mode. The execution of the event depends on the step in the logon process, as follows:

- The user requests the logon form — The event adds JavaScript to the logon page, enabling single sign-on to dynamic HTTP cloud services, and replaces the real password with a password token.
- The user submits the logon form — The event replaces the password token with the real password.

The SSO module executes this event with the following settings:

<Default> - Specifies settings for connecting to the SSO service provided by Web Gateway.

SSL Scanner rule set

The *SSL Scanner* rule set is the default rule set for SSL scanning.

Default rule set – SSL Scanner
Criteria – *Always*
Cycles – Requests (and IM)

The following rule sets are nested in this rule set:
• Handle CONNECT Call • Certificate Verification ◦ Verify Common Name (proxy setup) • Content Inspection • Verify Common Name (transparent setup)

Handle CONNECT Call

This nested rule set handles the CONNECT call in SSL-secured communication and enables certificate verification.

Nested library rule set – Handle CONNECT Call
Criteria – *Command.Name equals "CONNECT"*
Cycles – Requests (and IM)

The rule set criteria specifies that the rule set applies when a request is received on the appliance that contains the CONNECT command, which is sent in the opening phase of SSL-secured communication.

The rule set contains the following rules:

Set client context
Always –> Continue – Enable SSL Client Context with CA <Default CA>
The rule enables the use of a server certificate that is sent to a client.
The event settings specify the McAfee Web Gateway root certificate authority (CA), which is implemented on the appliance after the initial setup, as the default issuer of this certificate.
The Continue action lets processing continue with the next rule.

Tunneled hosts
URL.Host is in list SSL Host Tunnel List –> Stop Cycle
The rule lets requests for access to hosts with a URL that is on the specified whitelist skip SSL scanning.

Restrict destination ports to Allowed CONNECT Ports
URL.Port is not in list Allowed Connect Ports –> Block<Connect not allowed>
The rule blocks requests with destination ports that are not on the list of allowed CONNECT ports.

The action settings specify a message to the requesting user.

Enable certificate verification without EDH for hosts in no-EDH server list
URL.Host is in list No-EDH server –> Block<Connect not allowed> Stop Rule Set – Enable SSL Scanner<Certificate Verification without edh>
The rule enables the certificate verification for requests sent from a host on the no-EDH (Ephemeral Diffie-Hellman) server list.
The action settings specify a message to the requesting user.
The event settings specify running in verification mode for the SSL Scanner module and a special cipher string for data encryption on non-EDH hosts.

Enable certificate verification
Always –> Stop Rule Set – Enable SSL Scanner<Default certificate verification>
The rule enables certificate verification.
The event settings specify that the SSL Scanner module runs in verification mode.

Certificate Verification

This nested rule set handles the CERTVERIFY call in SSL-secured communication. It lets whitelisted certificates skip verification and blocks others according to particular criteria.

Nested library rule set – Certificate Verification
Criteria – *Command.Name equals "CERTVERIFY**
Cycles – Requests (and IM)

The rule criteria specifies that the rule set applies if a request is received on the appliance that contains the CERTVERIFY command, which is sent to request the verification of a certificate.

The following rule set is nested in this rule set:

- Verify Common Name (proxy setup)

The rule set contains the following rules:

Skip verification for certificates found in Certificate Whitelist
SSL.Server.Certificate.HostAndCertificate is in list Certificate Whitelist –> Stop Rule Set
The rule lets whitelisted certificates skip verification.

Block self-signed certificates
SSL.Server.Certificate.SelfSigned equals true –> Block <Certificate incident>
The rule blocks requests with self-signed certificates.

The action settings specify a message to the requesting user.

Block expired server (7 day tolerance) and expired CA certificates
SSL.Server.Certificate.DaysExpired greater than 7 OR SSL.Server.CertificateChain.ContainsExpiredCA<Default> equals true –> Block <Certificate incident>
The rule blocks requests with expired server and CA certificates.
The action settings specify a message to the requesting user.

Block too long certificate chains
SSL.Server.CertificateChain.PathLengthExceeded<Default> equals true –> Block <Certificate incident>
The rule blocks a certificate chain if it exceeds the path length.
The settings in the property specify a list for the module that checks the certificate authorities.
The action settings specify a message to the requesting user.

Block revoked certificates
SSL.Server.CertificateChain.ContainsRevoked<Default> equals true –> Block <Certificate incident>
The rule blocks a certificate chain if one of the included certificates has been revoked.
The settings in the property specify a list for the module that checks the certificate authorities.
The action settings specify a message to the requesting user.

Block unknown certificate authorities
SSL.Server.CertificateChain.FoundKnownCA<Default> equals false –> Block <Certificate incident>
The rule blocks a certificate chain if none of the certificate authorities (CAs) issuing the included certificates is a known CA.
The settings in the property specify a list for the module that checks the certificate authorities.
The action settings specify a message to the requesting user.

Block untrusted certificate authorities
SSL.Server.FirstKnownCAIsTrusted<Default> equals false –> Block <Certificate incident>
The rule blocks a certificate chain if the first known CA that was found is not trusted.
The settings in the property specify a list for the module that checks the certificate authorities.
The action settings specify a message to the requesting user.

Verify Common Name (proxy setup)

This nested rule set verifies the common name in a certificate. It applies to requests sent in explicit proxy mode.

Nested library rule set – Verify Common Name (proxy setup)
Criteria – *Connection.SSL.TransparentCNHandling equals false*
Cycles – Requests (and IM)

The rule criteria specifies that the rule set applies if a request is received on a connection used in SSL-secured communication and verification of the common name is not performed in transparent mode.

The rule set contains the following rules:

Allow matching hostname
URL.Host equals Certificate.SSL.CN –> Stop Rule Set
The rule allows a request if the URL of the requested host is the same as the common name in the certificate.

Allow wildcard certificates
Certificate.SSL.CN.HasWildcards equals true AND URL.Host matches.Certificate.SSL.CN.ToRegex(Certificate.SSL.CN) –> Stop Rule Set
The rule allows requests to hosts sending certificates that have wildcards in their common names matching the URLs of the hosts.
To verify that a common name containing wildcards matches a host, this name is converted into a regular expression.

Allow alternative common names
URL.Host is in list Certificate.SSL.AlternativeCNs –> Stop Rule Set
The rule allows requests to hosts with alternative common names in their certificates if the host matches at least one of them.

Block incident
Always –> Block <Common name mismatch>
If any of the rules for allowing matching common names applies, processing of the rule set stops and this rule is not processed. Otherwise, requests are blocked by this rule because it is then a common name mismatch.
The action settings specify a message to the requesting user.

Content Inspection

This nested rule set completes the handling of a CERTVERIFY call. It lets some requests skip content inspection according to particular criteria and enables inspection for all others.

Nested library rule set – Content Inspection
Criteria – *Command.Name equals "CERTVERIFY*"*

Nested library rule set – Content Inspection
Cycles – Requests (and IM)

The rule criteria specifies that the rule set applies if a request is received on the appliance that contains the CERTVERIFY command, which is sent to request the verification of a certificate.

The rule set contains the following rules:

Skip content inspection for hosts found in SSL Inspection Whitelist
Connection.SSL.Transparent equals false AND URL.Host matches in list SSL Inspection Whitelist –> Stop Rule Set
The rule lets requests sent to whitelisted hosts skip content inspection. It applies only in non-transparent mode.

Skip content inspection for CN found in SSL Inspection Whitelist
Connection.SSL.Transparent equals true AND Certificate.SSL.CN matches in list SSL Inspection Whitelist –> Stop Rule Set
The rule lets requests with whitelisted common names in their certificates skip content inspection. It applies only in transparent mode.
The rule is not enabled initially.

Do not inspect connections with client certificates
Connection.Client.CertificateIsRequested equals true –> Stop Rule Set
The rule lets requests skip inspection if they require the use of client certificates.
The rule is not enabled initially.

Enable content inspection
Always –> Continue – Enable SSL Scanner<Enable content inspection>
The rule enables content inspection.
The event settings specify that the SSL Scanner module runs in inspection mode.
If any of the rules for skipping content inspection applies, processing of the rule set stops and this last rule, which enables the inspection, is not processed. Otherwise, content inspection is enabled by this rule.

Verify Common Name (transparent setup)

This nested rule set verifies the common name in a certificate. It applies to requests sent in explicit proxy mode. It applies only to requests sent in transparent mode.

With requests sent in explicit proxy mode, the host name that is compared to the common name is taken from the CONNECT request that a client sends.

As in transparent mode no CONNECT request is sent, the host name is taken from the request for web access that a client sends.

Nested library rule set – Verify Common Name (transparent setup)
Criteria – *Connection.SSL.TransparentCNHandling equals true AND Command.Name does not equal "CONNECT" AND Command.Name does not equal "CERTVERIFY"*
Cycles – Requests (and IM)

The rule criteria specifies that the rule set applies if a request is received on a connection used in SSL-secured communication and verification of the common name is performed in transparent mode.

The rule set contains the following rules:

Allow matching hostname
URL.Host equals Certificate.SSL.CN –> Stop Rule Set
The rule allows a request if the URL of the requested host is the same as the common name in the certificate.

Allow wildcard certificates
Certificate.SSL.CN.HasWildcards equals true AND URL.Host matches.Certificate.SSL.CN.ToRegex(Certificate.SSL.CN) –> Stop Rule Set
The rule allows requests to hosts sending certificates that have wildcards in their common names matching the URLs of the hosts.
To verify that a common name containing wildcards matches a host, this name is converted into a regular expression.

Allow alternative common names
URL.Host is in list Certificate.SSL.AlternativeCNs –> Stop Rule Set
The rule allows requests to hosts with alternative common names in their certificates if the host matches at least one of them.

Block incident
Always –> Block <Common name mismatch>
If any of the rules for allowing matching common names applies, processing of the rule set stops and this rule is not processed. Otherwise, requests are blocked by this rule because it is then a common name mismatch.
The action settings specify a message to the requesting user.

Complete rules of the HTTPS Scanning rule set

When working with the complete rules of the HTTPS Scanning rule set, all rules and rule elements of this rule set can be viewed and configured.

Default rule set — HTTPS Scanning
Criteria — *Always*
Cycles — Requests (and IM)
This rule set is part of the default rule set system, but not enabled by default.

The following rule sets are nested in this rule set:
• Handle CONNECT Call • Certificate Verification ◦ Verify Signature Algorithms ◦ Verify Common Name (Proxy Setup) • Content Inspection • Verify Common Name (Transparent Setup)

Handle CONNECT Call

This nested rule set handles the CONNECT call in SSL-secured communication and enables certificate verification.

Nested library rule set — Handle CONNECT Call
Criteria — *Command.Name equals "CONNECT"*
Cycles — Requests (and IM)

The rule set criteria specifies that the rule set applies when a request is received on the appliance that contains the CONNECT command, which is sent in the opening phase of SSL-secured communication.

The rule set contains the following rules:

Set client context
Always –> Continue — Enable SSL Client Context with CA <Default CA>
The rule enables the use of a server certificate that is sent to a client.
The event settings specify the McAfee Web Gateway root certificate authority (CA), which is implemented on the appliance after the initial setup, as the default issuer of this certificate.
The Continue action lets processing continue with the next rule.

Tunneled hosts
URL.Host is in list SSL Host Tunnel List –> Stop Cycle
The rule lets requests for access to hosts with a URL that is on the specified whitelist skip HTTPS scanning.

Restrict destination ports to Allowed CONNECT Ports

URL.Port is not in list Allowed Connect Ports –> Block<Connect not allowed>
The rule blocks requests with destination ports that are not on the list of allowed CONNECT ports.
The action settings specify a message to the requesting user.

Enable certificate verification without EDH for hosts in no-EDH server list
URL.Host is in list No-EDH server –> Block<Connect not allowed> Stop Rule Set — Enable SSL Scanner<Certificate Verification without edh>
The rule enables the certificate verification for requests sent from a host on the non-EDH (Ephemeral Diffie-Hellman) server list.
The action settings specify a message to the requesting user.
The event settings specify running in verification mode for the SSL Scanner module and a special cipher string for data encryption on non-EDH hosts.

Enable certificate verification
Always –> Stop Rule Set — Enable SSL Scanner<Default certificate verification>
The rule enables certificate verification.
The event settings specify that the SSL Scanner module runs in verification mode.

Certificate Verification

This nested rule set handles the CERTVERIFY call in SSL-secured communication. It lets whitelisted certificates skip verification and blocks others according to particular criteria.

Nested library rule set — Certificate Verification
Criteria — *Command.Name equals “CERTVERIFY**
Cycles — Requests (and IM)

The rule criteria specifies that the rule set applies if a request is received on the appliance that contains the CERTVERIFY command, which is sent to request the verification of a certificate.

The following rule sets are nested in this rule set:

- Verify Signature Algorithms
- Verify Common Name (Proxy Setup)

The rule set contains the following rules:

Skip verification for certificates found in Certificate Whitelist
SSL.Server.Certificate.HostAndCertificate is in list Certificate Whitelist –> Stop Rule Set
The rule lets whitelisted certificates skip verification.

Block self-signed certificates
SSL.Server.Certificate.SelfSigned equals true –> Block <Certificate incident>
The rule blocks requests with self-signed certificates.
The action settings specify a message to the requesting user.

Block expired server (7 day tolerance) and expired CA certificates
SSL.Server.Certificate.DaysExpired greater than 7 OR SSL.Server.CertificateChain.ContainsExpiredCA<Default> equals true –> Block <Certificate incident>
The rule blocks requests with expired server and CA certificates.
The action settings specify a message to the requesting user.

Block too long certificate chains
SSL.Server.CertificateChain.PathLengthExceeded<Default> equals true –> Block <Certificate incident>
The rule blocks a certificate chain if it exceeds the path length.
The settings in the property specify a list for the module that checks the certificate authorities.
The action settings specify a message to the requesting user.

Block revoked certificates
SSL.Server.CertificateChain.ContainsRevoked<Default> equals true –> Block <Certificate incident>
The rule blocks a certificate chain if one of the included certificates has been revoked.
The settings in the property specify a list for the module that checks the certificate authorities.
The action settings specify a message to the requesting user.

Paranoid Certificate Chain Verification
SSL.Server.CertificateChain.AllRevocationStatusesKnown<Default> equals false OR SSL.Server.CertificateChain.IsComplete<Default> equals false –> Block <Certificate incident>
The rule blocks a certificate chain if the revocation status of at least one certificate is unknown or if the certificate chaiin is incomplete.
The settings in the property specify a list for the module that checks the certificate authorities.
The action settings specify a message to the requesting user.

Block unknown certificate authorities

SSL.Server.CertificateChain.FoundKnownCA<Default> equals false –> Block <Certificate incident>
The rule blocks a certificate chain if none of the certificate authorities (CAs) issuing the included certificates is a known CA.
The settings in the property specify a list for the module that checks the certificate authorities.
The action settings specify a message to the requesting user.

Block untrusted certificate authorities
SSL.Server.FirstKnownCAIsTrusted<Default> equals false –> Block <Certificate incident>
The rule blocks a certificate chain if the first known CA that was found is not trusted.
The settings in the property specify a list for the module that checks the certificate authorities.
The action settings specify a message to the requesting user.

Verify Signature Algorithms

This nested rule set verifies the algorithms that are used in creating signatures for certificates.

Nested library rule set – Verify Signature Algorithms
Criteria – *Always*
Cycles – Requests (and IM)

The rule criteria specifies that the rule set applies for all requests that are received.

The rule set contains the following rules:

Verify signature algorithms
SSL.Server.Certificate.SignatureMethod is in list Safe Signature Algorithms AND *SSL.Server.CertificateChain.SignatureMethods<Default> is in list Safe Signature Algorithms* –> Stop Rule Set
The rule uses the SSL.Server.Certificate.SignatureMethod and SSL.Server.CertificateChain.SignatureMethods properties to check whether a signature algorithm for a certificate that was sent with a request is on both of the two lists referred to in the rule criteria.
If a signature algorithm is on these lists, processing of the rule set stops, so the blocking rule that follows this rule is not processed anymore.

Block unsafe signature algorithms
Always –> Block <Certificate incident>
The rule blocks any request that has passed the filtering that was performed when processing the preceding rule. This means that blocking will occur whenever a signature algorithm is not on the lists used in that rule.
The action settings specify a message to the requesting user.

Verify Common Name (Proxy Setup)

This nested rule set verifies the common name in a certificate. It applies to requests sent in explicit proxy mode.

Nested library rule set — Verify Common Name (Proxy Setup)
Criteria — *Connection.SSL.TransparentCNHandling equals false*
Cycles — Requests (and IM)

The rule criteria specifies that the rule set applies if a request is received on a connection used in SSL-secured communication and verification of the common name is not performed in transparent mode.

The rule set contains the following rules:

Allow matching hostname
URL.Host equals Certificate.SSL.CN –> Stop Rule Set
The rule allows a request if the URL of the requested host is the same as the common name in the certificate.

Allow wildcard certificates
Certificate.SSL.CN.HasWildcards equals true AND URL.Host matches.Certificate.SSL.CN.ToRegex(Certificate.SSL.CN) –> Stop Rule Set
The rule allows requests to hosts sending certificates that have wildcards in their common names matching the URLs of the hosts.
To verify that a common name containing wildcards matches a host, this name is converted into a regular expression.

Allow alternative common names
URL.Host is in list Certificate.SSL.AlternativeCNs –> Stop Rule Set
The rule allows requests to hosts with alternative common names in their certificates if the host matches at least one of them.

Block incident
Always –> Block <Common name mismatch>
If any of the rules for allowing matching common names applies, processing of the rule set stops and this rule is not processed. Otherwise, requests are blocked by this rule because it is then a common name mismatch.
The action settings specify a message to the requesting user.

Content Inspection

This nested rule set completes the handling of a CERTVERIFY call. It lets some requests skip content inspection according to particular criteria and enables inspection for all others.

Nested library rule set — Content Inspection
Criteria — *Command.Name equals "CERTVERIFY**
Cycles — Requests (and IM)

The rule criteria specifies that the rule set applies if a request is received on the appliance that contains the CERTVERIFY command, which is sent to request the verification of a certificate.

The rule set contains the following rules:

Skip content inspection for hosts found in SSL Inspection Whitelist
Connection.SSL.Transparent equals false AND URL.Host matches in list SSL Inspection Whitelist –> Stop Rule Set
The rule lets requests sent to whitelisted hosts skip content inspection. It applies only in non-transparent mode.

Skip content inspection for CN found in SSL Inspection Whitelist
Connection.SSL.Transparent equals true AND Certificate.SSL.CN matches in list SSL Inspection Whitelist –> Stop Rule Set
The rule lets requests with whitelisted common names in their certificates skip content inspection. It applies only in transparent mode.
The rule is not enabled initially.

Do not inspect connections with client certificates
Connection.Client.CertificateIsRequested equals true –> Stop Rule Set
The rule lets requests skip inspection if they require the use of client certificates.
The rule is not enabled initially.

Enable content inspection
Always –> Continue — Enable SSL Scanner<Enable content inspection>
The rule enables content inspection.
The event settings specify that the SSL Scanner module runs in inspection mode.
If any of the rules for skipping content inspection applies, processing of the rule set stops and this last rule, which enables the inspection, is not processed. Otherwise, content inspection is enabled by this rule.

Verify Common Name (Transparent Setup)

This nested rule set verifies the common name in a certificate. It applies to requests sent in explicit proxy mode. It applies only to requests sent in transparent mode.

With requests sent in explicit proxy mode, the host name that is compared to the common name is taken from the CONNECT request that a client sends.

As in transparent mode no CONNECT request is sent, the host name is taken from the request for web access that a client sends.

Nested library rule set — Verify Common Name (Transparent Setup)
Criteria — *Connection.SSL.TransparentCNHandling equals true AND Command.Name does not equal "CONNECT" AND Command.Name does not equal "CERTVERIFY"*
Cycles — Requests (and IM)

The rule criteria specifies that the rule set applies if a request is received on a connection used in SSL-secured communication and verification of the common name is performed in transparent mode.

The rule set contains the following rules:

Allow matching hostname
URL.Host equals Certificate.SSL.CN –> Stop Rule Set
The rule allows a request if the URL of the requested host is the same as the common name in the certificate.

Allow wildcard certificates
Certificate.SSL.CN.HasWildcards equals true AND URL.Host matches.Certificate.SSL.CN.ToRegex(Certificate.SSL.CN) –> Stop Rule Set
The rule allows requests to hosts sending certificates that have wildcards in their common names matching the URLs of the hosts.
To verify that a common name containing wildcards matches a host, this name is converted into a regular expression.

Allow alternative common names
URL.Host is in list Certificate.SSL.AlternativeCNs –> Stop Rule Set
The rule allows requests to hosts with alternative common names in their certificates if the host matches at least one of them.

Block incident
Always –> Block <Common name mismatch>
If any of the rules for allowing matching common names applies, processing of the rule set stops and this rule is not processed. Otherwise, requests are blocked by this rule because it is then a common name mismatch.
The action settings specify a message to the requesting user.

SSO Log rule set

The SSO Log rule set is activated when the request is made by an SSO component, including the SSO.Client and SSO.Proxy components.

SSO Log rule set

Library rule set – SSO Log
Criteria – JSON.AsString (JSON.GetByName (SSO.LogAttributes, "origin")) matches SSO.*
Cycles – Requests (and IM), Responses, Embedded Objects

The SSO.LogAttributes property is a JSON object containing the SSO request attributes shown in the following table. The SSO Log rule set generates the SSO access log and optionally the SSO trace log from the attributes in the JSON object.

SSO.LogAttributes property

SSO request log attribute	Definition
action	Specifies the name of the internal action performed in response to the SSO request. Examples include: • LoadLaunchpad • GetServices • StartHTMLLogin, StartSAMLLogin, and StartIceTokenLogin • AddCredentials, UpdateCredentials, and DeleteCredentials
config	Specifies the name of the settings used by the internal action performed in response to the SSO request.
message	Describes the SSO request.
origin	Specifies the source of the values that the proxy copies to the SSO.LogAttributes property. The source can be one of the following SSO components: • SSO.Client — The proxy copies the values provided by the client (browser) to this property without checking them first. • SSO.Proxy — The proxy checks the values provided by the client (browser) before copying them to this property. SSO.Client values are used by developers when testing and debugging SSO features and are included in the SSO trace log. For security reasons, only values checked by the proxy (SSO.Proxy values) are included in the SSO access log.
level	Specifies the log level. Only SSO requests having a log level of four or less are included in the SSO access log. SSO requests having a log level higher than four are also included in the SSO trace log, which is more detailed. The log levels are: • Off (0) — Logging is turned off. • Error (1, 2) — Only error messages are logged. • Info (3, 4) — Error and info messages are logged to the SSO access log file. • Full (5, 6) — All messages are logged to the SSO trace log file.
service	Specifies the name of the cloud service in the SSO request.
outward	Specifies whether Web Gateway performs the web server role or the web server is external to Web Gateway. This attribute has one of the following values: • FALSE — Web Gateway is the destination of the SSO request and creates the SSO response. In this case, Web Gateway performs the role of a web server. For example, Web Gateway performs the web server role when the user accesses the launchpad. • TRUE — The SSO request is directed to an external web server, which creates the SSO response. In this case, Web Gateway does not perform the role of a web server.

SSO Access Log rule set

If the Access Log rule set's criteria are met, the rule in this rule set writes a log entry to the SSO access log file. Each SSO log entry corresponds to one SSO request. To meet the criteria, the SSO component making the request must be the proxy and the log level in the request must be less than or equal to four.

Nested library rule set – Access Log
Criteria – JSON.AsString (JSON.GetByName (SSO.LogAttributes, "origin")) matches SSO.Proxy* AND JSON.AsNumber (JSON.GetByName (SSO.LogAttributes, "level")) less than or equals 4
Cycles – Requests (and IM), Responses, Embedded Objects

This rule set contains the following rule.

Write sso_access.log

Rule element	Definition
Criteria	Always
Action	Continue
Events	Set User-Defined.logLine = DateTime.ToWebReporterString + " "" + Authentication.UserName + "" " + String.ReplaceIfEquals (IP.ToString (Client.IP), "", "-") + " " + String.ReplaceIfEquals (Number.ToString (Response.StatusCode), "", "-") + " "" + Request.Header.FirstLine + "" " + """" + JSON.AsString (JSON.GetByName (SSO.LogAttributes, "action")) + "" "" + JSON.AsString (JSON.GetByName (SSO.LogAttributes, "service")) + "" "" + JSON.AsString (JSON.GetByName (SSO.LogAttributes, "message")) + """" FileSystemLogging.WriteLogEntry (User-Defined.logLine)<SSO Access Log>

This rule creates the SSO access log entry, then writes the entry to the SSO access log file. The rule creates the log entry by retrieving the following information in string format and concatenating the strings:

- Date and time stamp in Web Reporter format
- User name
- Client IP address (if it exists)
- Status code in the response (if it exists)
- First line of the SSO request header
- Type of SSO request (action)
- Name of the cloud service in the SSO request (service)
- Description of the SSO request (message)

Note: To open and configure the file system log settings, click <SSO Access Log>.

SSO Trace Log rule set

The rules in the Trace Log rule set build an SSO trace log entry and write it to the SSO trace log file. The trace log is more detailed than the access log and is intended for debugging the SSO feature.

Note: The Trace Log rule set is disabled by default. When you enable trace logging, we recommend that you set the log level to Full. To locate the log level setting, select Policy → Settings → Engines → Single Sign On → Default → Advanced Settings.

Nested library rule set – Trace Log
Criteria – Always
Cycles – Requests (and IM), Responses, Embedded Objects

This rule set contains the following rules.

Web reporter timestamp

Rule element	Definition
Criteria	Always
Action	Continue
Events	Set User-Defined.logLine = DateTime.ToWebReporterString

This rule sets the SSO trace log entry equal to the date and time stamp in Web Reporter format.

Add all sso attributes

Rule element	Definition
Criteria	Always
Action	Continue
Events	Set User-Defined.logLine = User-Defined.logLine + " "" + JSON.ToString (SSO.LogAttributes) + """"

This rule adds the SSO log attributes in string format to the existing SSO trace log entry.

Add firstline for outward requests

Rule element	Definition
Criteria	JSON.AsBool (JSON.GetByName (SSO.LogAttributes, "outward")) equals true
Action	Continue
Events	Set User-Defined.logLine = User-Defined.logLine + " "" + Request.Header.FirstLine

Rule element	Definition
	+ "'"

If the SSO request is handled by an external web server, this rule adds the first line of the request header to the SSO trace log entry.

Add firstline

Rule element	Definition
Criteria	Always
Action	Continue
Events	Set User-Defined.logLine = User-Defined.logLine + " '" + Request.Header.FirstLine + "'"

This rule is disabled by default. When enabled, it adds the first line of the SSO request header to the SSO trace log entry for external and internal requests.

Write sso_trace.log

Rule element	Definition
Criteria	Always
Action	Continue
Events	FileSystemLogging.WriteLogEntry (User-Defined.logLine)<SSO Trace Log>

This rule writes the SSO trace log entry to the SSO trace log file. To open and configure the file system log settings, click <SSO Trace Log>.

SSO Stop Logging rule set

The SSO Stop Logging rule set stops the logging cycle after internal SSO requests are logged to the SSO access log and before they can be logged to the general access log.

Nested library rule set – Stop Logging
Criteria – Always
Cycles – Requests (and IM), Responses, Embedded Objects

This rule set contains one rule.

Avoid additional logging of internal SSO requests

Rule element	Definition
Criteria	JSON.AsBool (JSON.GetByName (SSO.LogAttributes, "outward")) equals false
Action	Stop Cycle

Rule element	Definition
Events	None

If the SSO request is handled by Web Gateway internally, this rule stops the current cycle of the SSO Log rule set. This action prevents internal SSO requests from being logged to the general access log.

Note: For this rule to be effective, you must add the SSO Log rule set to the Log Handler tree above the Default logging rule set.

Time Quota rule set

The Time Quota rule set is a library rule set for imposing time quotas on web usage.

Library rule set – Time Quota
Criteria – *SSL.Client.Context.IsApplied equals true OR Command.Name does not equal "CONNECT"*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies to SSL-secured communication and to any other communication, which does not use the CONNECT command at the beginning.

The following rule sets are nested in this rule set:

- Time Quota With URL Configuration
- Time Quota With IP Configuration

  This rule set is not enabled initially.
- Time Quota With Authenticated User Configuration

  This rule set is not enabled initially.

Time Quota With URL Configuration

This nested rule set handles time quotas related to URL categories.

Nested library rule set – Time Quota With URL Configuration
Criteria – *URL.Categories<Default> at least one in list URL Categories Blocklist for Time Quota*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies when a user sends a request for a URL that falls into a category on the blocking list for time quotas related to URL categories.

The rule set contains the following rules:

Redirecting after starting new time session
Quota.Time.IsActivationRequest equals true –> Redirect<Redirection After Time Session Activation>
The rule redirects a request to let a user again access a web object after session time has been exceeded and the user has chosen to continue with a new session.
The action settings specify a message to the requesting user.

Check if time session has been exceeded

Quota.Time.Session.Exceeded<URL Category Configuration> equals true –> Block<ActionTimeSessionBlocked>
The rule uses the *Quota.Time.SessionExceeded* property to check whether the configured session time has been exceeded for a user. If it has, the user's request for web access is blocked.
The *URL Category Configuration* settings, which are specified with the property, are the settings of the module that handles time quotas.
The action settings specify a message to the requesting user.

Check if time quota has been exceeded
Quota.Time.Exceeded<URL Category Configuration> equals true –> Block<ActionTimeQuotaBlocked>
The rule uses the *Quota.Time.Exceeded* property to check whether the configured time quota has been exceeded for a user. If it has, the user's request for web access is blocked.
The *URL Category Configuration* settings, which are specified with the property, are the settings of the module that handles time quotas.
The action settings specify a message to the requesting user.

Time Quota With IP Configuration

This nested rule set handles time quotas related to IP addresses.

Nested library rule set – Time Quota With IP Configuration
Criteria – *Client.IP is in list IP Blocklist for Time Quota*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies when a user sends a request from a client with an IP address that is on the blocking list for time quotas related to IP addresses.

The rules in this rule set are the same as in the Time Quota with URL Configuration rule set, except for the module settings that appear in the rule criteria, which are *IP Configuration*.

Time Quota With Authenticated User Configuration

This nested rule set handles time quotas related to user names.

Nested library rule set – Time Quota With Authenticated User Configuration
Criteria – *Authenticated.RawUserName is in list User Blocklist for Time Quota*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies when a request is sent by a user whose user name is on the blocking list for time quotas related to user names.

The rules in this rule set are the same as in the Time Quota with URL Configuration rule set, except for the module settings that appear in the rule criteria, which are *Authenticated User Configuration*.

URL Filtering rule set

The URL Filtering rule set is the default rule set for URL filtering.

When working with this rule set, you can use different views:

- **Key elements view** — Allows you to configure key elements of the rules in this rule set.

  Key elements are those parts of the rules that you will most likely want to work with when configuring your policy for a particular field of web security. You can also enable or disable some rules in this view.
- **Complete rules view** — Allows you to view all rules in the rule set and to configure all their elements, including the key elements.

  You can also enable or disable, move, copy, or delete any of the existing rules, as well as create new rules in this view.

General rule

The URL Filtering rule set includes a general rule and two nested rule sets for performing different kinds of URL filtering.

The general rule is by default processed before the work flow continues with the nested rule sets.

Name			
Set policy-filtered flag			
Criteria		Action	Event
Always	–>	Continue	Set User-Defined.alreadyFiltered = false

The rule uses an event to set a user-defined property for indicating whether the URL filtering rules were already processed for a given request to *false*.

The property serves as a flag, which is checked at the beginning of each of the two nested rule sets. When the first nested rule set is processed, a rule in this rule set the flag to *true*.

When processing of the first rule set is completed or the rule set was not processed because its criteria was not matched, the value of the flag is checked in the criteria of the second rule set.

If the value of the flag is *true*, the second rule set is not processed, as URL filtering has already been performed under the rules of the first rule set. Otherwise, the second rule set is processed.

Nested rule sets

The following nested rule sets are by default included in the URL Filtering rule set:

- Special URL Filtering Group rule set — Allows you to specify particular users, user groups, and ranges of IP addresses that URL filtering is performed for.
- Default rule set — Lets you perform URL filtering for all users, user groups, and IP addresses.

The key elements view and the complete rules view are both available for each of these nested rule sets.

Key elements of the Special URL Filtering Group rule set

The key elements of the Special URL Filtering Group rule set for URL filtering deal with important parts of this process.

Special URL Filtering

Key elements for performing URL filtering according to users, user groups, and IP address ranges.

Special URL Filtering

Option	Definition
User groups to include	Clicking Edit opens a window where you can edit a string list of user groups that URL filtering is to be performed for.

Option	Definition
Users to include	Clicking Edit opens a window where you can edit a string list of users that URL filtering is to be performed for.
IP ranges to include	Clicking Edit opens a window where you can edit a list of IP address ranges that URL filtering is to be performed for.

Basic Filtering

Key elements for performing basic URL filtering.

Basic Filtering

Option	Definition
URL whitelist	Clicking Edit opens a window to let you edit the URL whitelist that is used by a rule. You can add, modify, and remove entries on the list.
URL blocklist	Clicking Edit opens a window to let you edit the URL blocklist that is used by a rule. You can add, modify, and remove entries on the list.
URL category blocklist	Clicking Edit opens a window to let you edit the URL category blocklist that is used by a rule. You can add, modify, and remove entries on the list.

SafeSearch

Key elements for integrating SafeSearch in the URL filtering process.

SafeSearch

Option	Definition
Enable SafeSearch	When selected, a rule is enabled that controls the SafeSearch part of the URL filtering process.
SafeSearch settings	Clicking Edit opens a window to let you edit the settings for the SafeSearch Enforcer module (or engine). This module handles the integration of the SafeSearch Enforcer, which is an additional web security product, in the URL filtering process on Web Gateway.

GTI reputation

Key element for evaluating reputation scores retrieved from the Global Threat Intelligence service within the URL filtering process.

GTI reputation

Option	Definition
Block URLs with a High Risk reputation	When selected, a rule is enabled that blocks URLs with a reputation score that lets them appear to be a high or medium risk to web security. The reputation score of a URL is established by the Global Threat Intelligence service, which is provided by McAfee. It is retrieved from this service by the URL Filter module.

Uncategorized URLs

Key element for handling URLs that could not be categorized during the URL filtering process.

Uncategorized URLs

Option	Definition
Uncategorized URLs	Selecting Block enables a rule that blocks requests for access to web objects with URLs that could not be categorized during the URL filtering process. Selecting Allow means that no action is executed by this rule. URL filtering continues with processing the next rule.

Complete rules of the Special URL Filtering Group rule set

When working with the complete rules of the Special URL Filtering Group rule set for URL filtering, all rules and rule elements of this rule set can be viewed and configured.

Nested default rule set – Special URL Filtering Group
Criteria – *User-Defined.alreadyFiltered* = *false*
Cycles – Requests (and IM)

The rule set contains the following rules.

Allow URLs that match in URL WhiteList
URL matches in list URLWhiteList –> Stop Rule Set
The rule uses the *URL* property to check whether a given URL is on the specified whitelist. If it is, processing of the rule set stops and the blocking rules that follow the whitelisting rule are not processed.
You can use this rule to exempt URLs from filtering to make sure they are available to the users of your network and do not get blocked by any of the following blocking rules. Whitelisting also increases performance because it avoids the effort of retrieving information about the respective URLs.

Block URLs that match in URL BlockList
URL matches in list URL BlockList –> Block<URLBlocked> — Statistics.Counter.Increment ("BlockedByURLFilter",1)<Default>
The rules uses the *URL* property to check whether a given URL is on the specified blocking list. If it is, processing of all rules stops and the request for access to the URL is not passed on to the appropriate web server. Access to it is blocked this way.
The action settings specify a message to the requesting user.
The rule also uses an event to count blocking due to virus and malware infections. The event parameters specify the counter that is incremented and the size of the increment. The event settings specify the settings of the Statistics module, which executes the counting.

Enable SafeSearchEnforcer
Always –> Continue — Enable SafeSearchEnforcer<Default>
The rule enables the SafeSearchEnforcer, which is an additional module for filtering access to web sites with adult content.
The enabling is done by executing an event. The settings of the module are specified with the event.
Processing continues with the next rule.

Allow uncategorized URLs
List.OfCategory.IsEmpty(URL.Categories<Default>) equals true –> Stop Rule Set
The rule uses the *List.OfCategory.IsEmpty* property, which has the URL.Categories property as a parameter, to check whether the list of categories for categorizing a URL is empty. This would mean that the URL is uncategorized, as it could not be assigned to any of the existing categories. Specifying the URL.Categories property as a parameter ensures that it is a particular list of categories that is checked. It is the list that is the value of this property.
To provide a list of categories as the value for the URL.Categories property, the URL Filter module is called, which retrieves this list from the Global Threat Intelligence system. The module runs with the specified Default settings.
If a URL is uncategorized, processing of the rule set stops and the blocking rules that follow this rule are not processed. The request for the URL is forwarded to the appropriate web server and, unless access to the URL is blocked in the response or embedded object cycle, the user is allowed to access the web object that was requested by submitting the URL.

Block URLs whose category is in URL Category BlockList
URL.Categories<Default> at least one in list Category BlockList –> Block<URLBlocked> — Statistics.Counter.Increment (“BlockedByURLFilter”,1)<Default>
The rule uses the *URL.Categories* property to check whether one of the categories a given URL belongs to is on the specified blocking list. The URL Filter module, which is called to retrieve information on these categories, runs with the Default settings, as specified with the property.
If one of the URL’s categories is on the list, processing of all rules stops and the request for access to the URL is not passed on to the appropriate web server. Access to it is blocked this way.
The *URLBlocked* action settings specify that the user who requested this access is notified of the blocking.
The rule also uses an event to count blocking due to URL filtering in the same way as the blocking rule for individual URLs in this rule set.

Block URLs with bad reputation
URL.IsHighRisk<Default> equals true –> Block<URLBlocked> — Statistics.Counter.Increment (“BlockedByURLFilter”,1)<default>
The rules uses the *URL.IsHighRisk* property to find out whether a URL has a reputation that lets access to it appear as a high risk. If the value for this property is true, processing of all rules stops and the request for access to the URL is not passed on to the appropriate web server. Access to it is blocked this way.
The reputation score is retrieved by the URL Filter module, which runs with the settings specified after the property.

The *URLBlocked* action settings specify that the user who requested this access is notified of the blocking.
The rule also uses an event to count blocking due to URL filtering in the same way as the blocking rule for individual URLs in this rule set.

Key elements of the Default rule set for URL filtering

The key elements of the Default rule set for URL filtering deal with important parts of this process.

Basic Filtering

Key elements for performing basic URL filtering.

Basic Filtering

Option	Definition
URL whitelist	Clicking Edit opens a window to let you edit the URL whitelist that is used by a rule. You can add, modify, and remove entries on the list.
URL blocklist	Clicking Edit opens a window to let you edit the URL blocklist that is used by a rule. You can add, modify, and remove entries on the list.
URL category blocklist	Clicking Edit opens a window to let you edit the URL category blocklist that is used by a rule. You can add, modify, and remove entries on the list.

SafeSearch

Key elements for integrating SafeSearch in the URL filtering process.

SafeSearch

Option	Definition
Enable SafeSearch	When selected, a rule is enabled that controls the SafeSearch part of the URL filtering process.
SafeSearch settings	Clicking Edit opens a window to let you edit the settings for the SafeSearch Enforcer module (or engine). This module handles the integration of the SafeSearch Enforcer, which is an additional web security product, in the URL filtering process on Web Gateway.

GTI reputation

Key element for evaluating reputation scores retrieved from the Global Threat Intelligence service within the URL filtering process.

GTI reputation

Option	Definition
Block URLs with a High Risk reputation	When selected, a rule is enabled that blocks URLs with a reputation score that lets them appear to be a high or medium risk to web security.

Option	Definition
	The reputation score of a URL is established by the Global Threat Intelligence service, which is provided by McAfee. It is retrieved from this service by the URL Filter module.

Uncategorized URLs

Key element for handling URLs that could not be categorized during the URL filtering process.

Uncategorized URLs

Option	Definition
Uncategorized URLs	Selecting Block enables a rule that blocks requests for access to web objects with URLs that could not be categorized during the URL filtering process. Selecting Allow means that no action is executed by this rule. URL filtering continues with processing the next rule.

URL Filtering rule set

The URL Filtering rule set is the default rule set for URL filtering.

Default rule set – URL Filtering
Criteria – *Always*
Cycles – Requests (and IM), responses, embedded objects

The rule set contains the following rules.

Allow URLs that match in URL WhiteList
URL matches in list URLWhiteList –> Stop Rule Set
The rule uses the *URL* property to check whether a given URL is on the specified whitelist. If it is, processing of the rule set stops and the blocking rules that follow the whitelisting rule are not processed.
You can use this rule to exempt URLs from filtering to make sure they are available to the users of your network and do not get blocked by any of the following blocking rules. Whitelisting also increases performance because it avoids the effort of retrieving information about the respective URLs.

Block URLs that match in URL BlockList
URL matches in list URL BlockList –> Block<URLBlocked> — Statistics.Counter.Increment ("BlockedByURLFilter",1)<Default>
The rules uses the *URL* property to check whether a given URL is on the specified blocking list. If it is, processing of all rules stops and the request for access to the URL is not passed on to the appropriate web server. Access to it is blocked this way.
The action settings specify a message to the requesting user.

The rule also uses an event to count blocking due to virus and malware infections. The event parameters specify the counter that is incremented and the size of the increment. The event settings specify the settings of the Statistics module, which executes the counting.

Enable SafeSearchEnforcer
Always –> Continue — Enable SafeSearchEnforcer<Default>
The rule enables the SafeSearchEnforcer, which is an additional module for filtering access to web sites with adult content.
The enabling is done by executing an event. The settings of the module are specified with the event.
Processing continues with the next rule.

Allow uncategorized URLs
List.OfCategory.IsEmpty(URL.Categories<Default>) equals true –> Stop Rule Set
The rule uses the *List.OfCategory.IsEmpty* property, which has the URL.Categories property as a parameter, to check whether the list of categories for categorizing a URL is empty. This would mean that the URL is uncategorized, as it could not be assigned to any of the existing categories. Specifying the URL.Categories property as a parameter ensures that it is a particular list of categories that is checked. It is the list that is the value of this property.
To provide a list of categories as the value for the URL.Categories property, the URL Filter module is called, which retrieves this list from the Global Threat Intelligence system. The module runs with the specified Default settings.
If a URL is uncategorized, processing of the rule set stops and the blocking rules that follow this rule are not processed. The request for the URL is forwarded to the appropriate web server and, unless access to the URL is blocked in the response or embedded object cycle, the user is allowed to access the web object that was requested by submitting the URL.

Block URLs whose category is in URL Category BlockList
URL.Categories<Default> at least one in list Category BlockList –> Block<URLBlocked> — Statistics.Counter.Increment (“BlockedByURLFilter”,1)<Default>
The rule uses the *URL.Categories* property to check whether one of the categories a given URL belongs to is on the specified blocking list. The URL Filter module, which is called to retrieve information on these categories, runs with the Default settings, as specified with the property.
If one of the URL’s categories is on the list, processing of all rules stops and the request for access to the URL is not passed on to the appropriate web server. Access to it is blocked this way.
The *URLBlocked* action settings specify that the user who requested this access is notified of the blocking.
The rule also uses an event to count blocking due to URL filtering in the same way as the blocking rule for individual URLs in this rule set.

Block URLs with bad reputation
URL.IsHighRisk<Default> equals true –> Block<URLBlocked> — Statistics.Counter.Increment (“BlockedByURLFilter”,1)<default>

The rules uses the *URL.IsHighRisk* property to find out whether a URL has a reputation that lets access to it appear as a high risk. If the value for this property is true, processing of all rules stops and the request for access to the URL is not passed on to the appropriate web server. Access to it is blocked this way.
The reputation score is retrieved by the URL Filter module, which runs with the settings specified after the property.
The *URLBlocked* action settings specify that the user who requested this access is notified of the blocking.
The rule also uses an event to count blocking due to URL filtering in the same way as the blocking rule for individual URLs in this rule set.

Volume Quota rule set

The Volume Quota rule set is a library rule set for imposing volume quotas on web usage.

Library rule set – Volume Quota
Criteria – *SSL.Client.Context.IsApplied equals true OR Command.Name does not equal "CONNECT"*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies to SSL-secured communication and to other communication that does not use the CONNECT command at the beginning.
The following rule sets are nested in this rule set:
• Time Quota With URL Configuration • Time Quota With IP Configuration This nested rule set is not enabled initially. • Time Quota With Authenticated User Configuration This nested rule set is not enabled initially.

Library rule set – Volume Quota
Criteria – *SSL.Client.Context.IsApplied equals true OR Command.Name does not equal "CONNECT"*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies to SSL-secured communication and to any other communication, which does not use the CONNECT command at the beginning.

The following rule sets are nested in this rule set:

- Volume Quota With URL Configuration
- Volume Quota With IP Configuration

  This rule set is not enabled initially.
- Volume Quota With Authenticated User Configuration

  This rule set is not enabled initially.
- Volume Quota With Media Type Configuration

  This rule set is not enabled initially.

Volume Quota With URL Configuration

This nested rule set handles volume quotas related to URL categories.

Nested library rule set – Volume Quota With URL Configuration
Criteria – *URL.Categories<Default> at least one in list URL Categories Blocklist for Volume Quota*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies when a user sends a request for a URL that falls into a category on the blocking list for volume quotas related to URL categories.

The rule set contains the following rules:

Redirecting after starting new time session
Quota.Volume.IsActivationRequest<URL Category Configuration> equals true –> Redirect<Redirection After Volume Session Activation>
The rule redirects a request to let a user again access a web object after session time has been exceeded and the user has chosen to continue with a new session.
The *URL Category Configuration* settings, which are specified with the property, are the settings of the module that handles volume quotas.
The action settings specify a message to the requesting user.

Check if volume session has been exceeded
Quota.Volume.Session.Exceeded<URL Category Configuration> equals true –> Block<ActionVolumeSessionBlocked>
The rule uses the *Quota.Volume.SessionExceeded* property to check whether the configured session time has been exceeded for a user. If it has, the user's request for web access is blocked.
The *URL Category Configuration* settings, which are specified with the property, are the settings of the module that handles volume quotas.
The action settings specify a message to the requesting user.

Check if volume quota has been exceeded
Quota.Time.Exceeded<URL Category Configuration> equals true –> Block<ActionVolumeSessionBlocked>
The rule uses the *Quota.Volume.Exceeded* property to check whether the configured volume quota has been exceeded for a user. If it has, the user's request for web access is blocked.
The *URL Category Configuration* settings, which are specified with the property, are the settings of the module that handles volume quotas.
The action settings specify a message to the requesting user.

Volume Quota With IP Configuration

This nested rule set handles volume quotas related to IP addresses.

Nested library rule set – Volume Quota With IP Configuration
Criteria – *Client.IP is in list IP Blocklist for Volume Quota*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies when a user sends a request from a client with an IP address that is on the blocking list for volume quotas related to IP addresses.

The rules in this rule set are the same as in the Volume Quota with URL Configuration rule set, except for the module settings that appear in the rule criteria, which are *IP Configuration*.

Volume Quota With Authenticated User Configuration

This nested rule set handles volume quotas related to user names.

Nested library rule set – Volume Quota With Authenticated User Configuration
Criteria – *Authenticated.RawUserName is in list User Blocklist for Volume Quota*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies when a request is sent by a user whose user name is on the blocking list for volume quotas related to user names.

The rules in this rule set are the same as in the Volume Quota with URL Configuration rule set, except for the module settings that appear in the rule criteria, which are *Authenticated User Configuration*.

Volume Quota With Media Type Configuration

This nested rule set handles volume quotas related to media types.

Nested library rule set – Volume Quota With Media Type Configuration
Criteria – *MediaType.FromFileExtension at least one n list Media Type Blocklist for Volume Quota*
Cycle – Requests (and IM)

The rule set criteria specifies that the rule set applies when a request is sent to access a web object belonging to a media type that is on the blocking list for volume quotas related to media types.

The rules in this rule set are the same as in the Volume Quota with URL Configuration rule set, except for the module settings that appear in the rule criteria, which are *Media Type Configuration*.

Web Cache rule set

The Web Cache rule set is a library rule set for web caching.

Library rule set – Web Cache
Criteria – *Always*
Cycles – Requests (and IM) and responses

The following rule sets are nested in this rule set:

- Read from Cache
- Write to Cache

Read from Cache

This nested rule set enables the reading of web objects from the cache and forbids it for URLs that are on a bypassing list.

Nested library rule set – Read from Cache
Criteria – *Always*
Cycles – Requests (and IM)

The rule set contains the following rules.

Skip caching URLs that are in Web Cache URL Bypass List
URL matches in list Web Cache URL Bypass List –> Stop Rule Set
The rule uses the *URL property* to check for requested URLs whether they are on the specified bypass list.
If they are, processing of the rule set stops. The rule that enables reading from the cache is then not processed.
Processing continues with the next rule set.
The rule is not enabled by default.

Enable Web Cache
Always –> Continue — Enable Web Cache
The rule is always processed unless it is skipped because the bypassing rule placed before it in the rule set applies. It enables the web cache, so objects stored in it can be read.
Processing continues with the next rule set.

Write to Cache

This nested rule set enables the writing of web objects to the cache and forbids it for large objects, as well as for URLs and media types on particular bypassing lists.

Nested library rule set – Write to Cache
Criteria – *Always*
Cycles – Responses

The rule set contains the following rules.

Skip caching URLs that are in Web Cache URL Bypass List
URL matches in list Web Cache URL Bypass List –> Stop Rule Set
The rule uses the *URL property* to check for requested URLs whether they are on the specified bypass list.
If they are, processing of the rule set stops. The rule that enables reading from the cache is then not processed.
Processing continues with the next rule set.

The rule is not enabled by default.

Skip caching URLs that are in Web Cache URL Bypass List
URL matches in list Web Cache URL Bypass List –> Stop Rule Set
The rule uses the *URL property* to check for requested URLs whether they are on the specified bypass list.
If they are, processing of the rule set stops. The rule that enables reading from the cache is then not processed.
Processing continues with the next rule set.
The rule is not enabled by default.

Skip caching URLs that are in Web Cache URL Bypass List
URL matches in list Web Cache URL Bypass List –> Stop Rule Set
The rule uses the *URL property* to check for requested URLs whether they are on the specified bypass list.
If they are, processing of the rule set stops. The rule that enables reading from the cache is then not processed.
Processing continues with the next rule set.
The rule is not enabled by default.

Enable Web Cache
Always –> Continue — Enable Web Cache
The rule is always processed unless it is skipped because the bypassing rule placed before it in the rule set applies. It enables the web cache, so objects stored in it can be read.
Processing continues with the next rule set.

Configuration lists

Lists of items for configuring Web Gateway provide an overview and guidance on how to use them. Some items, such as IP addresses and ports, are used for configuring the appliance system that a web security policy is run on, others, such as properties and actions, are used for configuring this policy.

System configuration

The following list is important for system configuration:

- **List of open ports** — Several network ports must be open on the firewall if one exists in a configuration to enable communication between Web Gateway and update servers or databases outside the local network.

Policy configuration

The following lists are important for policy configuration:

- **List of actions** — Actions are configured in web security rules to protect your network against threats arising from the web.
- **List of block reason IDs** — Block reason IDs are configured in block messages to identify the reasons why user requests for web access were blocked.
- **List of error IDs** — Error IDs are configured in the criteria of web security rules to identify errors when measures are taken for handling them.
- **List of events** — Events are configured in web security rules to let activities happen in addition to the execution of rule actions.
- **List of incident IDs** — Incident IDs are configured in the criteria of web security rules to identify incidents when measures are taken for handling them.
- **List of operators** — Operators are configured in the criteria of web security rules to create meaningful connections between properties and their values on one side and operands on the other.
- **List of properties** — Properties are configured in the criteria of web security rules and evaluated in rule processing to determine whether criteria matches and rules apply.
- **List of statistics counters** — Statistics counters are configured in the events of web security rules to record the execution of rule actions.

List of open ports

Several network ports must be open on a firewall if one exists to enable communication between Web Gateway and update servers and databases outside the local network.

Web Gateway accesses these servers and databases to retrieve information in real time.

The following table lists the ports that are usually open by default. Some ports are, however, configurable. Also, not all inbound ports might be open by default, depending on your configuration.

Different directions of web traffic are indicated in the table as:

- **Inbound** — Connection is initiated by a remote system.
- **Outbound** — Connection is initiated by a local system.
- **Bidirectional** — Connection can be initiated from both directions.

List of open ports

Port	Direction	Transport protocol	Application protocol	Destination	Use	Note
22	Inbound	TCP	SSH	Local	Admin secure shell	
161	Inbound	TCP/UDP	SNMP	Local	SNMP	
1080	Inbound	TCP	SOCKS	Local	SOCKS proxy	

Port	Direction	Transport protocol	Application protocol	Destination	Use	Note
1344	Inbound	TCP	ICAP	Local	ICAP	
2121	Inbound	TCP	FTP	Local	FTP control port	
4005	Inbound	TCP	IFP	Local	IFP	
4711	Inbound	TCP	HTTP	Local	Admin interface	Also REST if enabled
4712	Inbound	TCP	HTTPS	Local	Admin interface	Also REST if enabled
4713	Inbound	TCP	HTTP	Local	File server	
4714	Inbound	TCP	HTTPS	Local	File server	
5050	Inbound	TCP	Yahoo	Local	Yahoo proxy	
5190	Inbound	TCP	ICQ	Local	ICQ proxy	
5222	Inbound	TCP	XMPP	Local	XMPP (Jabber) proxy	
9090	Inbound	TCP	HTTP	Local	HTTP(S) proxy	
9393	Inbound	TCP	HTTPS	Local	Intel Active System Console	
15000-20000	Inbound	TCP	FTP	Local	Passive FTP data connection	From FTP client to Web Gateway
16000-17000	Inbound	UDP		Local	SOCKS-UDP relay	
20001-25000	Inbound	TCP	FTP	Local	Active FTP data connection	From FTP server to Web Gateway
520	Bidirectional	UDP	RIP	Your RIP routers	IP routing	
12346	Bidirectional	TCP	Proprietray	Your Web Gateway appliances	Web Gateway cluster communication	
	Bidirectional	IP Protocol 47	GRE	Your Web Gateway appliances and WCCP routers	WCCP and traffic tunneling between Web Gateway cluster nodes	
	Bidirectional	IP Protocol 89	OSPF	Your OSPF routers	IP routing	
	Bidirectional	IP Protocol 112	VRRP	Your Web Gateway appliances	VIP failover	

Port	Direction	Transport protocol	Application protocol	Destination	Use	Note
	Bidirectional	IP Protocol 253	Proprietary	Your Web Gateway appliances	Network-driver cluster communication	
21	Outbound	TCP	FTP	Arbitrary FTP servers	File transfer protocol	Active and passive
25	Outbound	TCP	SMTP	Your email server	Email notifications	
53	Outbound	TCP/UDP	DNS	Your DNS server	Domain name system	
80	Outbound	TCP	HTTP	appliance1.webwasher.com appliance2.webwasher.com	System update	
80, 443	Outbound	TCP	HTTP(S)	Arbitrary HTTP(S) servers	User HTTP(S) traffic	Other ports depending on configuration
80, 443	Outbound	TCP	HTTP(S)	Update servers (tau.mcafee.com, tau-europe.mcafee.com, tau-usa.mcafee.com, tau-usa1.mcafee.com, tau-usa2.mcafee.com, tau-asia.mcafee.com, mwg-update.mcafee.com) CRL download servers, OCSP requests, telemetry	Centralized Updater	
80, 443	Outbound	TCP	HTTP(S)	Your customer-maintained subscribed lists servers	Subscribed Lists Manager	
80, 443	Outbound	TCP	HTTP(S)	Your scheduled-job servers (upload, download)	Scheduled Job Manager	
123	Outbound	TCP/UDP	NTP	Your NTP servers, ntp.webwasher.com	Time synchronization	

Port	Direction	Transport protocol	Application protocol	Destination	Use	Note
162	Outbound	TCP/UDP	SNMP	Your SNMP trap sink	SNMP traps	
389	Outbound	TCP	LDAP	Your directory servers	Directory service and Active Directory	
443	Outbound	TCP	HTTPS	tunnel.web.trustedsource.org (default, can be configured)	GTI cloud lookups (reputation, categories, geolocation, file reputation)	
443	Outbound	TCP	HTTPS	tunnel.web.trustedsource.org (default, can be configured)	GTI telemetry (Malicious URL feedback)	
514	Outbound	TCP/UDP	Syslog	Your syslog servers	Syslog	
636	Outbound	TCP	LDAP	Your directory servers	Secure directory and Active Directory	
1344	Outbound	TCP	ICAP	Your ICAP servers	ICAP	
2020 (Source)	Outbound	TCP	FTP	Local	Active FTP data connection	From Web Gateway to FTP client
8883	Outbound	TCP	DXL	Connection to the DXL broker	Communication between Web Gateway and DXL broker installed on ePO	
9111	Outbound	TCP	HTTP		Pushing logs from Web Gateway to CSR	
9112	Outbound	TCP	HTTPS		Pushing logs from Web Gateway to CSR	
9121	Outbound		FTP		Pushing logs from Web Gateway to CSR	
9200	Outbound	TCP	HTTP(S), FTP	Web Gateway clients	Forwarding web traffic from scanning nodes in a High Availability	

Port	Direction	Transport protocol	Application protocol	Destination	Use	Note
					cluster on Web Gateway	
Your proxy ports	Outbound	TCP	HTTP	Your parent proxies	HTTP proxy	For user traffic and various internal connections (AV update), configured individually

List of actions

The following table provides a list of the actions you can use in rules.

List of actions

Action	Description
Authenticate	Stops processing the rules in the current cycle. Sends an authentication request to the client of the user who requested access to a web object. Continues processing with the next cycle.
Block	Blocks access to a requested web object. Stops processing rules. Continues when the next request is received on the appliance.
Continue	Continues processing with the next rule.
Redirect	Redirects a client that requested access to a web object to another object.
Remove	Removes a requested web object. Stops processing the rules in the current cycle. Continues processing with the next cycle.
Stop Cycle	Stops processing the rules in the current cycle. Does not block access to a requested web object. Continues processing with the next cycle.
Stop Rule Set	Stops processing the rules of the current rule set. Continues processing with the next rule set.

List of block reason IDs

The following table provides a list of block reason IDs with descriptions of their meanings.

You can configure block reason IDs in user message templates to provide a value that identifies a block reason in a log entry.

List of block reason IDs

Block reason ID	Description
0	Allowed
1	Internal error
2	Default message template being used for an action
3	Internal URL filter error
10	Blocked due to an entry in the URL filter database
14	Blocked according to URL filtering by expression
15	Blocked by the Real-Time Classifier
20	Blocked due to lack of content type
22	Blocked due to the media type
30	Blocked due to a multi-part archive having been found
35	Blocked due to an archive not handled by the Archive Handler
80	Blocked due to a virus having been found
81	Blocked due to unauthorized access
82	Blocked due to a bad request
85	Blocked due to an internal anti-malware error
92	Blocked due to expiration of a certificate
93	Blocked due to a revoked certificate
94	Blocked due to a forbidden certificate authority (CA)
95	Blocked due to an unknown certificate authority (CA)
97	Blocked due to a self-signed certificate
98	Blocked due to a common name mismatch
102	Blocked due to an unspecified certificate incident
103	Blocked due to CONNECT not allowed
104	Blocked due to the reverse proxy destination not being allowed
140	Blocked due to an internal DLP filter error
150	Blocked due to an internal Application Control filter error
151	Blocked due to a request belonging to an application that is not allowed
160	Blocked due to missing policy for Web Hybrid

Block reason ID	Description
161	Blocked due to web access not being allowed by Web Hybrid
162	Blocked due to URL filtering by Web Hybrid
200	Blocked due to the coaching session of a user having been exceeded
201	Blocked due to the time quota session of a user having been exceeded
202	Blocked due to the time quota for a user having been exceeded
203	Blocked due to the volume quota session of a user having been exceeded
204	Blocked due to the volume quota for a user having been exceeded
205	Blocked due to the authorized override session of a user having been exceeded
206	Blocked due to the blocking session of a user being active
300	Blocked due to a quota redirect
301	Blocked due to an authentication redirect
400	Blocked due to an authorized override redirect

List of error IDs

The following table provides a list of the error IDs you can use in rules.

The error IDs are grouped in numerical ranges as follows.

10000–10049	Incorrect usage of properties or events
10050–10059	Errors of the rule processing module
10060–10069	Composite Opener errors
10070–10099	Other errors of the rule processing module
10100–10199	General errors
11000–11999	License Manager errors
12000–12999	Errors related to the appliance system
13000–13999	Persistent Database (PDStore) errors
14000–14999	Virus and malware filtering errors
15000–15999	URL filtering errors

16000–16999	ICAP client errors
20000–21000	Proxy module errors
25000–25999	External lists errors
26000–26999	Data loss prevention (DLP) errors
32000–32999	Cloud storage encryption errors
34000–34999	Single sign-on errors
35000–35999	DXL errors

List of error IDs

Error ID	Name	Description
10000	WrongPropParams	$onPosition$: Wrong parameters or types for property $propName$.
10001	UnknownProperty	$onPosition$: Error in rule ‘$ruleName$’: Property dispatcher does not know property $propName$.
10002	NoPropParam	$onPosition$: No parameter for property $propName$ given.
10003	WrongThirdPropParam	$onPosition$: Wrong type of third parameter for property $propName$.
10004	InvalidPropertyParameter	$onPosition$: Parameters for property $propName$ are invalid, reason: $reason$.
10005	InvalidPropertyParameter2	Parameters are invalid. Reason: $reason $.
10005	UnknownProperty2	$onPosition$: Unknown property $propName$.
10007	UnknownFunc	$onPosition$: Unknown function $funcName$. Details: $reason$.
10050	WrongOperator	$onPosition$: Error in rule '$ruleName$': wrong operator '$operator$' used on left hand side type $typeLeft$ and right hand side type $typeRight$.
10051	WrongOperatorNoNames	$onPosition$: $action$ failed. Type of $property$ is $typeName$, but it has to be $formatType$.
10052	FormatError	$onPosition$: User-defined property '$propName$' could not be found. Reason: it was not yet set (not initialized).

Error ID	Name	Description
10053	UserDefinedPropertyNotFound	$onPosition$: User-defined property '$propName$' could not be found. Reason: it was not yet set (not initialized).
10054	PropertyNotFound	$onPosition$: Property '$propName$' could not be found. Reason: it was not yet set (not initialized).
10055	NeedMoreDataOnLastCall	On computing property '$propName$' the filter returned 'NeedMoreData' though there is no more data.
10056	WrongPropState	$onPosition$: State of Property $propName$ is $propState$.
10057	ZombieRuleElemIsExecuted	$rule$ (name: '$name$', id: '$id$') could not be executed because it is a zombie. Reason: '$reason$'.
10058	SetPropertyFailed	$onPosition$: Error in Rule '$ruleName $': Event could not be evaluated. Reason: $reason$.
10059	EventError	$onPosition$: Error while $operation$ the $objName$. Reason: $reason$.
10063	Composite Opener Error	Maximum nested composite level reached.
10064	Composite Opener Error	Maximum compression size limit reached.
10065	Composite Opener Error	Maximum compression ratio limit reached.
10100	ErrorDuringOperation	$onPosition$: Error while $operation$ the $objName$. Reason: $reason$.
10101	InitializeFailed	$onPosition$: Could not initialize/create $objName$. Reason:$reason$.
11000	NoLicense	The requested functionality '$func$' is not covered by your license.
12000	CannotOpenPipe	Cannot open pipe.
12001	CannotOpenFile	Cannot open file '$name$' in mode '$mode$' with errno '$errno$'.
13000	NoUser	No user available.
14000	AVError	Error in AntivirusFilter: $reason$.
14001	AVScanFailedFull	Cannot call McAfee Gateway Anti-Malware engine. All connections in use.

Error ID	Name	Description
14002	AVError	Internal error in Anti-Malware filter. **Note:** As the IDs of error messages are used in the rules for error handling, you need to adapt these rules on your appliance to account for the new error messages and IDs (14003, 14004, 14005) that were introduced with McAfee Web Gateway version 7.3. The library rule set for error handling has been adapted to fit in with the new messages and IDs.
14003	AVError	Timeout occurred while filtering. **Note:** See also the note on error message 14002.
14004	AVError	Cannot filter because a special update is performed. **Note:** See also the note on error message 14002.
14005	AVError	Scanning failed. **Note:** See also the note on error message 14002.
14010	ATDError	Communication failed. Communication to a server that Advanced Threat Defense runs on failed. This can be due to several reasons, including network problems (the server is offline, a request timed out), to an issue with the HTTP protocol, or ton an unexpected or malformed server reply.
14011	ATDError	Timeout occurred while filtering. Advanced Threat Defense took longer to scan a web object than is allowed according to the configured time. The time allowed by default is 10 minutes.
14012	ATDError	File cannot be scanned. Advanced Threat Defense was not able to scan a web object. In the scanning report that is returned by Advanced Threat Defense, the value for Severity is set to *N/A*.
14013	ATDError	Background scan not started in time. Advanced Threat Defense was not started in time for scanning a web object.

Error ID	Name	Description
		This error occurs if the Antimalware.MATD.InitBackgroundScan property is not evaluated before the configured timeout has elapsed. The most likely reason for this evaluation failure error is that the MATD - Handle Offline Scan rule set has been deleted or is disabled or has not been placed in a proper position within the rule sets tree.
14014	ATDError	Invalid parameters in internal request for background scan. An internal request for passing on a web object to Advanced Threat Defense contained invalid parameters. This error occurs if the Antimalware.MATD.IsBackgroundScan property is evaluated and invalid parameters are detected in the internal request. The most likely reason for these invalid parameters to appear is that someone tried to simulate an internal request.
14015	ATDError	Already in background scan. The scanning process was already started for a web object that had been passed on to Advanced Threat Defense when another request for scanning the same object was submitted. This error occurs if the Antimalware.MATD.InitBackgroundScan property is evaluated in the course of processing a scanning request and another request regarding the same object is received at the same time.
15000	TSDatabaseExpired	Global Threat Intelligence system database expired error: Database is expired. '$desc$'.
15001	TSInvalidURL	The URL 'url' is invalid. In function $func$.
15002	TSBinaryNotProperlyLoaded	Binary could not be loaded from '$path $'. In function $func$.
15003	TSCommon	Global Threat Intelligence system error (code: $errorCode$). In function $func$.
15004	TSBinaryDoesNotExist	Global Threat Intelligence system library is not yet available. In function $func$.

Error ID	Name	Description
15005	TSDatabaseNotProperlyLoaded	Database was not properly loaded. In function $func$.
15006	TSNoMem	Global Threat Intelligence system is out of memory. In function $func$.
15007	TSInsufficientSpace	Insufficient space in buffer for Global Threat Intelligence system. In function $func$.
15008	TSNetLookup	Global Threat Intelligence system net error (code: TS_NET_ERROR). In function $func$.
15009	TSCommonNetLookup	Global Threat Intelligence system net error (code: $errorCode$). In function $func$.
15010	TSPipe	Cannot open Global Threat Intelligence system pipe. In function $func$.
16000	NoICAPServerAvailable	No ICAP server available from list: $list$.
16001	NoRespModPropInReqMod	Property $propName$ cannot be calculated in request cycle.
16002	ICAPBadResponse	ICAP client filter error: ICAP server sent bad response.
16003	ICAPMaxConnectionLimit	ICAP client filter error: Maximum number of connections reached.
16004	ICAPCannotConnectToServer	ICAP client filter error: Cannot connect to ICAP server.
16005	ICAPCommunicationFailure	ICAP client filter error: Failure in communication with ICAP server.
16006	ICAPSCertVerifyFailure	SSL certificate verification failure with ICAP server: $server IP$
20000	CheckLongRunningConnection	A timeout occurred on a long-running connection.
20001	CheckSizeOfConnection	The maximum amount of data that can be sent on a long-running connection has been exceeded.
25000	Unknown error happened	An uncategorized error was encountered by the External Lists module.
25001	Error during data fetch	An uncategorized error was encountered by the External Lists module during the data fetch.
25002	Error during data conversion	An error occurred while external list data was converted.

Error ID	Name	Description
25003	Too much data	The configured limit for the number of list entries that can be retrieved from an external source has been exceeded.
25004	Timeout during data fetch	The configured timeout for retrieving external list data has expired.
25005	Data access denied	The rights required for accessing a source of external list data have not been granted to the appliance.
25006	No such resource	A source of external list data, for example, a file or web server, could not be found.
26001	DLP engine not loaded	The DLP engine could not be loaded.
27001	AppRisk database not available	The AppRisk database is not available for filtering web traffic.
32002	Empty password is not allowed	An empty password was submitted, for example, when passwords were retrieved from an external data source.
32003	Invalid configuration for filter	The settings of the module for encryption and decryption are invalid. This error occurs very rarely. It could be caused by a general issue with policy configuration on Web Gateway.
32004	Encryption failed: Unknown content type	Data could not be encrypted because it was of an unknown type. This could be caused by an invalid description for a cloud storage service.
32005	Encryption failed: Parsing of message body failed	The data sent in the body of an upload request is in multi-part/form data format. Parsing this type of data, which is required for encryption, is not supported on Web Gateway.
32006	Encryption failed: Fetching of file name failed	The name of a file containing data that should be encrypted could not be fetched.
32007	Encryption failed: Cipher NNNN is not supported	The cipher that is provided for encrypting data is invalid. This is very unlikely to happen, as the administrator selects the encryption cipher from a pre-configured list.
32008	Encryption failed: Generation of salt failed	The process of salt generation, which is required for encrypting data, could not be performed successfully. This is

Error ID	Name	Description
		usually caused by an internal OpenSSL error.
32009	Encryption failed: Fetching of key failed	The key that is required for encrypting data could not be fetched.
32010	Encryption failed: Initialization of encryption failed	The encryption process could not be initialized.
32011	Encryption failed: Data encryption failed	An error occurred during the encryption process.
32012	Encryption failed: Finalization of decryption failed	The encryption process could not be completed.
32013	Encryption failed: Generic error	Other encryption-related error
32014	Decryption failed: Unknown content type	Data could not be decrypted because it was of an unknown type. This could be caused by an invalid description for a cloud storage service.
32015	Decryption failed: Multi-part message body is not supported	A cloud storage service sent data in the body of its response to a download request that is in multi-part/form data format. Decrypting this type of data is not supported on Web Gateway.
32016	Decryption failed: Cipher NNNN is not supported	The cipher that is provided for decrypting data is invalid. This is very unlikely to happen, as the administrator selects the decryption cipher from a pre-configured list.
32017	Decryption failed: Fetching of key failed	The key that is required for decrypting data could not be fetched.
32018	Decryption failed: Initialization of decryption failed	The decryption process could not be initialized.
32019	Decryption failed: Data decryption failed	An error occurred during the decryption process.
32020	Decryption failed: Finalization of decryption failed	The decryption process could not be completed.
32021	Decryption failed: Generic error	Other decryption-related error
34000	Generic SSO filter error	An error happened during the single sign-on process. Reason: 'General error...'
34001	Generic SSO filter error	A user tried to get single sign-on access using a non-existing cloud connector. Reason: 'No such connector'

Error ID	Name	Description
34003	Generic SSO filter error	No cloud connector was configured for the single sign-on process. Reason: 'There is no connector catalog'
34004	SSO service mismatch error	The value for a token did not match the value that was stored in a cloud connector: Service mismatch. Token ID: '$tokenid$', Service ID: '$serviceid$'
34005	SSO service not enabled	A cloud application was not available for a user: Realm: '$realm$', user: '$userid$', service ID: '$serviceid$'.
34006	SSO non-inline mode error	A cloud application was not available in the non-proxy (non-inline) mode of the single sign-on process: Service ID: '$serviceid$
34050	Credential store generic error	See the error log for details.
34051	Credential store generic error	This request is not allowed for current user.
34052	Credential store generic error	The credential store request could not be created.
34060	Credential store server HTTP error	The credential store server responded to a request with an HTTP error. See the error log for details.
34070	Credential store server error	The credential store server responded with an error. See the error log for details. The log includes the error code returned by the credential store server.
34080	Credential store connection error	A credential store request failed because of a connection error. See the error log for details.
34090	Credential store request error	An internal error occurred while a credential store request was performed. See the error log for details.
35000	DXLNotAvailable	No DXL messages can currently be sent.
37002	Generic application filtering error	A generic error occurred in application filtering. See the error log for details.

List of events

The following table provides a list of the events you can use in rules.

List of events

Name	Description	Parameters
Authentication.AddMethod	Adds an authentication method.	1. String: Name of an authentication method 2. String: Value for an authentication method 3. Boolean: If true, an existing method is overwritten.
Authentication.ClearCache	Clears the cache.	
Authentication.ClearMethodList	Clears the authentication methods list.	
Authentication.ClearNTMLCache	Clears the NTML cache.	
Authentication.GenerateICEResponse	Generates a token that is sent in response to McAfee Cloud Identity Manager to enable seamless authentication.	
Authentication.SendOTP	Sends a one-time password to an authenticated user.	
Bandwidth.FromClient	Limits the speed of data transfer from a client to the appliance.	String: Name of bandwidth class
Bandwidth.FromServer	Limits the speed of data transfer from a web server to the appliance.	String: Name of bandwidth class
Bandwidth.ToClient	Limits the speed of data transfer from the appliance to a client.	String: Name of bandwidth class
Bandwidth.ToServer	Limits the speed of data transfer from the appliance to a web server.	String: Name of bandwidth class
BlockingSession.Activate	Activates a blocking session.	
Body.Insert	Inserts a string into the body of the request or response that is currently processed.	1. Number: Byte position where insertion begins 2. String: Pattern a. string embedded in double quotes (“ ...”, can also contain hex values preceded by \) *or:* b. sequence of hex values
Body.Remove	Removes a number of bytes from the body of the request or response that is currently processed.	1. Number: Byte position where the removal begins 2. Number: Number of bytes to remove

Name	Description	Parameters
Body.Replace	Replaces a portion from the body of the request or response that is currently processed with a string.	1. Number: Byte position where replacement begins 2. String: Pattern a. string embedded in double quotes (“ ...”, can also contain hex values preceded by \) *or:* b. sequence of hex values
Body.ToFile	Writes the body of the request or response that is currently processed to the specified file. The file is stored in the directory */opt/mwg/log/debug/BodyFilterDumps*. The body is written to the file only after it has been completely loaded, even if the *Body.ToFile* event occurred when only one or more chunks of the body had been loaded. To prevent the stored files from filling up the hard disk of an appliance, enable their auto-deletion on the user interface under Configuration → <appliance> → Log File Manager → Advanced.	String: Name of the file that the body is written to
CloudEncryption.Encrypt	Performs the encryption of cloud storage data using the encryption algorithm configured in the settings and the password specified as a parameter of the event. This event can be triggered several times with different settings and passwords, so encryption is also performed several times.	
CloudEncryption.Decrypt	Performs the decryption of data using the decryption algorithm specified in the settings and the password specified as a parameter of the event. This event can be triggered several times with different settings and passwords, so decryption is also performed several times. Order of calls to this event should be the reverse of calls to the encryption event.	
CloudLogging.SetStorageRegion	Sets the storage region for web access data **Note:** This event takes effect only when the hybrid solution is enabled.	

Name	Description	Parameters
Connection.Mark	Sets a connection mark.	Number: Number of a connection
Discard.RuleEngine.Trace	Deletes a rule trace that has been generated by rule tracing on Web Gateway. The event can be used in a suitable rule to discard traces that are filtered according to particular rule criteria. For example, if a trace has been generated for a request that required less than ten seconds processing time, this trace can be considered not worthwhile storing and therefore be discarded. The Timer.TimeInTransaction property can be used in a rule like this to filter rule traces. The rule might be placed in a nested rule set of the Log Handler rule set that takes final position in this nesting rule set. Using the event in this way allows you to perform rule trace storing with a focus on traces that are considered relevant.	
DSCP.Mark.Request	Sets an IP address header field. This field is the *DSCP header* field. Setting this header is also known as *flagging*. The header can be evaluated by network devices supporting DSCP (Differentiated Services Code Point) for directing data packets sent from Web Gateway to a requested web server. Load balancing can, for example, be performed this way. The header can only be set for requests that are sent over an HTTP or HTTPS connection. Setting the header also works for tunneled SSL connections. It can be set here immediately after the CONNECT part of the process has completed. The value that the header is set to can be a number ranging from 0 to 63. **Note:** When using this header in configuring Web Gateway and connected network devices, be sure not to impact existing routes or connections. When multiple requests are sent to a web server over the same connection, a header value that is set at any point within the processing cycle, for example,	Number: Value of the header field

Name	Description	Parameters
	after the CONNECT or CERTVERIFY part of this cycle, will be used for directing the data packets of all following requests. So, when using the header, for example, in a rule for handling streaming media, setting the header inappropriately might lead to directing data packets in a way that throttles the connection.	
DSCP.Mark.Response	Sets an IP address header field. This field is the *DSCP header* field. Setting this header is also known as *flagging*. The header can be evaluated by network devices supporting DSCP (Differentiated Services Code Point) for directing data packets sent back in response from Web Gateway to a client. Load balancing can, for example, be performed this way. The header can only be set for responses that are sent over an HTTP or HTTPS connection. Setting the header also works for tunneled SSL connections. It can be set here immediately after the CONNECT part of the processing cycle has completed. The value that the header is set to can be a number ranging from 0 to 63. **Note:** When using this header in configuring Web Gateway and connected network devices, be sure not to impact existing routes or connections. When multiple responses are sent back to a client over the same connection, a header value that is set at any point within the processing cycle will be used for directing the data packets of all following responses. The same connection is, for example, used when persistent client connections have been configured. Also ACP packets requiring a longer processing time or buffered data packets from previously used connections that still exist in the TCP buffer, might use a header value even if it has been set at a later point in the processing cycle.	Number: Value of the header field

Name	Description	Parameters
DXL.Event	Sends a DXL message with information about a web security topic to the subscribers.	1. String: Topic to send information about 2. String: Information to send about topic
Email.Send	Sends an email.	1. String: Recipient 2. String: Subject 3. String: Body
Enable Cache	Enables the web cache. Using this event, web objects from traffic going on under HTTP or HTTPS can be cached. An event setting can be configured to enable caching for either of the two protocols. Default is HTTP. HTTP2 is not supported. Rules that use this event must specify the protocol that caching is configured for in their criteria. To increase the hit rate, the isssl and X-Forwarded-Proto request headers are ignored. The Accept-Encoding header is also ignored if the requested content can be extracted on Web Gateway. The default cache key is the URL for a web object with the protocol name added. An additional cache key can be configured using the Cache.AdditionalKey property in a rule.	
Enable CompositeOpener	Enables the composite opener.	
Enable Data Trickling	Enables data trickling.	
Enable FTP Upload Progress Indication	Enables the sending of responses to an FTP client, stating that processing of a file that has been sent for uploading to the web is still in progress. This is intended to prevent a timeout on the FTP client when processing on Web Gateway takes more time, for example, due to scanning the file that should be uploaded for infections by viruses and other malware.	
Enable HTML Opener	Enables the HTML opener.	

Name	Description	Parameters
Enable Media Stream Scanner	Enables the Media Stream Scanner, which is provided by the McAfee Gateway Anti-Malware engine.	
Enable Next Hop Proxy	Enables use of next-hop proxies.	
Enable Outbound Source IP Override	Enables the replacement of different outbound source IP addresses by a single IP address.	List of string: List of IP addresses for replacing other IP addresses in string format
Enable Progress Page	Enables display of a progress page.	
Enable RuleEngine Tracing	Enables tracing of the activities that are completed by the rule processing module (rule engine).	
Enable SSL Client Context with CA	Enables sending of client certificates issued by a certificate authority.	
Enable SSL Client Context without CA	Enables sending of client certificates not issued by a certificate authority.	
Enable SSL Scanner	Enables module for SSL scanning.	
Enable SafeSearchEnforcer	Enables the SafeSearchEnforcer.	
Enable Proxy Control	Enables proxy control	
FileSystemLogging.WriteDebugEntry	Writes a debugging entry.	1. String: Debugging entry 2. Boolean: If true, entry is written to stdout.
FileSystemLogging.WriteLogEntry	Writes an entry into a log.	String: Log entry
HTMLElement.InsertAttribute	Inserts an attribute into an HTML element.	1. String: Attribute name 2. String: Attribute value
HTMLElement.RemoveAttribute	Removes an attribute from an HTML element.	String: Attribute name
HTMLElement.SetAttributeValue	Sets an attribute to a value.	1. String: Attribute name 2. String: Value to set attribute to
Header.Add	Adds a header to a request or response.	1. String: Header name 2. String: Header value
Header.AddMultiple	Adds a header with a list of values to a request or response.	1. String: Header name 2. List of string: List of header values

Name	Description	Parameters
Header.Block.Add	Adds a block header to a request or response.	1. String: Header name 2. String: Header value
Header.Block.AddMultiple	Adds a block header with a list of values to a request or response.	1. String: Header name 2. List of string: List of header values
Header.Block.RemoveAll	Removes all block headers with a given name from a request or response.	String: Header name
Header.ICAP.Response.Add	Adds a header to an ICAP response.	1. String: Header name 2. String: Header value
Header.ICAP.Response.AddMultiple	Adds a header with a list of values to an ICAP response.	1. String: Header name 2. List of string: List of header values
Header.ICAP.Response.RemoveAll	Removes all headers with a given name from an ICAP response.	String: Header name
Header.RemoveAll	Removes all headers with a given name from a request or response.	String: Header name
Header.Response.Add	Adds a header to the page generated by a block action.	
HTTP.GenerateResponse	Generates a response to the request made in the request cycle.	String: Response body
HTTP.SetStatus	Sets the HTTP status code at the end of the response cycle.	Number: HTTP status code
ICAP.AddRequestInformation	Adds information to an ICAP request.	1. String: Name of the request 2. String: Added information
MediaType.Header.FixContentType	Replaces a media type header with an appropriate header when it is found after inspection of the media body that the original header does not match the body.	
Notice	Writes an entry with notice level into syslog.	String: Log entry
PDStorage.AddGlobalData.Bool	Adds global variable of type Boolean.	1. String: Variable key 2. Boolean: Variable value
PDStorage.AddGlobalData.Category	Adds global variable of type Category.	1. String: Variable key

Name	Description	Parameters
		2. Category: Variable value
PDStorage.AddGlobalData.Dimension	Adds global variable of type Dimension.	1. String: Variable key 2. Dimension: Variable value
PDStorage. AddGlobalData.Hex	Adds global variable of type Hex.	1. String: Variable key 2. Hex: Variable value
PDStorage. AddGlobalData.IP	Adds global variable of type IP.	1. String: Variable key 2. IP: Variable value
PDStorage.AddGlobalData.IPRange	Adds global variable of type IPRange.	1. String: Variable key 2. IPRange: Variable value
PDStorage.AddGlobalData.List.Category	Adds global variable of type List of Category.	1. String: Variable key 2. List of Category: Variable value
PDStorage. AddGlobalData.List. Dimension	Adds global variable of type List of Dimension.	1. String: Variable key 2. List of Dimension: Variable value
PDStorage.AddGlobalData.List.Hex	Adds global variable of type List of Hex.	1. String: Variable key 2. List of Hex: Variable value
PDStorage. AddGlobalData.List.IP	Adds global variable of type List of IP.	1. String: Variable key 2. List of IP: Variable value
PDStorage. AddGlobalData.List.IPRange	Adds global variable of type List of IPRange.	1. String: Variable key 2. List of IPRange: Variable value
PDStorage.AddGlobalData.List.MediaType	Adds global variable of type List of MediaType.	1. String: Variable key 2. List of MediaType: Variable value
PDStorage. AddGlobalData.List. Number	Adds global variable of type List of Number.	1. String: Variable key 2. List of Number: Variable value
PDStorage. AddGlobalData.List. String	Adds global variable of type List of String.	1. String: Variable key

Name	Description	Parameters
		2. List of String: Variable value
PDStorage. AddGlobalData.List. Wildcard	Adds global variable of type List of Wildcard Expression.	1. String: Variable key 2. List of Wildcard Expression: Variable value
PDStorage. AddGlobalData. MediaType	Adds global variable of type MediaType.	1. String: Variable key 2. MediaType: Variable value
PDStorage. AddGlobalData.Number	Adds global variable of type Number.	1. String: Variable key 2. Number: Variable value
PDStorage. AddGlobalData.String	Adds global variable of type String.	1. String: Variable key 2. String: Variable value
PDStorage. AddGlobalData. Wildcard	Adds global variable of type Wildcard Expression.	1. String: Variable key 2. Wildcard Expression: Variable value
PDStorage. AddUserData.Bool	Adds user variable of type Boolean.	1. String: Variable key 2. Boolean: Variable value
PDStorage. AddUserData.Category	Adds user variable of type Category.	1. String: Variable key 2. Category: Variable value
PDStorage. AddUserData. Dimension	Adds user variable of type Dimension.	1. String: Variable key 2. Dimension: Variable value
PDStorage. AddUserlData.Hex	Adds user variable of type Hex.	1. String: Variable key 2. Hex: Variable value
PDStorage. AddUserData.IP	Adds user variable of type IP.	1. String: Variable key 2. IP: Variable value
PDStorage. AddUserData.IPRange	Adds user variable of type IPRange.	1. String: Variable key 2. IPRange: Variable value
PDStorage. AddUserData.List. Category	Adds user variable of type List of Category.	1. String: Variable key

Name	Description	Parameters
		2. List of Category: Variable value
PDStorage. AddUserData.List. Dimension	Adds user variable of type List of Dimension.	1. String: Variable key 2. List of Dimension: Variable value
PDStorage. AddUserData.List.Hex	Adds user variable of type List of Hex.	1. String: Variable key 2. List of Hex: Variable value
PDStorage. AddUserData.List.IP	Adds user variable of type List of IP.	1. String: Variable key 2. List of IP: Variable value
PDStorage.AddUserData.List.IPRange	Adds user variable of type List of IPRange.	1. String: Variable key 2. List of IPRange: Variable value
PDStorage.AddUserData.List.MediaType	Adds user variable of type List of MediaType.	1. String: Variable key 2. List of MediaType: Variable value
PDStorage.AddUserData.List.Number	Adds user variable of type List of Number.	1. String: Variable key 2. List of Number: Variable value
PDStorage.AddUserData.List.String	Adds user variable of type List of String.	1. String: Variable key 2. List of String: Variable value
PDStorage.AddUserData.List.Wildcard	Adds user variable of type List of Wildcard Expression.	1. String: Variable key 2. List of Wildcard Expression: Variable value
PDStorage.AddUserData.MediaType	Adds user variable of type MediaType.	1. String: Variable key 2. MediaType: Variable value
PDStorage.AddUserData.Number	Adds user variable of type Number.	1. String: Variable key 2. Number: Variable value
PDStorage.AddUserData.String	Adds user variable of type String.	1. String: Variable key 2. String: Variable value
PDStorage.AddUserData.Wildcard	Adds user variable of type Wildcard Expression.	1. String: Variable key

Name	Description	Parameters
		2. Wildcard Expression: Variable value
PDStorage.Cleanup	Cleans up persistently stored data.	
PDStorage. DeleteAllGlobalData	Deletes all permanently stored global data.	
PDStorage. DeleteAllUserData	Deletes all permanently stored user data.	
PDStorage.DeleteGlobalData	Deletes all permanently stored global variables of a given type.	String: Variable key
PDStorage.DeleteUserData	Deletes all permanently stored user variables of a given type.	String: Variable key
ProtocolDetector.ApplyFiltering	Applies processing of web filtering rules on web traffic that has been found to follow a protocol that is supported on Web Gateway.	
SNMP.Send.Trap.Application	Sends an SNMP trap message with application information.	
SNMP.Send.Trap.System	Sends an SNMP trap message with system information.	
SNMP.Send.Trap.User	Sends an SNMP trap message with user information.	1. Number: User ID 2. String: Message body
SNMP.Send.Trap.UserHost	Sends an SNMP trap message with information on the host of a user.	1. Number: User ID 2. String: Message body 3. IP: IP address of the host
SSO.AddCredentials	Creates new credentials for a user who attempts to log on in a single sign-on process to a cloud application. To authenticate a user, the credentials are evaluated by an authentication instance, which is also known as identity provider (IdP), for example, an LDAP or NTLM database. The new credentials are stored in the database of the identity provider.	1. String: Identity provider 2. String: User name 3. String: Cloud application 4. JSON: Credentials in JSON format
SSO.AddServices	Prepares the availability of cloud applications for a user who attempts to select one of them for logon in a single sign-on process. **Note:**	1. String: Identity provider 2. String: User name 3. List: List of cloud applications

Name	Description	Parameters
	A cloud application is also referred to as *cloud service*.	
SSO.DeleteCredentials	Deletes credentials of a user who attempts to logon in a single sign-on process to a cloud application. To authenticate a user the credentials are evaluated by an authentication instance, which is also known as identity provider (IdP), for example, an LDAP or NTLM database. The new credentials are stored in the database of the identity provider.	1. String: Identity provider 2. String: User name 3. String: Cloud application 4. JSON: Credentials in JSON format
SSO.ProcessFormLogin	Processes the data that was submitted for a user in a form on a logon page to perform logon to a cloud application in a single sign-on process. One of the following is executed for the logon form: • When a logon form is sent with a POST request to a cloud application, the password token that had been inserted into the logon form before is replaced by the real password of the user who requests single sign-on access. • When a logon form is requested for a user with a GET request that is sent from a browser, script code is inserted into the form to fill it out and forward it to the cloud application. This event is only executed when the proxy (inline) mode is configured for the single sign-on process.	
SSO.UpdateCredentials	Updates credentials of a user who attempts to log on in a single sign-on process to a cloud application. To authenticate a user, the credentials are evaluated by an authentication instance, which is also known as identity provider (IdP), for example, an LDAP or NTLM database. The new credentials are stored in the database of the identity provider.	1. String: Identity provider 2. String: User name 3. String: Cloud application 4. JSON: Credentials in JSON format
Statistics.Counter.Increment	Increments a counter.	
Statistics.Counter.Reset	Resets a counter.	String: Counter name
Stopwatch.Reset	Sets an internal watch that measures processingtime for rule sets to zero.	String: Rule set name

Name	Description	Parameters
Stopwatch.Start	Starts an internal watch that measures processing time for rule sets.	String: Rule set name
Stopwatch.Stop	Stops an internal watch that measures processing time for rule sets.	String: Rule set name
Syslog	Writes an entry into syslog.	1. Number: Log level ◦ 0 – Emergency ◦ 1 – Alert ◦ 2 – Critical ◦ 3 – Error ◦ 4 – Warning ◦ 5 – Notice ◦ 6 – Info ◦ 7 – Debugging 2. String: Log entry
Throttle.Client	Limits the speed (in Kbps) of data transfer from a client to the appliance.	Number: Speed limit
Throttle.Server	Limits the speed (in Kbps) of data transfer from a web server to the appliance.	Number: Speed limit
TIE: Report File Reputation	Sends a file reputation score to a TIE server.	Number: File reputation score

List of incident IDs

The following table provides a list of the incident IDs you can use in rules.

The incident IDs are grouped in numerical ranges as follows.

1-199	Incidents related to the appliance system
200-299	Core subsystem incidents
300-399	Update module incidents
400-499	Virus and malware filtering incidents
500-599	Log File Manager incidents
600-699	*sysconfd* daemon incidents
700-799	Proxy module incidents
800-899	Virus and malware filtering incidents
900-999	Authentication incidents
1000-1099	URL filtering incidents

1100-1199	Quota management incidents
1200-1299	SSL certificate incidents
1300-1399	ICAP client incidents
1400-1499	Media type filtering incidents
1500-1599	Opener incidents
1600-1699	SSL certificate chain incidents
1700-1799	User interface incidents
1800-1849	External lists incidents
1850-1899	Application filtering incidents
1900-1999	Data Loss Prevention (DLP) incidents
2000-2099	Streaming media filtering incidents
2100-2199	Media type filtering incidents
2200-2299	Dynamic Content Classifier incidents
2300-2399	Single sign-on service incidents
2400-2499	Cloud storage encryption incidents
2500-2549	Credential store incidents
2550-2599	Single Sign On (SSO) incidents
2650-2699	Cloud Access Security Broker (CASB) catalog incidents
2800-2899	Update Server Certificate Authority (CA) incidents
3000-3200	Central Management incidents
3200-3399	Web Hybrid incidents
3400-3499	Web SaaS connector incidents
3500-3599	Protocol Detector incidents

List of incident IDs

Incident ID	Description	Origin number and name	Severity
5	A rule that uses an incident property was executed.	1 System	7
20	RAID monitoring reported critical status or failure of one or more hard disks.	1 Health Monitor	4 (or 3 for hard-disk failure)
21	S.M.A.R.T health check reported an error on an HDD hard disk.	1 Health Monitor	4

Incident ID	Description	Origin number and name	Severity
22	File system usage has exceeded a configured limit.	1 Health Monitor	4
23	Memory usage has exceeded a configured limit.	1 Health monitor	4
24	System load has exceeded a configured limit.	1 Health Monitor	4
26	A check has been executed to detect a BBU RAID error. The checking interval is 30 minutes.	1 Health Monitor	4
200	The license expiration date has been checked.	2 Core	6
201	The appliance has successfully completed all FIPS 140-2 self-tests.	2 Core	6
211	The maximum number of entries in dashboard report x has been exceeded.	2 Statistics	4
298	Update of product x succeeded.	2 Core	6
299	Update of product x failed.	2 Core	3
250	An entry in a list is invalid and will be ignored.	2 Core	3
301	Download of update files was stopped because there is not enough disk space.	3 Updater	3
302	Download of product x failed on node y.	3 Updater	3
303	Update of product x failed on node y.	3 Updater	3
304	Status of product x on node y is up to date.	3 Updater	3
305	The update module could not connect to an update server.	3 Updater	3
321	Download of product x succeeded on node y.	3 Updater	6
322	Download of product x succeeded on node y.	3 Updater	6

Incident ID	Description	Origin number and name	Severity
323	Update of customer subscribed list x succeeded on node y.	3 Customer Subscribed List Manager	6
324	Update of customer subscribed list x failed on nodes y, z, ...	3 Customer Subscribed List Manager	3
325	Status of customer subscribed list x on node y is up to date.	3 Customer Subscribed List Manager	6
326	Download of customer subscribed list x failed on nodes y, z, ...	3 Customer Subscribed List Manager	3
327	Download of McAfee subscribed list x failed on nodes y, z, ...	3 Updater	3
328	Update of McAfee subscribed list x failed on nodes y, z, ...	3 Updater	3
329	Status of McAfee subscribed list x on nodes y, z, ... is up to date.	3 Updater	6
330	Update of McAfee subscribed list x succeeded on node y.	3 Updater	6
331	Processing scheduled job x succeeded	3 Scheduled Job Manager	6
332	Processing scheduled job x failed.	3 Scheduled Job Manager	3
333	Update of updatable system lists failed on node y.	3 Central Updater	3
334	Update of updatable system lists succeeded on node y.	3 Central Updater	6
335	Status of updatable system lists on node y is up to date.	3 Central Updater	6
340-349	Migration failed for different reasons.	3 Migration	6
500	The log manager experienced an unrecoverable internal error and will terminate.	5 Log File Manager	2
501	Log File Manager failed to push log files.	5 Log File Manager	3
600	A yum update contained packages that require a	6 mwg-update	4

Incident ID	Description	Origin number and name	Severity
	restart of the appliance to become effective.		
601	A yum update was successfully completed.	6 mwg-update	5
602	A yum update failed.	6 mwg-update	3
620	A major distribution upgrade was successfully completed.	6 mwg-dist-upgrade	5
621	A major distribution upgrade is in progress. The appliance will restart automatically.	6 mwg-dist-upgrade	4
622	A major distribution upgrade failed. Check the upgrade log file.	6 mwg-dist-upgrade	3
666	A FIPS 140-2 self-test failed on node y. The node is running in non-FIPS mode.	1 FIPS	0
700	The number of concurrent connections has exceeded the configured overload limit. The appliance has entered overload status. Requests sent to the appliance are accepted with delay.	2 Proxy	2
701	The appliance is in overload status for more than 30 seconds. Requests sent to the appliance are accepted with delay.	2 Proxy	2
702	The appliance has left overload status. Requests sent to the appliance are again accepted without delay.	2 Proxy	4
703	The number of concurrent connections has exceeded the configured high-load limit. The appliance has entered high-load status. Requests sent to the appliance are accepted with a delay.	2 Proxy	4
704	The appliance is in high-load status for more than 30 seconds. Requests sent to the appliance are accepted with a delay.	2 Proxy	4

Incident ID	Description	Origin number and name	Severity
705	The number of concurrent connections has dropped below 85 % of the configured high-load limit. The appliance is still in high-load status. Requests sent to the appliance are accepted with a delay.	2 Proxy	6
710	A next-hop proxy server is down and will not be available for n seconds.	2 Proxy	4
711	The appliance could not connect to a next-hop proxy server.	2 Proxy	4
712	A next-hop proxy server has moved back from error status to normal operation.	2 Proxy	6
720	The listener on IP address x, port y could not be opened.	2 Proxy	2
730	A changed proxy mode configuration requires a restart of the appliance.	2 Proxy	2
740	The number of concurrent connections has exceeded the overload limit that is configured for an IFP proxy. Overload status has been entered. New requests are not processed.	2 Proxy	2
741	Overload status lasts more than 30 seconds for an IFP proxy. New requests are not processed.	2 Proxy	2
742	Overload status has been left for an IFP proxy. Requests are again accepted without delay.	2 Proxy	4
743	The number of concurrent connections has exceeded the high-load limit that is configured for an IFP proxy. High-load status has been entered. New requests are not processed.	2 Proxy	4
744	High-load status lasts more than 30 seconds for an IFP	2 Proxy	4

Incident ID	Description	Origin number and name	Severity
	proxy. New requests are not processed.		
745	The number of concurrent connections has dropped below 85 % of the high-load limit that is configured for an IFP proxy. High-load status is still on. Requests are accepted with a delay.	2 Proxy	6
750	A key for the HSM Agent could not be loaded due to an error on the appliance side.	2 Proxy	2
751	A key for the HSM Agent could not be loaded due to an error on the agent side.	2 Proxy	2
752	The ID of a key for an HSM Agent could not be retrieved due to an error on the appliance side.	2 Proxy	2
753	The ID of a key for an HSM Agent could not be retrieved due to an error on the agent side.	2 Proxy	2
760	The WCCP listener could not be started.	2 Proxy	2
761	WCCP could not start send and listerner threads.	2 Proxy	2
762	WCCP could not resolve the router address <host>	2 Proxy	3
763	WCCP could not join the multicast group <host>	2 Proxy	3
764	An error occurred when reading WCCP sockets or writing to them.	2 Proxy	3
765	Authentication with the WCCP router <host> failed.	2 Proxy	3
766	WCCP message parsing failed and malformed packets were created.	2 Proxy	3
767	The WCCP service ID or group could not be found	2 Proxy	3

Incident ID	Description	Origin number and name	Severity
768	A WCCP router for a service ID was added.	2 Proxy	6
769	A WCCP router for a service ID was removed.	2 Proxy	6
850	An update of the MGAM module for virus and malware filtering was successfully completed.	2 Anti-Malware Filter	6
851	An update of the MGAM module for virus and malware filtering failed.	2 Anti-Malware Filter	3
852	Download or verification of the update files for the MGAM module failed.	2 Anti-Malware Filter	3
853	The version of the MGAM module for virus and malware filtering is up to date.	2 Anti-Malware Filter	6
854	An update of the Avira module for virus and malware filtering was successfully completed.	2 Anti-Malware Filter	6
855	An update of the Avira module for virus and malware filtering failed.	2 Anti-Malware Filter	3
856	Download or verification of the update files for the Avira module failed.	2 Anti-Malware Filter	3
857	The version of the Avira module for virus and malware filtering is up to date.	2 Anti-Malware Filter	6
901	The appliance is connected to n servers for NTML authentication in Windows domain x.	2 NTLM Authentication Filter	6
902	The appliance could not connect to n servers for NTML authentication in Windows domain x.	2 NTLM Authentication Filter	4
903	The appliance could not contact Windows domain x for NTLM authentication.	2 NTLM Authentication Filter	3

Incident ID	Description	Origin number and name	Severity
910	The appliance is connected to the LDAP server with configuration ID n.	2 LDAP Authentication Filter	6
912	The appliance was disconnected from the LDAP server with configuration ID n.	2 LDAP Authentication Filter	4
913	The appliance could not connect to any LDAP server with configuration ID n.	2 LDAP Authentication Filter	3
920	A response has been received from RADIUS server x after attempting to start communication to retrieve information for authenticating users.	2 RADIUS Authentication Filter	6
921	A response has again been received from RADIUS server x after communication had been interrupted.	2 RADIUS Authentication Filter	6
923	An authentication request sent to RADIUS server x has led to a timeout.	2 RADIUS Authentication Filter	3
931	The appliance is connected to NTLM-Agent server x.	2 NTLM-Agent Authentication Filter	6
932	The appliance has been disconnected from NTLM-Agent server x.	2 NTLM-Agent Authentication Filter	3
933	The appliance could not connect to NTLM-Agent server x.	2 NTLM-Agent Authentication Filter	3
940	An update of a Certificate Revocation List was successfully completed.	2 Authentication Filter	6
941	An update of a Certificate Revocation List failed.	2 Authentication Filter	4
942	A download of a Certificate Revocation List failed.	2 Authentication Filter	4
943	The status of a Certificate Revocation List is up to date.	2 Authentication Filter	6
1050	An update of the URL Filter module was successfully completed.	2 URL Filter	6

Incident ID	Description	Origin number and name	Severity
1051	An update of the URL Filter module failed.	2 URL Filter	3
1052	Download or verification of update files for the URL Filter module failed.	2 URL Filter	3
1053	Status of the URL Filter module is up to date.	2 URL Filter	6
1650	An updated Certificate Revocation List was downloaded and loaded successfully.	2 Certificate Chain Filter	6
1651	An updated Certificate Revocation List was downloaded, but could not be loaded.	2 Certificate Chain Filter	4
1652	An updated Certificate Revocation List could not be downloaded.	2 Certificate Chain Filter	3
1653	Status of all Certificate Revocation Lists is up to date.	2 Certificate Chain Filter	6
1700	An admin user logged on successfully to the user interface.	7 User interface	4
1701	Logon of an admin user to the user interface failed.	7 User interface	3
1702	The IP address of a client that an end user sent a request from changed.	7 User interface	4
1703	An admin user logged off successfully from the user interface.	7 User interface	6
1704	A logoff from the user interface was forced upon an admin user after a restart of an appliance, a timeout, or a similar incident had occurred.	7 User interface	6
1710	An admin user saved changes successfully.	7 User interface	6
1711	An attempt by an admin user to save changes failed.	7 User interface	3
1800	The number of entries that can be retrieved from an	2 External Lists Filter	4

Incident ID	Description	Origin number and name	Severity
	external list has exceeded the configured limit.		
1801	The amount of data of entries that can be retrieved from an external list has exceeded the configured limit.	2 External Lists Filter	4
1802	An error occurred when data was retrieved from an external list.	2 External Lists Filter	4
1803	An error occurred when data that had been retrieved from an external list was converted.	2 External Lists Filter	4
1804	A time-out occurred when data was retrieved from an external list.	2 External Lists Filter	4
1805	Permission to retrieve data from an external list was denied.	2 External Lists Filter	4
1806	A resource for retrieving external list data could not be found.	2 External Lists Filter	4
1850	An update of the database for application filtering was successfully completed.	2 Application Control	6
1851	An update of the database for application filtering failed.	2 Application Control	3
1852	A download of the database for application filtering failed.	2 Application Control	3
1853	Status of the database for application filtering is up to date.	2 Application Control	6
1854	Loading the database for application filtering failed.	2 Application Control	3
1855	Loading the database for application filtering was successfully completed.	2 Application Control	6
1950	An update of the Data Loss Prevention (DLP) module was successfully completed.	2 Data Loss Prevention	6
1951	An update of the Data Loss Prevention (DLP) module failed.	2 Data Loss Prevention	3

Incident ID	Description	Origin number and name	Severity
1952	Download or verification of the update files for the Data Loss Prevention (DLP) module failed.	2 Data Loss Prevention	3
1953	Status of the Data Loss Prevention (DLP) is up to date.	2 Data Loss Prevention	6
2001	An error occurred with the Stream Detector module.	2 Stream Detector	2
2101	The database for media type filtering could not be loaded.	2 Media Type Filter	2
2200	An update of the Dynamic Content Classifier was successfully completed.	2 Dynamic Content Classifier	6
2201	An update of the Dynamic Content Classifier failed.	2 Dynamic Content Classifier	3
2202	A download or verification of the update files for the Dynamic Content Classifier failed.	2 Dynamic Content Classifier	3
2203	Status of the Dynamic Content Classifier is up to date.	2 Dynamic Content Classifier	6
2350	An update of the files for the single sign-on process was successfully completed.	3 Single Sign On Service	6
2351	An update of the files for the single sign-on process failed.	3 Single Sign On Service	3
2352	A download or verification of the updated files for the single sign-on process failed.	3 Single Sign On Service	3
2353	Status oft he files for the single sign-on process are up to date.	3 Single Sign On Service	
2401	Failed to load services database. This incident is reported when the Cloud Storage Encryption module cannot load files with a description of supported cloud storage services.	3 Cloud Storage Encryption	2
2501	CFM error A CFM error occurred.	2 SSOS Filter	6
2502	MAS export incident	2 SSOS Filter	6

Incident ID	Description	Origin number and name	Severity
	Export of data from the credential store failed.		
2503	MAS store import incident Import of data into the credential store failed.	2 SSOS Filter	6
2550	SSO update success The SSO module was successfully updated.	2 SSOS Filter	6
2551	SSO update failure The SSO module could not successfully be updated. See the errors log for more details.	2 SSOS Filter	3
2552	SSO download failed Files could not successfully be downloaded from the SSO server.	2 SSOS Filter	3
2553	SSO catalog up to date There is no new version of the SSO files on the update server.	2 SSOS Filter	6
2650	CASB catalog update success The CASB connector catalog was successfully updated.	2 SSOS Filter	6
2651	CASB catalog update failure The CASB connector catalog could not successfully be updated. See the errors log for more details.	2 SSOS Filter	3
2652	CASB catalog download failed CASB connector catalog files could not successfully be downloaded from the update server.	2 SSOS Filter	3
2653	CASB catalog up to date There is no new version of the CASB connector catalog files on the update server.	2 SSOS Filter	6
2800	The Update Certificate Authorities (CAs) are up to date.	2 Update CA plugin	6

Incident ID	Description	Origin number and name	Severity
2801	A download of the Update Certificate Authorities (CAs) failed.	2 Update CA plugin	3
2802	The Update Certificate Authorities (CAs) were succesfully updated.	2 Update CA plugin	6
2803	An update of the Update Certificate Authorities (CAs) failed.	2 Update CA plugin	3
3000	At least one node in a Central Management configuration is not in synchronized status (with regard to storage and configuration). The number of unsynchronized nodes changes. This incident is only recorded on the root node.	3 Central Management	3
3001	After incident 3000 occurred, all nodes in a Central Management configuration are again in synchronized status (with regard to storage and configuration).	3 Central Management	6
3005	At least one node in a Central Management configuration did not respond properly after shared data had been sent out. The number of nodes not properly responding changes. This incident is only recorded on the root node and only if the shared data was intended for all nodes.	3 Central Management	3
3006	After incident 3004 occurred, all nodes in a Central Management configuration responded properly again to the sending of shared data.	3 Central Management	6
3200	Sending lists to McAfee Web Gateway Cloud Service was successfully completed.	3 Web Hybrid	6
3201	Sending lists to McAfee Web Gateway Cloud Service failed.	3 Web Hybrid	3

Incident ID	Description	Origin number and name	Severity
3205	Lists were successfully downloaded from McAfee Web Gateway Cloud Service and stored.	3 Web Hybrid	6
3206	Lists could not be downloaded from McAfee Web Gateway Cloud Service and stored.	3 Web Hybrid	3
3210	Synchronization status could not be determined.	3 Web Hybrid	3
3211	An error occurred with the API for McAfee Web Gateway Cloud Service, for example, a mismatch of the API version.	3 Web Hybrid	3
3250	Status of synchronization with McAfee Web Gateway Cloud Service is OK.	3 Web Hybrid	6
3300	The list for Web Service Access is not available for an unknown reason.	2 Web Hybrid	2
3301	The list for Web Service Access does not exist.	2 Web Hybrid	2
3302	The settings for Web Service Access are not available for an unknown reason.	2 Web Hybrid	2
3303	The settings for Web Service Access do not exist.	2 Web Hybrid	2
3400	A policy could not be synchronized to McAfee Web Gateway Cloud Service.	8 SaaS Connector	3
3500	The Protocol Detector rule set could not be found and loaded.	2 Protocol Detector Filter	2
3501	The Protocol Detector rule set was broken or corrupt and could not be loaded.	2 Protocol Detector Filter	2

List of operators

The following table provides a list of the operators that you can use in rules.

The operators are listed in alphabetical order.
The part that precedes the operator in the criteria of a rule is referred to as *property* and the part that follows it as *operand*.

Note: Some properties are of the list type, which means they can have more than one value at the same time.

List of operators

Operator	Description
all in list	All values of the property must be entries in the list of the operand. **Note:** This operator is for use with values of the string type only. Example: URL.Categories<Default> all in list Category Blocklist The criteria matches if, for example, the values of URL.Categories are Entertainment, Media Downloads, and Streaming Media, and all of them are entries in the list Category Blocklist.
at least one in list	One of the values of the property must be an entry in the list of the operand. **Note:** This operator is for use with values of the string type only. Example: URL.Categories<Default> at least one in list Category Blocklist The criteria matches if, for example, one of the values of URL.Categories is Nudity and this is also an entry in the list Category Blocklist.
contains	The value of the operand must be a part of the value of the property. **Note:** This operator is for use with values of the string type only. The string for the operand is submitted by typing it in a suitable field of the user interface. Example: Authentication.UserGroups contains "Domain Users" The criteria matches if the string "Domain Users" can be found in the list of strings that are the values of Authentication.UserGroups.
does not contain	The value of the operand must not be a part of the value of the property. **Note:** This operator is for use with values of the string type only. The string for the operand is submitted by typing it in a suitable field of the user interface. Example: Authentication.UserGroups does not contain "Domain Users" The criteria matches if the string "Domain Users" cannot be found in the list of strings that are the values of Authentication.UserGroups.
does not equal	The value of the property must not be the same as the value of the operand. Example: Antimalware.Infected<Gateway Anti-Malware> does not equal false The criteria matches if the value of Antimalware.Infected is true. Or: Cycle.TopName does not equal "Response"

Operator	Description
	The criteria matches, for example, if the value of Cycle.TopName is "Request". **Note:** Wildcards are not allowed as operands when this operator is used. Even using a blank at the beginning or end of an operand will prevent this operator from working properly.
does not match	The value of the property must not be: • the same as the value of the operand • *or:* covered by the wildcard (regular or glob expression) that is the value of the operand Example: URL.Host does not match *.mcafee.com The criteria matches if the value of URL.Host is, for example, `www.cisco.com`.
does not match in list	The value of the property must not be: • the same as one of the entries in the list of the operand • *or:* covered by one of the wildcards (regular or glob expressions) in the list of the operand Example: URL.Host matches in list URL.Whitelist The criteria matches, for example, if the value of URL.Host is `www.mcafee.com`, and this is not an entry in the list URL.Whitelist. The criteria also matches if the value of URL.Host is `www.mcafee.com` and no regular or glob expression that would cover this value is found in the list URL.Whitelist.
equals	The value of the property must be the same as the value of the operand. Example: Antimalware.Infected<Gateway Anti-Malware> equals true The criteria matches if the value of Antimalware.Infected is true. Or: Cycle.TopName equals "Request" The criteria matches if the value of Cycle.TopName is "Request". **Note:** Wildcards are not allowed as operands when this operator is used. Even using a blank at the beginning or end of an operand will prevent this operator from working properly.
greater than	The value of the property must be above the value of the operand. Example: Body.Size greater than 20000000 The criteria matches if the value of Body.Size is, for example, 20000001 bytes.
greater than or equals	The value of the property must be above or the same as the value of the operand. Example: Body.Size greater than or equals 20000000

Operator	Description
	The criteria matches if the value of Body.Size is, for example, 20000001 or 20000000 bytes.
is in list	The value of the property must be an entry in the list of the operand. **Note:** This operator is for use with values of the string type only. Example: Client.IP is in list Allowed Clients The criteria matches if, for example, the client IP address is 181.153.30.0 and this is an entry in the list Allowed Clients.
is in range list	The value of the property must be within one of the ranges of values that are entries in the list of the operand. **Note:** This operator is for use with values of the string type only. Example: Client.IP is in range list Anti-Malware Quarantine IPRange The criteria matches if, for example, the client IP address is 207.183.100.0 and this value can be found within one of the ranges of values in the list Anti-Malware Quarantine IPRange.
is not in list	The value of the property must not be an entry in the list of the operand. **Note:** This operator is for use with values of the string type only. Example: Client.IP is not in list Allowed Clients The criteria matches if, for example, the client IP address is 174.199.0.0 and this is not an entry in the list Allowed Clients.
is not in range list	The value of the property must not be within one of the ranges of values that are entries in the list of the operand. **Note:** This operator is for use with values of the string type only. Example: Client.IP is not in range list Anti-Malware Quarantine IPRange The criteria matches if, for example, the client IP address is 207.183.100.0 and this value is not found within any of the ranges of values in the list Anti-Malware Quarantine IPRange.
less than	The value of the property must be below the value of the operand. Example: Body.Size less than 20000000 The criteria matches if the value of Body.Size is, for example, 19999999 bytes.
less than or equals	The value of the property must be below or the same as the value of the operand. Example: Body.Size less than or equals 20000000

Operator	Description
	The criteria matches if the value of Body.Size is, for example, 19999999 or 20000000 bytes.
matches	The value of the property must be: • the same as the value of the operand • *or:* covered by the wildcard (regular or glob expression) that is the value of the operand Example: URL.Host matches *.mcafee.com The criteria matches if the value of URL.Host is, for example, `www.mcafee.com`.
matches in list	The value of the property must be: • the same as one of the entries in the list of the operand • *or:* covered by one of the wildcard (regular or glob expressions) in the list of the operand Example: URL.Host matches in URL.Whitelist The criteria matches if the value of URL.Host is, for example, `www.mcafee.com`, and this is an entry in the list URL.Whitelist. The criteria also matches if the value of URL.Host is `www.mcafee.com`, and, for example, `regex(www.mcafee.*)` is an entry in the list URL.Whitelist.
none in list	None of the values of the property must be entries in the list of the operand. **Note:** This operator is for use with values of the string type only. Example: URL.Categories<Default> none in list Category Blocklist The criteria matches if, for example, the values of URL.Categories are Entertainment, Media Downloads, and Streaming Media, and none of them can be found in the list Category Blocklist.

List of properties

The following tables provides a list of the properties you can use in rules.

Order of properties

The properties are listed in alphabetical order. However, the listing takes into consideration the parts of the property names. Name parts begin with a capital letter, in many cases they are also separated by periods.

For example, *Body.HasMimeHeaderParameter* is listed before *Body.Hash*.

Note: There are no properties with names that begin with K, O, V, X, Y, or Z.

SaaS compatibility

Properties that are SaaS-compatible can be used when creating security rules for the web usage of on-premise users as well as of cloud users. Most properties are actually SaaS-compatible, however, some are not, which means they can only be used in rules for on-premise users.

Note: More properties will be made available as SaaS-compatible items in future releases of Web Gateway.

If you use a property that is not SaaS-compatible in a rule that you create on Web Gateway, you are informed on the user interface that you cannot synchronize this rule for use in the cloud.

For a few properties, synchronization can be performed, but when the rules that contain them are executed for use in the cloud, only default values are retrieved for these properties.

These default values are usually meaningless with regard to web security purposes. For example, for the Proxy.Port property, 0 is retrieved as a value instead of a real port number when this property is processed within a rule for use in the cloud.

In the following list, a note is added to the description of a property if it is not SaaS-compatible. If a property can be synchronized together with the rule that contains it, but only a default value is retrieved, this is also indicated.

Properties in context

You can easily find out about the rules and rule sets that use a property.

1. On the user interface, click Search, and under Search for objects referring to, select Property and the property you are interested in.

   The rules that use the property are shown. For example, for *Antimalware.Infected*, the rule Block if virus was found is shown.
2. Select a rule and click Show in context.

   The rule and the property are shown within in its rule set. For example, the rule for *Antimalware.Infected* is shown within the Gateway Anti-Malware rule set.

Properties - A

The following table describes the properties that have names beginning with A.

Properties – A

Name	Type	Description	Parameters
Action.Names	List of String	List with names of the actions that were performed when processing a request, including the response received upon the request	
Antimalware.Avira.VersionString	String	Version of the Avira engine that was used to perform a scanning job	
Antimalware.Infected	Boolean	If true, a web object has been found to be infected.	
Antimalware.Proactive.Probability	Number	Probability that a web object is malware The probability is a percentage, indicated by a number from 1 to 100.	
Antimalware.MATD.Error.MessageDetails	String	Details about an error that occurred when running Advanced Threat Defense (ATD) to interact with Web Gateway in anti-malware filtering.	

Name	Type	Description	Parameters
		Details are shown in a message on the following errors: • Malformed reply • Missing/unexpected mandatory values • No ATD appliances defined • Timeout occurred while scanning • ATD failed to send status report to MWG • Network error • and others	
Antimalware.MATD.GetReport	Boolean	If true, a scanning report exists already for a web object that is to be scanned by Advanced Threat Defense. **Note:** This property is not SaaS-compatible.	
Antimalware.MATD.Hash	String	Hash value used to identify a file that was received from a web server in response to a download request and scanned by Advanced Threat Defense.	
Antimalware.MATD.InitBackgroundScan	Boolean	If true, data for the current transaction is recorded, including data that is related to a request for web access and the response from the web server. The data is recorded in preparation of the scanning that is performed by Advanced Threat Defense when the web object that should be scanned has already been forwarded to the user who requested it. An internal request is also sent to initiate the scanning. If this request is not accepted before the timeout (in seconds) has elapsed that is configured as a parameter of the property, the attempt to let additional scanning be performed by Advanced Threat Defense has failed.	Number: Maximum number of seconds that can elapse before an internal request to initiate scanning is accepted

Name	Type	Description	Parameters
		Note: This property is not SaaS-compatible.	
Antimalware.MATD.IsBackgroundScan	Boolean	If true, the data that was recorded in preparation of the additional scanning is used by Advanced Threat Defense to scan the web object specified by the data. **Note:** This property is not SaaS-compatible.	
Antimalware.MATD.Probability	Number	Severity grade indicating how malicious a web object is on a scale from 1 (low severity grade) to 5 The severity grade is found when an object is scanned by Advanced Threat Defense.	
Antimalware.MATD.Report	String	Report for a web object that was scanned by Advanced Threat Defense. The report is provided in JSON data format.	
Antimalware.MATD.Server	String	Server that Advanced Threat Defense was running on when scanning a web object The server is identified by a URL, for example, *http://matdserver300*.	
Antimalware.MATD.TaskID	String	Identifier for the task that was performed by Advanced Threat Defense when scanning a web object	
Antimalware.MATD.VersionString	String	Version of Advanced Threat Defense that was used to perform a scanning job	
Antimalware.MGAM.VersionString	String	Version of the McAfee Gateway Anti-Malware engine that was used to perform a scanning job	
Antimalware.VersionString	String	Version information referring to all engines for virus and malware filtering that were used by Web Gateway to perform a scanning job	

Name	Type	Description	Parameters
Antimalware.VirusNames	List of String	List with names of the viruses that a web object has been found to be infected with	
AnyText.Language	String	Name of the language that a given text is written in The languages are identified according to ISO-639-1.	String: Text to find language name for
Application.IsHighRisk	Boolean	If true, access to an application is considered to be a high risk for web security.	
Application.IsMediumRisk	Boolean	If true, access to an application is considered to be a medium risk for web security.	
Application.IsMinimalRisk	Boolean	If true, access to an application is considered to be a minimal risk for web security.	
Application.IsUnverified	Boolean	If true, it has not been verified that access to an application is a risk for web security	
Application.Name	Applcontrol	Name of an application	
Application.Reputation	Number	Reputation score for an application	
Application.ToString	String	Name of an application converted into a string	Applcontrol: Application name to convert
Authentication.Authenticate	Boolean	If true, the authentication engine has been called to apply the configured method, for example, NTLM, to the credentials of a user and the user has been authenticated successfully. Values have also been set for the *Authentication.IsAuthenticated* and *Authentication.UserName* properties. If false, it was not possible to apply the configured authentication method successfully, for example, because no credentials or incorrect credentials were submitted.	

Name	Type	Description	Parameters
		Note: This property is not SaaS-compatible.	
Authentication.CacheRemainingTime	Number	Time (in seconds) that remains until authentication credentials are cleared from the cache	
Authentication.Failed	Boolean	If true, credentials were provided by a user, but authentication has failed.	
Authentication.FailureReason.ID	Number	Number identifying the reason why authentication has failed for a user	
Authentication.FailureReason.Message	String	Message text explaining the reason why authentication has failed for a user	
Authentication.GetAzureUserGroups	List of String	List of user groups that the authentication process is applied to, which is retrieved from an Azure AD. **Note:** This property is not SaaS-compatible.	String: User name submitted by Web Gateway when connecting to an Azure AD server
Authentication.GetUserGroups	List of String	List of user groups that the authentication process is applied to **Note:** This property is not SaaS-compatible.	
Authentication.GetUserGroupsJSON	JSON	List of user groups that the authentication process is applied to provided as a JSON object **Note:** This property is not SaaS-compatible.	
Authentication.ICEToken.Attributes	List	List of additional attributes that are retrieved from an ICE token	
Authentication.ICEToken.Audiences	List	List of audiences that are retrieved from an ICE token	
Authentication.ICEToken.Subject	String	Subject that is retrieved from an ICE token	
Authentication.IsAuthenticated	Boolean	If true, a user has been successfully authenticated.	
Authentication.IsLandingOnServer	Boolean	If true, cookie authentication has been applied for a user.	

Name	Type	Description	Parameters
Authentication.IsServerRequest	Boolean	If true, authentication has been requested for a user under the Authentication Server method.	
Authentication.Method	String	Method used for authenticating a user, for example, LDAP	
Authentication.OTP.Context	String	Information required for verifying a one-time password user in encrypted format The property is set to this value when the *Authentication.SendOTP* event is executed. When the rules of the Authentication Server (Time/IP Based Session or Authorized Override with OTP library rule sets are processed, the information is sent in the header of a response under the HTTP protocol. **Note:** This property is not SaaS-compatible.	
Authentication.RawCredentials	String	Credentials of a user in the format originally received on the appliance from a client or other instances of the network Using this property for rule configuration will speed up processing because it saves the time used for converting user credentials to a human readable format, as it is done for the simple *Authentication.UserName* property.	
Authentication.RawUserName	String	Name of a user in the format originally received on the appliance from a client or other instances of the network Using this property for rule configuration will speed up processing because it saves the time used for converting the user name to a human readable format, as it is done	

Name	Type	Description	Parameters
		for the simple *Authentication.UserName* property.	
Authentication.Realm	String	Authentication realm, for example, a Windows domain	
Authentication.SAML.Attributes	List of String	Stores a list of attribute name-value pairs extracted from the <saml2:Attribute> tag in the SAML response. When there are multiple values for one attribute name, the values are separated by commas. **Note:** This property is not SaaS-compatible.	
Authentication.SAML.CreateAuthnRequest			
see above	HTTP POST form	Creates the SAML authentication request which is sent to the external Identity Provider and sets the Authentication.SAML.IDPSSOEndpoint property to the URL of the external Identity Provider. **Note:** This property is not SaaS-compatible.	
Authentication.SAML.Error	String	Describes the error that occurred when the authentication server failed to validate the SAML response. **Note:** Errors messages are provided by the OpenSAML library.	
Authentication.SAML.IDPSSOEndpoint	String	Specifies the SSO URL of the external Identity Provider. If an error occurs, the user is redirected to this URL. **Note:** This property is not SaaS-compatible.	
Authentication.SAML.ParseAuthnResponse			
see above	String	Parses the SAML authentication response that is received from the external Identity Provider. If the response is valid, this property returns a list of attribute name-value pairs in the Authentication.SAML.Attributes	

Name	Type	Description	Parameters
		property. If the response is invalid, this property returns an error in the property Authentication.SAML.Error. **Note:** This property is not SaaS-compatible.	
Authentication.SAML.RelayState	String	Stores the value of the ACS URL at the time that the authentication server creates the SAML authentication request. The authentication server sends the RelayState parameter to the external Identity Provider in the authentication request. The Identity Provider returns the parameter unchanged in the authentication response. The proxy can use the value stored in the RelayState to construct the ACS URL when the external Identity Provider does not support dynamic URLs. **Note:** This property is not SaaS-compatible.	
Authentication.SOCKSKerberosProtectionLevel			
see above	Number	Number representing the protection level that is used when the SOCKS Kerberos authentication method is configured	
Authentication.Token	String	Stores the SAML assertion returned by the external Identity Provider.	
Authentication.UserGroups	List of string	List of user groups that the authentication process is applied to	
Authentication.UserName	String	Name of a user that the authentication process is applied to	

Properties - B

The following table describes the properties that have names beginning with B.

Properties – B

Name	Type	Description	Parameters
Block.ID	Number	ID of an action that blocked a request	
Block.Reason	String	Name of the reason for an action that blocked a request	
BlockingSession.IsBlocked	Boolean	If true, a blocking session has been activated for a user. **Note:** This property is not SaaS-compatible.	
BlockingSession.RemainingSession	Number	Remaining time of a blocking session (in minutes) **Note:** This property is not SaaS-compatible.	
BlockingSession.SessionLength	Number	Time length of a blocking session (in minutes) **Note:** This property is not SaaS-compatible.	
Body.ChangeHeaderMime	Boolean	If true, the header sent in MIME format with the body of a web object has been changed.	
Body.ClassID	String	ID for a class of web objects	
Body.Equals	Boolean	If true, the body of a web object matches the pattern specified by the property parameters.	1. Number: Position of byte where pattern begins 2. String: Pattern a. String embedded in double quotes (“ ...”, can also contain hex values preceded by \) *or:* b. Sequence of hex values
Body.FileName	String	Name of a file that is embedded in the body of a web object, for example, an archived file	
Body.FileReputationBad	Boolean	If true, a web object sent with a request or response as its body, for example, a file, is rated as *bad* by the Global Threat Intelligence (GTI) service.	

Name	Type	Description	Parameters
Body.FileReputationGood	Boolean	If true, a web object sent with a request or response as its body, for example, a file, is rated as *good* by the Global Threat Intelligence (GTI) service.	
Body.FileReputationKnown	Boolean	If true, the Global Threat Intelligence (GTI) service provides a rating for a web object, for example, a file, sent with a request or response as its body.	
Body.FullFileName	String	Name of a file that is embedded in the body of a web object, including also the names of the embedding entities, such as documents or archives Name parts are separated by the | (pipe) symbol, for example, *test.zip|test.doc*.	
Body.HasMimeHeader	Boolean	If true, the body of an extracted multi-part object sent in MIME format has a specified header.	String: Header name
Body.HasMimeHeaderParameter	Boolean	If true, the body of an extracted multi-part object sent in MIME format has a specified header parameter.	1. String: Header name 2. String: Header parameter name
Body.Hash	String	Hash value of the type specified by the property parameter for the body of a web object Hash types can be *md5*, *sha1*, *sha256*, *sha512*, and others.	String: Hash type
Body.HashSHA1	String	Hash value of the SHA1 type for the body of a web object	
Body.IsAboveSizeLimit	Boolean	If true, the body of a web object is above a size limit.	
Body.IsAccessRestrictedObject	Boolean	If true, access restrictions, such as read or write access, are configured for the body of a web object, for example, a PDF file. Password protection is not counted among the methods	

Name	Type	Description	Parameters
		of access restriction with regard to this property. So if the body of a web object is password-protected, the value of the property is set to *false*.	
Body.IsCompleteWithTimeout	Boolean	If true, the body of a web object has been completely sent to the appliance before the time (in milliseconds) specified by the property parameter has elapsed.	Number: Time allowed to send object completely)
Body.IsCorruptedObject	Boolean	If true, an archive contained in the body of a web object is corrupted.	
Body.IsEncryptedObject	Boolean	If true, an archive contained in the body of a web object is encrypted.	
Body.IsMultiPartObject	Boolean	If true, an archive contained in the body of a web object is complex, including multiple parts.	
Body.IsSupportedByOpener	Boolean	If true, an opener device is available on the appliance for the body of a web object that is composite, for example,the body of an archive.	
Body.MimeHeaderParameterValue	String	Value of a header parameter in the body of a web object sent in MIME format	1. String: Header name 2. String: Header parameter value
Body.MimeHeaderValue	String	Value of a header in the body of a web object sent in MIME format	String: Header value
Body.Modified	Boolean	If true, an appliance module has modified the body of a web object.	
Body.NestedArchive Level	Number	Current level of an archive part in an archive	
Body.NotEquals	Boolean	If false, the body of a web object matches the pattern specified by the property parameters.	1. Number: Position of byte where pattern begins 2. String: Pattern a. String embedded in double quotes (" ...", can

Name	Type	Description	Parameters
			also contain hex values preceded by \) *or:* b. Sequence of hex values
Body.NumberOfChildren	Number	Number of objects embedded in the body of a web object	
Body.PositionOfPattern	Number	Position of the byte where the search for a pattern in the body of a web object begins Returns -1 if the pattern is not found.	1. String: Pattern to search for a. String embedded in double quotes (“ ...”, can also contain hex values preceded by \) *or:* b. Sequence of hex values 2. Number: Position of byte where search for pattern begins 3. Number: Search length (in bytes, 0 means search from offset to end of object)
Body.Size	Number	Size of the body of a web object (in bytes)	
Body.Text	String	Text in the body of a web object	
Body.ToNumber	Number	Part of the body of a web object converted into a number (maximum 8 bytes beginning at a specified position) The big-endian or little-endian format can be used for the conversion.	1. Number: Position of byte where converted part begins 2. Number: Length of converted part (in bytes, maximum 8) 0 for the first parameter and the respective value of the *Body.Size* property for the second means the whole body is converted. 3. Boolean: If true, little-endian format is used for conversion,otherwise big-endian
Body.ToString	String	Part of the body of a web object converted into a string	1. Number: Position of byte where converted part begins 2. Number: Length of converted part (in bytes)

Name	Type	Description	Parameters
			0 for the first parameter and the respective value of the *Body.Size* property for the second means the whole body is converted.
Body.UncompressedSize	Number	Size of the body of an archived web object (in bytes) after having been extracted from the archive	
BooleanToString	String	Boolean value converted into a string	Boolean: Boolean value to convert
BytesFromClient	Number	Number of bytes received in a request from a client	
BytesFromServer	Number	Number of bytes received in a response from a web server	
BytesToClient	Number	Number of bytes in a web server response that is forwarded to a client	
BytesToServer	Number	Number of bytes in a client request that is forwarded to a web server	

Properties - C

The following table describes the properties that have names beginning with C.

Properties – C

Name	Type	Description	Parameters
Cache.AdditionalKey	String	Key that can be used in addition to the default key for web caching	
Cache.IsCacheable	Boolean	If true, an object sent in response from a web server can be stored in the web cache.	
Cache.IsFresh	Boolean	If true, an object stored in the web cache has either been downloaded from the web or has been verified.	
Cache.Status	String	Cache status for a web object Values:	

Name	Type	Description	Parameters
		• *TCP_HIT* – A web object was requested by a user and found in the cache. • *TCP_MISS* – A web object was requested by a user and not found in the cache. • *TCP_MISS_RELOAD* – A web object was requested by a user, but was not taken from the cache because the user required it to be fetched directly from the web server in question by clicking the Refresh button. The object was then entered into the cache again. • *TCP_MISS_VERIFY* – A web object was requested by a user and existed in the cache, but verification information from the web server in question showed it was outdated. An updated version of the object was received from the server and entered into the cache.	
Category.ToShortString	String	URL category converted into a string that is the category abbreviation	Category: Category to convert
Category.ToString	String	URL category converted into a string	Category: Category to convert
Client.IM.Login	String	ID used by a client to log on to the appliance under an instant messaging protocol	
Client.IM.ScreenName	String	Screen name of a client communicating with the appliance under an instant messaging protocol	
Client.SystemInfo	String	JSON string providing detailed information about the endpoint The information is relayed by the MCP client using the X-SWPS-SystemInfo header.	
Client.IP	IP	IP address of a client	
Client.NumberOfConnections	Number	Number of connections from a client to the appliance that are open at the same time	

Name	Type	Description	Parameters
Client.OriginalSourceIP	IP	IP address of the connection with your network having one of these values: • IP address of a firewall or other device between your network and the cloud — The property has the same value as Connection.IP. • IP address of the endpoint where the web request originated — The endpoint IP address is available when the deployment is hybrid and Client Proxy or IPsec site-to-site authentication is configured in McAfee WGCS. *Endpoint* is the term used for the client or user computers in your organization that are managed with McAfee ePO or McAfee ePO Cloud. **Note:** You can use this property to write policy rules that apply to particular endpoints in a hybrid deployment. For on-premise deployments, this property retains its default value.	
Client.ProcessName	String	Name of the process that initiated a request sent from a client and redirected to Web Gateway by McAfee Client Proxy.	
CloudEncryption.IsEncryptionSupported	Boolean	If true, encryption can be performed for the data that is uploaded to a cloud storage service with the request that is currently processed. The Cloud Storage Encryption module finds out whether this is true by evaluating service description files for cloud storage services and the settings that are configured on Web Gateway, for example, the Cloud Storage Encryption Support settings, which specify the supported cloud storage services.	

Name	Type	Description	Parameters
CloudEncryption.IsDecryptionSupported	Boolean	If true, decryption can be performed for the data that is downloaded from a cloud storage service with the request that is currently processed. For the method of finding out whether this is true, see the description of the CloudEncryption.IsEncryptionSupported property.	
CloudEncryption.ServiceName	String	Name of the cloud storage service that data is uploaded to or downloaded from with the request that is currently processed. The property is always filled with a value when request are received on Web Gateway for uploading or downloading cloud storage data. However, the property should not be used in rule criteria to trigger an encryption or decryption event upon a match of the criteria. For this purpose, the CloudEncryption.IsEncryptionSupported and CloudEncryption.IsDecryptionSupported properties are provided.	
CloudEncryption.CipherName	String	Name of the algorithm (cipher) used for encrypting or decrypting the cloud storage data that is uploaded or downloaded with the request that is currently processed.	
Command.Categories	List of String	List of categories that a command belongs to, for example, to the FTP command category	
Command.Name	String	Name of a command	
Command.Parameter	String	Parameter of a command	
Connection.Aborted	Boolean	If true, communication on a connection has finally failed and the connection is closed.	

Name	Type	Description	Parameters
Connection.IP	IP	IP address used on a connection	
Connection.IPSec	Boolean	If true, an IPsec VPN tunnel is enabled and configured between McAfee WGCS and an IPsec device on your network. **Note:** You can use this property when writing policy rules for a hybrid deployment. For on-premise deployments, this property retains its default value of *false*.	
Connection.IPSec.Device	String	Name that you assign to the location of the IPsec device on your network in the McAfee WGCS interface **Note:** You can use this property when writing policy rules for a hybrid deployment. For on-premise deployments, this property retains its default value, an empty string.	
Connection.OriginalDestinationIP	IP	IP address of the destination that a request was originally sent to over a given connection The default value is 0. **Note:** This property is not SaaS-compatible. A rule with this property can, however, be synchronized for use in the cloud, but only the default value is then retrieved for this property.	
Connection.Port	Number	Port number of the port that a request sent by a client over a given connection is received on	
Connection.Protocol	String	Protocol used for communication on a connection, for example, HTTP	
Connection.Protocol.IsIM	Boolean	If true, communication on a connection uses an instant messaging protocol.	

Name	Type	Description	Parameters
Connection.Protocol.Parent	String	The embedding protocol for the protocols that are used in communication with the clients when Web Gateway runs as a proxy under the SOCKS protocol. This protocol is the SOCKS protocol, while various protocols can be embedded, for example, HTTP or HTTPS.	
Connection.RunTime	Number	Time (in seconds) that a connection has been running since it was opened until the current second	
Connection.SSL.TransparentCNHandling	Boolean	If true, communication on a connection is SSL-secured and runs in transparent mode.	
Connection.Socketmark	Number	Numerical value, which is the socket mark for the socket of a connected client	
Connection.Variables.GetStringValue	String	Object in string format, which is stored on Web Gateway as long as a given connection to a client persists. The stored string can, for example, be the value of another string-formatted property.	String: Key to identify stored string
Connection.Variables.HasString	Boolean	If true, an object in string format is stored on Web Gateway as long as a given connection to a client persists. The stored string can, for example, be the value of another string-formatted property.	String: Key to identify stored string
Connection.VlanID	Number	VLAN ID of the network that a client uses to communicate with Web Gateway	
Cycle.LastCall	Boolean	If true, processing of data is complete for a cycle.	
Cycle.Name	String	Name of a processing cycle	
Cycle.TopName	String	Name of a cycle (Requests or Responses) that is processed before a web object is	

Name	Type	Description	Parameters
		processed in the Embedded Objects cycle	

Properties - D

The following table describes the properties that have names beginning with D.

Properties – D

Name	Type	Description	Parameters
DataTrickling.Enabled	Boolean	If true, data trickling is used for downloading web objects.	
DateTime.Date.MonthDayNumber	Number	Number of day in month	
DateTime.Date.MonthNumber	Number	Number of month	
DateTime.Date.ToString	String	String representing current date (in the format specified by the property parameters)	String including the following three parts: 1. %YYYY (for the year) *or:* %YY (last two digits) *or:* %Y (last two digits, but only one digit if the last two digits begin with 0, for example, 9 for 2009) 2. %MM (for the month number with 0 inserted before one-digit numbers) *or:* %M (0 is not inserted, for example, 3 for March and 12 for December) 3. %DD (for the day) *or:* %D If no parameter is specified, the format is: %YYYY/%MM /%DD
DateTime.Date.WeekDayNumber	Number	Number of day in week (1 is Sunday)	
DateTime.Date.Year	Number	Year (four digits)	
DateTime.Date.YearTwoDigits	Number	Year (last two digits)	
DateTime.GMTString.FromEpoch	String	String representing current time (in GMT format, converted from number of	Number: Current time in UNIX epoch seconds

Name	Type	Description	Parameters
		UNIX epoch seconds specified by the property parameter) The property can be used with the *DateTime.IsInRangeGMT* property in a rule that checks whether a time range has expired, for example, the time range set for cookie expiration.	

DateTime.IsInRangeGMT	Boolean	If true, the current time is in the range specified by one string in GMT format for the beginning of the range and another for the end.
		The strings can be provided using the *DateTime.GMTString.FromEpoch* property with different values.
		When an irregular time value is specified in a parameter, it is corrected as follows.
		An irregular value for a day, such as Feb 31, is corrected to the regular value that matches it.
		For example:
		`Feb 31 00:00:00 GMT 2018`
		is corrected to:
		`Mar 3 00:00:00 GMT 2018`
		Irregular values for hours and minutes are rejected as invalid.
		Irregular values for seconds are handled as follows.
		• 61 is corrected to 1:01
		• 62 to 69 are rejected as invalid.
		• 70 and higher are corrected by deleting the last digit or digits, so that a regular value is created.
		70 becomes 7 this way, 96 becomes 9, 100 becomes 10, and so on.
		So, for example:
		`Jun 15 00:00:61 GMT 2018`
		is corrected to:
		`Jun 15 00:01:01 GMT 2018`
		And:
		`Jun 15 00:00:96 GMT 2018`
		is corrected to:
			1. String: Date and time in GMT format
			2. String: Date and time in GMT format

Name	Type	Description	Parameters
		`Jun 15 00:00:09 GMT 2018`	

DateTime.IsInRangeISO	Boolean	If true, the current time is in the range specified by one string in ISO format for the beginning of the range and another for the end
		The strings can be provided using the *DateTime.ISOString.FromEpoch* property with different values.
		When an irregular time value is specified in a parameter, is corrected as follows.
		An irregular value for a day, such as Feb 31, is corrected to the regular value that matches it.
		For example:
		`2018-02-31 00:00:00`
		is corrected to:
		`2018-03-03 00:00:00`
		Irregular values for hours and minutes are rejected as invalid.
		Irregular values for seconds are handled as follows.
		• 61 is corrected to 1:01
		• 62 to 69 are rejected as invalid.
		• 70 and higher are corrected by deleting the last digit or digits, so that regular values are created.
		70 becomes 7 this way, 96 becomes 9, 100 becomes 10, and so on.
		So, for example:
		`2018-06-15 00:00:61`
		is corrected to:
		`2018-06-15 00:01:01`
		And:
		`2018-06-15 00:00:96`
		is corrected to:
		`2018-06-15 00:00:09`
			1. String: Date and time in ISO format
			2. String: Date and time in ISO format

DateTime.ISOString.FromEpoch	String	String representing current time (in ISO format, converted from number of UNIX epoch seconds specified by the property parameter) The property can be used with the *DateTime.IsInRangeISO*	Number: Current time in UNIX epoch seconds

Name	Type	Description	Parameters
		property in a rule that checks whether a time range has expired, for example, the time range set for cookie expiration.	
DateTime.Time.Hour	Number	Hour (in 24-hours format, for example, 1 p. m. is 13)	
DateTime.Time.Minute	Number	Minute in hour	
DateTime.Time.Second	Number	Second in minute	
DateTime.Time.ToString	String	String representing current time (in the format specified by the property parameters)	String including the following three parts: 1. %h (for the hour) *or:* %hh (with 0 inserted before a one-digit hour) 2. %m (for the minute) *or:* %mm 3. %s (for the second) *or:* %ss If no parameter is specified, the format is: %hh:%mm:%ss
DateTime.ToGMTString	String	String representing current date and time in Greenwich Mean Time format For example, “Mon, 22 March 2012 11:45:36 GMT”	
DateTime.ToISOString	String	String representating current date and time in ISO format For example, "2012-03-22 11:45:12"	
DateTime.ToNumber	Number	Current time in number of seconds since beginning of 1/1/1970 (UNIX epoch time)	
DateTime.ToString	String	String representing current date and time (in the format specified by the property parameters)	String including the part of the *DateTime.Date.ToString* and *DateTime.Time. ToString* properties If no parameter is specified, the format is: %YYYY/%MM /%DD %hh:%mm:%ss

Name	Type	Description	Parameters
DateTime.ToWebReporterString	String	String representing current date and time in Web Reporter format For example, “29/Oct/2012:14:28:15 +0000”	
DecimalNumber.ToString	String	Decimal number converted to a string The string is truncated according to a parameter. For example, 10.12345 is truncated to 10.12 if this parameter is 2.	1. Number: Decimal number to convert 2. Number: Number of places after the decimal point
Dimension.ToString	String	Dimension converted into a string	Dimension:Dimension to convert
DLP.Classification.AnyText.Matched	Boolean	If true, a given text string is specified as sensitive or inappropriate content by one or more entries in classification lists.	String: Text checked for being sensitive or inappropriate
DLP.Classification.AnyText.MatchedClassifications			
see above	List of String	List of entries in classification lists that specify a given text string as sensitive or inappropriate The list is filled when *DLP.Classification.AnyText.Matched* has been set to *true*.	String: Text checked for being sensitive or inappropriate
DLP.Classification.AnyText.MatchedTerms			
see above	List of String	List of terms including a given text string that is specified as sensitive or inappropriate by one or more entries in classification lists The list is filled when *DLP.Classification.AnyText.Matched* has been set to *true*.	String: Text checked for being sensitive or inappropriate
DLP.Classification.BodyText.Matched	Boolean	If true, the text of a request or response body includes content that is specified as sensitive or inappropriate by one or more entries in classification lists.	
DLP.Classification.BodyText.MatchedClassifications			
see above	List of String	List of entries in classification lists that specify the sensitive	

Name	Type	Description	Parameters
		or inappropriate content found in the body text of requests or responses The list is filled when *DLP.Classification.BodyText.Matched* has been set to *true* .	
DLP.Classification.BodyText.MatchedTerms			
see above	List of String	List of terms in request or response body text that are sensitive or inappropriate content according to one or more entries in classification lists. The list is filled when *DLP.Classification.BodyText.Matched* has been set to *true*.	
DLP.Dictionary.AnyText.Matched	Boolean	If true, a given text string is specified as sensitive or inappropriate content on a dictionary list.	String: Text checked for being sensitive or inappropriate
DLP.Dictionary.AnyText.MatchedTerms			
see above	List of String	List of terms including a given text string that is specified as sensitive or inappropriate on a dictionary list The list is filled when *DLP.Dictionary .AnyText.Matched* has been set to *true*.	String: Text checked for being sensitive or inappropriate
DLP.Dictionary.BodyText.Matched	Boolean	If true, the text of a request or response body includes content that is specified as sensitive or inappropriate by an entry you made in a dictionary list.	
DLP.Dictionary.BodyText.MatchedTerms			
see above	List of String	List of the terms in request or response body text that are sensitive or inappropriate content according to the entries you made in a dictionary list The list is filled when *DLP.Dictionary.BodyText.Matched* has been set to *true*.	
DNS.Lookup	List of IP	List of IP addresses found in a DNS lookup for a host name	String: Host name

Name	Type	Description	Parameters
DNS.Lookup.Reverse	List of String	List of host names found in a reverse DNS lookup for an IP address	IP: IP address
DXL.Query	String	Information retrieved about a topic by sending a DXL query to a service	1. String: Topic that the query is about 2. String: Information about the topic that the query retrieves as response

Properties - E

The following table describes the properties that have names beginning with E.

Properties – E

Name	Type	Description	Parameters
Error.ID	Number	ID of an error	
Error.Message	String	Message text describing an error	
ExtLists.Boolean	Boolean	Boolean value **Note:** This property is not SaaS-compatible.	1. String: Value holding the place of a term that identifies an external list source, for example, in a URL 2. String: as above 3. String: as above
ExtLists.Category	Category	URL category **Note:** This property is not SaaS-compatible.	as above
ExtLists.CategoryList	List of Category	List of URL categories **Note:** This property is not SaaS-compatible.	as above
ExtLists.Double	Double	Double value **Note:** This property is not SaaS-compatible.	as above
ExtLists.DoubleList	List of Double	List of Double values **Note:** This property is not SaaS-compatible.	as above
ExtLists.Integer	Integer	Integer	as above

Name	Type	Description	Parameters
		Note: This property is not SaaS-compatible.	
ExtLists.IntegerList	List of Integer	List of integers **Note:** This property is not SaaS-compatible.	as above
ExtLists.IP	IP	IP address **Note:** This property is not SaaS-compatible.	as above
ExtLists.IPList	List of IP	List of IP addresses **Note:** This property is not SaaS-compatible.	as above
ExtLists.IPRange	IPRange	IP address range **Note:** This property is not SaaS-compatible.	as above
ExtLists.IPRangeList	List of IPRange	List of IP address ranges **Note:** This property is not SaaS-compatible.	as above
ExtLists.JSON	JSON	List of JSON elements **Note:** This property is not SaaS-compatible.	as above
ExtLists.LastUsedListName	String	String representing name of settings for the External Lists module that were used last	
ExtLists.MediaType	MediaType	Media type **Note:** This property is not SaaS-compatible.	as above
ExtLists.MediaTypeList	List of MediaType	List of media types **Note:** This property is not SaaS-compatible.	as above
ExtLists.String	String	String **Note:** This property is not SaaS-compatible.	as above
ExtLists.StringList	List of String	List of strings **Note:** This property is not SaaS-compatible.	as above
ExtLists.StringMap	List of String	List of strings representing map type pairs of keys and values **Note:** This property is not SaaS-compatible.	as above

Name	Type	Description	Parameters
ExtLists.Wildcard	Wildcard Expression	Wildcard (regular) expression **Note:** This property is not SaaS-compatible.	as above
ExtLists.WildcardList	List of Wildcard Expression	List of wildcard (regular) expressions **Note:** This property is not SaaS-compatible.	as above

Properties - F

The following table describes the properties that have names beginning with F.

Properties – F

Name	Type	Description	Parameters
FileSystemLogging.MakeAnonymous	String	String made anonymous by encryption The default values is an empty string. **Note:** This property is not SaaS-compatible. A rule with this property can, however, be synchronized for use in the cloud, but only the default value is then retrieved for this property.	String: String to encrypt

Properties - G

The following table describes the properties that have names beginning with G.

Properties – G

Name	Type	Description	Parameters
GTI.RequestSentToCloud	Boolean	If true, a lookup request for URL category information was sent to the Global Threat Intelligence server.	

Properties - H

The following table describes the properties that have names beginning with H.

Properties – H

Name	Type	Description	Parameters
Header.Block.Exists	Boolean	If true, a header with the specified name is sent with a block page. The header and block page are sent in the response cycle responding to a client request.	String: Header name
Header.Block.ExistsMatching	Boolean	If true, a header with the specified name is sent with a block page and matches a given wildcard expression. The header and block page are sent in the response cycle responding to a client request.	1. String: Header name 2. Wildcard expression
Header.Block.Get	String	First value found for a header with the specified name that is sent with a block page. The header and block page are sent in the response cycle responding to a client request.	String: Header name
Header.Block.GetMatching	String	First value found for a header with the specified name that is sent with a block page and matches a given wildcard expression. The header and block page are sent in the response cycle responding to a client request.	1. String: Header name 2. Wildcard expression
Header.Block.GetMultiple	List of String	List of all values found for a header with the specified name that is sent with a block page. The header and block page are sent in the response cycle responding to a client request.	String: Header name
Header.Block.GetMultipleMatching	List of String	List of all values found for a header with the specified name that is sent with a block page and matches a given wildcard expression. The header and block page are sent in the response cycle responding to a client request.	1. String: Header name 2. Wildcard expression
Header.Exists	Boolean	If true, a specified header is contained in a request or response that is processed on the appliance.	String: Header name

Name	Type	Description	Parameters
		It depends on the current processing cycle whether it is actually a request or response that contains the header.	
Header.Get	String	First value found for the specified header in a request or response that is processed on the appliance It depends on the current processing cycle whether it is actually a request or response that contains the header.	String: Header name
Header.GetMultiple	List of String	List of values found for a specified header in a request or response that is processed on the appliance It depends on the current processing cycle whether it is actually a request or response that contains the header.	String: Header name
Header.ICAP.Request.Exists	Boolean	If true, a specified header is contained in a request sent in ICAP communication. **Note:** This property is not SaaS-compatible.	String: Header name
Header.ICAP.Request.ExistsMatching	Boolean	If true, a specified header is contained in a request sent in ICAP communication and matches a given wildcard expression. **Note:** This property is not SaaS-compatible.	1. String: Header name 2. Wildcard expression
Header.ICAP.Request.Get	String	First value found for a specified header in a request sent in ICAP communication **Note:** This property is not SaaS-compatible.	String: Header name
Header.ICAP.Request.GetMatching	String	First value found for a specified header in a request sent in ICAP communication that also matches a given wildcard expression **Note:** This property is not SaaS-compatible.	1. String: Header name 2. Wildcard expression
Header.ICAP.Response.Exists	Boolean	If true, a specified header is contained in a response	String: Header name

Name	Type	Description	Parameters
		received in ICAP communication. **Note:** This property is not SaaS-compatible.	
Header.ICAP.Response.ExistsMatching	Boolean	If true, a specified header is contained in a response received in ICAP communication and matches a given wildcard expression. **Note:** This property is not SaaS-compatible.	1. String: Header name 2. Wildcard expression
Header.ICAP.Response.Get	String	First value found for a specified header in a response received in ICAP communication **Note:** This property is not SaaS-compatible.	String: Header name
Header.ICAP.Response.GetMatching	String	First value found for a specified header in a response received in ICAP communication that also matches a given wildcard expression **Note:** This property is not SaaS-compatible.	1. String: Header name 2. Wildcard expression
Header.Request.Exists	Boolean	If true, a specified header is contained in a request.	String: Header name
Header.Request.Get	String	First value found for a specified header in a request	String: Header name
Header.Request.GetAll	String	Concatenated string consisting of all original headers in requests received from a client Original values are the values that the headers had when the requests were received on Web Gateway. The headers are concatenated in the order in which they were received. They are separated by \r\n. The value for this property is truncated if its length exceeds 100000 bytes. Example (two concatenated headers):	

Name	Type	Description	Parameters
		Host: www.google.com\r \nUser-Agent: curl/7.55.1\r\n	
Header.Request.GetMultiple	List of String	List of values found for a specified header in a request	String: Header name
Header.Response.Exists	Boolean	If true, a specified header is contained in a response.	String: Header name
Header.Response.Get	String	First value found for a specified header in a response	String: Header name
Header.Response.GetAll	String	Concatenated string consisting of all original headers in responses received from a server Original values are the values that the headers had when the responses were received on Web Gateway. The headers are concatenated in the order in which they were received. They are separated by \r\n. The value for this property is truncated if its length exceeds 100000 bytes. Example (two concatenated headers): Content-Type: text/html \r \nConnection: Keep-Alive\r\n	
Header.Response.GetMultiple	List of String	List of values found for a specified header in a response	String: Header name
Hex.ToString	String	Hex value converted into a string	Hex: Hex value to convert
HTML.Element.Attribute	String	String representing an attribute of an HTML element	
HTML.Element.Dimension	Dimension	Dimension of an HTML element (width and height)	
HTML.Element.HasAttribute	Boolean	If true, an HTML element has a specified attribute.	String: Attribute name
HTML.Element.Name	String	Name of an HTML element	
HTML.Element.ScriptType	String	Script type of an HTML element, for example, JavaScript or Visual Basic Script	

Properties - I

The following table describes the properties that have names beginning with I.

Properties – I

Name	Type	Description	Parameters
ICAP.Policy	String	Name of a policy included in an ICAP request for a URL	
ICAP.ReqMod.ResponseHeader.Exists	Boolean	If true, a response sent from an ICAP server in REQMOD mode contains a specified header. **Note:** This property is not SaaS-compatible.	String: Header name
ICAP.ReqMod.ResponseHeader.ExistsMatching	Boolean	If true, a response sent from an ICAP server in REQMOD mode contains a specified header and matches a given wildcard expression. **Note:** This property is not SaaS-compatible.	1. String: Header name 2. Wildcard expression
ICAP.ReqMod.ResponseHeader.Get	String	First value found for a specified header in a REQMOD response **Note:** This property is not SaaS-compatible.	String: Header name
ICAP.ReqMod.ResponseHeader.GetMatching	String	First value found for a specified header in a REQMOD response that also matches a given wildcard expression **Note:** This property is not SaaS-compatible.	1. String: Header name 2. Wildcard expression
ICAP.ReqMod.ResponseHeader.GetMultiple	List of String	List of values found for a specified header in a REQMOD response **Note:** This property is not SaaS-compatible.	String: Header name
ICAP.ReqMod.ResponseHeader.GetMultipleMatching	List of String	List of values found for a specified header in a REQMOD response that also match a given wildcard expression **Note:** This property is not SaaS-compatible.	1. String: Header name 2. Wildcard expression

Name	Type	Description	Parameters
ICAP.ReqMod.Satisfaction	Boolean	If true, an ICAP server has replaced a request with a response. The ICAP server does this after sending a message that a particular request is blocked. **Note:** This property is not SaaS-compatible.	
ICAP.RespMod.EncapsulatedHTTPChanged	Boolean	If true, an ICAP server has changed the HTTP state for a response sent in RESPMOD mode. **Note:** This property is not SaaS-compatible.	
ICAP.RespMod.ResponseHeaderExists	Boolean	If true, a response sent from an ICAP server in RESPMOD mode contains a specified header. **Note:** This property is not SaaS-compatible.	String: Header name
ICAP.RespMod.ResponseHeaderExistsMatching	Boolean	If true, a response sent from an ICAP server in RESPMOD mode contains a specified header that also matches a given wildcard expression. **Note:** This property is not SaaS-compatible.	1. String: Header name 2. Wildcard expression
ICAP.RespMod.ResponseHeaderGet	String	First value found for a specified header in a RESPMOD response **Note:** This property is not SaaS-compatible.	String: Header name
ICAP.RespMod.ResponseHeaderGetMatching	String	First value found in a RESPMOD response for a specified header that also matches a given wildcard expression **Note:** This property is not SaaS-compatible.	1. String: Header name 2. Wildcard expression
ICAP.RespMod.ResponseHeaderGetMultiple	List of String	List of values found for a specified header in a RESPMOD response **Note:** This property is not SaaS-compatible.	String: Header name

Name	Type	Description	Parameters
ICAP.RespMod.ResponseHeader.GetMultipleMatching	List of String	List of values found in a RESPMOD response for a specified header that also matches a given wildcard expression **Note:** This property is not SaaS-compatible.	1. String: Header name 2. Wildcard expression
IM.Direction	String	Direction of a chat message sent or a file transferred under an instant messaging protocol and processed on the appliance For a chat message sent from a client to the appliance, the direction could, for example, be specified as *out*, for a message sent from a server to the appliance it could be specified as *in*. **Note:** This property is not SaaS-compatible.	
IM.FileName	String	Name of a file transferred under an instant messaging protocol **Note:** This property is not SaaS-compatible.	
IM.FileSize	Number	Size of a file transferred under an instant messaging protocol (in bytes) **Note:** This property is not SaaS-compatible.	
IM.MessageCanSendBack	Boolean	If true, a block message or other message can be sent from the appliance to a user of an instant messaging service. A block message is, for example, sent back to a user who submitted a chat message during a time interval that is not allowed for chatting. A message can typically not be sent before a user has completed the procedure for logging on to the instant messaging service.	

Name	Type	Description	Parameters
		Note: This property is not SaaS-compatible.	
IM.Notification	String	Name of a template used for sending a notification from the appliance to a user of an instant messaging service, for example, a block message **Note:** This property is not SaaS-compatible.	
IM.Recipient	String	Name of a client that receives a chat message or file under an instant messaging protocol This name can also be a group name (group ID) when a chat message is sent to a group of recipients. **Note:** This property is not SaaS-compatible.	
IM.Sender	String	Name of a client that sends a chat message or file under an instant messaging protocol **Note:** This property is not SaaS-compatible.	
Incident.AffectedHost	IP	IP address of a host that is involved in an incident, for example, a web server that the appliance cannot connect to **Note:** This property is not SaaS-compatible.	
Incident.Description	String	Plain-text description of an incident **Note:** This property is not SaaS-compatible.	
Incident.ID	Number	ID of an incident For a list of these IDs, refer to the *List of incident IDs*. **Note:** This property is not SaaS-compatible.	
Incident.Origin	Number	Number specifying the appliance component that is the origin of an incident • 1 – Appliance system • 2 – Core subsystem • 3 – Coordinator subsystem	

Name	Type	Description	Parameters
		• 4 – Anti-Malware process • 5 – Log File Manager • 6 – sysconf daemon • 7 – User interface • 8 – SaaS connector • 9 – Unidentified origin The origin of an incident is further specified by the *Incident.OriginName* property. For the origin of an incident with a particular ID, refer to the *List of incident IDs*. **Note:** This property is not SaaS-compatible.	
Incident.OriginName	String	Name of an appliance component that is the origin of an incident, for example, Core or Log File Manager The name can be that of one of the main components that are listed under *Incident.Origin*. It can also be the name of a subcomponent, which appears together with the *Incident.Origin* number for the related main component. For example, the value of *Incident.OriginName* could be *2 Proxy*. For the origin name of an incident with a particular ID, refer to the *List of incident IDs*. **Note:** This property is not SaaS-compatible.	
Incident.Severity	Number	Severity of an incident Severity levels: • 0 – Emergency • 1 – Alert • 2 – Critical • 3 – Error • 4 – Warning • 5 – Notice • 6 – Informational • 7 – Debug These levels are the same as those used in syslog entries. For the severity level of an incident with a particular ID, refer to the *List of incident IDs*.	

Name	Type	Description	Parameters
		Note: This property is not SaaS-compatible.	
InTheCloud	Boolean	if true, a rule that is currently processed is executed in the cloud	
IP.ToString	String	IP address converted into a string	IP: IP address to convert
IPRange.ToString	String	Range of IP addresses converted into a string	IPRange: Range of IP addresses to convert

Properties - J

The following table describes the properties that have names beginning with J.

Properties – J

Name	Type	Description	Parameters
JSON.ArrayAppend	JSON	JSON array with specified element appended	1. JSON: Array 2. JSON: Element to append
JSON.AsBool	Boolean	Value of specified JSON element returned as Boolean value **Note:** The element value must be a Boolean value.	JSON: Element
JSON.AsNumber	Number	Value of specified JSON element returned as number **Note:** The element value must be a number in Long, Double, or Hexadecimal format.	JSON: Element
JSON.AsString	String	Value of specified JSON element returned as string **Note:** The element value must be a string.	JSON: Element
JSON.CreateArray	JSON	New empty JSON array	
JSON.CreateObject	JSON	New empty JSON object	
JSON.CreateNull	JSON	JSON element value null	
JSON.FromBool	JSON	JSON element value created from Boolean value	Boolean: Boolean value to create JSON element value from

Name	Type	Description	Parameters
JSON.FromNumber	JSON	JSON element value created from number	Number: Number to create JSON element value from
JSON.FromNumberList	String	JSON element value created from number list	List of Number: Number list to create JSON element value from
JSON.FromString	JSON	JSON element value created from string	String: String to create JSON element value from
JSON.FromStringList	JSON	JSON element value created from string list	List of String: String list to create JSON element value from
JSON.GetAt	JSON	JSON element value retrieved from specified position in specified array	1. JSON: Array 2. Number: Position of element
JSON.GetByName	JSON	JSON element identified by key retrieved from specified object	1. JSON: Object 2. String: Element key
JSON.GetType	String	Type of specified JSON element	JSON: Element
JSON.PutAt	JSON	JSON array with element inserted in specified position	1. JSON: Array 2. Number: Position of element 3. JSON: Element
JSON.ReadFromString	JSON	JSON element created from specified string	String: String to create element from
JSON.RemoveAt	JSON	JSON array with element at specified position removed	1. JSON: Array: 2. Number: Position of element
JSON.RemoveByName	JSON	JSON object with element identified by specified key removed	1. JSON: Object 2. String: Element key
JSON.Size	Number	Number of elements in specified JSON object or array	JSON: Object or array
JSON.StoreByName	JSON	JSON object with element value stored under specified key If the object does not exist yet, it is created under the name that is specified for the object.	1. JSON: Object 2. String: Element key 3. JSON: Element value

Name	Type	Description	Parameters
JSON.ToString	String	JSON element value converted into a string **Note:** The element value can be a string or in any of the other data formats for element values.	JSON: Element value to convert

Properties - L

The following table describes the properties that have names beginning with L.

Properties – L

Name	Type	Description	Parameters
License.RemainingDays	Number	Remaining time until a license expires (in days)	
List.LastMatches	String	String containing all elements that have been found to match when two lists are compared using an operator such as *at least one in list* or *all in list* Matches are only added to the list as long it has not yet been decided whether the relationship between the lists that the operator evaluates exists or not. For example, list A contains the elements 1, 2, 3, whereas list B contains 1, 2, 4. Both lists are compared using the *at least one in list* operator. To find out that list A actually contains at least one element of list B, the operator only needs to compare element 1 in both lists and detect that they match. *List.LastMatches* then contains 1 because it has been found to be a match. 2 is also a match in the two lists, but is not contained in *List.LastMatches* because it was not evaluated by the operator and found to be a match.	

Name	Type	Description	Parameters
		It was not evaluated because the operator had already found out after evaluating the 1 in both lists that at least one element of list A was also in list B. If the property *String.BelongsToDomains* has "true" as its value, the string that is specified as its first parameter is set as the value of *List.LastMatches*. This means *List.LastMatches* then provides a string that matched in a list of domain names, being either the name of a domain or a subdomain. The same applies for the property *URL.Host.BelongsToDomains* and *List.LastMatches*.	
List.OfCategory.Append	List of Category	List of URL categories that a category is appended to	1. List of Category: List to append category to 2. Category: Category to append
List.OfCategory.ByName	List of Category	List of URL categories (specified by its name)	String: List name
List.OfCategory.Erase	List of Category	List of URL categories with specified category erased	1. List of Category: List with category to erase 2. Number: Position of category to erase
List.OfCategory.EraseElementRange	List of Category	List of URL categories with specified range of categories erased	1. List of Category: List with categories to erase 2. Number: Position of first category to erase 3. Number: Position of last category to erase
List.OfCategory.EraseList	List of Category	List of URL categories with categories that are also on other list erased	1. List of Category: List with categories to erase 2. List of Category: List of categories to erase on first list

Name	Type	Description	Parameters
List.OfCategory.Find	Number	Position of a URL category on a list	1. List of Category: List with category to find position for 2. Category: Category to find position for
List.OfCategory.Get	Category	URL category specified by its position on a list	1. List of Category: List containing category 2. Number: Position of category on list
List.OfCategory.GetElementRange	List of Category	List of URL categories extracted from other list	1. List of Category: List with categories to extract 2. Number: Position of first category to extract 3. Number: Position of last category to extract
List.OfCategory.Insert	List of Category	List of URL categories with specified category inserted	1. List of Category: List to insert category in 2. Category: Category to insert
List.OfCategory.IsEmpty	Boolean	If true, the specified list is empty.	List of Category: List to check for being empty
List.OfCategory.Join	List of Category	List of URL categories created by joining two lists	1. List of Category: First list to join 2. List of Category: Second list to join
List.OfCategory.Reverse	List of Category	List of URL categories that has its original order reverted	List of Category: List in original order
List.OfCategory.Size	Number	Number of URL categories on a list	List of Category: List to provide number of categories for
List.OfCategory.Sort	List of Category	List of URL categories sorted in alphabetical order	List of Category: List to sort
List.OfCategory.ToShortString	String	List of URL categories converted into a list of their abbreviated name forms	List of Category: List to convert
List.OfCategory.ToString	String	List of URL categories converted into a string	List of Category: List to convert

Name	Type	Description	Parameters
List.OfDimension.Append	List of Dimension	List of dimensions that a dimension is appended to	1. List of Dimension: List to append dimension to 2. Dimension: Dimension to append
List.OfDimension.ByName	List of Dimension	List of dimensions specified by its name	String: List name
List.OfDimension.Erase	List of Dimension	List of dimensions with specified dimension erased	1. List of Dimension: List with dimension to erase 2. Number: Position of dimension to erase
List.OfDimension.EraseElementRange	List of Dimension	List of dimensions with specified range of dimensions erased	1. List of Dimension: List with dimension range to erase 2. Number: Position of first dimension to erase 3. Number: Position of last dimension to erase
List.OfDimension.EraseList	List of Dimension	List of dimensions with dimensions that are also on other list erased	1. List of Dimension: List with dimensions to erase 2. List of Dimension: List of dimensions to erase on first list
List.OfDimension.Find	Number	Position of a dimension on a list	1. List of Dimension: List with dimension to find position for 2. Dimension: Dimension to find position for
List.OfDimension.Get	Dimension	Dimension specified by its position on a list	1. List of Dimension: List containing dimension 2. Number: Position of dimension on list
List.OfDimension.GetElementRange	List of Dimension	List of dimensions extracted from other list	1. List of Dimension: List with dimensions to extract 2. Number: Position of first dimension to extract 3. Number: Position of last dimension to extract

Name	Type	Description	Parameters
			4. Dimension: Dimension to insert
List.OfDimension.Insert	List of Dimension	List of dimensions with specified dimension inserted	1. List of Dimension: List to insert dimension in 2. Dimension: Dimension to insert
List.OfDimension.IsEmpty	Boolean	If true, the specified list is empty.	List of Dimension: List to check for being empty
List.OfDimension.Join	List of Dimension	List of dimensions created by joining two lists	1. List of Dimension: First list to join 2. List of Dimension: Second list to join
List.OfDimension.Reverse	List of Dimension	List of dimensions that has its original order reverted	List of Dimension: List in original order
List.OfDimension.Size	Number	Number of dimensions on a list	List of Dimension: List to provide number of dimensions for
List.OfDimension.Sort	List of Dimension	List of dimensions sorted in alphabetical order	List of Dimension: List to sort
List.OfDimension.ToString	String	List of dimensions converted into a string	List of Dimension: List to convert
List.OfHex.Append	List of Hex	List of hex values that a hex value is appended to	1. List of Hex: List to append Hex value to 2. Hex: Hex value to append
List.OfHex.ByName	List of Hex	List of hex values specified by its name	String: List name
List.OfHex.Erase	List of Hex	List of hex values with specified value erased	1. List of Hex: List with hex value to erase 2. Number: Position of hex value to erase
List.OfHex.EraseElementRange	List of Hex	List of hex values with specified range of values erased	1. List of Hex: List with hex values to erase 2. Number: Position of first hex value to erase 3. Number: Position of last hex value to erase

Name	Type	Description	Parameters
List.OfHex.EraseList	List of Hex	List of hex values with values that are also on other list erased	1. List of Hex: List with hex values to erase 2. List of Hex: List of hex values to erase on first list
List.OfHex.Find	Number	Position of a hex value on a list	1. List of Hex: List with hex value to find position for 2. Hex: Hex value to find position for
List.OfHex.Get	Hex	Hex value specified by its position on a list	1. List of Hex: List containing hex value 2. Number: Position of hex value on list
List.OfHex.GetElementRange	List of Hex	List of hex values extracted from other list	1. List of Hex: List with hex values to extract 2. Number: Position of first hex value to extract 3. Number: Position of last hex value to extract
List.OfHex.Insert	List of Hex	List of hex values with specified value inserted	1. List of Hex: List to insert hex value in 2. Hex: Hex value to insert
List.OfHex.IsEmpty	Boolean	If true, the specified list is empty.	List of Hex: List to check for being empty
List.OfHex.Join	List of Hex	List of hex values created by joining two lists	1. List of Hex: First list to join 2. List of Hex: Second list to join
List.OfHex.Reverse	List of Hex	List of hex values that has its original order reverted	List of Hex: List in original order
List.OfHex.Size	Number	Number of hex values on a list	List of Hex: List to provide number of hex values for
List.OfHex.Sort	List of Hex	List of sorted hex values	List of Hex: List to sort
List.OfHex.ToString	String	List of hex values converted into a string	List of Hex: List to convert

Name	Type	Description	Parameters
List.OfIP.Append	List of IP	List of IP addresses that an IP address is appended to	1. List of IP: List to append IP address to 2. IP: IP address to append
List.OfIP.ByName	List of IP	List of IP addresses (specified by its name)	String: List name
List.OfIP.Erase	List of IP	List of IP addresses with specified address erased	1. List of IP: List with IP address to erase 2. Number: Position of IP address to erase
List.OfIP.EraseElementRange	List of IP	List of IP addresses with specified range of addresses erased	1. List of IP: List with IP addresses to erase 2. Number: Position of first IP address to erase 3. Number: Position of last IP address to erase
List.OfIP.EraseList	List of IP	List of IP addresses with addresses that are also on other list erased	1. List of IP: List with IP addresses to erase 2. List of IP: List of IP addresses to erase on first list
List.OfIP.Find	Number	Position of an IP address on a list	1. List of IP: List with IP address to find position for 2. IP: IP address to find position for
List.OfIP.Get	IP	IP address specified by its position on a list	1. List of IP: List containing IP address 2. Number: Position of IP address on list
List.OfIP.GetElementRange	List of IP	List of IP addresses extracted from another list	1. List of IP: List with IP addresses to extract 2. Number: Position of first IP address to extract 3. Number: Position of last IP address to extract

Name	Type	Description	Parameters
List.OfIP.Insert	List of IP	List of IP addresses with specified address inserted	1. List of IP: List to insert IP address in 2. IP: IP address to insert
List.OfIP.IsEmpty	Boolean	If true, the specified list is empty.	List of IP: List to check for being empty
List.OfIP.Join	List of IP	List of IP addresses created by joining two lists	1. List of IP: First list to join 2. List of IP: Second list to join
List.OfIP.Reverse	List of IP	List of IP addresses that has its original order reverted	List of IP: List in original order
List.OfIP.Size	Number	Number of IP addresses on a list	List of IP: List to provide number of IP addresses for
List.OfIP.Sort	List of IP	List of sorted IP addresses	List of IP: List to sort
List.OfIP.ToString	String	List of IP addresses converted into a string	List of IP: List to convert
List.OfIPRange.Append	List of IPRange	List of IP address ranges that an IP address range is appended to	1. List of IPRange: List to append IP address range to 2. IPRange: IP address range to append
List.OfIPRange.ByName	List of IPRange	List of IP address ranges specified by its name	String: List name
List.OfIPRange.Erase	List of IPRange	List of IP address ranges with specified range erased	1. List of IPRange: List with IP address range to erase 2. Number: Position of IP address range to erase
List.OfIPRange.EraseElementRange	List of IPRange	List of IP address ranges with specified ranges erased	1. List of IPRange: List with IP address ranges to erase 2. Number: Position of first IP address range to erase 3. Number: Position of last IP address range to erase
List.OfIPRange.EraseList	List of IPRange	List of IP address ranges with ranges that are also on other list erased	1. List of IPRange: List with IP address ranges to erase 2. List of IPRange: List of IP address ranges to erase on first list

Name	Type	Description	Parameters
List.OfIPRange.Find	Number	Position of an IP address range on a list	1. List of IPRange: List with IP address range to find position for 2. IPRange: IP address range to find position for
List.OfIPRange.Get	IPRange	IP address range specified by its position on a list	1. List of IPRange: List containing IP address range 2. Number: Position of IP address range on list
List.OfIPRange.GetElementRange	List of IPRange	List of IP address ranges extracted from other list	1. List of IPRange: List with IP address ranges to extract 2. Number: Position of first IP address range to extract 3. Number: Position of last IP address range to extract
List.OfIPRange.Insert	List of IPRange	List of IP address ranges with specified range inserted	1. List of IPRange: List to insert IP address range in 2. IPRange: IP address range to insert
List.OfIPRange.IsEmpty	Boolean	If true, the specified list is empty.	List of IPRange: List to check for being empty
List.OfIPRange.Join	List of IPRange	List of IP address ranges created by joining two lists	1. List of IPRange: First list to join 2. List of IPRange: Second list to join
List.OfIPRange.Reverse	List of IPRange	List of IP address rangess that has its original order reverted	List of IPRange: List in original order
List.OfIPRange.Size	Number	Number of IP address ranges on a list	List of IPRange: List to provide number of IP address ranges for
List.OfIPRange.Sort	List of IPRange	List of sorted IP address ranges	List of IPRange: List to sort
List.OfIPRange.ToString	String	List of IP address ranges converted into a string	List of IPRange: List to convert
List.OfMediaType.Append	List of MediaType	List of media types that a media type is appended to	1. List of MediaType: List to append media type to

Name	Type	Description	Parameters
			2. MediaType: Media type to append
List.OfMediaType.ByName	List of MediaType	List of media types specified by its name	String: List name
List.OfMediaType.Erase	List of MediaType	List of media types with specified type erased	1. List of MediaType: List with media type to erase 2. Number: Position of media type to erase
List.OfMediaType.EraseElementRange	List of MediaType	List of media types with specified range of types erased	1. List of MediaType: List with media types to erase 2. Number: Position of first media type to erase 3. Number: Position of last media type to erase
List.OfMediaType.EraseList	List of MediaType	List of media types with types that are also on other list erased	1. List of MediaType: List with media types to erase 2. List of MediaType: List of media types to erase on first list
List.OfMediaType.Find	Number	Position of a media type on a list	1. List of MediaType: List with media type to find position for 2. MediaType: Media type to find position for
List.OfMediaType.Get	MediaType	Media type specified by its position on a list	1. List of MediaType: List containing media type 2. Number: Position of media type on list
List.OfMediaType.GetElems	List of MediaType	List of media types extracted from other list	1. List of MediaType: List with media types to extract 2. Number: Position of first media type to extract 3. Number: Position of last media type to extract
List.OfMediaType.Insert	List of MediaType	List of media types with specified type inserted	1. List of MediaType: List to insert media type in

Name	Type	Description	Parameters
			2. MediaType: Media type to insert
List.OfMediaType.IsEmpty	Boolean	If true, the specified list is empty.	List of MediaType: List to check for being empty
List.OfMediaType.Join	List of MediaType	List of media types created by joining two lists	1. List of MediaType: First list to join 2. List of MediaType: Second list to join
List.OfMediaType.Reverse	List of MediaType	List of media types that has its original order reverted	List of MediaType: List in original order
List.OfMediaType.Size	Number	Number of media types on a list	List of MediaType: List to provide number of media types for
List.OfMediaType.Sort	List of MediaType	List of media types sorted in alphabetical order	List of MediaType: List to sort
List.OfMediaType.ToString	String	List of media types converted into a string	List of MediaType: List to convert
List.OfNumber.Append	List of Number	List of numbers that a number is appended to	1. List of Number: List to append number to 2. Number: Number to append
List.OfNumber.ByName	List of Number	List of numbers specified by its name	String: List name
List.OfNumber.Erase	List of Number	List of numbers with specified number erased	1. List of Number: List with number to erase 2. Number: Position of number to erase
List.OfNumber.EraseElementRange	List of Number	List of numbers with specified range of numbers erased	1. List of Number: List with numbers to erase 2. Number: Position of first number to erase 3. Number: Position of last number to erase
List.OfNumber.EraseList	List of Number	List of numbers with numbers that are also on other list erased	1. List of Number: List with numbers to erase

Name	Type	Description	Parameters
			2. List of Number: List of numbers to erase on first list
List.OfNumber.Find	Number	Position of a number on a list	1. List of Number: List with number to find position for 2. Number: Number to find position for
List.OfNumber.Get	Number	Number specified by its position on a list	1. List of Number: List containing number 2. Number: Position of number on list
List.OfNumber.GetElementRange	List of Number	List of numbers extracted from other list	1. List of Number: List with numbers to extract 2. Number: Position of first number to extract 3. Number: Position of last number to extract
List.OfNumber.Insert	List of Number	List of numbers with specified number inserted	1. List of Number: List to insert number in 2. Number: Number to insert
List.OfNumber.IsEmpty	Boolean	If true, the specified list is empty.	List of Number: List to check for being empty
List.OfNumber.Join	List of Number	List of numbers created by joining two lists	1. List of Number: First list to join 2. List of Number: Second list to join
List.OfNumber.Reverse	List of Number	List of numbers that has its original order reverted	List of Number: List in original order
List.OfNumber.Size	Number	Number of numbers on a list	List of Number: List to provide number of numbers for
List.OfNumber.Sort	List of Number	List of sorted numbers	List of Number: List to sort
List.OfNumber.ToString	String	List of numbers converted into a string	List of Number: List to convert
List.OfSSOConnectors.Append	List of SSOConnector	List of cloud connectors with specified cloud connector appended	1. List of SSOConnec: List to append cloud connector to

Name	Type	Description	Parameters
			2. SSO Connector: Cloud connector to append
List.OfSSOConnectors.ByName	List of SSOConnector	List of cloud connectors specified by its name	String: List name
List.OfSSOConnectors.Erase	List of SSOConnector	List of cloud connectors with specified connector erased	1. List of SSOConnector: List with cloud connector to erase 2. Number: Position of cloud connector to erase
List.OfSSOConnectors.EraseElementRange	List of SSOConnector	List of cloud connectors with specified range of connectors erased	1. List of SSOConnector: List with range of cloud connectors to erase 2. Number: Position of first cloud connector to erase 3. Number: Position of last cloud connector to erase
List.OfSSOConnectors.EraseList	List of SSOConnector	List of cloud connectors with connectors that are also on other list erased	1. List of SSOConnector: List with cloud connectors to erase 2. List of SSOConnector: List of cloud connectors to erase on first list
List.OfSSOConnectors.Exists	Boolean	If true, the list of cloud connectors with the specified name exists.	String: List name
List.OfSSOConnectors.Find	Number	Position of cloud connector in a list	1. List of SSOConnector: List containing cloud connector 2. SSOConnector: Cloud connector to find position for
List.OfSSOConnectors.Get	SSOConnector	Cloud connector specified by its position on a list	1. List of SSOConnector: List containing cloud connector 2. Number: Position of cloud connector on list
List.OfSSOConnectors.GetElementRange	List of SSOConnector	List of cloud connectors extracted from other list	1. List of SSOConnector: List with cloud connectors to extract

Name	Type	Description	Parameters
			2. Number: Position of first cloud connector to extract 3. Number: Position of last cloud connector to extract
List.OfSSOConnectors.Insert	List of SSOConnector	List of cloud connectors with specified connector inserted	1. List of SSO Connect or: List to insert cloud connecto in 2. SSO Connector: Cloud connector to insert 3. Number: Position to insert cloud connector in
List.OfSSOConnectors.IsEmpty	Boolean	If true, the specified list is empty.	List of SSOConnector: List to check for being empty
List.OfSSOConnectors.Join	List of SSOConnector	List of single sign-on connectors created by joining two lists	1. List of SSOConnector: First list to join 2. List of SSOConnector: Second list to join
List.OfSSOConnectors.Reverse	List of SSOConnector	List of cloud connectors that has its original order reverted	List of SSOConnector: List in original order
List.OfSSOConnectors.Set	List of SSOConnector	List of cloud connectors with specified connector set	1. List of SSOConnector: List to set cloud connector on 2. SSOConnector: Cloud connector to set 3. Number: Position to set cloud connector on
List.OfSSOConnectors.Size	Number	Number of cloud connectors on a list	List of SSOConnector: List to provide number of cloud connectors for
List.OfSSOConnectors.Sort	List of SSOConnector	List of cloud connectors sorted in alphabetical order of names	List of SSOConnector: List to sort
List.OfSSOConnectors.ToString	String	List of cloud connectors converted into a string	List of SSOConnector: List to convert
List.OfString.Append	List of String	List of strings that a string is appended to	1. List of String: List to append string to 2. String: String to append
List.OfString.ByName	List of String	List of strings specified by its name	String: List name

Name	Type	Description	Parameters
List.OfString.Erase	List of String	List of strings with specified string erased	1. List of String: List with string to erase 2. Number: Position of string to erase
List.OfString.EraseElementRange	List of String	List of strings with specified range of strings erased	1. List of String: List with strings to erase 2. Number: Position of first string to erase 3. Number: Position of last string to erase
List.OfString.EraseList	List of String	List of strings with strings that are also on other list erased	1. List of String: List with strings to erase 2. List of String: List of strings to erase on first list
List.OfString.Find	Number	Position of a string on a list	1. List of String: List with string to find position for 2. String: String to find position for
List.OfString.Get	String	String specified by its position on a list	1. List of String: List containing string 2. Number: Position of string on list
List.OfString.GetElementRange	List of String	List of strings extracted from other list	1. List of String: List with strings to extract 2. Number: Position of first string to extract 3. Number: Position of last string to extract
List.OfString.Insert	List of String	List of strings with specified string inserted	1. List of String: List to insert string in 2. String: String to insert
List.OfString.IsEmpty	Boolean	If true, the specified list is empty.	List of String: List to check for being empty
List.OfString.Join	List of String	List of strings created by joining two lists	1. List of String: First list to join

Name	Type	Description	Parameters
			2. List of String: Second list to join
List.OfString.JSON.AsStringList	List of String	List of strings created from the element values of a JSON array If a value is null, an empty string is created.	JSON: Array
List.OfStringMapInList	List of String	String specified by a parameter and contained in a list with an index for the position this string has in another list If the specified string is not contained in the first list or does not exist as a position in the second list, the string is empty.	1. List of String: First list containing string 2. List of String: Second list containing string 3. String: String contained in first and second list or empty string
List.OfString.Reverse	List of String	List of strings that has its original order reverted	List of String: List in original order
List.OfString.Size	Number	Number of strings on a specified list	List of String: List to provide number of strings for
List.OfString.Sort	List of String	List of strings sorted in alphabetical order	List of String: List to sort
List.OfString.ToString	String	List of strings converted into a string	List of String: List to convert
List.OfWildcard.Append	List of Wildcard Expression	List of wildcard expressions that an expression is appended to	1. List of Wildcard Expression: List to append wildcard expression to 2. Wildcard Expression: Wildcard expression to append
List.OfWildcard.ByName	List of Wildcard Expression	List of wildcard expressions specified by its name	String: List name
List.OfWildcard.Erase	List of Wildcard Expression	List of wildcard expressions with specified expression erased	1. List of Wildcard Expression: List with wildcard expression to erase 2. Number: Position of wildcard expression to erase

Name	Type	Description	Parameters
List.OfWildcard.EraseElementRange	List of Wildcard Expression	List of wildcard expressions with specified range of expressions erased	1. List of Wildcard Expression: List with wildcard expressions to erase 2. Number: Position of first wildcard expression to erase 3. Number: Position of last wildcard expression to erase
List.OfWildcard.EraseList	List of Wildcard Expression	List of wildcard expressions with expressions that are also on other list erased	1. List of Wildcard Expression: List with wildcard expressions to erase 2. List of Wildcard Expression: List of wildcard expressions to erase on first list
List.OfWildcard.Find	Number	Position of a wildcard expression on a list	1. List of Wildcard expression: List with wildcard expression to find position for 2. Wildcard expression: Wildcard expression to find position for
List.OfWildcard.Get	Wildcard Expression	Wildcard expression specified by its position on a list	1. List of Wildcard Expression: List containing wildcard expression 2. Number: Position of wildcard expression on list
List.OfWildcard.GetElementRange	List of Wildcard Expression	List of wildcard expressions extracted from other list	1. List of Wildcard Expression: List with wildcard expressions to extract 2. Number: Position of first wildcard expression to extract 3. Number: Position of last wildcard expression to extract
List.OfWildcard.Insert	List of Wildcard Expression	List of wildcard expressions with specified expression inserted	1. List of Wildcard Expression: List to insert wildcard expression in

Name	Type	Description	Parameters
			2. Wildcard Expression: Wildcard expression to insert
List.OfWildcard.IsEmpty	Boolean	If true, the specified list is empty.	List of Wildcard Expression: List to check for being empty
List.OfWildcard.Join	List of Wildcard Expression	List of wildcard expressions created by joining two lists	1. List of Wildcard Expression: First list to join 2. List of Wildcard Expression: Second list to join
List.OfWildcard.Reverse	List of Wildcard Expression	List of wildcard expressions that has its original order reverted	List of Wildcard Expression: List in original order
List.OfWildcard.Size	Number	Number of wildcard expressions on a list	List of Wildcard Expression: List to provide number of wildcard expressions for
List.OfWildcard.Sort	List of Wildcard Expression	List of sorted wildcard expressions	List of Wildcard Expression: List to sort
List.OfWildcard.ToString	String	List of wildcard expressions converted into a string -	List of Wildcard Expression: List to convert
Location.Name	String	Name of a location or other source that requests for web access are sent from Names of locations where users reside or names of users or user groups are usually set as values of this property. As no value is by default provided for this property, you must set its value using a rule to work with it. For example, if you know that a particular range of IP addresses has been allotted to an office of your organization, you can create this rule: Client.IP is in range 10.140.226.173-10.140.226.183 —> Continue — Set Location.Name ="Downtown Office" Once the property has been set to a value, you can use it, for example, in logging or blocking rules.	

Name	Type	Description	Parameters
		Web Gateway and McAfee WGCS share this property, which is named **Location** in the McAfee WGCS interface. For cloud-only deployments, **Location** retains its default value, an empty string. For hybrid deployments, **Location** has the same value as **Location.Name**.	

Properties - M

The following table describes the properties that have names beginning with M.

Properties – M

Name	Type	Description	Parametersss
Map.ByName	List of MapType	Already existing Map Type list that has the specified name	String: List name
Map.CreateStringMap	List of MapType	Newly created Map Type list The list is still empty.	
Map.DeleteKey	List of MapType	Map Type list, in which the specified key and the related value are deleted	1. List of MapType: Map Type list 2. String: Key
Map.GetKeys	List of MapTYpe	List of keys that are contained in the specified Map Type list	List of MapType: Map Type list
Map.GetStringValue	String	String that is the value for the specified key in the specified Map Type list	1. List of MapType: Map Type list 2. String: Key
Map.HasKey	Boolean	If true, the specified key exists in the specified Map Type list	1. List of MapType: Map Type list 2. String: Key
Map.SetStringValue	List of MapType	Map Type list, in which the specified value is set for the specified key	1. List of MapType: Map Type list 2. String: Key 3. String: Value
Map.Size	Number	Number of key-value pairs in the specified Map Type list	List of MapType: Map Type list

Name	Type	Description	Parametersss
Map.ToString	String	Map Type list converted into a string	List of MapType: Map Type list
Math.Abs	Number	Absolute value of specified number	Number: Number that absolute value is provided for
Math.Modulo	Number	Integer that is the remainder after dividing integer *a* by integer *b* when only an integer is accepted as the resulting quotient. For example, if a = 14 and b = 3, the value of *Math.Modulo* is 2. The integer that is the result of dividing 14 by 3 is 4 and, since 3 x 4 = 12, this leaves 2 as the remainder.	1. Number: Value for *a* 2. Number: Value for *b*
Math.Random	Number	Random number between specified minimum and maximum values (including these values)	1. Number: Minimum value 2. Number: Maximum value
MediaStreamProbability	Number	Probability that the streaming media in question matches the found media type (in percent)	
MediaType.EnsuredTypes	List of MediaType	List of media types that are ensured for the respective media with a probability of more than 50%	
MediaType.FromFileExtension	List of MediaType	List of media types that are found using the file name extension of the media	
MediaType.FromHeader	List of MediaType	List of media types that are found using the content-type header sent with the media	
MediaType.HasOpener	Boolean	If true, an opener module is available on the appliance for media of a given type.	
MediaType.IsArchive	Boolean	If true, the media that is being processed is an archive.	
MediaType.IsAudio	Boolean	If true, the media that is being processed is of the audio type.	
MediaType.IsCompositeObject	Boolean	If true, the media that is being processed is a composite object.	

Name	Type	Description	Parametersss
MediaType.IsDatabase	Boolean	If true, the media that is being processed is a database.	
MediaType.IsDocument	Boolean	If true, the media that is being processed is a document.	
MediaType.IsExecutable	Boolean	If true, the media that is being processed is an executable file.	
MediaType.IsImage	Boolean	If true, the media that is being processed is an image.	
MediaType.IsText	Boolean	If true, the media that is being processed is of the text type.	
MediaType.IsVideo	Boolean	If true, the media that is being processed is of the video type.	
MediaType.MagicBytesMismatch	Boolean	If true, the media type specified in the header sent with the media does not match the type that was found on the appliance by examining the magic bytes actually contained in the media.	
MediaType.NotEnsuredTypes	List of MediaType	List of media types that are ensured for the respective media with a probability of less than 50%	
MediaType.ToString	String	Media type converted into a string	MediaType: Media type to convert
Message.Language	String	Name of language for messages sent to users in short form, for example, en, de, ja	
Message.TemplateName	String	Name of a template for messages sent to users	

Properties - N

The following table describes the properties that have names beginning with N.

Properties – N

Name	Type	Description	Parameters
NextHopProxy.StickinessAttribute	String	Part of a request that qualifies it for being handled in next-hop proxy stickiness mode **Note:** This property is not SaaS-compatible.	
Number.ToDecimalNumber	Number	Integer converted into decimal format For example, 10 is converted to 10.0.	Number: Integer to convert
Number.ToString	String	Number converted into a string	Number: Number to convert
Number.ToVolumeString	String	Number of bytes that a volume amounts to converted into a string	Number: Number of bytes to convert
NumberOfClientConnections	Number	Number of connections to clients that are open on an appliance at the same time **Note:** This property is not SaaS-compatible.	

Properties - P

The following table describes the properties that have names beginning with P.

Properties – P

Name	Type	Description	Parameters
PDStorage.GetAllData	List of String	List containing all persistently stored data in string format **Note:** This property is not SaaS-compatible.	
PDStorage.GetAllGlobalData	List of String	List containing all persistently stored global data in string format **Note:** This property is not SaaS-compatible.	
PDStorage.GetAllUserData	List of String	List containing all persistently stored user data in string format **Note:** This property is not SaaS-compatible.	

Name	Type	Description	Parameters
PDStorage.GetGlobalData.Bool	Boolean	Global variable of type Boolean **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.Category	Category	Global variable of type Category **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.Dimension	Dimension	Global variable of type Dimension **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.Hex	Hex	Global variable of type Hex **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.IP	IP	Global variable of type IP **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.IPRange	IPRange	Global variable of type IPRange **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.List.Category	List of Category	Global variable of type List of Category **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.List.Dimension	List of Dimension	Global variable of type List of Dimension **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.List.Hex	List of Hex	Global variable of type List of Hex **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.List.IP	List of IP	Global variable of type List of IP **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.List.IPRange	List of IPRange	Global variable of type List of IPRange	String: Variable key

Name	Type	Description	Parameters
		Note: This property is not SaaS-compatible.	
PDStorage.GetGlobalData.List.MediaType	List of MediaType	Global variable of type List of MediaType **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.List.Number	List of Number	Global variable of type List of Number **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.List.String	List of String	Global variable of type List of String **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.List.WildcardExpression	List of WildcardExpression	Global variable of type List of WildcardExpression **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.MediaType	MediaType	Global variable of type MediaType **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.Number	Number	Global variable of type Number **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.String	String	Global variable of type String **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetGlobalData.WildcardExpression	WildcardExpression	Global variable of type WildcardExpression **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.Bool	Boolean	User variable of type Boolean **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.Category	Category	User variable of type Category **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.Dimension	Dimension	User variable of type Dimension	String: Variable key

Name	Type	Description	Parameters
		Note: This property is not SaaS-compatible.	
PDStorage.GetUserData.Hex	Hex	User variable of type Hex **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.IP	IP	User variable of type IP **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.IPRange	IPRange	User variable of type IPRange **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.List.Category	List of Category	User variable of type List of Category **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.List.Dimension	List of Dimension	User variable of type List of Dimension **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.List.Hex	List of Hex	User variable of type List of Hex **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.List.IP	List of IP	User variable of type List of IP **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.List.IPRange	List of IPRange	User variable of type List of IPRange **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.List.MediaType	List of MediaType	User variable of type List of MediaType **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.List.Number	List of Number	User variable of type List of Number **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.List.String	List of String	User variable of type List of String	String: Variable key

Name	Type	Description	Parameters
		Note: This property is not SaaS-compatible.	
PDStorage.GetUserData.List.WildcardExpression	List of WildcardExpression	User variable of type List of WildcardExpression **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.MediaType	MediaType	User variable of type MediaType **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.Number	Number	User variable of type Number **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.String	String	User variable of type String **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.GetUserData.WildcardExpression	WildcardExpression	User variable of type WildcardExpression **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.HasGlobalData	Boolean	If true, permanently stored global data is available. **Note:** This property is not SaaS-compatible.	String: Variable key
PDStorage.HasGlobalDataWait	Boolean	If true, a request is kept waiting until the requested global variable exists in the storage or the specified time interval has elapsed. The value of the property is then set to false. It is true by default. **Note:** This property is not SaaS-compatible.	1. String: Variable key 2. Number: Timeout (in seconds)
PDStorage.HasUserData	Boolean	If true, persistently stored user data is available. **Note:** This property is not SaaS-compatible.	String: Variable key
ProgressPage.Enabled	Boolean	If true, download progress is indicated to the user by a progress page.	

Name	Type	Description	Parameters
ProgressPage.Sent	Boolean	f true, a progress page is displayed when a requested web object is downloaded.	
ProtocolDetector.DetectedProtocol	String	String containing name of a protocol that has been detected as being used for traffic on a connection between Web Gateway and a client	
ProtocolDetector.ProtocolFilterable	Boolean	If true, filtering is supported for a protocol that has been detected as being used for web traffic	
Protocol.FailureDescription	String	String containing description of a connection error under the current protocol	
Proxy.EndUserURL	String	String representing URL for display to a user	
Proxy.IP	IP	IP address of Web Gateway The default value is 0. **Note:** This property is not SaaS-compatible. A rule with this property can, however, be synchronized for use in the cloud, but only the default value is then retrieved for this property.	
Proxy.Outbound.IP	IP	Source IP address that Web Gateway uses when connecting to web servers or next-hop proxies **Note:** Do not confuse this property with the *Proxy.OutboundIP* property, which has no dot before *IP*.	
Proxy.Outbound.IPList	List of IP	List of source IP addresses that Web Gateway selects an address from when connecting to web servers or next-hop proxies. **Note:** This property is not SaaS-compatible.	
Proxy.Outbound.Port	Number	Number of source port that Web Gateway uses when	

Name	Type	Description	Parameters
		connecting to web servers or next-hop proxies	
Proxy.OutboundIP	IP	Source IP address for replacing multiple source IP addresses that Web Gateway might use when connecting to web servers or next-hop proxies The address is selected from a list, using a number parameter to identify its position in the list. **Note:** This property is not SaaS-compatible. Do not confuse it with the *Proxy.Outbound.IP* property, which has a dot before *IP*.	Number: Position of source IP address in list
Proxy.Port	Number	Number of a port used by Web Gateway The default value is 0. **Note:** This property is not SaaS-compatible. A rule with this property can, however, be synchronized for use in the cloud, but only the default value is then retrieved for this property.	

Properties - Q

The following table describes the properties that have names beginning with Q.

Properties – Q

Name	Type	Description	Parameters
Quota.AuthorizedOverride.GetLogin	String	User name submitted for performing an authorized override **Note:** This property is not SaaS-compatible.	
Quota.AuthorizedOverride.IsActivationRequest	Boolean	If true, an authorized user has chosen to continue with a authorized override session after session time has been exceeded.	

Name	Type	Description	Parameters
		Note: This property is not SaaS-compatible.	
Quota.AuthorizedOverride.IsActivationRequest.Strict	Boolean	If true, an authorized user has chosen to continue with an Authorized Override session and the request for continuing the session applies to the current settings. **Note:** This property is not SaaS-compatible.	
Quota.AuthorizedOverride.JS.ActivateSession	String	String in JavaScript code calling the function that is executed when an authorized user chooses to start a new session by clicking the appropriate button in the authorized override template. The code is provided when the template is created and displayed to the user. **Note:** This property is not SaaS-compatible.	
Quota.AuthorizedOverride.LastAuthorizedPerson	String	User name of the last person who performed an authorized override to provide additional session time for a user **Note:** This property is not SaaS-compatible.	
Quota.AuthorizedOverride.RemainingSession	Number	Remaining time (in seconds) for an authorized override session **Note:** This property is not SaaS-compatible.	
Quota.AuthorizedOverride.SessionExceeded	Boolean	If true, the time allowed for an authorized override session has been exceeded. **Note:** This property is not SaaS-compatible.	
Quota.AuthorizedOverride.SessionLength	Number	Time length (in seconds) for an authorized override session **Note:** This property is not SaaS-compatible.	
Quota.Coaching.IsActivationRequest	Boolean	If true, a user has chosen to continue with a new coaching	

Name	Type	Description	Parameters
		session after session time has been exceeded. **Note:** This property is not SaaS-compatible.	
Quota.Coaching.IsActivationRequestStrict	Boolean	If true, a user has chosen to continue with a Coaching session and the request for continuing the session applies to the current settings. **Note:** This property is not SaaS-compatible.	
Quota.Coaching.JS.ActivateSession	String	String in JavaScript code calling the function that is executed when a user chooses to start a new session by clicking the appropriate button in the coaching session template. The code is provided when the template is created and displayed to the user. **Note:** This property is not SaaS-compatible.	
Quota.Coaching.RemainingSession	Number	Remaining time (in seconds) for a coaching session **Note:** This property is not SaaS-compatible.	
Quota.Coaching.SessionExceeded	Boolean	If true, the time allowed for a coaching session has been exceeded. **Note:** This property is not SaaS-compatible.	
Quota.Coaching.SessionLength	Number	Time length (in seconds) for a coaching session **Note:** This property is not SaaS-compatible.	
Quota.Time.Exceeded	Boolean	If true, the time quota has been exceeded. **Note:** This property is not SaaS-compatible.	
Quota.Time.IsActivationRequest	Boolean	If true, a user has chosen to continue with a new time session after session time has been exceeded.	

Name	Type	Description	Parameters
		Note: This property is not SaaS-compatible.	
Quota.Time.IsActivationRequestStrict	Boolean	If true, a user has chosen to continue with a new Time session and the request for continuing the session applies to the current settings. **Note:** This property is not SaaS-compatible.	
Quota.Time.JS.ActivateSession	String	String in JavaScript code calling the function that is executed when a user chooses to start a new session by clicking the appropriate button in the time session template. The code is provided when the template is created and displayed to the user. **Note:** This property is not SaaS-compatible.	
Quota.Time.RemainingDay	Number	Time (in seconds) remaining from the configured time quota for the current day **Note:** This property is not SaaS-compatible.	
Quota.Time.RemainingDay.ReducedAtActivation	Number	Time (in seconds) remaining from the configured time quota for the current day when a user has just started a session **Note:** This property is not SaaS-compatible.	
Quota.Time.RemainingDay.ReducedAtDeactivation	Number	Time (in seconds) remaining from the configured time quota for the current day when a user has just closed a session **Note:** This property is not SaaS-compatible.	
Quota.Time.RemainingMonth	Number	Time (in seconds) remaining from the configured time quota for the current month **Note:** This property is not SaaS-compatible.	

Name	Type	Description	Parameters
Quota.Time.RemainingMonth.ReducedAtActivation	Number	Time (in seconds) remaining from the configured time quota for the current month when a user has just started a session **Note:** This property is not SaaS-compatible.	
Quota.Time.RemainingMonth.ReducedAtDeactivation	Number	Time (in seconds) remaining from the configured time quota for the current month when a user has just closed a session **Note:** This property is not SaaS-compatible.	
Quota.Time.RemainingSession	Number	Remaining time (in seconds) for a time session **Note:** This property is not SaaS-compatible.	
Quota.Time.RemainingWeek	Number	Time (in seconds) remaining from the configured time quota for the current week **Note:** This property is not SaaS-compatible.	
Quota.Time.RemainingWeek.ReducedAtActivation	Number	Time (in seconds) remaining from the configured time quota for the current week when a user has just started a session **Note:** This property is not SaaS-compatible.	
Quota.Time.RemainingWeek.ReducedAtDeactivation	Number	Time (in seconds) remaining from the configured time quota for the current week when a user has just closed a session **Note:** This property is not SaaS-compatible.	
Quota.Time.SessionExceeded	Boolean	If true, the time allowed for a time session has been exceeded. **Note:** This property is not SaaS-compatible.	
Quota.Time.SessionLength	Number	Time length (in seconds) for a time session	

Name	Type	Description	Parameters
		Note: This property is not SaaS-compatible.	
Quota.Time.SizePerDay	Number	Time (in seconds) allowed per day under the configured quota **Note:** This property is not SaaS-compatible.	
Quota.Time.SizePerMonth	Number	Time (in seconds) allowed per month under the configured quota **Note:** This property is not SaaS-compatible.	
Quota.Time.SizePerWeek	Number	Time (in seconds) allowed per week under the configured quota **Note:** This property is not SaaS-compatible.	
Quota.Volume.Exceeded	Boolean	If true, the volume quota has been exceeded. **Note:** This property is not SaaS-compatible.	
Quota.Volume.IsActivationRequest	Boolean	If true, a user has chosen to continue with a new volume session after session time has been exceeded. **Note:** This property is not SaaS-compatible.	
Quota.Volume.IsActivationRequestStrict	Boolean	If true, a user has chosen to continue a session when the configured volume has been exceeded and the request for continuing the session applies to the current settings. **Note:** This property is not SaaS-compatible.	
Quota.Volume.JS.ActivateSession	String	String in JavaScript code calling the function that is executed when a user chooses to start a new session by clicking the appropriate button in the volume session template. The code is provided when the template is created and displayed to the user.	

Name	Type	Description	Parameters
		Note: This property is not SaaS-compatible.	
Quota.Volume.RemainingDay	Number	Volume (in bytes) remaining from the configured volume quota for the current day **Note:** This property is not SaaS-compatible.	
Quota.Volume.RemainingMonth	Number	Volume (in bytes) remaining from the configured volume quota for the current month **Note:** This property is not SaaS-compatible.	
Quota.Volume.RemainingSession	Number	Remaining time (in seconds) for a volume session **Note:** This property is not SaaS-compatible.	
Quota.Volume.RemainingWeek	Number	Volume (in bytes) remaining from the configured volume quota for the current week **Note:** This property is not SaaS-compatible.	
Quota.Volume.SessionExceeded	Boolean	If true, the time allowed for a volume session has been exceeded. **Note:** This property is not SaaS-compatible.	
Quota.Volume.SessionLength	Number	Time length (in seconds) for a volume session **Note:** This property is not SaaS-compatible.	
Quota.Volume.SizePerDay	Number	Volume (in bytes) allowed per day under the configured quota **Note:** This property is not SaaS-compatible.	
Quota.Volume.SizePerMonth	Number	Volume (in bytes) allowed per month under the configured quota **Note:** This property is not SaaS-compatible.	
Quota.Volume.SizePerWeek	Number	Volume (in bytes) allowed per week under the configured quota	

Name	Type	Description	Parameters
		Note: This property is not SaaS-compatible.	

Properties - R

The following table describes the properties that have names beginning with R.

Properties – R

Name	Type	Description	Parameters
Redirect.URL	String	String representing a URL that a user is redirected to by an authentication or quota rule	
Reporting.URL.Categories	List of Category	List of all URL categories used on the appliance	
Reporting.URL.Reputation	List of Number	List of all reputation score values used on the appliance	
Request.Header.FirstLine	String	First line of a header sent with a request	
Request.POSTForm.Get	String	Retrieves URL encoded data in the POST form sent by the external Identity Provider.	
Request.ProtocolAndVersion	String	Protocol and protocol version used when a request is sent	
Response.ProtocolandVersion	String	Protocol and protocol version used when a response is sent	
Response.Redirect.URL	String	URL that a user is redirected to when a response is sent	
Response.StatusCode	String	Status code of a response	
Rules.CurrentRuleID	String	ID of the rule that is currently processed	
Rules.CurrentRuleName	String	Name of the rule that is currently processed	
Rules.CurrentRuleSetName	String	Name of the rule set that is currently processed	
Rules.EvaluatedRules	List of String	List of all rules that have been processed	
Rules.EvaluatedRules.Names	List of String	List with names of all rules that have been processed	

Name	Type	Description	Parameters
Rules.FiredRules	List of String	List of all rules that have applied	
Rules.FiredRules.Names	List of String	List with names of all rules that have applied	

Properties - S

The following table describes the properties that have names beginning with S.

Properties – S

Name	Type	Description	Parameters
SecureReverseProxy.EmbeddedHost	String	Host name of a URL in an HTTP request that is embedded in an HTTPS request **Note:** This property is not SaaS-compatible	
SecureReverseProxy.EmbeddedProtocol	String	Protocol of a URL in an HTTP request that is embedded in an HTTPS request **Note:** This property is not SaaS-compatible	
SecureReverseProxy.EmbeddedURL	String	URL in an HTTP request that is embedded in an HTTPS request This is the URL for the host specified by the value of the *SecureReverseProxy.EmbeddedHost* property. **Note:** This property is not SaaS-compatible	String: Host name of the URL
SecureReverseProxy.GetDomain	String	Domain specified in the settings for the SecureReverseProxy module **Note:** This property is not SaaS-compatible	
SecureReverseProxy.IsValidReverseProxyRequest			
see above	Boolean	If true, the URL submitted in a request has the format required in a SecureReverseProxy configuration.	

Name	Type	Description	Parameters
		Note: This property is not SaaS-compatible	
SecureReverseProxy.URLToEmbed	String	URL submitted in a HTTP request that is embedded in an HTTPS request **Note:** This property is not SaaS-compatible	
SecureToken.CreateToken	String	Encrypted string This string serves as a token for securing an IP address. An AES-128-bit algorithm is used to create the token. Depending on the value of a parameter in the settings of the SecureReverseProxy module, the string includes a time stamp.	String: String to encrypt
SecureToken.IsValid	Boolean	If true, the specified token is valid and has not expired. Depending on the on the value of a parameter in the settings of the SecureReverse Proxy module, the token string includes no time stamp. Expiration of the token is then not checked.	1. String: Token to be checked 2. Number: Time (in seconds) to elapse until the token expires
SecureToken.GetString	String	String serving as a token for securing an IP address If the token is invalid or has expired, the string is empty.	1. String: Token to be checked 2. Number: Time (in seconds) to elapse until the token expires
Server.DownloadBandwidth	Number	Bandwidth (in bytes per second) consumed for downloads from web servers	
Server.UploadBandwidth	Number	Bandwidth (in bytes per second) consumed for uploads to web servers	
SNMP.Trap.Additional	String	Additional message sent to a trap under the SNMP protocol	
SOCKS.Version	String	Version of the SOCKS protocol that is used when a client requests access to a web object under this protocol	

Name	Type	Description	Parameters
SSL.Certificate.CN.ToWildcard	Wildcard Expression	Common name in an SSL certificate converted into a wildcard expression	String: Common name to convert
SSL.Client.Certificate.Serial	String	Serial of a client certificate	
SSL.ClientContext.IsApplied	Boolean	If true, parameters for setting the client context in SSL-secured communication have been configured.	
SSL.Server.Certificate.AlternativeCNs			
see above	List of Wildcard Expression	List of alternative common names for a web server as used in SSL certificates	
SSL.Server.Certificate.CN	String	Common name of a web server provided in a certificate for SSL-secured communication	
SSL.Server.Certificate.CN.HasWildcards			
see above	Boolean	If true, the common name for a web server in an SSL certificate includes wildcards.	
SSL.Server.Certificate.DaysExpired	Number	Number of days that an SSL certificate for a web server has expired	
SSL.Server.Certificate.HostAndCertificate			
see above	HostAnd Certificate	Host name and certificate for connecting to web server in SSL-secured communication	
SSL.Server.Certificate.OnlyCertificate			
see above	HostAnd Certificate	Certificate for connecting to a web server in SSL-secured communication	
SSL.Server.Certificate.SelfSigned	Boolean	If true, an SSL certificate for a web server is self-signed.	
SSL.Server.Certificate.SHA1Digest	String	String representing an SHA1Digest of a SSL certificate for a web server	
SSL.Server.Certificate.SignatureMethod			
see above	String	Text describing the method used for signing the certificate	
SSL.Server.CertificateChain.AllRevocationStatusesKnown			

Name	Type	Description	Parameters
see above	Boolean	If true, it is known of all SSL certificates in a certificate chain for a web server whether they are revoked or not.	
SSL.Server.CertificateChain.ContainsExpiredCA			
see above	Boolean	If true, an SSL certificate in a certificate chain for a web server has expired.	
SSL.Server.CertificateChain.ContainsRevoked			
see above	Boolean	If true, an SSL certificate in a certificate chain for a web server has been revoked.	
SSL.Server.CertificateChain.FirstKnownCAIsTrusted			
see above	Boolean	If true, a the certificate authority for issuing SSL certificates that has been found first in a certificate chain for a web server is trusted.	
SSL.Server.CertificateChain.FoundKnownCA			
see above	Boolean	If true, a known certificate authority for issuing SSL certificates has been found in a certificate chain for a web server.	
SSL.Server.CertificateChain.IsComplete	Boolean	If true, the chain of SSL certificates for a web server is complete.	
SSL.Server.CertificateChain.IssuerCNs	List of String	List of common names for the issuers that issued an SSL certificate in a certificate chain for a web server The list is sorted in bottom-up order. It ends with the common name of the issuer that issued the certificate for the self-signed root certificate authority (CA).	
SSL.Server.CertificateChain.Length	Number	Number of SSL certificates in a certificate chain for a web server	
SSL.Server.CertificateChain.ContainsViolation			

Name	Type	Description	Parameters
see above	Boolean	If true, at least one of the following violations has been found during verification of the chain of SSL certificates for a web server: • The chain exceeds the allowed path length • The intended usage is not observed, for example, a server sends a client certificate. • An issuer constraint is ignored, for example, a certificate authority (CA) issues a certificate, although the issuer of the CA forbids issuing this certificate. • A certificate contains an unknown extension that is considered critical	
SSL.Server.CertificateChain.SignatureMethods			
see above	List of String	List of texts describing the methods used for signing the certificates in the chain	
SSL.Server.Cipher.KeyExchangeBits	Number	Normalized strength of the weakest link involved in a key exchange performed in SSL-secured communication	
SSL.Server.Handshake.CertificateIsRequested			
see above	Boolean	If true, a handshake is requested for setting up a connection to a web server in SSL-secured communication.	
SSL.Server.SkypeForBusiness.IsByPassed			
see above	Boolean	If true, the option for bypassing Skype for business traffic is enabled for HTTPS scanning.	
SSO.Action	String	Returns the name of an internal action performed in response to an SSO request. **Note:** This property is not SaaS-compatible	
SSO.Config	String	Returns the name of the settings used by an internal	

Name	Type	Description	Parameters
		action performed in response to an SSO request. **Note:** This property is not SaaS-compatible	
SSO.Debug	String	Returns an SSO debug message. **Note:** This property is not SaaS-compatible	
SSO.GetConnectorInfo	Variable	Returns information about the SSO connector to the service the user is requesting. This information is stored as a JSON object in a local variable named sso-conn-info. **Note:** This property is not SaaS-compatible	String: Service ID
SSO.GetData	JSON object	Returns additional information needed for SAML single sign-on. **Note:** This property is not SaaS-compatible	
SSO.GetDatFile	String	Retrieves the specified DAT file from the update server and returns the contents of the file in a string. The Single Sign On module uses the collection of SSO DAT files to create the launchpad. **Note:** This property is not SaaS-compatible	String: Name of the SSO DAT file
SSO.GetIceTokenLoginAction	String	Returns the user information needed to complete single sign-on to the requested service or application. **Note:** This property is not SaaS-compatible	1. String: Service ID 2. Variable: sso-user-data
SSO.GetPostLoginAction	String	Returns the information needed to complete single sign-on to the requested HTTP service or application. **Note:** This property is not SaaS-compatible	1. String: Identity Provider 2. String: User name 3. String: Service ID 4. String: User account
SSO.GetSAMLLoginAction	String	Returns the user information needed to complete single	1. String: Service ID 2. Variable: sso-user-data

Name	Type	Description	Parameters
		sign-on to the requested SAML service or application. **Note:** This property is not SaaS-compatible	
SSO.GetServices	JSON object	Returns all information about the current user added by the SSO Select Services rule set. This information is returned in JSON format and includes the names of cloud services the user is allowed to access and all account information. **Note:** This property is not SaaS-compatible	Variable: "conditions"
SSO.GetTools	String	Returns a string of JavaScript tools. **Note:** This property is not SaaS-compatible	
SSO.IsManagementRequest	Boolean	Returns a true value if the current request is an SSO request and one or both of the following conditions are met: • Web Gateway has received an SSO request. • The SSO.Action property is processed with valid settings. **Note:** This property is not SaaS-compatible	
SSO.LogProperties	JSON object	Stores information about each SSO request that is used to generate the SSO access and SSO trace logs. **Note:** This property is not SaaS-compatible	
SSO.ManagementHost	String	Returns the host name of the SSO service specified in the configuration. Typically, this value is the name of the server hosting the SSO service provided by Web Gateway. **Note:** This property is not SaaS-compatible	

Name	Type	Description	Parameters
SSO.OTPRequired	Boolean	Returns a true value if the SSO action requires OTP authentication. **Note:** This property is not SaaS-compatible	
SSO.ProcessTask	Boolean	Processes common SSO tasks, such as credential management, using the Single Sign On settings. If the SSO tasks are processed successfully, this property returns a true value. **Note:** This property is not SaaS-compatible	**Note:** The parameters are passed in URLs.
SSO.UserHasAccessToService	Boolean	Returns a true value if the user is allowed to access the cloud service or manage the account. **Note:** This property is not SaaS-compatible	
SSOConnector.ToString	String	Converts the name of a cloud connector to the Service ID that identifies the corresponding cloud service or application.	String: Name of cloud connector
Statistics.Counter.Get	Number	Number of occurrences of an activity or situation recorded on a counter **Note:** This property is not SaaS-compatible	String: Name of counter
Statistics.Counter.GetCurrent	Number	Number of occurrences of an activity or situation recorded on a counter (fully completed) during the last minute **Note:** This property is not SaaS-compatible	String: Name of counter
Stopwatch.GetMicroSeconds	Number	Time measured for rule set processing in microseconds	String: Name of rule set
Stopwatch.GetMilliSeconds	Number	Time measured for rule set processing in milliseconds	String: Name of rule set
StreamDetector.IsMediaStream	Boolean	If true, a requested web object is streaming media. This is the basic property used in streaming media filtering.	

Name	Type	Description	Parameters
StreamDetector.MatchedRule	String	Name of a streaming media filtering rule that has matched This property is given a value if the *StreamDetector.IsMediaStream* property is set to *true*.	
StreamDetector.Probability	Number	Probability for a web object that it is streaming media Values range from 1 to 100. This property is given a value if the *StreamDetector.IsMediaStream* property is set to *true*.	
String.BackwardFind	Number	Position where a substring begins that is found in a string by a backward search Returns -1 if the substring is not found.	1. String: String containing substring 2. String: Substring 3. Number: Position where backward search for substring begins
String.Base64DecodeAsBinary	String	String of binary digits that is the result of decoding a base-64 encoded string	String: String in encoded format
String.Base64DecodeAsText	String	Text string that is the result of decoding a base-64 encoded string	String: String in encoded format
String.Base64Encode	String	String that is the result of using the base-64 encoding method to encode a string	String: String to encode
String.BelongsToDomains	Boolean	If true, a specified string is found in a list of domain names The value of the property is "true" if the string matches a list entry, which means it is a domain name. The value of the property is also "true" if the string is a character or sequence of characters followed by a dot and a substring that matches a list entry (*.<list entry>), which means it is the name for a subdomain of a domain in the list.	1. String: String to be found in list 2. List of string: List of domain names

Name	Type	Description	Parameters
		In both cases the string is set as the value of the *List.LastMatches* property.	
String.Concat	String	Concatenation of two specified strings	1. String: First string to concatenate 2. String: Second string to concatenate
String.CRLF	String	Carriage-return line-feed	
String.Find	Number	Position where a substring begins that is found in a string by a forward search Returns -1 if the substring is not found.	1. String: String containing substring 2. String: Substring 3. Number: Position where forward search for substring begins
String.FindFirstOf	Number	Position of the first character of a substring found in a string Returns -1 if the substring is not found.	1. String: String containing substring 2. String: Substring 3. Number: Position where search for substring begins
String.FindLastOf	Number	Position of the last character of a substring found in a string Returns -1 if the substring is not found.	1. String: String containing substring 2. String: Substring 3. Number: Position where search for substring begins
String.GetWordCount	Number	Number of words in a string	String: String to get number of words for
String.Hash	String	Hash value of a particular type for a given string	1. String: String to find hash value for 2. String: Hash type
String.IsEmpty	Boolean	If true, the specified string is empty.	String: String checked for being empty
String.Length	Number	Number of characters in a string	String: String to count characters for
String.LF	String	Line-feed	
String.MakeAnonymous	String	String that has been made anonymous and requires one	String: String to anonymize

Name	Type	Description	Parameters
		or two passwords for reverting anonymization The string that is to be anonymized is specified as a parameter of the property. The passwords are set within the Anonymization settings, which are provided as settings of the property. You can use the property in a rule to anonymize sensitive data, for example, the user name that is retrieved as the value of the Authentication.UserName property. An event in this rule sets the authentication property to the value of String.MakeAnonymous, which takes the authentication property as its parameter, so its value is the anonymized user name. After the set event has been executed, the anonymized user name is also the value of Authentication.UserName. Sensitive information is protected this way. For the rule to work, a rule with the authentication property must have been processed before. Otherwise the string that is to be anonymized would not be known.	
String.MatchWildcard	List of String	List of terms in a string that match a wildcard expression	1. String: String with matching terms 2. Wildcard Expression: Wildcard expression to match 3. Number: Position where search for substring begins
String.Replace	String	String having a substring replaced by a string as specified	1. String: String containing substring to replace 2. Number: Position where replacement begins 3. Number: Number of characters to replace

Name	Type	Description	Parameters
			4. String: Replacing string
String.ReplaceAll	String	String having each occurrence of a substring replaced by string as specified	1. String: String containing substring to replace 2. String: Replacing substring 3. String: Substring to replace
String.ReplaceAllMatches	String	String having each occurrence of a substring that matches a wildcard expression replaced by a string as specified	1. String: String containing substring to replace 2. Wildcard Expression: Wildcard expression to match 3. String: Substring to replace
String.ReplaceFirst	String	String having first occurrence of a substring replaced by a string as specified	1. String: String containing substring to replace 2. String: Substring to replace 3. String: Replacing string
String.ReplaceFirstMatch	String	String having first occurrence of a substring that matches a wildcard expression replaced by a string as specified	1. String: String containing substring to replace 2. Wildcard Expression: Wildcard expression to match 3. String: Replacing substring
String.ReplaceIfEquals	String	String having every occurrence of a substring replaced by a string as specified	1. String: String containing substring to replace 2. String: Substring to replace 3. String: Replacing string
String.SubString	String	Substring contained in a string specified by start position and length	1. String: String containing substring 2. Number: Position where substring begins 3. Number: Number of characters in substring If no number is specified, the substring extends to the end of the original string
String.SubStringBetween	String	Substring of string extending between two other substrings of this string	1. String: String containing substrings

Name	Type	Description	Parameters
		The search for this substring begins with looking for the first of the other substrings. If this string is found, the search is continued with looking for the second substring. If the first substring is not found, the search has no result. If the second substring is not found, the wanted substring extends from the end of the first substring to the end of the main string.	2. String: Substring ending immediately before the wanted substring 3. String: Substring beginning immediately after the wanted substring
String.ToCategory	Category	String converted into a category	String: String to convert
String.ToDimension	Dimension	String converted into a dimension	String: String to convert
String.ToHex	Hex	String converted into a hex value	String: String to convert
String.ToIP	IP	String converted into an IP address	String: String to convert
String.ToIPRange	PRange	String converted into a range of IP addresses	String: String to convert
String.ToMediaType	MediaType	String converted into a media type	String: String to convert
String.ToNumber	Number	String converted into a number	String: String to convert
String.ToSSOConnector	String	Converts the Service ID that identifies a cloud service or application to the name of the corresponding cloud connector.	String: Service ID
String.ToStringList	List of String	String converted into a string list The string list is a list of the elements in the string to convert. For example, the string to convert can be a text and the string list a list of the words in this text. The delimiter is a substring that separates elements in the string to convert. For example, in a normal text, the delimiter is the whitespace. The substring can be a single	1. String: String to convert 2. String: Delimiter 3. String: Substring beginning immediately after the wanted substring

Name	Type	Description	Parameters
		character, such as the whitespace, or multiple characters. To specify the whitespace, hit the space bar. A trim character is a character that appears at the beginning or end of an element in the string to convert, but not in the string list. A trim character can, for example, be a comma, a period, or a single quotation mark. It can also be an “invisible” character, such as a tab stop or a line feed. To specify trim characters, type them in the input field that is provided on the user interface without separating them from each other. Use the following combinations to type invisible characters: \t – tab stop \r – carriage return \n – line feed \b – backspace \\ – backslash If you specify a character as a delimiter, it is also deleted from the resulting string list, so you need not specify it as a trim character.	
String.ToWildcard	Wildcard Expression	String converted into a wildcard expression	String: String to convert
String.URLDecode	String	Standard format of a URL that was specified in encoded format	String: URL in encoded format
String.URLEncode	String	Encoded format of a URL	String: URL to encode
System.HostName	String	Host name of an appliance	
System.UUID	String	UUID (Universal Unique Identifier) of an appliance	

Properties - T

The following table describes the properties that have names beginning with T.

Properties – T

Name	Type	Description	Parameters
TIE.Filereputation	Number	File reputation score that has been retrieved from a TIE server **Note:** This property is not SaaS-compatible.	
Timer.FirstReceivedFirstSentClient	Number	Processing time consumed between receiving the first byte from a client on the appliance and sending the first byte to this client within a transaction Using this property is only supported when HTTP or HTTPS connections are involved, but not for FTP connections.	
Timer.FirstSentFirstReceivedServer	Number	Processing time consumed between sending the first byte from the appliance to a web server and receiving the first byte from this server within a transaction Using this property is only supported when HTTP or HTTPS connections are involved, but not for FTP connections.	
Timer.HandleConnectToServer	Number	Processing time consumed for connecting to a web server within a transaction	
Timer.LastReceivedLastSentClient	Number	Processing time consumed between receiving the last byte from a client on the appliance and sending the last byte to this client within a transaction Using this property is only supported when HTTP or HTTPS connections are involved, but not for FTP connections.	
Timer.LastSentLastReceivedServer	Number	Processing time consumed between sending the last byte from the appliance to a web server and receiving the last	

Name	Type	Description	Parameters
		byte from this server within a transaction Using this property is only supported when HTTP or HTTPS connections are involved, but not for FTP connections.	
Timer.ResolveHostNameViaDNS	Number	Processing time consumed for looking up a host name on a DNS server within a transaction Only lookups on external servers are considered. Cache lookups are disregarded.	
Timer.TimeInExternals	Number	Time (in milliseconds) consumed when processing a request in waiting for responses by components other than the rule engine that are involved in the process, for example, domain controllers or anti-malware scanning engines. This time is the time that has already been consumed in waiting when the property is evaluated. Waiting periods in all relevant processing cycles are considered when calculating this time.	
Timer.TimeInRuleEngine	Number	Time (in milliseconds) consumed by the rule engine for processing a request, including activities in all relevant processing cycles, at the time when the property is evaluated. Processing a request through all relevant processing cycles is also referred to as a transaction. When the property is evaluated within a rule for log handling, its value is the time that was used by the rule engine for the complete transaction.	
Timer.TimeInTransaction	Number	Time (in milliseconds) consumed for processing a	

Name	Type	Description	Parameters
		request, including activities in all relevant processing cycles, at the time when the property is evaluated. Time used for rule engine activities and waiting times are summed up in this property value. Processing a request through all relevant processing cycles is also referred to as a transaction. When the property is evaluated within a rule for log handling, its value is the time that was used for the complete transaction.	
Tunnel.Enabled	Boolean	If true, an HTTP or HTTPS tunnel is enabled	

Properties - U

The following table describes the properties that have names beginning with U.

Properties – U

Name	Type	Description	Parameters
URL	String	URL of a web object	
URL.Categories	List of Category	List of URL categories that a URL belongs to	
URL.CategoriesForURL	List of Category	List of URL categories that a specified URL belongs to	String: URL in string format
URL.CategorySetVersion	Number	Version number of the category set that is used for URL filtering	
URL.CloudLookupLedToResult		If true, the rating for a URL was retrieved by a cloud lookup that was performed using the Global Threat Intelligence service.	
URL.DestinationIP	IP	IP address for a URL as found in a DNS lookup	
URL.DiscardedHost	String	Name of a host that was discarded when conflicting host names occurred in a	

Name	Type	Description	Parameters
		request sent in HTTP(S) or SSL communication. A conflict of this kind is also known as *domain fronting*. It is resolved by the proxy on Web Gateway, which prefers one of the conflicting host names over the other, depending on what is configured. By querying the value of this property in the criteria of a rule or by logging it, you can detect a host name conflict. If no conflict arises, the value of the property is an empty string. Conflicting host names might occur in the following scenarios: • Under HTTP(S): The first-line part of the communication does not match the host header that is sent with a request. This conflict does not arise under HTTPS2, where no first-line part is sent in any single stream. • Under SSL: The host name sent in a CONNECT request does not match the host information read from a client hello.	
URL.Domain	String	Name of the domain that access was requested to	
URL.DomainSuffix	String	Suffix appended to the name of the domain that access was requested to	
URL.FileExtension	String	Extension of the file name for a requested file	
URL.FileName	String	Name of a file that can be accessed through a URL	
URL.ForwardDNSLedToResult	Boolean	If true, the rating for a URL was retrieved by performing a forward DNS lookup.	
URL.Geolocation	String	ISO 3166 code for the country where the host that a URL belongs to is located	

Name	Type	Description	Parameters
		If a value is to be assigned to this property, the following option of the settings for the URL Filter module must be enabled: *Only use online GTI web reputation and categorization services.*	
URL.Geolocation	String	Name of the country where the host that a given URL belongs to is located The URL is the URL that was sent with the request that is currently processed. The country is identified according to ISO 3166. **Note:** The name can only be found if the following option of the settings for URL filtering is selected: Disable local GTI database	
URL.GeolocationForURL	String	Name of the country where the host that a given URL belongs to is located The URL is specified as a parameter of the property. The country is identified according to ISO 3166. **Note:** The name can only be found if the following option of the settings for URL filtering is selected: Disable local GTI database	String: URL that country name is to be found for
URL.GetParameter	String	Parameter of a URL in string format	String: Parameter name
URL.HasParameter	Boolean	If true, a specified parameter belongs to the parameters of a URL.	String: Parameter name
URL.Host	String	Host that a URL belongs to	
URL.Host.BelongsToDomains	Boolean	If true, a host that access was requested to by submitting a particular URL belongs to one of the domains in a list. The name of a host that was found to belong to one of the domains is set as the value of the *List:LastMatches* property.	List of string: List of domain names

Name	Type	Description	Parameters
		You can use the _URL.Host.BelongsToDomains_ property to match anything to the domain name in a URL or anything to the left of a dot of a domain name (*.domain.com). Terms including the domain name (*domain.com) are not counted as matches. _Example:_ _Domain List_ is the string list specified as the property parameter. It contains the following entries (dots preceding a domain name in a URL are omitted): _twitter.com_ _mcafee.com_ _dell.com_ _k12.ga.us_ _xxx_ Then the criteria: _URL.Host.BelongsToDomains("Domain List") equals true_ matches for the following URLs: _http://twitter.com_ _http://www.twitter.com_ _http://my.mcafee.com_ _http://my.support.dell.com_ _http://www.dekalb.k12.ga.us_ _any.site.xxx_ but not for: _http://malicioustwitter.com_ _http://www.mymcafee.com_ _http://www.treasury.ga.us_ Using this property avoids the effort of creating more complicated solutions to accomplish the same, for example: • Using two entries in a list of wildcard expressions, such as: _twitter.com_ and _*twitter.com_ • Using a single, complex entry in a list of wildcard expressions, such as: _regex((.*\.|.?)twitter\.com)_ If these entries were contained in the list _Other_	

Name	Type	Description	Parameters
		Domain List, the following criteria would match for the *twitter.com* domain: *URL.Host matches in list "Other Domain List"*	
URL.HostIsIP	Boolean	If true, the URL that is submitted for access to a host is an IP address.	
URL.IsHighRisk	Boolean	If true, the reputation score of a URL falls in the high risk range.	
URL.IsMediumRisk	Boolean	If true, the reputation score of a URL falls in the medium risk range.	
URL.IsMinimalRisk	Boolean	If true, the reputation score of a URL falls in the minimal risk range.	
URL.IsUnverified	Boolean	If true, the reputation score of a URL falls in the unverified risk range.	
URL.Parameters	List of String	List of URL parameters	
URL.ParametersString	String	String containing the parameters of a URL If the URL has parameters, the string begins with the ? character.	
URL.Path	String	Path name for a URL	
URL.Port	Number	Number of a port for a URL	
URL.Protocol	String	Protocol for a URL	
URL.Raw	String	URL in the format originally received on the appliance from a client or other network components Using this property for rule configuration will speed up processing because it saves the time used for converting URL code to a human readable format, as it is done for the simple *URL* property.	
URL.Reputation	Number	Reputation score for a given URL	

Name	Type	Description	Parameters
		The URL is the URL sent with the request that is currently processed.	
URL.ReputationForURL	Number	Reputation score for a given URL The URL is specified as a parameter of the property.	String: URL that reputation score is to be found for
URL.ReputationString	String	Reputation score for a given URL in string format The URL is the URL sent with the request that is currently processed.	
URL.ReputationStringForURL	String	Reputation score for a given URL The URL is specified as a parameter of the property.	String: URL that reputation score is to be found for
URL.ReverseDNSLedToResult	Boolean	If true, the rating for a URL was retrieved by performing a reverse DNS lookup.	
URL.SmartMatch	Boolean	If true, a URL matches one or more of the URL parts that are specified in string format in any of the entries in the list of URL parts that is given as the parameter of this property **Note:** Use of a very long string list here can impact performance. An entry in this string list must specify at least the *domain* or the *path* part of a URL as a substring. It can specify both. The domain part matches also if a URL only contains a subdomain of the specified domain. For the path part, it is sufficient if the beginning of the path in a URL matches it. Additionally, a list entry can specify the *protocol* and *port* of a URL. The value of the property is *true* if a URL matches the domain or the path part (or both) in an entry of the string list and also matches the protocol part (if specified) and the port part (if specified).	List of String: List with parts of URLs in string format

Name	Type	Description	Parameters
		If a port is specified in an entry of the string list, but not in the URL, there is no match. For example, with the following URL: *http://www.mycompany.com/samplepath/xyz* The below list entries will produce matches or not as follows: *mycompany.com* (match) *http://mycompany.com* (match) *https://mycompany.com* (no match) *http://www.mycompany.com/* (match) *host.mycompany.com* (no match) *http://www.mycompany.com:8080/* (no match) *http://www.mycompany.com/samplepath/* (match) */samplepath/* (match) *mycompany.com/samplepath/* (match) *com* (match) You can use this property to search for matches in a complex URL whitelist or blocklist, for example, in a list that contains both entries for URL hosts and for complete URLs.	
URLFilter.DatabaseVersion	Number	Version number of the database on an appliance	
URLFilter.EngineVersion	String	String identifying the version of the URL filtering module (engine)	
User-Defined.cacheMessage	String	Message text providing information on web cache usage	
User-Defined.eventMessage	String	Message text providing information on an event	
User-Defined.loadMessage	String	Message text providing information on CPU overload	
User-Defined.logLine	String	Entry written into a log file	

Name	Type	Description	Parameters
User-Defined.monitorLogMessage	String	Entry written into a log file	
User-Defined.notificationMessage	String	Text of a notification message	
User-Defined.requestLoadMessage	String	Message text providing information on request overload	
User-Defined.requestsPerSecond	Number	Number of requests processed on an appliance per second	

Properties - W

The following table describes the properties that have names beginning with W.

Properties – W

Name	Type	Description	Parameters
Wildcard.ToString	String	Wildcard expression converted into a string	Wildcard Expression: Wildcard expression to convert

List of statistics counters

The following table provides a list of the statistics counters that you can use in rules.

You can implement each of these counters by configuring it as a parameter of a particular rule event. Some of them are already implemented in rules of the default rule set system.

List of statistics counters

Name	Description
AMLoad	Percentage of CPU resources that is currently used by anti-malware filtering
AMUsed	Number of bytes in the virtual memory that are currently used by anti-malware filtering
AMUsedPhys	Number of bytes in the physical memory that are currently used by anti-malware filtering
AMJobQueueLength	Number of jobs in the anti-malware job queue by applications running on Web Gateway
ApplHighRisk	Number of applications that are considered a high risk
ApplMediumRisk	Number of applications that are considered a medium risk

Name	Description
ApplMinimalRisk	Number of applications that are considered a minimal risk
ApplUnverified	Number of applications that no risk level could be verified for
ApplicationMemoryUsage	Percentage of memory that is currently in use
AuthNTLMCacheRequests	Number of NTLM authentication requests that were granted based on user information in the cache
AuthUserCacheRequests	Number of authentication requests that were granted based on user information in the cache
BlockedByAntiMalware	Number of requests blocked by anti-malware filtering
BlockedByApplControl	Number of requests blocked by application filtering
BlockedByDLPMatch	Number of requests blocked by the DLP process
BlockedByMediaFilter	Number of requests blocked by media type filtering
BlockedByURLFilter	Number of requests blocked by URL filtering
Categories	Number of URLs that were processed in each of the categories used in URL filtering
CertNameMismatch	Number of mismatches that occurred in certificate verification
CertNameWildCardMatch	Number of matches that occurred in certificate verification when wildcards had been submitted
CertExpired	Number of expired certificates
CertRevoked	Number of revoked certificates
CertSelfSigned	Number of self-signed certificates
CertUnresolvable	Number of certificates that could not be resolved
ClientCount	Number of clients that are currently communicating with Web Gateway
CloudEnc.DecryptionBytesAll	Number of bytes for all web objects that cloud decryption was applied to
CloudEnc.DecryptionErrorsAll	Number of bytes for all web objects that had cloud decryption resulting in an error
CloudEnc.DecryptionHitsAll	Number of bytes for all web objects that cloud decryption was successfully applied to
CloudEnc.EncryptionBytesAll	Number of bytes for all web objects that cloud encryption was applied to
CloudEnc.EncryptionErrorsAll	Number of bytes for all web objects that had cloud encryption resulting in an error
CloudEnc.EncryptionHitsAll	Number of bytes for all web objects that cloud encryption was successfully applied to

Name	Description
CloseWaits	Number of sockets that are in CLOSE WAIT status
ConnectedSockets	Number of sockets that are connected to Web Gateway
ConnectionsBlocked	Number of blocked connections
ConnectionsLegitimate	Number of legitimate connections
CoreLoad	Percentage of CPU resources that is currently used by the core process
CoreUsed	Number of bytes in the virtual memory that are currently used by the core process
CoreUsedPhys	Number of bytes in the physical memory that are currently used by the core process
CoreThreads	Number of threats that currently processed in the core
CoordLoad	Percentage of CPU resources that is currently used by the Coordinator subsystem
CoordUsed	Number of bytes in the virtual memory that are currently used by the Coordinator subsystem
CoordUsedPhys	Number of bytes in the physical memory that are currently used by the Coordinator subsystem
CPULoad	Percentage of CPU resources that are currently in use
CPUIdle	Percentage of CPU resources that are currently not in use
CPUUser	Percentage of CPU resources that are currently used by user-related functions
CPUSystem	Percentage of CPU resources that are currently used by system functions
DLPMatches	Number of matches that were achieved in DLP filtering
FilesystemUsage	Percentage of the opt system partition that is currently in use
FtpBytesFromServer	Number of bytes for all web objects that were received from a web server under FTP
FtpBytesToServer	Number of bytes for all web objects sent to a web server under FTP
FtpRequests	Number of requests received under FTP
FtpTraffic	Number of bytes for all web objects sent and received under FTP
GTICloudTimedOut	Number of timeouts that occurred on the Global Threat Intelligence server when cloud lookups were performed in URL filtering

Name	Description
GTIFileRepCloudLookupDone	Number of cloud lookups that were performed by Global Threat Intelligence to retrieve file reputations
GTIRequestSentToCloud	Number of requests that were sent to Global Threat Intelligence to retrieve URL category information (not file reputations)
HandleConnectToServer	Average time (in milliseconds) spent on connecting to a server
HarddiskUsage	Percentage of hard-disk space that is currently available
HttpBytesFromClient	Number of bytes for all web objects that were received from a client under HTTP
HttpBytesFromServer	Number of bytes for all web objects that were received from a web server under HTTP
HttpBytesToClient	Number of bytes for all web objects that were sent to a client under HTTP
HttpBytesToServer	Number of bytes for all web objects that were sent to a web server under HTTP
HttpRequests	Number of requests received under HTTP
HttpTraffic	Number of bytes for all web objects sent and received under HTTP
HttpsBytesFromClient	Number of bytes for all web objects that were received from a client under HTTPS
HttpsBytesFromServer	Number of bytes for all web objects that were received from a web server under HTTPS
HttpsBytesToClient	Number of bytes for all web objects sent to a client under HTTPS
HttpsBytesToServer	Number of bytes for all web objects sent to a web server under HTTPS
HttpsRequests	Number of requests received under HTTPS
HttpsTraffic	Number of bytes for all web objects sent and received under HTTPS
ICAPReqmodRequests	Number of requests received in the Reqmod mode of ICAP
ICAPReqmodTraffic	Number of bytes for all web objects sent and received in the Reqmod mode of ICAP
ICAPRespmodRequests	Number of requests received in the Respmod mode of ICAP
ICAPRespmodTraffic	Number of bytes for all web objects sent and received in the Respmod mode of ICAP
IfpRequests	Number of requests received under IFP

Name	Description
KerberosRequests	Number of requests for authentication using the Kerberos method
LDAPRequests	Number of requests for authentication using the LDAP method
LoadPerCPU	Load on a Web Gateway appliance divided by number of CPU cores (rounded integer)
MalwareDetected	Number of malicious objects found by anti-malware filtering
MATDInfected	Number of viruses found by Advanced Threat Defense
MATDRequests	Number of requests sent to Advanced Threat Defense
MATDScanTime	Number of seconds used by the Advanced Threat Defense process
MemoryUsage	Percentage of memory that is currently in use
MemUsed	Number of bytes in the memory that are currently in use system-wide
MemFree	Number of bytes in the memory that are currently not in use system-wide
MT.Archives	Number of archives that are processed
MT.Audio	Number of audio files that are processed
MT.Database	Number of database files that are processed
MT.Document	Number of documents that are processed
MT.Executable	Number of executable files that are processed
MT.Image	Number of images that are processed
MT.Stream	Number of data streams that are processed
MT.Text	Number of text files that are processed
MT.Video	Number of video files that are processed
NetworkBytesReceived	Number of bytes received in network communication
NetworkBytesSent	Number of bytes sent in network communication
NTLMAgentRequests	Number of requests for authentication using an agent system to apply the NTLM method
NTLMAgentRequestProcTime	Average time (in milliseconds) for processing an NTLM Agent request
NTLMRequests	Number of requests for authentication using the NTLM method
NTLMRequestsProcTime	Average time (in milliseconds) for processing an NTLM request

Name	Description
OTPSendProcTime	Average time (in milliseconds) for processing an OTP request
OTPSendRequests	Number of requests received submitting a One-Time Password (OTP)
OTPVerifyProcTim	Average time (in milliseconds) for OTP verification
OTPVerifyRequests	Number of requests received in OTP verification
RADIUSRequests	Number of requests for authentication using the RADIUS method
RADIUSRequestsProcTime	Average time (in milliseconds) for processing a RADIUS request
RepHighRisk	Number of URLs with a reputation that is considered a high risk
RepMediumRisk	Number of URLs with a reputation that is considered a medium risk
RepMinimalRisk	Number of URLs with a reputation that is considered a minimal risk
RepUnverified	Number of URLs with a reputation that could not be verified
ReputationMalicious	Number of URLs with a reputation of being malicious
ReputationNeutral	Number of URLs with a reputation that is considered neutral regarding its risk level
ReputationTrusted	Number of URLs with a reputation that is trusted
ReputationUnverified	Number of URLs with a reputation that could not be verified
SOCKSRequests	Number of requests received under SOCKS
SOCKSTraffic	Number of bytes for all web objects sent and received under SOCKS
SOCKSv4Requests	Number of requests received under SOCKS version 4
SOCKSv4Traffic	Number of bytes for all web objects sent and received under SOCKS version 4
SOCKSv5Requests	Number of requests received under SOCKS version 5
SOCKSv5Traffic	Number of bytes for all web objects sent and received under SOCKS version 5
SSO.AllLogins	Number of logons performed using cloud single sign-on
SSO.IncorrectTokens	Number of invalid tokens submitted when logon was performed using cloud single sign-on
StatDBSize	Number of bytes stored in the statistics database
SwapUsed	Number of bytes in the swap space that are currently in use

Name	Description
SwapFree	Number of bytes in the swap space that are currently not in use
TimeConsumedByGTIFileRepCloudLookup	Average time (in milliseconds) spent for a cloud lookup performed by Global Threat Intelligence to retrieve a file reputation
TimeConsumedByGTIURLCloudLookup	Average time (in milliseconds) spent for a cloud lookup performed by Global Threat Intelligence to retrieve category information for a particular URL
TimeForRegex	Average time (in millseconds) spent for Regex processing in a transaction
WebCacheDiskUsage	Percentage of disk space that is currently used by the web cache
WebCacheHits	Number of objects that were requested and found in the web cache
WebCacheMisses	Number of objects that were requested and not found in the web cache
WebCacheObjectsCount	Number of objects in the web cache

REST interface

An interface is provided for Web Gateway that allows you to administer an appliance without being logged on to the standard user interface. This alternative interface is known as the REST (Representational State Transfer) interface.

Using the REST interface, you can perform various kinds of activities on a particular appliance or on others that are connected to it.

- **Actions** — Turn off an appliance, restart it, flush the cache, create a configuration backup, and perform several other activities
- **File handling** — Access system, log, and troubleshooting files to perform activities such as downloading, modifying, or deleting
- **Policy configuration** — Configure settings for engines and rule actions, manage rule sets and lists by performing activities such as enabling, adding, deleting, exporting, or importing
- **Updates** — Perform manual engine updates and trigger automatic yum and engine updates

Running a suitable script is the usual way to perform these activities.

Prepare use of the REST interface

To let users work with the REST interface, you need to enable it on the standard user interface of an appliance and permit access to it.

Enable use of the interface

You can enable the use of the REST interface for completing administration activities on an appliance.

Task

1. Select Configuration → Appliances.
2. On the appliances tree, select the appliance you want to administer using the REST interface and click User Interface.
3. Under UI Access, select Enable REST interface over HTTP or Enable REST interface over HTTPS as needed.
4. Click Save Changes.

Give permission to access the interface

You must add permission to access the REST interface to an administrator role for those users who are supposed to work with the interface.

Task

1. Select Accounts → Administrator Accounts.
2. In the Roles area, select an administrator role and click Edit.

   The Edit Role window opens.
3. Select REST interface accessible.
4. Click OK to close the window.
5. Click Save Changes.

Results

You can now assign this administrator role to the appropriate users.

Instead of adding access permission to an existing role, you can also create a new role with this permission and name it, for example, `REST Admin`.

Working with the REST interface

When working with the REST interface that is provided for a Web Gateway appliance, you send requests to this interface to have particular activities completed.

You can send single HTTP or HTTPS requests that are immediately processed or run these requests in a script, for example, in a bash script. The latter is the typical use.

You can also send requests for completing activities on other appliances that are connected as nodes in a Central Management cluster to the appliance where you are working

Requests are sent using a client of the appliance, which in turn provides a server for processing the requests and sending responses. You are assigned a particular amount of work space on this server. For some types of changes, you also need to send a commit request.

As this client, you can- use a data transfer tool, for example, *curl* (Client for URLs).

Before you can send requests for completing any activities, you must log on to the REST interface, authenticate and receive a session ID.

Note: The REST interface is provided in a particular format known as the ATOM format.

Sample script for sending a request

The following is an example of a bash script that sends a request to the REST interface using curl. The purpose of the request is to create a configuration backup.

The script does basically the following:

- Logs on and authenticates to the REST interface on an appliance
- Sends a request to create a backup file
- Logs off again

The script also uses a variable for the URL that is specified in the request for logging on to the REST interface. The variable is set at the beginning.

Note: When a sample command or a script with commands is shown in this documentation, a command can extend over two or more lines. When working with the REST interface, you must enter any command completely within a single line.

```
#!bin/bash ## Set URL variable for access to REST interface REST="http://localhost:4711/Konfigurator/REST" ##
Log on and authenticate curl -c cookies.txt -H "Authorization: Basic YWRtaW46d2ViZ2F0ZXdheQ==" -X POST "$REST/
login" ## Create backup file curl -b cookies.txt -X POST "$REST/backup" -o filename.backup ## Log off again curl
-b cookies.txt -X POST "$REST/logout"
```

Using curl as the data transfer tool

To send requests to the REST interface on an appliance, you can use curl as the data transfer tool.

A request sent with curl usually has three main parts: the `curl` command, one or multiple options, and a URL.

For example, in the following backup request, the `curl` command appears with the `-b` option for sending cookies that have been collected in a text file and the `-o` option, which stores the output of the request in another file. The `-X` option is for the request method.

```
curl -b cookies.txt -X POST "$REST/backup" -o filename.backup
```

The URL is specified as a variable that has the IP address, port number, and other information needed for access to the REST interface on an appliance as its value. It is followed by the name of the activity that is to be performed.

Using these and other options of curl together with the appropriate URLs, you can send requests to the REST interface on an appliance to perform activities as needed.

The curl data transfer tool is available under Linux and other UNIX operating systems and described in full detail, for example, on the *curl* man page.

Request methods

The request method is specified in curl by the `-X` option. When working with the REST interface on an appliance, the GET, POST, PUT, and DELETE methods can be used, for example, as follows:

```
curl -X POST <URL>
```

If no request method is specified, GET is the default method.

Headers

When a header is sent with a request, it is specified by the `-H` option, for example, as follows:

```
curl -H " <header name>:<header value>" -X POST <URL>
```

You can send multiple headers within one request, repeating the `-H` option letter before each header.

```
curl -H "<header name 1>:<header value 1>" -H "<header name 2>:<header value 2>" -X POST <URL>
```

A request normally includes an `Accept` header that has `application/atom+xml` as its value. In curl, `Accept: */*` is sent as a default, which is accepted by the REST interface, so you can leave out this header in many cases.

However, if you send data in the body of a request, you must include the `Content-Type` header with `application/atom+xml` as its value. You must also include the `Content-Length` header and set it correctly. The latter is done in curl by default, so you need not do it explicitly when using this tool.

If you want to include the header of the response that you receive upon a request in its output, you must insert the `-i` option.

```
curl -i -c cookies.txt -H "Authorization: Basic YWRtaW46d2ViZ2F0ZXdheQ==" -X POST "$REST/login"
```

The `-v` option creates verbose output, which means that the request header is included.

URLs

A URL in a request specifies a protocol, which can be HTTP or HTTPS in communication with the REST interface, the IP address or host name and the port number of the appliance that a request is sent to, and the internal path on the appliance to the REST interface.

This is followed by the name of the activity that should be performed and further parameters if there are any.

As the REST interface is located within the configurator subsystem of an appliance, the internal name of this subsystem, which is *Konfigurator*, appears in the URL.

A URL in, for example, a logon request, could therefore look as follows:

```
curl -X POST "http://localhost:4711/Konfigurator/$REST /login?userName=myusername&pass=mypassword"
```

In this request, the URL also has query parameters for the logon credentials. Query parameters are introduced by a `?` (question mark) and separated by an `&` (ampersand), as shown. A URL can also have matrix parameters, which are introduced by a `;` (semicolon).

For correct URL encoding, spaces in a URL must be filled with the symbols `%20`. So, for example, `Bob Smith` becomes `Bob %20Smith`.

You can use a variable within a URL for easier code writing and reading. For example, if you have set the `$REST` variable accordingly, the above request could look as follows:

```
curl -X POST "$REST/login?userName=myusername&pass=mypassword"
```

Sending data in the request body

For sending data in the body of a request, the `-d` option is used, followed by the name of the file that contains the data.

```
curl -b cookies.txt -X POST -d @file.txt "$REST/list?name=newlist&type=string"
```

If you are sending only binary data, the option to use is `- - data-binary`.

```
curl -b cookies.txt --data-binary @file.backup -X POST "$REST/restore" -H "Content-Type: text/plain;
charset=UTF-8"
```

You can use the `@` symbol after the option name to indicate a file name.

Authenticating to the interface

Before you can use the REST interface to perform any activities on an appliance you need to authenticate.

To authenticate, you submit user name and password in the logon request that you send to the REST interface.

There are two ways to submit them:

- Using query parameters
- Using an authentication header

After a successful authentication, the response contains the session ID, which you must include in each of your following requests.

Using query parameters for authentication

You can submit your credentials with query parameters that you add to the URL in your logon request.

```
curl -i -X POST "$REST/login?userName=myusername&pass=mypassword"
```

Using an authentication header

You can also use the Basis Access Authentication method to authenticate, which requires that you submit your credentials in an authentication header.

```
curl -i -H "Authorization: Basic YWRtaW46d2ViZ2F0ZXdheQ==" -X POST "$REST/login"
```

In the authentication header, the string after `Authorization: Basic` is the Base64-encoded representation of your user name and password.

Session ID

The session ID is sent to you in the response to your logon request. A session ID looks, for example, like this:

```
D0EFF1F50909466159728F28465CF763
```

It is either contained in the response body:

```
<entryxmlns="http://www.w3.org/2005/Atom"> <contenttype="text">D0EFF1F50909466159728F28465CF763</content></
entry>
```

or in a Set-Cookie header:

```
Set-Cookie: JSESSIONID=D0EFF1F50909466159728F28465CF763
```

In the requests of the sessions that follow the logon request, you must include the session ID as `JSESSIONID`.

For easier code writing and reading, you can set a variable to the value of the ID and use it for including the ID.

```
export SESSIONID=D0EFF1F50909466159728F28465CF763
```

You can append the ID as a matrix parameter to the URL, preceded by a semicolon.

```
curl -i "$REST/appliances;jsessionid=$SESSIONID"
```

Alternatively, you can send the ID in a Cookie header.

```
curl -i -H "Cookie: JSESSIONID=$SESSIONID" "$REST/appliances"
```

The `-c` option in curl allows you to collect all cookies in a text file, which is then sent with subsequent requests.

```
curl -i -c cookies.txt -H "Authorization: Basic YWRtaW46d2ViZ2F0ZXdheQ==" -X POST "$REST/login"
```

For sending a cookie file with a request, the `-b` option is used:

```
curl -i -b cookies.txt "$REST/appliances"
```

Requesting resources

A request sent to the REST interface regarding system files, log files, lists, and some other items is considered to be a request for resources.

The response to a request for resources can be one of the following:

- **Entry** — An entry delivers information in xml format about an individual resource, such as its ID, name, or the URL that can be used to access it
- **Feed** — A feed delivers information in xml format about a collection of resources.

  A feed can, for example, be a list of appliances that are available as nodes in a Central Management configuration, or a list of all lists that exist on an appliance, or a list of all lists of a particular type.

- **Binary data** — Binary data is delivered in a file that you requested for downloading.

A response can also be empty. This is the case when the requested data is not available.

Reducing xml data overhead

You can reduce the xml data overhead that you receive with a response, by including an appropriate `Accept` header in a request for resources. For this purpose, the header value must be `application/mwg+xml`.

Instead of an entry in the normal Atom format, you will then receive only the xml data from the content part of that format.

Instead of a feed in Atom format, you will only receive a list of IDs for the resources you asked for.

Similarly, you can reduce xml data overhead when working with the resources, for example, when modifying them. For this, you need to set the `Content-Type` header to `application/mwg+xml`.

Paging a feed

When requesting a feed, you can use paging, which means you can ask for a feed that is divided into pages.

Paging information is specified by query parameters that are added to the URL in a request. The following two parameters can be used:

- **pageSize** — Maximum number of elements on a page
- **page** — Page number

A request for a feed that uses paging could look as follows:

```
curl -i -b cookies.txt "$REST/list?pageSize=10&page=4"
```

If a feed is, for example, a list of 35 lists, the `pageSize` parameter in the above request divides it up into four pages, three of which contain ten lists, while the last one contains only five. The last page is also the one that is delivered.

Navigating within a feed

To allow navigation within a feed, the xml file that you receive contains appropriate links.

Using these links, you can go to the current, next, previous, first, and last page, respectively.

Performing basic REST operations

When working with the REST interface, you can perform several basic operations such as logging on and off, committing changes, and creating a configuration backup.

Use a POST request for these operations and specify each operation by a parameter that you add to the URL of the request.

For example, this is a request to log off from the REST interface on an appliance, using a curl command:

Note: When a sample command or script with commands is shown in this documentation, a command can extend over two or more lines. When working with the REST interface, you must enter any command completely within a single line.

```
curl -i -b cookies.txt -X POST "$REST/logout"
```

Parameters for basic REST operations include:

- `login` — Log on
- `logout` — Log off
- `heartbeat` — Keep a session alive
- `commit` — Commit changes
- `discard` — Discard changes
- `backup` — Back up the configuration
- `restore` — Restore the configuration
- `updateEngines` — Update the filter engines

Logging on

Request format:

URL/login?userName=<user name>&pass=<password>

To log on to the REST interface on an appliance, use the `login` parameter. Within the request, you also submit your credentials for authentication. If authentication is successful, the response to the logon request provides a session ID.

Sample command:

```
curl -i -X POST "$REST/login?userName=myusername&pass=mypassword"
```

Request parameters:

Parameter	Type	Description
userName	String	User name submitted for authenticating to the REST interface Default: None
pass	String	Password submitted for authenticating to the REST interface Default: None

Logging off

Request format:

URL/logout

To log off from the REST interface on an appliance, use the `logout` parameter. Logging off deletes the session information and discards the changes made in a session that have not been committed.

Sample command:

```
curl -i -X POST "$REST/logout"
```

Request parameters:

None

Keeping a session alive

Request format:

URL/heartbeat

Using the `heartbeat` parameter in a request keeps the current session alive.

Sample command:

```
curl -i -b cookies.txt -X POST "$REST/heartbeat"
```

Request parameters:

None

Committing changes

Request format:

URL/commit

To commit changes that have been made to items such as system files, log files, and lists on an appliance, use the `commit` parameter.

Sample command:

```
curl -i -b cookies.txt -X POST "$REST/commit"
```

Request parameters:

None

Discarding changes

Request format:

URL/discard

To discard changes that have been made to items such as system files, log files, and lists on an appliance, use the `discard` parameter.

Sample command:

```
curl -i -b cookies.txt -X POST "$REST/discard"
```

Request parameters:

None

Backing up the configuration

Request format:

URL/backup or **URL/backup?password=string** or **URL/backup?backUpMAS=boolean&password=string**

To create a configuration backup for the appliance where you are currently working, use the `backup` parameter. When backing up or restoring a configuration, no response header is required as part of the output, so the `-i` option is not included in the command for performing the request.

The configuration backup is stored in the file that is specified as output in the command.

Sample commands:

```
curl -b cookies.txt -X POST "$REST/backup" -o filename.backup or: curl -b cookies.txt -X POST "$REST/backup?
password=yourpassword" -o filename.backup or: curl -b cookies.txt -X POST "$REST/backup?
backUpMAS=true&password=yourpassword" -o filename.backup
```

Request parameters:

Parameter	Type	Description
backUpMAS	Boolean	If true, a password is used to encrypt the backup. Default: false
password	String	Password used to encrypt the backup. If no password is specified, the backup is not encrypted. Default: None

Restoring the configuration

Request format:

URL/restore?fullrestore=boolean&restoreMAS=boolean&password=string

To restore the configuration of the appliance you are currently working on, use the `restore` parameter. You must also specify a Content-Type header for the type of the backup file.

Sample command:

```
curl -b cookies.txt --data-binary @filename.backup -X POST "$REST/restore" -H "Content-Type: text/
plain;charset=UTF-8"
```

Request parameters:

Parameter	Type	Description
fullrestore	Boolean	If true, a configuration is restored completely. If false, only the policy is restored, omitting the system settings for an appliance. Default: false
restoreMAS	Boolean	If true, MAS is also restored if available. Default: false
password	String	Password used to decrypt the backup. If no password is specified, the backup is not decrypted.

Updating the filter engines

Request format:

URL/updateEngines

To perform an update of the filter engines on an appliance with data that is provided offline, use the *updateEngines* parameter. You must also specify a Content-Type header for the type of the update file.

Sample command:

```
curl -b cookies.txt --data-binary @mwg7-linux-mix-small.upd -X POST "$REST/updateEngines" -H "Content-Type:
text/plain;charset=UTF-8"
```

Request parameters:

None

Sample script for performing basic REST operations

The following bash script performs several basic operations: logging on and authenticating to the REST interface on an appliance, creating a backup file, and logging off again.

Before these operations are performed, the script sets a URL variable for accessing the REST interface.

```
#!/bin/bash ## Set URL variable for accessing REST interface REST=http://localhost:4711/Konfigurator/REST ## Log
on and authenticate curl -i -c cookies.txt -X POST "$REST/login?userName=myUserName&pass=myPassword" ## Create
backup file curl -b cookies.txt -X POST "$REST/backup" -o filename.backup ## Log off again curl -b cookies.txt -
X POST "$REST/logout"
```

Requesting version information

You can request the version of the REST interface for the Web Gateway appliance where you are currently working, as well as the version of the standard user interface.

You can request both versions at once or each version separately.

Requesting the interface versions

Request format:

URL/version

To request version information, use the `version` parameter. When used without further specification, the response to this request is a feed with the version numbers of both interfaces in XML format.

Sample commands:

```
curl -i -b cookies.txt .X GET "$REST/version"
```

Request parameters:

None

Requesting the version of a particular interface

Request formats:

URL/version/mwg-rest

URL/version/mwg-ui

To request version information for a particular interface, the interface name is added to the *version* parameter. The response to this request is a feed with the respective version number in XML format.

Sample commands:

```
curl -i -b cookies.text .X GET "$REST/version/mwg-rest" curl -i -b cookies.text .X GET "$REST/version/mwg-ui"
```

Request parameters:

Parameter	Type	Description
name	String	Name of the interface that version information is requested for

Parameter	Type	Description
		Default: None

Sample script for requesting version information

The following bash script requests version information for both the REST interface and the standard user interface of the appliance where you are currently working.

Before specifying this request, the script sets a URL variable for accessing the REST interface.

```
#!/bin/bash ## Set URL variable for accessing REST interface REST=http://localhost:4711/Konfigurator/REST ## Log
on and authenticate curl -c cookies.txt -H "Authorization: Basic YWRtaW46d2ViZ2F0ZXdheQ==" -X POST "$REST/login"
## Request version information curl -b cookies.txt -X GET "$REST/version" ## Log off again curl -b cookies.txt -
X POST "$REST/logout"
```

Working on appliances

After logging on to the REST interface on one appliance, you can complete activities on any other appliance that is connected. You can also complete some activities on all appliances at once.

An activity that is completed on all appliances at once is, for example, to import a license for all of them.

Individual appliances are identified in requests by their *UUIDs (Universal Unique Identifiers)*

The UUID of an appliance looks like this:

081EEDBC-7978-4611-9B96-CB388EEFC4BC

To find out about the UUID of an appliance, you can request a feed of all appliances that are connected as nodes in a Central Management cluster to the one where you are currently working.

The feed that is sent in response to your request includes a list of the UUIDs for all nodes. You can then identify an appliance by its UUID and, for example, shut down this appliance.

Importing a license

Request format:

<URL>/appliances/license

To import a license onto the appliances that are connected in a Central Management cluster, use the `appliances` and `license` parameters in a request.

You also need to specify a Content-Type header for the type of the license file.

Sample command:

```
curl -i -b cookies.txt " -H "Content-Type: text/plain; charset=UTF-8" -X POST "$REST/appliances/license" --data-
binary @license.xml
```

Requesting a feed with UUIDs for all appliances

Request format:

<URL>/appliances

To request a feed with a list of UUIDs for all appliances that run as nodes in a cluster, use the `appliances` parameter.

Sample command:

```
curl -i -b cookies.txt -X GET "$REST/appliances"
```

Request parameters:

None

Requesting a page in a feed for all appliances

Request format:

<URL>/appliances?page=<int>&pageSize=<int>

To request a particular page in a feed for all appliances that run as nodes in a cluster, use the *appliances* parameter with a page parameter appended. You can also append another parameter to request a particular page size.

Sample command:

```
curl -i -b cookies.txt -X GET "$REST/appliances?page=3&pageSize=2"
```

Request parameters:

Parameter	Type	Description
page	Integer	Number of the page in a feed for all appliances Default: 1
pageSize	Integer	Size of the page in a feed for all appliances Default: –1

Actions

Actions that are performed on individual appliances are specified in a request by their names, which are preceded by the `action` parameter.

These actions do not involve a modification of resources and are performed instantly, which means no request to commit them is required.

You can perform the following actions:

- `restart` — Restart an appliance
- `shutdown` — Shut down an appliance
- `flushcache` — Flush the cache
- `rotateLogs` — Rotate log files
- `rotateAndPushLog`s — Rotate and push log files
- `license` — Import a license

Restarting an appliance

Request format:

<URL>/<UUID>/action/restart

To restart an appliance, use `restart` as the action name in a request.

Sample command:

```
curl -i -b cookies.txt -X POST "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /action/restart"
```

Request parameters:

Parameter	Type	Description
UUID	String	UUID of the appliance that you want to restart Default: None

Shutting down an appliance

Request format:

<URL>/<UUID>/action/shutdown

To shut down an appliance, use `shutdown` as the action name in a request.

Sample command:

```
curl -i -b cookies.txt -X POST "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /action/shutdown"
```

Request parameters:

Parameter	Type	Description
UUID	String	UUID of the appliance that you want to shutdown

Parameter	Type	Description
		Default: None

Flushing the cache

Request format:

<URL>/<UUID>/action/flushcache

To flush the cache on an appliance, use `flushcache` as the action name in a request.

Sample command:

```
curl -i -b cookies.txt -X POST "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /action/flushcache"
```

Request parameters:

Parameter	Type	Description
UUID	String	UUID of the appliance where you want to flush the cache Default: None

Rotating log files

Request format:

<URL>/<UUID>/action/rotateLogs

To rotate the log files on an appliance, use `rotateLogs` as the action name in a request.

Sample command:

```
curl -i -b cookies.txt -X POST "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /action/rotateLogs"
```

Request parameters:

Parameter	Type	Description
UUID	String	UUID of the appliance where you want to rotate log files Default: None

Rotating and pushing log files

Request format:

<URL>/<UUID>/action/rotateAndPushLogs

To rotate the log files on an appliance and push them to a remote server, use *rotateAndPushLogs* as the action name in a request.

Sample command:

```
curl -i -b cookies.txt -X POST "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /action/rotateAndPushLogs"
```

Request parameters:

Parameter	Type	Description
UUID	String	UUID of the appliance where you want to rotate and push log files Default: None

Sample script for working on individual appliances

The following bash script rotates logs as an example for completing an action on an individual appliance.

Before this action is completed, the script sets a URL variable for accessing the REST interface.

```
#!/bin/bash ## Set URL variable for accessing REST interface REST=http://localhost:4711/Konfigurator/REST ## Log
on and authenticate curl -i -c cookies.txt -X POST "$REST/login?userName=myUserName&pass=myPassword" ## Rotate
```

```
log files curl -b cookies.txt -X POST "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /action/rotateLogs"
## Log off again curl -b cookies.txt -X POST "$REST/logout"
```

Working with configurations and settings

Using the REST interface, you can work with the configurations that were created for Web Gateway individual appliances and appliance clusters. You can also work with the settings for the filter modules on the appliances.

Identifying configurations and settings

When working with configuration and settings, you must identify them in the commands to complete particular activities.

The `cfgID` parameter serves as an identifier. Its value is usually the name of a file that contains an xml representation of a configuration or settings, for example, `com.scur.engine.coaching.configuration`.

You can look up the identifiers in the feed that you receive when retrieving a configuration or settings. In the entries of the feed, identifiers are tagged differently for configurations and settings.

- **Identifier for a configuration** — Last string of the section tagged as `<id>` in an entry, immediately following the `/cfg/` section.

  For example, `com.scur.engine.cmclusternode.configuration` is the identifier for a cluster configuration in this entry:

  ```
  <entry><id>7284F2DE-BE26-0CC9-E825-000000468CBE/cfg/com.scur.engine.cmclusternode .configuration</
  id><title>Central Management</title><type>com.scur.engine.cmclusternode</type> <link href="http://localhost:
  4711/Konfigurator/REST/appliances/7284F2DE-BE26-0CC9 -E825-000000468CBE/configuration/
  com.scur.engine.cmclusternode.configuration" rel="self"/></entry>
  ```

  The `<title>` section serves as an alternative identifier. In the above example, `com.scur.engine.cmclusternode` also identifies the cluster configuration.

- **Identifier for settings** — Tagged as `<id>` in an entry.

  For example, `com.scur.mainaction.block.11376` is an identifier for settings in this entry:

  ```
  <entry><id>com.scur.mainaction.block.11376</id><title>Unknown Certificate Authorities</
  title><type>com.scur.mainaction.block</type><link href="http://localhost:4711 Konfigurator/REST/setting/
  com.scur.mainaction.block.11376" rel="self"/></entry>
  ```

  The filter module that the settings belong to is tagged as `<title>`.

  In this example, the `com.scur.mainaction.block.11376` settings are configured for the module that filters unknown certificate authorities.

Retrieving a configuration or settings

Request formats

<URL>/appliances/<UUID>/configuration

<URL>/appliances/<UUID>/configuration?page=<int>&pageSize=<int>

<URL>/appliances/<UUID>/configuration?type=<string>&name=<string>

<URL>/appliances/<UUID>/configuration?type=<string>&name=<string>&page=<int>&pageSize=<int>

<URL>/cluster/configuration

<URL>/cluster/configuration?type=<string>&name=<string>&page=<int>&pageSize=<int>

<URL>/setting

<URL>/setting?type=<string>&name=<string>

You can retrieve these configurations and settings:

- Configuration for an appliance that is connected to the appliance where you are currently working
- Configuration for a cluster of appliances including the appliance where you are currently working
- Settings of the appliance where you are currently working

You can also retrieve configurations and settings for filter modules. You can retrieve all configurations and settings that exist for a module or only particular configurations and settings.

The response to this request is a feed with the configuration or settings in xml format. You can request a particular page of the feed and specify the page size.

Sample commands

Retrieve the configuration of an appliance:

```
curl -i -b cookies.txt -X GET "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /configuration"
```

Retrieve a cluster configuration:

```
curl -i -b cookies.txt -X GET "$REST/cluster/configuration"
```

Retrieve the settings of the appliance where you are currently working:

```
curl -i -b cookies.txt -X GET "$REST/setting"
```

Retrieve all settings that exist for a filter:

```
curl -i -b cookies.txt -X GET "$REST/setting?type=com.scur.engine.antivirus"
```

Retrieve particular filter settings:

```
curl -i -b cookies.txt -X GET "$REST /setting?type=com.scur.engine.antivirus&name=Gateway%20Anti-Malware"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
page	Integer	Number of a page in a feed with a configuration or settings Default: 1
pageSize	Integer	Size of a page in a feed Default: –1
type	String	Name of a filter module on an appliance, for example, `com.scur.engine.antivirus` You can look up the names of the filter modules in the feed that you receive when retrieving a configuration or settings. In this feed, filter names are tagged as `<type>`. Default: None
name	String	Name of particular settings for a filter module You can look up the settings names on the standard user interface of Web Gateway or in the feed that you receive when retrieving a configuration or settings. In this feed, settings names are tagged as `<title>`. **Note:** Spaces in the settings names must be filled with `%20`, for example, `Gateway%20Anti-Malware` Default: None

Retrieving an xml representation

Request formats

<URL>/appliances/<UUID>/configuration/<cfgID>

<URL>/cluster/configuration/<cfgID>

<URL>/setting/<cfgID>

You can retrieve a file in xml format that has been created on a Web Gateway appliance to represent a configuration or settings.

When retrieving an xml representation, you use an identifying string, also known as cfgID, to identify a configuration or settings.

The response to this request includes the xml representation in the response body.

Sample commands

Retrieve the xml representation of a configuration for an appliance:

```
curl -i -b cookies.txt -X GET "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /configuration/
com.scur.engine.coaching.configuration"
```

Retrieve the xml representation of a cluster configuration:

```
curl -i -b cookies.txt -X GET "$REST/cluster/configuration /com.scur.cm_cluster_global.internal.configuration"
```

Retrieve the xml representation of the settings for the appliance where you are currently working:

```
curl -i -b cookies.txt -X GET "$REST/setting/com.scur.mainaction.block.11376"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
cfgID	String	String used as an identifier for a configuration or settings This string is usually the name of a file that contains an xml representation of the configuration or settings, for example, `com.scur.engine.coaching.configuration`. Default: None

Modifying a configuration or settings

Request formats

<URL>/appliances/<UUID>/configuration/<cfgID>

<URL>/cluster/configuration/<cfgID>

<URL>/setting/<cfgID>

You can modify a configuration or settings by applying changes to the file that contains an xml representation of them.

When specifying the changes, you provide the name of the file that is to be changed and the name of the file that contains the changes. You must also specify a Content-Type header.

After specifying the changes, you must commit them using a separate command.

The response to this request includes the modified configuration or settings in the response body.

Sample commands

Modify the xml representation of a configuration for an appliance:

```
curl -i -b cookies.txt -H "Content-Type: application/xml" -d @changesToModifyConfiguration.xml -X PUT "$REST/
appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /configuration/com.scur.engine.coaching.configuration"
```

Modify the xml representation of a cluster configuration:

```
curl -i -b cookies.txt -X PUT -d @changesToModifyClusterConfiguration.xml -H "Content-Type: application/xml"
"$REST/cluster/configuration /com.scur.cm_cluster_global.internal.configuration"
```

Modify the xml representation of the settings for the appliance where you are currently working:

```
curl -i -b cookies.txt -H "Content-Type: application/xml" -d @changesToModifySettings.xml -X PUT "$REST/setting/
com.scur.mainaction.block.11376"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
cfgID	String	String used as an identifier for a configuration or settings This string is usually the name of a file that contains an xml representation of the configuration or settings, for example, `com.scur.engine.coaching.configuration`. Default: None

Retrieving a list of properties

Request formats

<URL>/appliances/<UUID>/configuration/<cfgID>/property

<URL>/appliances/<UUID>/configuration/<cfgID>/property?page=<int>&pageSize=<int>

<URL>/cluster/configuration/<cfgID>/property

<URL>/cluster/configuration/<cfgID>/property?page=<int>&pageSize=<int>

<URL>/setting/<cfgID>/property

<URL>/setting/<cfgID>/property?page=<int>&pageSize=<int>

You can retrieve a list with the properties of a configuration or settings.

When retrieving this list, you provide the name of the file with the xml representation that includes the properties.

The response to this request is a feed that includes the list in xml format. You can retrieve a particular page of this feed and specify the page size.

Sample commands

Retrieve a list of the properties in the configuration for an appliance:

```
curl -i -b cookies.txt -X GET "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /configuration/
com.scur.engine.coaching.configuration/property"
```

Retrieve a list of the properties in a cluster configuration:

```
curl -i -b cookies.txt -X GET "$REST/cluster/configuration /com.scur.cm_cluster_global.internal.configuration/
property"
```

Retrieve a list of the properties in the settings of the appliance where you are currently working:

```
curl -i -b cookies.txt -X GET "$REST/setting/com.scur.mainaction.block.11376/property"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
cfgID	String	String used as an identifier for a configuration or settings This string is usually the name of a file that contains an xml representation of the configuration or settings, for example, `com.scur.engine.coaching.configuration`.

Parameter	Type	Description
		Default: None
page	Integer	Number of a page in a feed for a configuration or settings Default: 1
pageSize	Integer	Size of a page in a feed Default: -1

Retrieving a property

Request formats

<URL>/appliances/<UUID>/configuration/<cfgID>/property/<propertyname>

<URL>/cluster/configuration/<cfgID>/property/<propertyname>

<URL>/setting/<cfgID>/property/<propertyname>

You can retrieve a property from a configuration or settings.

When retrieving this property, you provide its name and the name of the file with the xml representation that includes the property.

The response to this request includes the property in xml format.

Sample commands

Retrieve a property from a configuration for an appliance:

```
curl -i -b cookies.txt -X GET "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /configuration/
com.scur.engine.coaching.configuration/property/sendsync"
```

Retrieve a property from a cluster configuration:

```
curl -i -b cookies.txt -X GET "$REST/cluster/configuration /com.scur.cm_cluster_global.internal.configuration/
property/DataUsageStatementVersionAccepted"
```

Retrieve a property from the settings of the appliance where you are currently working:

```
curl -i -b cookies.txt -X GET "$REST/setting/com.scur.mainaction.block.11376/property /TemplateName"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
cfgID	String	String used as an identifier for a configuration or settings This string is usually the name of a file that contains an xml representation of the configuration or settings, for example, `com.scur.engine.coaching.configuration.` Default: None
propertyname	String	Name of a property in a configuration or settings Default: None

Modifying a property value

Request formats

<URL>/appliances/<UUID>/configuration/<cfgID>/property/<propertyname>

<URL>/cluster/configuration/<cfgID>/property/<propertyname>

<URL>/setting/<cfgID>/property/<propertyname>

You can modify the value of a property in a configuration or settings.

When modifying this value, you provide the name of the file with the property that has its value changed and the name of the file with the new value. You must also specify a Content-Type header.

After modifying the value, you must commit this change using a separate command.

The response to this request includes the modified property value in xml format.

Sample commands

Modify the value of a property in a configuration for an appliance:

```
curl -i -b cookies.txt -H "Content-Type: application/xml" -d @changesToModifyProperty.xml -X PUT "$REST/
appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC/configuration /com.scur.engine.coaching.configuration/property/
sendsync"
```

Modify the value of a property in a cluster configuration:

```
curl -i -b cookies.txt -H "Content-Type: application/xml" -d @changesToModifyProperty.xml -X PUT "$REST/cluster/
configuration/com.scur.cm_cluster_global.internal.configuration /property/DataUsageStatementVersionAccepted"
```

Modify the value of a property in the settings of the appliance where you are currently working:

```
curl -i -b cookies.txt -H "Content-Type: application/xml" -d @changesToModifyProperty.xml -X PUT "$REST/setting/
com.scur.mainaction.block.11376/property/TemplateName"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
cfgID	String	String used as an identifier for a configuration or settings This string is usually the name of a file that contains an xml representation of the configuration or settings, for example, `com.scur.engine.coaching.configuration`. Default: None
propertyname	String	Name of a property in a configuration or settings Default: None

Retrieving the default value of a property

Request formats

<URL>/appliances/<UUID>/configuration/<cfgID>/property/<propertyname>/default

<URL>/cluster/configuration/<cfgID>/property/<propertyname>/default

<URL>/setting/<cfgID>/property/<propertyname>/default

You can retrieve the default value of a property from a configuration or settings.

When retrieving this value, you provide the property name and the name of the file with the xml representation that includes the property.

The response to this request includes the default value in xml format.

Sample commands

Retrieve the default value of an individual property in a configuration for an individual appliance.

```
curl -i -b cookies.txt -X GET "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC/configuration /
com.scur.engine.coaching.configuration/property/sendsync/default"
```

Retrieve the default value of a property in a cluster configuration.

```
curl -i -b cookies.txt -X GET "$REST/cluster/configuration /com.scur.cm_cluster_global.internal.configuration/
property/DataUsageStatementVersionAccepted /default"
```

Retrieve the default value of a property in the settings of the appliance where you are currently working.

```
curl -i -b cookies.txt -X GET "$REST/setting/com.scur.mainaction.block.11376/property /TemplateName/default"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
cfgID	String	String used as an identifier for a configuration or settings This string is usually the name of a file that contains an xml representation of the configuration or settings, for example, `com.scur.engine.coaching.configuration.` Default: None
propertyname	String	Name of a property in a configuration or settings Default: None

Setting a property to its default value

Request formats

<URL>/appliances/<UUID>/configuration/<cfgID>/property/<propertyname>/default

<URL>/cluster/configuration/<cfgID>/property/<propertyname>/default

<URL>/setting/<cfgID>/property/<propertyname>/default

You can set a property to its default value in a configuration or settings.

When setting this value, you provide the property name and the name of the file with the xml representation that includes the property.

The response to this request includes the property set to its default value in xml format.

Sample commands

Set a property to its default value in a configuration for an appliance:

```
curl -i -b cookies.txt -X POST "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /configuration/
com.scur.engine.coaching.configuration/property/sendsync/default"
```

Set a property to its default value in a cluster configuration:

```
curl -i -b cookies.txt -X POST "$REST/cluster/configuration /com.scur.cm_cluster_global.internal.configuration /
property/DataUsageStatementVersionAccepted/default"
```

Set a property to its default value in the settings of the appliance where you are currently working:

```
curl -i -b cookies.txt -X POST "$REST/setting/com.scur.mainaction.block.11376/property /TemplateName/default"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None

Parameter	Type	Description
cfgID	String	String used as an identifier for a configuration or settings This string is usually the name of a file that contains an xml representation of the configuration or settings, for example, `com.scur.engine.coaching.configuration.` Default: None
propertyname	String	Name of a property in a configuration or settings Default: None

Retrieving a list with the actions of a configuration

Request format

<URL>/appliances/<UUID>/configuration/<cfgID>/action

You can retrieve a list with the actions that are executed according to the rules of a configuration for an appliance.

When retrieving this list, you provide the name of the file with the xml representation that includes the actions.

The response to this request is a feed that includes the list in xml format.

Sample command

Retrieve a list with the actions in a configuration for an appliance:

```
curl -i -b cookies.txt -X GET "$REST/appliances/7284F2DE-BE26-0CC9-E825-0000004A637C /configuration/
com.scur.engine.cmclusternode.configuration/action"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an individual appliance
cfgID	String	String used as an identifier for a configuration or settings This string is usually the name of a file that contains an xml representation of the configuration or settings, for example, `com.scur.engine.coaching.configuration.` Default: None

Triggering an action

Request format

<URL>/appliances/<UUID>/configuration/<cfgID>/action/*actionname*

You can trigger an action that is included in a configuration for an appliance.

When triggering this action, you provide its name and the name of the file with the xml representation that includes the action.

Sample command

Trigger an action in a configuration for an appliance:

```
curl -i -b cookies.txt -X POST "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /configuration/
com.scur.engine.cmclusternode/action/update_engines_all_standalone"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an individual appliance
cfgID	String	String used as an identifier for a configuration or settings This string is usually the name of a file that contains an xml representation of the configuration or settings, for example, `com.scur.engine.coaching.configuration`. Default: None
actionname	String	Name of an action that is included in the xml representation of a configuration or settings Default: None

Deleting settings

Request format

<URL>/setting/<cfgID>

You can delete settings for the appliance where you are currently working.

When deleting these settings, you provide the name of the file with the xml representation that includes these settings.

After deleting the settings, you must commit this change using a separate command.

The response to this request includes the deleted settings in xml format.

Sample command

Delete settings for the appliance where you are currently working:

```
curl -i -b cookies.txt -X DELETE "$REST/setting/com.scur.mainaction.block.11376"
```

Variable request parameters

Parameter	Type	Description
cfgID	String	String used as an identifier for a configuration or settings This string is usually the name of a file that contains an xml representation of the configuration or settings, for example, `com.scur.engine.coaching.configuration`. Default: None

Adding settings with content

Request format

<URL>/setting?type=<string>&name=<string>

You can add new settings for a filter module on the appliance where you are currently working.

When adding these settings, you provide the names of the filter module and the settings.

You must also specify a Content-Type header and append a file in xml format that provides the content of the settings.

After adding the settings, you must commit this change using a separate command.

The response to this request includes the added settings in xml format.

Sample command

Add new settings for a filter module on the appliance where you are currently working:

```
curl -i -b cookies.txt -H "Content-Type: application/xml" -d @newSettingwithoutNameAndType.xml -X POST "$REST/
setting?type=com.scur.mainaction.block&name=Malware%20Detected"
```

Variable request parameters

Parameter	Type	Description
type	String	Name of a filter module on an appliance, for example, `com.scur.engine.antivirus` You can look up the names of the filter modules in the feed that you receive when retrieving a configuration or settings. In this feed, filter names are tagged as `<type>`. Default: None
name	String	Name of new settings that are added for a filter module. **Note:** Spaces in the settings names must be filled with `%20`, for example, `Gateway%20New%20Anti-Malware`. Default: None

Adding settings with content including type and name

Request format

<URL>/setting

You can add new settings for a filter module on the appliance where you are currently working with the names of the filter module and the settings already included in the settings content.

When adding settings in this way, you do not provide the names of the filter module and the settings as request parameters. You specify a Content-Type header and append a file in xml format that provides the content of the settings.

After adding the settings, you must commit this change using a separate command.

The response to this request includes the added filter settings in xml format.

Sample command

Add new settings for a filter module on the appliance where you are currently working with content including the filter and settings names:

```
curl -i -b cookies.txt -H "Content-Type: application/xml" -d @newSettingWithNameAndType.xml -X POST "$REST/
setting"
```

Variable request parameters

None

Adding new default settings

Request format

<URL>/setting?type=<string>&name=<string>

You can add new settings for a filter module on the appliance where you are currently working with default values configured for all settings options.

When adding these settings, you provide the name of the filter module and a name for the new default settings.

After adding the settings, you must commit this change using a separate command.

The response to this request includes the new settings with their default values in xml format.

Sample command

Add new settings with default values for a filter module on the appliance where you are currently working:

```
curl -i -b cookies.txt -X POST "$REST /setting?type=com.scur.mainaction.block&name=New%20Blocking"
```

Variable request parameters

Parameter	Type	Description
type	String	Name of a filter module on an appliance, for example, `com.scur.engine.antivirus` You can look up the names of the filter modules in the feed that you receive when retrieving a configuration or settings. In this feed, filter names are tagged as `<type>`. Default: None
name	String	Name of new default settings that are added for a filter module **Note:** Spaces in the settings names must be filled with `%20`, for example, `Gateway%20New%20Default%20Anti-Malware`. Default: None

Sample script for working with configurations and settings

The following bash script modifies the value of a property in a configuration for an appliance.

Before performing this operation, the script sets a URL variable for accessing the REST interface.

Some of the operations performed with configurations and settings require running still another command to commit a change that you requested, for example, modifying a property value. When a commit is required, this is mentioned in the description of the operation.

```
#!/bin/bash ## Set URL variable for accessing REST interface REST=http://localhost:4711/Konfigurator/REST ## Log
on and authenticate curl -i -c cookies.txt -X POST "$REST/login?userName=myUserName&pass=myPassword" ## Modify
property value curl -i -b cookies.txt -H "Content-Type: application/xml" -d @changesToModifyProperty.xml -X PUT
"$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC/configuration /com.scur.engine.coaching.configuration/
property/sendsync" ## Commit modification curl -b cookies.txt -X POST "$REST/commit" ## Log off curl -b
cookies.txt -X POST "$REST/logout"
```

Working with system files

You can use the REST interface to work with system files on any appliance in a cluster that includes the appliance where you are currently working.

Caution: Modifying system files inadequately can impact the proper operation of an appliance.

With system files, you can:

- Retrieve a list of system files
- Download a system file
- Modify a system file

  Note: When running an appliance in FIPS-compliant mode, you cannot modify system files.

Retrieving a list of system files

Request formats

<URL>/appliances/<UUID>/system

<URL>/appliances/<UUID>/system?page=<int>&pageSize=<int>

You can retrieve a list of system files from any appliance in a cluster that includes the appliance where you are currently working.

When retrieving a list of system files from an appliance, you append the `appliances` parameter, the appliance ID, and the system parameter.

You can request a particular page of the feed and specify the page size.

In response to your request, you receive an xml file with all system files listed.

Sample command

Retrieve a list of system files from an appliance in a cluster that includes the appliance where you are currently working:

```
curl -i -b cookies.txt -X GET "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC/system"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
page	Integer	Number of a page in a feed with a list of system files Default: 1
pageSize	Integer	Size of a page in a feed Default: –1

Downloading a system file

Request format

<URL>/appliances/<UUID>/system/<file name>

<URL>/appliances/<UUID>/system/<path>/<file name>

You can download a system file from an appliance in a cluster that includes the appliance where you are currently working.

When downloading a system file from an appliance, you append the `appliances` parameter and the appliance ID, as well as the path to the system file if there is any and the file name. You must also specify an Accept header.

Using the `-o` parameter, you can store the downloaded system file as a local file under its name on the appliance where you downloaded it from.

Sample command

Download a system file from an appliance in a cluster that includes the appliance where you are currently working.

```
curl -b cookies.txt -H "Accept: application/x-download" -X GET "$REST/appliances/081EEDBC-7978-4611-9B96-
CB388EEFC4BC/system/hosts" -o hosts.txt
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
<path>	String	Path to a system file that is downloaded Default: None
<file name>	String	Name of a system file that is downloaded Default: None

Modifying a system file

Request format

<URL>/appliances/<UUID>/system/<file name>

<URL>/appliances/<UUID>/system/<path>/<file name>

You can modify a system file on an appliance in a cluster that includes the appliance where you are currently working.

Note: When running an appliance in FIPS-compliant mode, you cannot modify system files.

When modifying a system file on an appliance, you append the appliances parameter and the appliance ID, as well as the path to the system file if there is any and the file name.

You must also specify a Content-Type header and append a file with the modified data in binary format.

After modifying the system file, you must commit this change using a separate command.

Sample command

Modify a system file on an appliance in a cluster that includes the appliance where you are currently working.

```
curl -b cookies.txt -H "Content-Type: */*" -X PUT "$REST/appliances /081EEDBC-7978-4611-9B96-CB388EEFC4BC/
system/hosts" -d @hosts.txt
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
<path>	String	Path to a system file that is modified Default: None
<file name>	String	Name of a system file that is modified Default: None

Sample script for working with system files

The following bash script modifies a system file on an appliance in a cluster that includes the appliance where you are currently working.

Before performing this operation, the script sets a URL variable for accessing the REST interface.

Modifying a system file requires running a separate command to commit the modification. Other operations, for example, retrieving a list of system files, do not require a commit.

```
#!/bin/bash ## Set URL variable for accessing REST interface REST=http://localhost:4711/Konfigurator/REST ## Log
on and authenticate curl -i -c cookies.txt -X POST "$REST/login?userName=myUserName&pass=myPassword" ## Modify
system file curl -i -b cookies.txt -H "Content-Type: */* -X PUT "$REST/appliances /081EEDBC-7978-4611-9B96-
CB388EEFC4BC/system/hosts" -d @hosts.txt ## Commit modification curl -b cookies.txt -X POST "$REST/commit" ##
Log off curl -b cookies.txt -X POST "$REST/logout"
```

Working with log files

You can use the REST interface to work with log files on any appliance in a cluster that includes the appliance where you are currently working.

When working with log files, you include the `appliances` parameter and identify an appliance in all requests.

You also append the `log` parameter and other parameters as needed to complete particular activities with log files.

With log files, you can:

- Retrieve a list of log files
- Download a log file
- Delete a log file

Retrieving a list of log files

Request formats

<URL>/appliances/<UUID>/log

<URL>/appliances/<UUID>/log?page=<int>&pageSize=<int>

You can retrieve a list of log files on an appliance in a cluster that includes the appliance where you are currently working.

You can request a particular page of the feed that you receive in response and specify the page size.

The feed provides MIME type information in XML format, indicating for every list item whether it is a log file or a directory.

- *"application/x-download"* — log file
- *"application/atom+xml; type=feed"* — directory

Sample command

Retrieve a list of log files on an appliance in a cluster that includes the appliance where you are currently working:

```
curl -i -b cookies.txt -X GET "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC/log"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
page	Integer	Number of a page in a feed with a list of log files Default: 1
pageSize	Integer	Size of a page in a feed Default: –1

Downloading a log file

Request format

<URL>/appliances/<UUID>/log/<subresources>

You can download a log file from an appliance in a cluster that includes the appliance where you are currently working.

When downloading a log file, you provide its path and name. You must also specify an Accept header.

Using the `-O` parameter, you can store the downloaded log file as a local file under its name on the appliance where you downloaded it from.

Sample command

Download a log file from an appliance in a cluster that includes the appliance where you are currently working.

```
curl -b cookies.txt -H "Accept: application/x-download" -X GET "$REST/appliances/081EEDBC-7978-4611-9B96-
CB388EEFC4BC/log/debug/debug_1234.log" -O
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
subresources	String	Path to a file that is downloaded and file name Default: None

Deleting a log file

Request format

<URL>/appliances/<UUID>/log/<subresources>

You can delete a log file on an appliance in a cluster that includes the appliance where you are currently working.

When deleting a log file, you provide its path and name.

Sample command

Delete a log file on an appliance in a cluster that includes the appliance where you are currently working.

```
curl -i -b cookies.txt -X DELETE "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /log/debug/
debug_1234.log"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
subresources	String	Path to a file that is deleted and file name Default: None

Sample script for working with log files

The following bash script deletes a log file on an appliance in a cluster that includes the appliance where you are currently working.

Before performing this operation, the script sets a URL variable for accessing the REST interface.

Log file operations do not require a commit.

```
#!/bin/bash ## Set URL variable for accessing REST interface REST=http://localhost:4711/Konfigurator/REST ## Log
on and authenticate curl -i -c cookies.txt -X POST "$REST/login?userName=myUserName&pass=myPassword" ## Delete
log file curl -i -b cookies.txt -X DELETE "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /log/debug/
debug_1234.log" ## Log off curl -b cookies.txt -X POST "$REST/logout"
```

Working with files uploaded for troubleshooting

You can use the REST interface to work with files that have been uploaded to an appliance for troubleshooting.

On the standard Web Gateway interface, you can perform this upload using the Troubleshooting top-level menu.

When working with uploaded files on the REST interface, you identify an appliance and append the `files` parameter.

With files uploaded for troubleshooting, you can:

- Retrieve a list of uploaded files
- Download an uploaded file
- Add a file to the uploaded files
- Modify an uploaded file
- Delete an uploaded file

Retrieving a list of uploaded files

Request formats

<URL>/appliances/<UUID>/files

<URL>/appliances/<UUID>/files?page=<int>&pageSize=<int>

You can retrieve a list of uploaded files on an appliance in a cluster that includes the appliance where you are currently working.

You can request a particular page of the feed that contains the list and specify the page size.

Sample commands

Retrieve a list of uploaded files from an appliance:

```
curl -i -b cookies.txt -X GET "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /files"
```

Request a particular page of the feed with the list and specify the page size:

```
curl -i -b cookies.txt -X GET "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /files?page=4&pageSize=1"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
page	Integer	Number of a page in a feed with uploaded files Default: 1
pageSize	Integer	Size of a page in a feed Default: –1

Downloading an uploaded file

Request format

<URL>/appliances/<UUID>/files/<filename>

You can download an uploaded file from an appliance in a cluster that includes the appliance where you are currently working.

When downloading an uploaded file, you specify an Accept header. Using the `-O` parameter, you can store the downloaded data in a local file under its name on the appliance where you downloaded it from.

Sample command

Download an uploaded file from an appliance:

```
curl -b cookies.txt -H "Accept: application/x-download" -X GET "$REST/appliances /081EEDBC-7978-4611-9B96-
CB388EEFC4BC/files/troubleshooting.zip" -O
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
filename	String	Name of an uploaded file Default: None

Adding a file to the uploaded files

Request format

<URL>/appliances/<UUID>/files/<filename>

You can add a file to the uploaded files on an appliance in a cluster that includes the appliance where you are currently working.

When adding a file, you provide its name and append the file in binary format as the request body.

You must also specify a Content-Type header. Do not specify `application/x-www-form-urlencoded`, as the curl tool already appends this type.

Sample command

Add a file to the uploaded files on an appliance:

```
curl -i -b cookies.txt -H "Content-Type: */*" -X POST "$REST/appliances /081EEDBC-7978-4611-9B96-CB388EEFC4BC/
files/moreTroubleshooting.zip" --data-binary @moreTroubleshooting.zip
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
filename	String	Name of a file that is added to the uploaded files on an appliance Default: None

Modifying an uploaded file

Request format

<URL>/appliances/<UUID>/files/<filename>

You can modify an uploaded file on an appliance in a cluster that includes the appliance where you are currently working.

When modifying an uploaded file, you provide its name and append the file it in binary format with the data required for the modification as the request body.

You must also specify a Content-Type header. Do not specify `application/x-www-form-urlencoded`, as the curl tool already appends this type.

Sample command

Modify an uploaded file on an appliance:

```
curl -i -b cookies.txt -H "Content-Type: */*" -X PUT "$REST/appliances /081EEDBC-7978-4611-9B96-CB388EEFC4BC/
files/moreTroubleshooting.zip" --data-binary @modificationData.zip
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
filename	String	Name of a file that is modified Default: None

Deleting an uploaded file

Request format

<URL>/appliances/<UUID>/files/<filename>

You can delete an uploaded file on an appliance in a cluster that includes the appliance where you are currently working.

When deleting an uploaded file, you provide its name.

Sample command

Delete an uploaded file on an appliance:

```
curl -i -b cookies.txt -X DELETE "$REST/appliances/081EEDBC-7978-4611-9B96-CB388EEFC4BC /files/
troubleshooting.zip"
```

Variable request parameters

Parameter	Type	Description
UUID	String	Universally unique identifier for an appliance Default: None
filename	String	Name of a file that is deleted

Parameter	Type	Description
		Default: None

Sample script for working with uploaded files

The following bash script modifies an uploaded file on an appliance.

Before performing this operation, the script sets a URL variable for accessing the REST interface.

```
#!/bin/bash ## Set URL variable for accessing REST interface REST=http://localhost:4711/Konfigurator/REST ## Log
on and authenticate curl -i -c cookies.txt -X POST "$REST/login?userName=myUserName&pass=myPassword" ## Modify
uploaded file curl -i -b cookies.txt -H "Content-Type:*/*" -X PUT "$REST/appliances /081EEDBC-7978-4611-9B96-
CB388EEFC4BC/files/moreTroubleshooting.zip" --data-binary @modificationData.zip ## Log off curl -b cookies.txt -
X POST "$REST/logout"
```

Working with lists

You can use the REST interface to work with lists and their entries on the appliance where you are currently working.

When working with lists, you include the `list` parameter in all requests. For list entries, you add the `entry` parameter.

With lists, you can:

- Retrieve a list of lists
- Retrieve a list
- Add a list with content
- Add a list with content including type and name
- Add an empty list
- Modify a list
- Rename a list
- Copy a list
- Delete a list

With list entries, you can:

- Retrieve a list of list entries
- Retrieve a list entry
- Insert a list entry
- Modify a list entry
- Move a list entry
- Delete a list entry

Retrieving a list of lists

Request formats

<URL>/list

<URL>/list?page=<int>&pageSize=<int>

<URL>/list?type=<string>

<URL>/list?type=<string>&page=<int>&pageSize=<int>

<URL>/list?name=<string>

<URL>/list?name=<string>&page=<int>&pageSize=<int>

<URL>/list?type=<string>&name=<string>

<URL>/list?type=<string>&name=<string>&page=<int>&pageSize=<int>

You can retrieve a list of all lists from the appliance where you are currently working.

You can also specify a list type, for example, string, to retrieve a list that includes only lists of this type. Similarly, you can use a list name to retrieve all lists with this name. You can combine these requests.

You can request a particular page of the feed that is returned and specify the page size.

The xml file in the feed provides a list ID for each list. This ID allows you to identify a list that you want to work with.

Sample commands

Retrieve a list of all lists on the appliance where you are currently working:

```
curl -i -b cookies.txt -X GET "$REST/list"
```

Retrieve a list of all string lists on the appliance where you are currently working:

```
curl -i -b cookies.txt -X GET "$REST/list?type=string"
```

Retrieve a list of all lists named "default" on the appliance where you are currently working:

```
curl -i -b cookies.txt -X GET "$REST/list?name=default"
```

Variable request parameters

Parameter	Type	Description
type	String	List type, which can be: • category • ip • iprange • mediatype • number • regex • string Default: None
name	String	Name of a list, for example, `default` Default: None
page	Integer	Number of a page in a feed with a list of lists Default: 1
pageSize	Integer	Size of a page in a feed Default: –1

Retrieving a list

Request format

<URL>/list/<list ID>

You can retrieve a list from the appliance where you are currently working.

When retrieving this list, you provide its ID.

The response to this request includes the retrieved list in xml format.

Sample command

Retrieve a list:

```
curl -i -b cookies.txt -X GET "$REST/list/com.scur.type.regex.4537"
```

Variable request parameter

Parameter	Type	Description
list ID	String	List identifier, for example, `com.scur.type.regex.4537` Default: None

Adding a list with content

Request format

<URL>/list?type=<string>&name=<string>

You can add a list with content to the lists on the appliance where you are currently working.

When adding this list, you provide its type and name and append it in xml format as the request body. You must also specify a Content-Type header.

After adding the list, you must commit this change using a separate command.

The response to this request includes the added list in xml format as the response body.

Sample command

Add a list with content:

```
curl -i -b cookies.txt -H "Content-Type: application/xml" -d @categoryListWithContent.xml -X POST "$REST/list?
type=category&name=newlist"
```

Variable request parameters

Parameter	Type	Description
type	String	List type, which can be: • category • ip • iprange • mediatype • number • regex • string Default: None
name	String	Name of an added list, for example, `newlist` Default: None

Adding a list with content including type and name

Request format

<URL>/list

You can add a list with content that includes its type and name to the lists on the appliance where you are currently working.

When adding a list in this way, you do not provide a type and name, but simply append the list in xml format as the request body. You must also specify a Content-Type header.

After adding the list, you must commit this change using a separate command.

The response to this request includes the added list in xml format.

Sample command

Add a list with content that includes type and name:

```
curl -i -b cookies.txt -H "Content-Type: application/xml" -d @listWithTypeAndNameInside.xml -X POST "$REST/list"
```

Request parameters

None

Adding an empty list

Request format

<URL>/list?type=<string>&name=<string>

You can add an empty list to the lists on the appliance where you are currently working.

When adding this list, you provide its type and name.

After adding the list, you must commit this change using a separate command.

The response to this request includes the empty list in xml format.

Sample command

Add an empty list:

```
curl -i -b cookies.txt -X POST "$REST/list?type=ip&name=emptyiplist"
```

Variable request parameters

Parameter	Type	Description
type	String	List type, which can be: • category • ip • iprange • mediatype • number • regex • string Default: None
name	String	Name of a list, for example, `emptylist` Default: None

Modifying a list

Request format

<URL>/list/<list ID>

You can modify a list on the appliance where you are currently working.

When modifying this list, you provide its ID and append the modified content in xml format as the request body. You must also specify a Content-Type header.

After modifying the list, you must commit this change using a separate command.

The response to this request includes the modified list in xml format.

Sample command

Modify a list:

```
curl -i -b cookies.txt -H "Content-Type: application/xml" -d @modifiedContent.xml -X PUT "$REST/list/
com.scur.type.regex.4537"
```

Variable request parameter

Parameter	Type	Description
list ID	String	List identifier, for example, `com.scur.type.regex.4537` Default: None

Renaming a list

Request format

<URL>/list/<list ID>/rename?name=<string>

You can rename a list on the appliance where you are currently working.

When renaming this list, you provide its ID and the new name for the list.

After renaming the list, you must commit this change using a separate command.

The response to this request includes the renamed list in xml format.

Sample command

Rename a list:

```
curl -i -b cookies.txt -X POST "$REST/list/com.scur.type.regex.4537/rename?name=newname"
```

Variable request parameters

Parameter	Type	Description
list ID	String	List identifier, for example, `com.scur.type.regex.4537` Default: None
name	String	Name of a renamed list, for example, `newname` Default: None

Copying a list

Request format

<URL>/list/<list ID>/copy?name=<string>

You can copy a list on the appliance where you are currently working.

When copying this list, you provide its ID and a name for the copied list.

After copying the list, you must commit this change using a separate command.

The response to this request includes the copied list in xml format.

Sample command

Copy a list:

```
curl -i -b cookies.txt -X POST "$REST/list/com.scur.type.regex.4537/copy?name=newname"
```

Variable request parameters

Parameter	Type	Description
list ID	String	List identifier, for example, `com.scur.type.regex.4537` Default: None
name	String	Name of a copied list, for example, `othername` Default: None

Deleting a list

Request format

<URL>/list/<list ID>

You can delete a list on the appliance where you are currently working.

When deleting a list, you provide its ID.

Sample command

Delete a list:

```
curl -i -b cookies.txt -X DELETE "$REST/list/com.scur.type.regex.4537"
```

Variable request parameter

Parameter	Type	Description
list ID	String	List identifier, for example, `com.scur.type.regex.4537` Default: None

Retrieving a list of list entries

Request formats

<URL>/list/<list ID>/entry

<URL>/list/<list ID>/entry?page=<int>&pageSize=<int>

You can retrieve a list of the entries in a list on the appliance where you are currently working.

When retrieving this list, you provide the ID of the list that contains the entries.

You can request a particular page of the feed that is returned and specify the page size.

The feed provides a list of the entries in xml format with a number for each entry to indicate its position. This number allows you to identify an entry that you want to access.

Sample command

Retrieve the entries of a list on the appliance where you are currently working:

```
curl -i -b cookies.txt -X GET "$REST/list/com.scur.type.regex.4537/entry"
```

Variable request parameters

Parameter	Type	Description
list ID	String	List identifier, for example, `com.scur.type.regex.4537` Default: None
page	Integer	Number of a page in a feed with list entries Default: 1
pageSize	Integer	Size of a page in a feed Default: –1

Retrieving a list entry

Request format

<URL>/list/<list ID>/entry/<position>

You can retrieve an entry from a list on the appliance where you are currently working.

When retrieving an entry, you provide the list ID and the position of the entry in the list.

The response to this request includes the entry in xml format.

Sample command

Retrieve an entry in a list:

```
curl -i -b cookies.txt -X GET "$REST/list/com.scur.type.regex.4537/entry/2"
```

Variable request parameters

Parameter	Type	Description
list ID	String	List identifier, for example, `com.scur.type.regex.4537` Default: None
position	Integer	Position of an entry in a list The position is specified as i –1. For example, for the third entry, specify 2. Default: None

Inserting a list entry

Request format

<URL>/list/<list ID>/entry/<position>/insert

You can insert an entry in a list on the appliance where you are currently working.

When inserting an entry, you provide the list ID and the position for the entry in the list.

You must also specify a Content-Type header and append the entry in xml format as the request body.

After inserting the entry, you must commit this change using a separate command.

The response to this request includes the inserted entry in xml format.

Sample command

Insert an entry in a list:

```
curl -i -b cookies.txt -H "Content-Type: application/xml" -d @newEntry.xml -X POST "$REST/list/
com.scur.type.regex.4537/entry/1/insert"
```

Variable request parameters

Parameter	Type	Description
list ID	String	List identifier, for example, `com.scur.type.regex.4537` Default: None
position	Integer	Position for an entry in a list The position is specified as i – 1. For example, to insert an entry in third position, specify 2. Default: None

Modifying a list entry

Request format

<URL>/list/<list ID>/entry/<position>

You can modify an entry in a list where you are currently working.

When modifying this entry, you provide the list ID and the position of the entry in the list.

You must also specify a Content-Type header and append the modified content in xml format as the request body.

After modifying the entry, you must commit this change using a separate command.

The response to this request includes the modified entry in xml format.

Sample command

Modify an entry in a list:

```
curl -i -b cookies.txt -H "Content-Type: application/xml" -d @modifiedEntry.xml -X PUT "$REST/list/
com.scur.type.regex.4537/entry/3"
```

Variable request parameters

Parameter	Type	Description
list ID	String	List identifier, for example, `com.scur.type.regex.4537` Default: None
position	Integer	Position of an entry in a list The position is specified as i – 1. For example, when modifying the third entry, specify 2. Default: None

Moving a list entry

Request format

<URL>/list/<list ID>/entry/<position>/move?newpos=<int>

You can move an entry in a list on the appliance where you are currently working.

When moving this entry, you provide the list ID and the old and new positions of the entry in the list.

After moving the entry, you must commit this change using a separate command.

The response to this request includes the entry on its new position in xml format.

Sample command

Move an entry in a list:

```
curl -i -b cookies.txt -X POST "$REST/list/com.scur.type.regex.4537/entry/4/move?newpos=1"
```

Variable request parameters

Parameter	Type	Description
list ID	String	List identifier, for example, `com.scur.type.regex.4537` Default: None
position	Integer	Position of an entry in a list The position is specified as i – 1. For example, when moving an entry from second position, specify 1. Default: None
newpos	Integer	New position of an entry in a list The position is specified as i – 1. For example, to move an entry to fourth position, specify 3. Default: None

Deleting a list entry

Request format

<URL>/list/<list ID>/entry/<position>

You can delete an entry from a list on the appliance where you are currently working.

When deleting this entry, you provide the list ID and the position of the entry in the list.

Sample command

Delete an entry in a list:

```
curl -i -b cookies.txt -X DELETE "$REST/list/com.scur.type.regex.4537/entry/0"
```

Variable request parameters

Parameter	Type	Description
list ID	String	List identifier, for example, `com.scur.type.regex.4537` Default: None
position	Integer	Position of an entry in a list The position is specified as i – 1. For example, to delete the third entry, specify 2. Default: None

Sample script for working with lists

The following bash script modifies an entry in a list on the appliance you are currently working on.

Before performing this operation, the script sets a URL variable for accessing the REST interface.

Some list operations require a commit, for example, modifying a list entry. When a commit is required for an operation, this is mentioned in the description.

```
#!/bin/bash ## Set URL variable for accessing REST interface REST=http://localhost:4711/Konfigurator/REST ## Log
on and authenticate curl -i -c cookies.txt -X POST "$REST/login?userName=myUserName&pass=myPassword" ## Modify
list entry curl -i -b cookies.txt -H "Content-Type: application/xml" -d @modifiedEntry.xml -X PUT "$REST/list/
com.scur.type.regex.4537/entry/3" ## Commit modification curl -b cookies.txt -X POST "$REST/commit" ## Log off
curl -b cookies.txt -X POST "$REST/logout"
```

Working with rule sets

You can use the REST interface to work with rule sets and their rules on the appliance where you are currently working.

When working with rule sets, you include the `rulesets` parameter in all requests.

With rule sets, you can:

- Retrieve a list of rule sets
- Retrieve a rule set
- Export rule sets
- Export a rule set
- Import a rule set into a top-level position
- Import a rule set into a nested position
- Enable a rule set
- Disable a rule set
- Move a rule set
- Delete a rule set

Retrieving a list of rule sets

Request formats

<URL>/rulesets?topLevelOnly=<Boolean>

<URL>/rulesets?topLevelOnly=<Boolean>&page=<int>&pageSize=<int>

You can retrieve a list of all rule sets on the appliance where you are currently working.

When retrieving this list, set the value of the `topLevelOnly` parameter to `true` to include only top-level rule sets. Rule sets that are nested in a top-level rule set are not shown on the list.

You can request a particular page of the feed that is returned and specify the page size.

The feed provides a list in xml format with an ID for each rule set. This ID allows you to identify a rule set that you want to work with.

Sample commands

Retrieve a list of all rule sets on the appliance where you are currently working:

```
curl -i -b cookies.txt -X GET "$REST/rulesets?topLevelOnly=false"
```

Retrieve a list of all top-level rule sets on the appliance where you are currently working, omitting nested rule sets:

```
curl -i -b cookies.txt -X GET "$REST/rulesets?topLevelOnly=true"
```

Variable request parameters

Parameter	Type	Description
topLevelOnly	Boolean	If true, only top-level rule sets are shown on a list of rule sets. Otherwise, also nested rule sets are shown. Default: false
page	Integer	Number of a page in a feed with a list of rule sets

Parameter	Type	Description
		Default: 1
pageSize	Integer	Size of a page in a feed Default: –1

Retrieving a rule set

Request formats

<URL>/rulesets/rulegroups/<rule set ID>

<URL>/rulesets/rulegroups/<rule set ID>/successor?topLevelOnly=<Boolean>

<URL>/rulesets/rulegroups/<rule set ID>/successor?topLevelOnly=<Boolean>&page=<int>&pageSize=<int>

You can retrieve a rule set on the appliance where you are currently working.

When retrieving a rule set, you append the `rulegroups` parameter in addition to the `rulesets` parameter and provide the rule set ID. You can look up rule set IDs in the feed that is returned when you retrieve a list of all rule sets.

You can retrieve a rule set with all rule sets following it on the rule sets tree by appending the `successor` parameter.

To retrieve a top-level rule set without nested rule sets, set the value of the `topLevelOnly` parameter to `true`.

You can request a particular page of the feed that is returned and also specify the page size.

The feed provides the rule set in xml format with nested rule sets and successors according to what you specified.

Sample commands

Retrieve an individual rule set on the appliance where you are currently working:

```
curl -i -b cookies.txt -X GET "$REST/rulesets/rulegroups/5234"
```

Retrieve an individual rule set on the appliance where you are currently working, including its successors and nested rule sets:

```
curl -i -b cookies.txt -X GET "$REST/rulesets/rulegroups/5234/successor?topLevelOnly=false"
```

Variable request parameters

Parameter	Type	Description
rule set ID	Integer	Rule set identifier, for example, `5234` Rule set IDs are included in the feed that is returned when you retrieve a list of all rule sets. Default: None
topLevelOnly	Boolean	If true, only top-level rule sets are shown on a list of rule sets Otherwise, also nested rule sets are shown. Default: false
page	Integer	Number of a page in a feed with a list of rule sets Default: 1
pageSize	Integer	Size of a page in a feed Default: –1

Exporting rule sets

Request format

<URL>/rulesets/export

You can export all rule sets that exist on the appliance where you are currently working. The exported data includes all rules, lists, settings, and properties pertaining to each of the rule sets.

When exporting rule sets, you append the `export` parameter in addition to the `rulesets` parameter.

The exported data is stored in a file with the name that you provide using the `-o` parameter.

The feed that is returned contains the exported data in xml format under the file name that you provided.

Sample command

Export all rule sets that exist on the appliance where you are currently working:

```
curl -b cookies.txt -X POST "$REST/rulesets/export" -o rulesetInMWGLibraryXMLForm.xml
```

Variable request parameters

None

Exporting a rule set

Request format

<URL>/rulesets/rulegroups/<rule set ID>/export

You can export a rule set from the appliance where you are currently working. The exported data includes all rules, lists, settings, and properties pertaining to the rule set. Nested rule sets are also included.

When exporting a rule set, you append the `export` and `rulegroups` parameters in addition to the rulesets parameter and provide the rule set ID. You can look up rule set IDs in the feed that is returned when you retrieve a list of all rule sets.

The exported data is stored in a file with the name that you provide, using the *-o* parameter.

The xml file that you receive as a feed in response to your request contains the exported data under the file name that you provided.

Sample command

Export an individual rule set that exists on the appliance where you are currently working:

```
curl -b cookies.txt -X POST "$REST/rulesets/rulegroups/5234/export" -o rulesetInMWGLibraryXMLForm.xml
```

Variable request parameter

Parameter	Type	Description
rule set ID	Integer	Rule set identifier, for example, `5234` Rule set IDs are included in the feed that is returned when you retrieve a list of all rule sets. Default: None

Importing a rule set into a top-level position

Request formats

<URL>/rulesets/import

<URL>/rulesets/import?position=<int>&autoResolveBy=<string>

You can import an individual rule set into a top-level position of the rule set system on the appliance where you are currently working. The imported data includes all rules, lists, settings, and properties pertaining to the rule set.

When importing a rule set into a top-level position, you append the import `parameter` in addition to the `rulesets` parameter.

You can specify a position for the rule set and the method of resolving conflicts with existing configuration items, for example, lists and settings. If you do not specify a position or method, default values are applied.

You must also specify a Content-Type header and provide the name of the xml file that contains the rule set data, using the `-d` parameter.

The feed that is returned contains the imported rule set data in xml format.

Sample command

Import a rule set into a top-level position on the appliance where you are currently working:

```
curl -i -b cookies.txt H "Content-Type: application/xml" -d @rulesetInMWGLibraryXMLForm.xml -X POST "$REST/
rulesets/import?position=3&autoResolveBy=copy"
```

Variable request parameters

Parameter	Type	Description
position	Integer	Number of the position that an imported top-level rule set takes in the rule set system The rule set system can be viewed on the standard Web Gateway interface. Numbering begins with 0, which means if you specify 0, the rule set is imported into first position among the existing top-level rule sets. If you specify –1 or nothing at all, the rule set is imported into the last position. Default: –1
autoResolveBy	String	Method used for resolving conflicts that might arise when a rule set is imported Conflicts arise when a rule set uses configuration items, such as lists or settings, that already exist in the rule set system on your appliance. The resolution method can be: • off — No conflict resolution is performed. • copy — If items used by the imported rule set already exist under the same names in the rule set system, they are copied and renamed. The renamed items are used by the imported rule set. • refer — If items used by the imported rule set already exist under the same names in the rule set system, the rule set uses these items. • auto — If items used by the imported rule set already exist under the same name in the rule set system, the refer method is tried first, then the copy method. Usually, this solves all conflicts. Default: off If not all conflicts could be solved by applying the selected method, an error message with code number 409 is sent in response to your request. The response body then includes the conflicting data.

Importing a rule set into a nested position

Request formats

<URL>/rulesets/rulegroups/<parent rule set ID>/import

<URL>/rulesets/rulegroups/<parent rule set ID>import?position=<int>&autoResolveBy=<string>

You can import a rule set into a nested position of the rule set system on the appliance where you are currently working. The imported data includes all rules, lists, settings, and properties pertaining to the rule set.

When importing a rule set into a nested position, you append the `import` and `rulegroups` parameters in addition to the `rulesets` parameter and provide the ID of the rule set that serves as parent of the nested rule set. You can look up rule set IDs in the feed that is returned when you retrieve a list of all rule sets.

You can specify a position among the nested rule sets for the imported rule set and the method of resolving conflicts with existing configuration items, for example, lists and settings. If you do not specify a position or method, default values are applied.

You must also specify a Content-Type header and provide the name of the xml file that contains the rule set data, using the `-d` parameter.

The feed that is returned contains the imported rule set data in xml format.

Sample command

Import a rule set into a nested position on the appliance where you are currently working:

```
curl -i -b cookies.txt H "Content-Type: application/xml" -d @rulesetInMWGLibraryXMLForm.xml -X POST "$REST/
rulesets/rulegroups/4224/import?position=0&autoResolveBy=auto"
```

Variable request parameters

Parameter	Type	Description
parent rule set ID	Integer	Rule set identifier for a parent rule set, for example, `422`4 Rule set IDs are included in the feed that is returned when you retrieve a list of all rule sets. Default: None
position	Integer	Number of the position that an imported rule set takes among the existing nested rule sets of its parent rule set. Numbering begins with 0, which means if you specify 0, the rule set is imported into first position among the existing nested rule sets. If you specify –1 or nothing at all, the rule set is imported into the last position. Default: –1
autoResolveBy	String	Method used for resolving conflicts that might arise when a rule set is imported Conflicts arise when a rule set uses configuration items, such as lists or settings, that already exist in the rule set system on your appliance. The resolution method can be: • off — No conflict resolution is performed. • copy — If items used by the imported rule set already exist under the same names in the rule set system, they are copied and renamed. The renamed items are used by the imported rule set. • refer — If items used by the imported rule set already exist under the same

Parameter	Type	Description
		names in the rule set system, the rule set uses these items. • auto — If items used by the imported rule set already exist under the same name in the rule set system, the refer method is tried first, then the copy method. Usually, this solves all conflicts. Default: off If not all conflicts could be solved by applying the selected method, an error message with code number 409 is sent in response to your request. The response body then includes the conflicting data.

Enabling a rule set

Request format

<URL>/rulesets/rulegroups/<rule set ID>/enable

You can enable a rule set on the appliance where you are currently working.

If this rule set is nested within a parent rule set, make sure this rule set is also enabled before enabling the nested rule set.

When enabling a rule set, you append the `enable` and `rulegroups` parameters in addition to the `rulesets` parameter and provide the rule set ID. You can look up rule set IDs in the feed that is returned when you retrieve a list of all rule sets.

You must also specify a Content-Type header.

After enabling the rule set, you must commit this change using a separate command.

Sample command

Enable a rule set on the appliance where you are currently working:

```
curl -i -b cookies.txt H "Content-Type: application/xml" -X POST "$REST/rulesets/rulegroups/1929/enable"
```

Variable request parameters

Parameter	Type	Description
rule set ID	Integer	Rule set identifier, for example, `5234` Rule set IDs are included in the feed that is returned when you retrieve a list of all rule sets. Default: None

Disabling a rule set

Request format

<URL>/rulesets/rulegroups/<rule set ID>/disable

You can disable a rule set on the appliance where you are currently working.

When disabling a rule set, you append the `disable` and `rulegroups` parameters in addition to the `rulesets` parameter and provide the rule set ID. You can look up rule set IDs in the feed that is returned when you retrieve a list of all rule sets.

You must also specify a Content-Type header.

After disabling the rule set, you must commit this change using a separate command.

Sample command

Disable a rule set on the appliance where you are currently working:

```
curl -i -b cookies.txt H "Content-Type: application/xml" -X POST "$REST/rulesets/rulegroups/1929/disable"
```

Variable request parameters

Parameter	Type	Description
rule set ID	Integer	Rule set identifier, for example, `5234` Rule set IDs are included in the feed that is returned when you retrieve a list of all rule sets. Default: None

Moving a rule set

Request formats

<URL>/rulesets/rulegroups/<rule set ID>/move?position=<int>

<URL>/rulesets/rulegroups/<rule set ID>/move?parentId=<int>&position=<int>

You can move a rule set to a different position within the rule set system on the appliance where you are currently working. You can move it to a top-level or a nested position.

When moving a rule set to a top-level position, you append the `move` and `rulegroups` parameters in addition to the `rulesets` parameter and provide the rule set ID. You can look up rule set IDs in the feed that is returned when you retrieve a list of all rule sets.

You must also provide a position number to specify the top-level position, for example, the first position among all existing top-level rule sets.

When moving a rule set to a nested position, you append the `move` and `rulegroups` parameters in addition to the `rulesets` parameter and provide the rule set IDs of the moved rule set and the rule set that serves as its parent.

You must also provide a position number to specify the position of the moved rule set among the existing nested rule sets of the parent rule set.

After moving the rule set, you must commit this change using a separate command.

Sample commands

Move a rule set to a top-level position on the appliance where you are currently working:

```
curl -i -b cookies.txt -X POST "$REST/rulesets/rulegroups/4224/move?position=2"
```

Move a rule set to a nested position on the appliance where you are currently working:

```
curl -i -b cookies.txt -X POST "$REST/rulesets/rulegroups/6326/move?parentId=2159&position=4"
```

Variable request parameters

Parameter	Type	Description
rule set ID	Integer	Rule set identifier, for example, `5234` Rule set IDs are included in the feed that is returned when you retrieve a list of all rule sets. Default: None
parentId	Integer	Rule set identifier for a parent rule set, for example, `4224` Default: None
position	Integer	Number of the position that an moved rule set takes among the existing top-level rule sets or nested rule sets of its parent rule set. Numbering begins with 0, which means if you specify 0, the rule set is moved

Parameter	Type	Description
		into first position among the existing top-level or nested rule sets. If you specify –1 or nothing at all, the rule set is imported into the last position. Default: –1

Deleting a rule set

Request format

<URL>/rulesets/rulegroups/<rule set ID>/delete

You can delete a rule set on the appliance where you are currently working.

When deleting a rule set, you append the `delete` and `rulegroups` parameters in addition to the `rulesets` parameter and provide the rule set ID. You can look up rule set IDs in the feed that is returned when you retrieve a list of all rule sets.

After deleting the rule set, you must commit this change using a separate command.

Sample command

Delete a rule set on the appliance where you are currently working:

```
curl -i -b cookies.txt H "Content-Type: application/xml" -X DELETE "$REST/rulesets/rulegroups/2037/delete"
```

Variable request parameters

Parameter	Type	Description
rule set ID	Integer	Rule set identifier, for example, `5234` Rule set IDs are included in the feed that is returned when you retrieve a list of all rule sets. Default: None

Sample script for working with rule sets

The following bash script deletes a rule set on the appliance you are currently working on.

Before performing this operation, the script sets a URL variable for accessing the REST interface.

```
#!/bin/bash ## Set URL variable for accessing REST interface REST=http://localhost:4711/Konfigurator/REST ## Log
on and authenticate curl -i -c cookies.txt -X POST "$REST/login?userName=myUserName&pass=myPassword" ## Delete
rule set curl -i -b cookies.txt -X DELETE "$REST/rulesets/rulegroups/com.scur.type.regex.4537/entry/3" ## Commit
deletion curl -b cookies.txt -X POST "$REST/commit" ## Log off curl -b cookies.txt -X POST "$REST/logout"
```

Third-party software

The following list provides information about third-party software used in developing the McAfee Web Gateway appliance software.

Third-party software list

Information on third-part software is provided in this list following the alphabetical order of names.

Apache Jakarta Commons IO

Used in portions

Made available under an Apache License, version 2.0

Copyright © 2002-2012 The Apache Software Foundation

Apache log4j

Used in portions

Made available under an Apache License, version 2.0

Copyright © 2007 The Apache Software Foundation.

Apache ORO

Used in portions

Made available under an Apache License, version 1.1

Copyright © 2002-2003 The Apache Software Foundation.

Apache Tomcat

Used in portions

Made available under an Apache License , version 2.0

Copyright © 2012 The Apache Software Foundation.

Apache-Jakarta Codec

Used in portions

Made available under an Apache License, version 2.0

Copyright © 2000-2009 The Apache Software Foundation

Apache-Jakarta Fileupload

Used in portions.

Made available under an Apache License, version 2.0

Copyright © 2002-2010 The Apache Software Foundation

Apache-Jakarta Lang

Used in portions

Made available under an Apache License, version 2.0

Copyright © 2001-2011 The Apache Software Foundation

Arabica XML and HTML Toolkit for C++

Made available under a Berkeley Software Distribution (BSD) License, version 2.0

Used in portions

Copyright © 2001-2013 Jez UK Ltd

ASM

Used in portions

Made available under a Berkeley Software Distribution (BSD) License, version 2.0

Copyright © 2000-2005 INRIA France Telekom.

Boost C++ Libraries

Used in portions

Made available under a Boost Software License, version 1.0

Copyright © miscelleaneous

Bzip2

Used in portions

Made available under a Bzip2 License

Copyright © 1996-2013 julian@bzip.org

Chromium Source

Used in portions

Made available under a Berkeley Software Distribution (BSD) License, version 2.0

Copyright © 2010

Code Project - Walking the callstack

Used in portions

Made available under a Berkeley Software Distribution (BSD) License, version 2.0

Copyright © 2005 Jochen Kalmbach

Dynamic Drive - DD Tooltip

Used in portions

Made available under a Dynamic Drive DHTML Scripts License

Copyright © 1998-2004 Dynamic Drive

Eclipse

Used in portions

Made available under an Eclipse Public License, version 1.0, and a Common Public License

Copyright © 2005-2009 Eclipse contributors and others

ftpparse

Used in portions

Made available under an Ftp Parse License

Copyright © 2000 D. J. Bernstein

fugue icons

Used in portions

Made available under a Creative Commons Attribution License, version 3.0

Copyright © 2013 Yusuke Kamiyamane

Glazed Lists

Used in portions

Made available under a Mozilla Public License, version 1.1

Copyright © 2003-2006 publicobject.com O'Dell Engineering Ltd

googletest

Used in portions

Made available under a Berkeley Software Distribution (BSD) License, version 2.0

Copyright © 2008 Google Inc

Info-ZIP project - source-UnZip

Used in portions

Made available under an Info-ZIP Updated License

Copyright © 1999-2005 Greg Roelofs

Jackson JSON Processor Core Annotations

Used in portions

Made available under an Apache License, version 2.0

Copyright © 2000-2005 INRIA France Telecom

jersey-bundle

Used in portions

Made available under a Common Development and Distribution License, version 1.0

Copyright © 2000-2005 INRIA France Telecom

JFreeChart

Used in portions

Made available under a GNU Lesser General Public License, version 2.1

Copyright © 2000-2009 Object Refinery Limited and Contributors

JIDE Common Layer

Used in portions

Made available under a GNU Lesser General Public License, version 2.1

Copyright © 2002-2011 JIDE Software, Inc

JSON

Used in portions

Made available from a public domain

jsprogressBarHandler

Used in portions

Made available under a Creative Commons Attribution Share-Alike License, version 2.5

Copyright © 2007 - 2008 Bram Van Damme

JSR-311 - JAX-RS - The Java API for RESTful Web Services

Used in portions

Made available under a Common Development and Distribution License, version 1.0

Copyright © 2009 Sun Microsystems, Inc

jQuery

Used in portions

Made available under a Massachusetts Institute for Technology (MIT) License

Copyright © 2005, 2013 jQuery Foundation, Inc

jQuery UI

Used in portions

Made available under a Massachusetts Institute for Technology (MIT) License

Copyright © 2013 jQuery Foundation and other contributors

Kerberos 5

Used in portions

Made available under a Kerberos 5 Massachusetts Institute of Technology (MIT) License

Copyright © 1985-2013 Massachusetts Institute of Technology and contributors

libuuid

Used in portions

Made available under a Theodore Ts'o License

Copyright © 1999 Theodore Ts'o

Mozilla Rhino JavaScript for Java

Used in portions

Made available under a Mozilla Public License, version 1.1

Copyright © 2012 Mozilla Foundation

msgpack

Used in portions

Made available under an Apache License, version 2.0

Copyright © 2004 The Apache Software Foundation

Open BSD

Used in portions

Made available under a Berkeley Software Distribution (BSD) License, version 2.0

Copyright © 1982, 1986, 1990, 1991, 1993 The Regents of the University of California

opencsv

Used in portions

Made available under an Apache License, version 2.0

Copyright © 2005 Bytecode Pty Ltd

OpenSSL

Used in portions

Made available under an OpenSSL License, version 1.1

Copyright © 1999-2011 The OpenSSL Project

Paho

Used in portions

Made available under an Eclipse Public License, version 1.0

POCO

Used in portions

Made available under a Boost Software License, version 1.0

Prototype JavaScript Framework

Used in portions

Made available under a Massachusetts Institute of Technology (MIT) License, version 2.0

Copyright © 2005-2010 Sam Stephenson

rapidjson

Used in portions

Made available under a Massachusetts Institute of Technology (MIT) License, version 2.0

Copyright © 2011 Milo Yip

RapidXml

Used in portions

Made available under a Massachusetts Institute of Technology (MIT) License, version 2.0

Copyright © 2008 2006, 2007 Marcin Kalicinski

RARLAB-UnRAR

Used in portions

Made available under an unRAR License

Copyright ©1993-2012

RDialog

Used in portions

Made available under a Ruby License

Copyright © 2007 Aleks Clarks

RegExFormatter Tutorial

Used in portions

Made available under a Creative Commons Attribution License, version 2.5

Copyright © 2008, 2010, Oracle and/or its affiliates

Ruby

Used in portions

Made available under a Ruby License, version 2.5

Copyright © 1995-2013 Yukihiro Matsumoto

Silk Icons

Used in portions

Made available under a Creative Commons Attribution License, version 2.5

Copyright © 2008 Mark James

StAX2

Used in portions

Made available under an Apache License, version 2.0

Copyright © 2004 Alexander Slominski 2006 Chris Fry

test/unit

Used in portions

Made available under a Ruby License, version 2.0

Copyright © 2000-2003, Nathaniel Talbott

The ASN.1 Compiler

Used in portions

Made available under a Berkeley Software Distribution (BSD) Two Clause License (BSD -) License, version 2.0

Copyright © 2003, 2004, 2005, 2006, 2007 Lev Walkin

The Legion of the Bouncy Castle

Used in portions

Made available under a Massachusetts Institute of Technology (MIT) License, version 2.0

Copyright © 2000-2012 2000 - 2012 The Legion Of The Bouncy Castle

The prefuse visualization toolkit

Used in portions

Made available under a Berkeley Software Distribution (BSD) License, version 2.0

Copyright © 2000-2012 Regents of the University of California

Trove for Java

Used in portions

Made available under a Massachusetts Institute of Technology (MIT) License, version 2.0

Copyright © 2000-2012 The Legion of the Bouncy Castle

Valgrind Instrumentation Framework

Used in portions

Made available under a Massachusetts Institute of Technology (MIT) License, version 2.0

Copyright © 2000-2012 The Legion of the Bouncy Castle

Woodstox

Used in portions

Made available under an Apache License, version 2.0

Copyright © 2000-2012 2004 Tatu Saloranta

XStream Library

Used in portions

Made available under a Berkeley Software Distribution (BSD) License, version 2.0

Copyright © 2003-2006 Joe Walnes 2006-2007 XStream Committers.

COPYRIGHT