Getting Started

- Overview, page 1
- Operation, page 1
- Choosing a Connector Mode and Authentication Key Type, page 2
- Deployment Scenarios, page 3
- Authentication Process, page 5
- Groups and Policy Application, page 7

Overview

This chapter helps you decide when to deploy Connector and the most appropriate deployment method for your network infrastructure. Connector is used to deliver web traffic from a client computer to Cisco Cloud Web Security. You may not need to use Connector. It is only required in the following configurations:

- When user-level granularity is required for policy and reporting.
- When accessing Cloud Web Security from a device with a dynamic IP address.
- When accessing Cloud Web Security off-site or with a roaming device without Cisco AnyConnect Secure Mobile Security installed.

Operation

Connector identifies users by merging their details from Active Directory using LDAP, or Windows Domain integration and an authentication key. With Connector, users with a dynamic IP address can connect to Cloud Web Security with a company, group, or user authentication key. Users with a static IP address do not require a key.

Connector encrypts user information which Cloud Web Security uses to apply specific user or group policy information. Connector passes user web traffic requests through Cloud Web Security for filtering, scanning, and policy enforcement, before providing the cleansed web content to the user.
Company, group, and user authentication keys are created in Cisco ScanCenter, the administrator portal to Cloud Web Security. These authentication keys enable Cloud Web Security to identify and authenticate a user. For more information, see the Cisco ScanCenter Administrator Guide.

Note

Group authentication keys provide more detailed user behavior reporting and policy management but may require additional key management by the administrator.

Choosing a Connector Mode and Authentication Key Type

Before installing Connector, use the following flow chart to decide the most appropriate Connector mode (standalone or enterprise) and authentication key type (company, group, or user) for your organization.
Deployment Scenarios

When you have chosen your appropriate solution and deployment method based on your network infrastructure, you are ready to proceed. The flow chart provides for a variety of deployment scenarios, but in practice, the three most common are:

- Company authentication key and Active Directory
- Group authentication key
- User authentication key

Company Authentication Key and Active Directory

The most common scenario uses a single company authentication key for all users in the organization and Active Directory to provide user and group granularity for policy and reporting. For example:

In this scenario, Policy A is applied to Mohan and Joe, while Policy B is applied to Louise. If a policy causes a block event for Mohan or Joe, it is registered against WinNT://.../Marketing with user information for Mohan or Joe. Block events for Louise are registered against WinNT://.../Engineering with her user information.

To create this configuration:

**Step 1** Create two Active Directory groups in ScanCenter:
- WinNT://.../Marketing
- WinNT://.../Engineering

**Step 2** Install Connector and the company authentication key on the Domain Controller.

**Step 3** Apply Policy A to the Marketing group and Policy B to the Engineering group.
**Group Authentication Key**

The next most common scenario uses a group authentication key without Active Directory. In this example, Connector is installed in standalone mode with a location-based key (LK), a unique group authentication key, in each of three branch offices:

In this scenario, Policy A is applied to Sinead and Joe, Policy B is applied to Adeola, and Policy C is applied to Yuki. If a policy causes a block event for Sinead or Joe, it is registered against NYGROUP group. Block events for Adeola are registered against the LDNGROUP group. Block events for Yuki are registered against TKYGROUP.

---

**Note**

This deployment method is suitable only when Active Directory is not used.

To create this configuration:

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Create a group authentication key for each location group: NYGROUP, LDNGROUP, and TKYGROUP.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 2</td>
<td>Install Connector in standalone mode and the relevant group authentication key at each location.</td>
</tr>
<tr>
<td>Step 3</td>
<td>Apply Policy A to group NYGROUP, Policy B to LDNGROUP, and Policy C to TKYGROUP.</td>
</tr>
</tbody>
</table>

---

**User Authentication Key**

This scenario provides an alternative off-site solution when it is not possible to deploy Cisco AnyConnect Secure Mobile Security or Passive Identity Management. For example, it could be used for non-Windows users. In the following example, Connector and a user authentication key are installed on each user’s computer.
In this scenario, Policy A is applied to Bob and Ling, and Policy B is applied to Isaac. If a policy causes a block event for Bob, Ling, or Isaac, it is registered against their user and custom group.

⚠️ Caution

The user authentication key overrides all reported user information. Therefore, you must deploy the key only once with a separate Connector installation for each computer.

To create this configuration:

Step 1
Create custom groups and assign users to those groups in ScanCenter.

Step 2
Create a user authentication key for each user in ScanCenter.

Step 3
Install Connector in standalone mode and a unique user authentication key on each user's computer.

Summary

When Active Directory is in use, a company key installed with Connector in standalone or enterprise mode on the Domain Controller provides user and group granularity for policy and reporting.

If you have satellite offices where you want to apply group policy, you should use a group authentication key.

For portable computers or remote connections where Cisco AnyConnect Secure Mobile Security or Passive Identity Management are not viable, a user authentication key and a local installation of Connector in standalone mode should be used.

Authentication Process

Connector uses one of several possible authentication resources to annotate web requests with end-user data. The supported data sources are:

- Active Directory using the LDAP protocol
- Windows Domain (for example, using CIFS/SMB protocols)
- Authentication key
When a company authentication key is used in conjunction with either Active Directory or Windows Domain lookup, data needs to be merged. See Company Authentication Key and Active Directory, on page 3.

There are several ways to use the authentication key:

- As end-user identification (with Cisco AnyConnect Secure Mobile Security, where appropriate)
  - To control access to Cloud Web Security
  - To identify an organization
  - To identify groups within an organization
  - To identify users within groups

- As group identification (with Cisco AnyConnect Secure Mobile Security, where appropriate)
  - To control access to Cloud Web Security
  - To identify an organization
  - To identify groups

- As organization identification (in enterprise mode for dynamic IP access)
  - To control access to Cloud Web Security
  - To identify an organization

The authentication key has a dual purpose:

- To control access to Cloud Web Security (as opposed to using static IP lockdown)
- To provide some identification data

Data embedded in requests can be classified as:

- Service authentication data (optional, if you use static IP addresses)
  - Authentication key

- User identification data
  - Internal IP address
  - User name
  - Groups (for example, from AD or Windows Domain)

- Session data
  - Local time
  - Tallies

---

**Note**

Data is combined and transmitted securely with every web request in data headers.
Groups and Policy Application

The following sections describe how user data is derived, how groups work, and how policies are applied in ScanCenter.

Deriving User Data

ScanCenter derives user data from the maximum granularity available for a given request. The user name will be chosen from the first available data item:

- Connector-supplied user name either from Active Directory or from user authentication key (if both, Connector uses user authentication key user name)
- Basic digest auth user name (for customers who have Squid)
- Connector-supplied internal IP
- Squid internal IP (for customers who have Squid)
- External IP

Applying Policy to Groups

Within ScanCenter, there are two basic group types:

- Active Directory groups: must be created to match those returned by the Connector (from a customer’s Active Directory server)
- Custom groups: collections of other identification (for example, user names, internal IPs, external IPs)

A group authentication key can be assigned to either an Active Directory group or a custom group.
Installing Connector on Linux

- Overview, page 9
- Requirements, page 10
- Accessing Your Server, page 10
- Installing the Runtime Environment, page 11
- Increasing the Open Files Limit, page 12
- Running the Installer, page 13
- Operation, page 13
- Configuring Connector, page 13
- Applying an Exception, page 14
- Upgrading Connector, page 15

Overview

This chapter provides a step-by-step guide to installing the Linux Connector on x86 and x86-64 servers running either Red Hat Enterprise Linux or CentOS Linux version 5 or greater. Connector is used to deliver web traffic from a client to Cisco Cloud Web Security.

To enable Connector to integrate with the widest possible variety of software and devices, it has two modes of operation determined during installation.

- Standalone mode should be used when there are no edge devices on your corporate network.
- Enterprise mode should be used when an edge device (for example, Microsoft ISA) is already present in your corporate network.

Cisco does not support the installation of the Linux Connector onto any other distributions of Linux or UNIX operating system, or any custom configuration beyond the instructions in this guide.

Before beginning installation, ensure that your server is suitably sized and capable of running the Linux operating system and Connector with the expected number of users.
Requirements

The Linux operating system has basic hardware requirements. The following requirements are based on the number of users expected to send their web traffic through Connector.

For deployments of 500 or more users, Cisco strongly recommends multiple servers are deployed behind a hardware load balancer to ensure that there is no interruption of service in the event of a server failure. DNS load balancing (also known as round-robin) is not recommended due to the failover delay caused by caching of DNS responses by local computers.

A TCP/IP network connection and outbound Internet access on TCP ports 80 and 8080 are required for all installations. Other requirements vary depending on the number of intended users. For current sizing requirements, see the Connector release notes at http://www.cisco.com/en/US/products/ps11720/prod_release_notes_list.html.

Before installing Connector, determine where you want it installed. Connector is lightweight and does not require its own dedicated server. You must also ensure your firewall is configured correctly and SELinux is switched off.

To prepare to install Connector:

---

**Step 1**
Determine which mode the Connector will use; Standalone or Enterprise.

**Step 2**
In ScanCenter, generate the authentication keys, as necessary.

*Note* Keys are required for users with dynamic IP addresses. Keys can also be used with static IP addresses.

**Step 3**
If you require Connector to perform group lookup with Active Directory or a Windows domain, create a dedicated user within the 'Domain Users' group of the primary Domain Controller.

**Step 4**
If you have stopped ISA Server to remove a previously installed version of Connector, restart it.

---

Accessing Your Server

To connect to your Linux server:

**Before You Begin**

Confirm with your network administrator that your network firewall allows connections from your desktop computer to the Linux server on TCP port 22.

To access your Linux server from a client computer, you need a secure shell (SSH) client.

For example, to download and use the free PuTTY SSH client for Microsoft Windows:

- In the Binaries section, right-click the putty.exe hyperlink, and click Save Link As.
- Save the putty.exe file to your local desktop.
**Installing the Runtime Environment**

Connector is written in the Java language, which enables it to run on multiple platforms. However, like any Java application, it requires a Java runtime environment (JRE) to be installed on the host server.

**Caution**

You must install the official Sun JRE. Alternatives such as JRockit and IcedTea are not supported and may prevent the Connector from running.

When you have logged in to your Linux server, the command prompt is displayed, for example:

```
[root@localhost ~]#
```

The default server name is localhost. You may see a different hostname in your command prompt depending on the settings you entered during installation.

To install the Sun JRE:

**Step 1**
Change to the root directory to begin installation.

```
cd /root/
```

**Step 2**
If you have a 32-bit server, download the 32-bit Java installer from the official Oracle web site.

```
wget http://80.254.145.118/linux/jre-8u60-linux-i586-rpm.bin
```

Alternatively, if you have a 64-bit server, download the 64-bit Java installer from the official Oracle web site.

```
wget http://80.254.145.118/linux/jre-8u60-linux-x64-rpm.bin
```

If you are unsure if your server is 32-bit or 64-bit, download the 32-bit installer.

**Step 3**
Once the download has completed, make the program file executable.

```
chmod a+x ./jre-* .bin
```
Step 4 Run the program file to start the installation process.
   
   ./.jre-*.bin

Step 5 The Sun JRE license agreement is displayed. Read through the agreement. Press the space bar to display the next page.
Step 6 After you have read through the agreement, accept the terms by entering yes.
Step 7 Add the JAVA_HOME environment variable to the global server settings so it knows where to find the Java runtime environment.
   
   echo export JAVA_HOME=/usr/java/latest >> /etc/profile

Step 8 Reload the profile.
   
   source /etc/profile

Step 9 Test that Java has been installed correctly. This command should display the Java runtime environment version. If you see any other output, please review the steps above before calling support.
   
   java -version

Step 10 In some instances, it may be necessary to manually create a symbolic link.
   
   ln -s /usr/java/latest/bin/java /usr/bin/java

---

**Increasing the Open Files Limit**

The Linux default open files limit is too low and must be increased to support high traffic levels before installation.

To increase the open files limit:

---

Step 1 At the command prompt, open the limits.conf file for editing.
   
   vi /etc/security/limits.conf

Step 2 Use the Down Arrow key to move the cursor to the end of the file.

Step 3 Press the A key to enter insert (or edit) mode.

Step 4 Type conn and press the Tab key.

Step 5 Type hard and press the Tab key.

Step 6 Type nofile and press the Tab key.

Step 7 Type 32768 and press the Tab key.

Step 8 Confirm that the text you have entered appears as:
   
   conn hard nofile 32768

Step 9 Press the Esc key to exit editing mode.

Step 10 Type :wq to save your changes and exit the limits.conf file.
Running the Installer

When you have increased the open files limit, you can install Connector.

Step 1
At the command prompt, change to the root directory to begin installation.
```
cd /root/
```

Step 2
Download and execute the Connector installer (approximately 700 kb).
```
rpm -ivh http://80.254.145.118/linux/connector.noarch.rpm
```

Step 3
Confirm that the installation was successful.
```
ls /opt/connector/
```

What to Do Next
You should see a listing of files including agent.properties. If you do not see a listing of files or see an error, check that you have Internet connectivity, and try the steps above again before contacting support.

Operation
To stop the Connector, at the command prompt, type:
```
/etc/init.d/connector stop
```
To start the Connector, at the command prompt, type:
```
/etc/init.d/connector start
```
To restart the Connector, at the command prompt, type:
```
/etc/init.d/connector restart
```

Configuring Connector
Configuring Connector is achieved by editing the main configuration file agent.properties and restarting Connector. There is no graphical user interface for configuring the Linux Connector.

This section is intended as a basic overview to show you how to open the file for editing and how to apply your changes.

Note
Your technical account manager or a member of the customer support team can provide a pre-configured agent.properties file which you can upload to the server. This is the easiest approach for new installations.
Before you can configure the Connector, you need to connect to your server either from the console or using an SSH connection. See Accessing Your Server, on page 10. You should now be logged in to your Linux server and see a prompt similar to:

[root@localhost ~]#

To configure Connector:

Step 1  Open the agent.properties file for editing.
vi /opt/connector/agent.properties

Step 2  Use the arrow keys to locate the configuration option you wish to edit, as directed by the customer support engineer.

Step 3  Press A to enter insert (or edit) mode.

Step 4  Use the Delete key to remove the existing configuration option, and type in your required modification.

Step 5  Press the Esc (escape) key to exit editing mode.

Step 6  Type :wq to save your changes and exit the agent.properties file.

Step 7  You must restart the Connector to apply your changes. This can take up to 60 seconds, and should be done during a network quiet time or maintenance window.
/etc/init.d/connector restart

Step 8  When complete, type exit to close your SSH session.

Applying an Exception

An exception (or bypass) is used when you do not wish a particular web site to be filtered by Cloud Web Security. For example, you may use a secure site that restricts access to your offices egress IP address. Therefore, you would not want the web request to be routed through the shared filtering tower.

Before you can add an exception, you need to connect to your server either from the console or using an SSH connection. See Accessing Your Server, on page 10. You should now be logged in to your Linux server, and see a prompt similar to:

[root@localhost ~]#

To apply an exception:

Step 1  Open the agent.properties file for editing.
vi /opt/connector/agent.properties

Step 2  Press the Page Down key to scroll to the bottom of the file.

Step 3  Press A to enter insert (or edit) mode.

Step 4  Type in the exception. For example:
hotmail.com-exception_pattern=*.hotmail.com
hotmail.com-primary_allowed=80,443
Step 5  Press the Esc (escape) key to exit editing mode.
Step 6  Type `:wq` to save your changes and exit the `agent.properties` file.
Step 7  You must restart the Connector to apply your changes. This can take up to 60 seconds, and should be done during a network quiet time or maintenance window.

```
/etc/init.d/connector restart
```

Step 8  When complete, type `exit` to close your SSH session.

---

## Upgrading Connector

To upgrade Connector to the latest General Availability (GA) version:

Step 1  Change to the root directory to begin installation.

```
cd /root/
```

Step 2  Backup the Connector configuration to the root directory.

```
cp /opt/connector/agent.properties agent.properties-`date -I`
```

Step 3  Download and execute the Connector upgrade (approximately 700 kb).

```
rpm -Uvh http://80.254.145.118/linux/connector.noarch.rpm
```

Step 4  Restart the Connector to ensure the upgrade completes.

```
/etc/init.d/connector restart
```

Step 5  

### What to Do Next

You should now be able to browse the Internet through Connector. If you are unable to do so, you should contact support for assistance. You must provide a copy of the log files in `/opt/connector/logs/`. 
Installing Connector on Linux

Upgrading Connector
Installing Connector on Windows

Overview

This chapter provides a step-by-step guide to installing the Windows Connector on servers running the Microsoft Windows Server operating system. It should help you select the appropriate Connector mode, apply the right authentication key if necessary, and install, configure, and operate Connector.

To enable Connector to integrate with the widest possible variety of software and devices, it has two modes of operation determined during installation.

• Standalone mode should be used when there are no edge devices on the corporate network.
• Enterprise mode should be used when an edge device (for example, Microsoft TMG) is already present in the corporate network.

Note

If you are using Microsoft Forefront TMG 2010, you must upgrade to Forefront TMG 2010 SP1.

Requirements

Connector is supported on the following Microsoft operating systems:

• Windows Server 2012 R2 x64 (64-bit)
• Windows Server 2008 R2 x64 (64-bit)
Windows Server 2008 x86 (32-bit) and x64 (64-bit)

Cisco Cloud Web Security Connector is not supported on Windows Server 2012 x64 (64-bit).

For deployments of 500 or more users, Cisco strongly recommends multiple servers are deployed behind a hardware load balancer to ensure there is no interruption of service in the event of a server failure. DNS load balancing (also known as round-robin) is not recommended due to the failover delay caused by caching of DNS responses by local computers.

At least one GB of available disk space, a TCP/IP network connection, and outbound Internet access on TCP ports 80 and 8080 are required for all installations. Other requirements vary depending on the number of intended users. For current sizing requirements, see the Connector release notes at http://www.cisco.com/en/US/products/ps11720/prod_release_notes_list.html.

Windows Firewall must be enabled on the server where Connector is installed.

Before installing Connector, determine where you want it installed. Connector is lightweight and does not require its own dedicated server. For standalone servers, Cisco recommends installing Connector on either a Primary Domain Controller (PDC) or Backup Domain Controller (BDC) within your network. If you are using Microsoft Forefront TMG, Cisco recommends installing Connector on the same server.

To prepare to install Connector:

**Step 1** Determine which mode Connector will use; Standalone or Enterprise.

**Step 2** In ScanCenter, generate the authentication keys, as necessary.

**Note** Keys are required for users with dynamic IP addresses. Keys can also be used with static IP addresses.

**Step 3** If you require Connector to perform group lookup with Active Directory or a Windows domain, create a dedicated user within the ‘Domain Users’ group of the PDC.

**Step 4** If you are using Windows Server 2003 or higher with SMB signing enabled (default setting), create a dedicated user on the PDC. If you have already created a user to enable group lookups, you do not need to create a new user. When configuring Connector, include the details of this user in the config file with the following arguments:

- ntlm.preauth.domain=
- ntlm.preauth.username=
- ntlm.preauth.password=

**Step 5** Download the Connector installation program from ScanCenter.

**Step 6** Remove any previously installed versions of Connector, including Proxy Agent. See Removing Connector, on page 67.

**Step 7** If you will use Microsoft Forefront TMG, make sure it is installed and running. The server where Microsoft TMG and Connector are installed must meet the minimum requirements for the version of Microsoft TMG you are using.
What to Do Next
Following all installations, apply the relevant Windows registry patches. See Applying the Windows Registry Patches, on page 34.

Installing in Standalone Mode

To install Connector in Standalone mode:

Note

The following screenshots are examples from Connector release 3.0.3.0. Your actual screenshots may vary slightly depending on the release. Most or all of the functionality should be similar or the same.

Step 1
Double-click the Connector program file to run the installation wizard.

Step 2
Click Next to display the License Agreement dialog.
Step 3  Read the End User License Agreement. If you agree to the terms, click **I accept the agreement** and click **Next** to display the Select Destination Location dialog. Alternatively, if you do not agree to the terms, click **Cancel** to stop the installation.
Step 4 Click **Next** to accept the default installation folder. Alternatively, enter a new path, or click **Browse** and navigate to the required folder, then click **Next** to display the Ready to Install dialog.
Step 5  Click **Install** to begin the installation.
Step 6 Welcome!
Step 7  
Click **Configure a Connector**, and click **Next** to display the Startup Settings dialog.
Step 8  From the Charset pull-down list, choose a character encoding set. The default is ISO-8859-1. Choose UTF-8 if your user or group names contain UTF-8 symbols.

Step 9  Click Next to display the Connector Type dialog.
Step 10  Click **Workgroup Connector**, and click **Next** to display the Authentication Configuration dialog.
Step 11  You can use IP-based or key based authentication. IP-based authentication requires a static IP address. To use IP-based authentication, click **I authenticate with my static IP**. Alternatively, click **Enter your authentication key here** and enter a company or group authentication key. For details of how to generate a key, refer to the Cisco ScanCenter Administrator Guide.

Step 12  Click **Next** to display the Service Settings dialog.
Step 13 Your proxy settings are contained in your provisioning email. If you have not received this email, contact your support representative. Normally, only primary and secondary proxies are provided. You can specify up to three proxy servers:

- The primary proxy is used in preference to the other proxies.
- The secondary proxy is used as a fallback in cases where the primary proxy is unreachable.
- The tertiary proxy is used as a fallback in cases where both the primary and secondary proxies are unreachable.

**Note** The value of the primary, secondary, or tertiary proxy can be set to DIRECT to cause all traffic to bypass the Cisco Cloud Web Security proxy and go directly to the origin server.

Step 14 Enter your proxy settings. Click **Next** to display the Host/IP Exceptions dialog.
Step 15  The Host/IP Exceptions dialog enables you to create exceptions that specify direct connections or alternate proxies for specific websites, domains, hosts, or networks. The exceptions are shown in a list. The ranges already included in the RFC1918 exception list are some of the most commonly used. They are not comprehensive. They do not include all local IP address ranges defined in RFC1918. Review the ranges for accuracy with your requirements. It is not necessary to configure the exceptions during the installation. If you decide to configure them later, see Adding Host Exceptions, on page 35. Click Next to display the Authentication dialog.
Step 16  Ensure the Use NTLM check box is selected. Do not clear this check box unless instructed to do so by your support representative.

Step 17  Clear the Verify authentication with the domain controller check box. Alternatively, select the check box to verify credentials provided by clients with the domain controller and enter the respective information.

Step 18  Enter any client IP addresses to be excluded from authentication in a comma separated list in the NTLM Exceptions box. This box should normally be left blank unless you have been otherwise instructed by your support representative.

Step 19  Click Next to display the Group Lookup Settings dialog.
Step 20  To use LDAP to gather group information:

1  Click Use Active Directory.

2  Enter an LDAP URL in the Provider URL box. Alternatively, if you are installing Connector on the Domain Controller, accept the default LDAP URL (ldap://127.0.0.1:3268).

3  To enable Connector to perform LDAP group lookups, an Active Directory user must be created. Enter the user name of the Active Directory account you created for Connector in the Username box. For example, cn=proxyagent,cn=users,dc=company,dc=com. See Requirements, on page 17.

4  Enter the password for the Active Directory account in the Password box.

Step 21  Alternatively, to use NTLM to gather group information:

1  Click Use Windows Domain.

2  Enter the user name of the Domain Controller account you created for Connector in the DC login username box.

3  Enter the password of the Domain Controller account in the DC login password box.

4  Click Next to display the NTLM Domain Controllers Settings dialog.
5 Enter the IP address of your PDC in the Primary DC box.

6 If you have a secondary or backup Domain Controller, enter its IP address in the Secondary DC box.

**Step 22** Cisco recommends using LDAP to gather group information. Do not click **Do not lookup user groups** unless instructed to do so by your support representative.

**Step 23** Click **Next** to begin the installation.
Step 24 When the installation tasks have completed successfully, the following dialog is displayed.
Applying the Windows Registry Patches

There are two registry patch files in the folder where Connector was installed:

- TCP-IP-BackLog.reg
- PortRangeAndSocketShutdownPatch.reg

For versions of Windows prior to Windows Server 2008 R2, the TCP-IP-BackLog.reg patch should be applied. This increases the maximum number of connections in the backlog queue from 250 to 1000 and prevents Connector from rejecting connections if there are already 250 'half open' connections.

For all versions of Windows, the PortRangeAndSocketShutdownPatch.reg patch should be applied. This increases the short-lived (ephemeral) port range from 1024-5000 to 1024-65535 and changes the default time-out for these ports from four minutes to 30 seconds. This prevents the number of available ports from being exhausted when a very large number of users are connecting to the service.

After you have applied the registry patches, restart the server for the OS to accept the new changes.
Post-Installation Firewall Configuration

You need to ensure Connector can forward all web traffic out of your network to Cisco Cloud Web Security. In most cases, this requires a simple change to your firewall settings to allow all TCP traffic on port 8080 originating from the IP address where Connector is running to go out to the Internet. The following diagram shows the path a web request must take to get to Cisco Cloud Web Security:

1. Web browser requests a URL.
2. Connector performs an NTLM challenge.
3. Web browser responds with NTLM user details.
4. Connector uses credentials to poll the domain controller (LDAP) for AD Groups. If the user exists, Connector performs a query based on user name to lookup groups.
5. Domain Controller sends group information to Connector.
6. URL request and encrypted user and group information forwarded from Connector to Cisco Cloud Web Security.
7. Content sent back to the user through Connector.

Adding Host Exceptions

Host exceptions are used to allow users to bypass Cisco Cloud Web Security when connecting to specified web sites. Exceptions can include wild cards, address ranges, and IP ports. They should not be used for connections to your own network because a proxy server (local exception) set in a user's web browser is more efficient for this task.

To add host exceptions to Connector:

Step 1  In the folder where you installed Connector, double-click the Wizard batch file to run the configuration wizard.
Step 2

The wizard imports the settings from your last session so it is not necessary to specify that you are using a standalone server, your method of authentication, or the service settings. At each dialog, click **Next** until the Host/IP Exceptions dialog is displayed.
Step 3  The Host/IP Exceptions dialog enables you to create exceptions that specify direct connections or alternate proxies for specific web sites, domains, hosts, or networks. The exceptions are shown in a list. You can click **Edit** to edit an existing exception or **Delete** to remove an exception.

Step 4  For each exception you want to add:

1. Click **Add**.
2 Enter a Name for the exception.

3 Enter the web sites which the exception should be applied to, separated by commas. Web sites can be entered:
   • in full (www.company.com)
   • with wildcards (*.company.com)
   • as an IP address (164.35.91.46)
   • as a range of IP addresses (164.35.91.*)
   • with a port (*.company.com/80, 164.35.91.*/8080)

4 You can provide up to three proxies. The secondary and tertiary proxies act as fallbacks in the event that the primary proxy is unavailable. Only the primary proxy is required. If no proxy is available, you cannot connect to the service. For each proxy, select the Direct check box to enable users to connect directly to the specified web sites. Alternatively, enter a Host (normally an internal proxy) and (optionally) IP port.

5 When you have entered the proxy details, click OK. Alternatively, click Cancel to abandon your changes.

**Step 5**

The wizard imports the settings from your last session so it is not necessary to specify authentication, NTLM, or Group Lookup settings. At each dialog, click Next until the Applying settings dialog is displayed.
When the configuration tasks have completed successfully, the following dialog is displayed:
Step 6

Click Finish to close the wizard.

Installing in Enterprise Mode

In Enterprise mode, Connector works with a device that uses the Internet Content Application Protocol (ICAP), such as Microsoft TMG or Blue Coat.

Note

The following screenshots are examples from Connector release 3.0.3.0. Your actual screenshots may vary slightly depending on the release. Most or all of the functionality should be similar or the same.

To install Connector in Enterprise mode:

Step 1

Double-click the Connector program file to run the installation wizard.
Step 2 Click **Next** to display the License Agreement dialog.
Step 3  Read the End User License Agreement. If you agree to the terms, click **I accept the agreement**, and click **Next** to display the Select Destination Location dialog. Alternatively, if you do not agree to the terms, click **Cancel** to stop the installation.
Step 4  Click **Next** to accept the default installation folder. Alternatively, enter a new path, or click **Browse** and navigate to the required folder, then click **Next** to display the Welcome dialog.
Step 5  Click **Configure a Connector**, and click **Next** to display the Startup Settings dialog.
**Step 6**  From the Charset pull-down list, choose a character encoding set. The default is ISO-8859-1. Choose UTF-8 if your user or group names contain UTF-8 symbols.

**Step 7**  Click Next to display the Connector Type dialog.
Step 8
Click **Enterprise Connector**, and click **Next** to display the Authentication Configuration dialog.
Step 9 You can use IP-based or key-based authentication. IP-based authentication requires a static IP address. To use IP-based authentication, click I authenticate with my static IP. Alternatively, click Enter your authentication key here and enter a company or group authentication key. For details of how to generate a key, refer to the Cisco ScanCenter Administrator Guide.

Step 10 Click Next to display the Enterprise Gateway Settings dialog.
Step 11 If you are using Microsoft Forefront TMG:

1. Select **Microsoft TMG server**.
2. Click **Next** to display the Settings dialog.
3 Choose **TMG will run on this computer**.

4 Enter the directory where the TMG server installation is located.

5 Click **Next**.

6 If the Microsoft Firewall service is running, you are prompted to stop the service. The service is restarted when the configuration is complete. Click **OK**.

**Step 12** Alternatively, if you are not using TMG Server, select **Other - ICAP capable gateway** and click **Next**.
Step 13 Click OK.
Step 14  Click **Finish** to display the Group Lookup Settings dialog.
**Step 15** To use LDAP to gather group information:

1. Click **Use Active Directory**.

2. Enter an LDAP URL in the **Provider URL** box. Alternatively, if you are installing Connector on the Domain Controller, accept the default LDAP URL (ldap://127.0.0.1:3268).

3. To enable Connector to perform LDAP group lookups, an Active Directory user must be created. Enter the user name of the Active Directory account you created for Connector in the **Username** box. For example, `cn=proxyagent,cn=users,dc=company,dc=com`. See Requirements, on page 17.

4. Enter the password for the Active Directory account in the **Password** box.

**Step 16** Alternatively, to use NTLM to gather group information:

1. Click **Use Windows Domain**.

2. Enter the user name of the Domain Controller account you created for Connector in the **DC login username** box.

3. To enable Connector to perform LDAP group lookups, an Active Directory user must be created. Enter the user name of the Active Directory account you created for Connector in the **Username** box. For example, `cn=proxyagent,cn=users,dc=company,dc=com`. See Requirements, on page 17.

4. Enter the password of the Domain Controller account in the **DC login password** box.

5. Click **Next** to display the NTLM Domain Controllers Settings dialog.

6. Enter the IP address of your PDC in the **Primary DC** box.
7 If you have a secondary or backup Domain Controller, enter its IP address in the **Secondary DC** box.

**Step 17** Click **Next** to begin the installation. If the Microsoft Firewall service is running, you are prompted to stop the service. The service is restarted when the installation is complete. When the installation tasks have completed successfully, the following dialog is displayed:

![Connector Configuration Wizard](image)

**Step 18** Click **Finish** to close the wizard.

---

**What to Do Next**

If you installed Connector on a different computer than Forefront TMG, you must now configure TMG. See *Configuring Microsoft Forefront TMG*, on page 54.

**Post-Installation Proxy Server Configuration**

You need to ensure the connector can forward all web traffic out of your network to Cisco Cloud Web Security. The changes you need to make are dependent on the proxy server, or firewall appliance, you are using. For more information, refer to the quick reference for your proxy server. The following diagram shows the path a user's web request must take to get to Cisco Cloud Web Security.
1 Web browser requests a URL.
2 Proxy server performs an NTLM challenge.
3 Web browser responds with NTLM user details.
4 URL request is forwarded with NTLM credentials to Connector.
5 Connector uses the credentials to poll the domain controller (LDAP) for AD Groups. If the user exists, Connector performs a query based on user name to lookup groups.
6 Domain Controller sends group information to Connector.
7 URL request forwarded from Connector to proxy server with encrypted headers.
8 URL request with encrypted group information forwarded from proxy server to Cisco Cloud Web Security.
9 Content sent back to the user through the proxy server.

**Configuring Microsoft Forefront TMG**

The recommended method for using Microsoft Forefront TMG (formerly known as Microsoft ISA Server) with Connector is to install Connector, Forefront TMG, and the Forefront TMG plug-in on a shared server. You must use separate folders to install the ICAP sender and receiver, for example C:\Program Files\ConnectorICAP and C:\Program Files\ConnectorLDAP. You should not use other configurations unless instructed to do so by customer support.

**Step 1**

Double-click the Connector program file to run the installation wizard.
Step 2  
Click **Next** to display the License Agreement dialog.
Step 3

Read the End User License Agreement. If you agree to the terms, click I accept the terms in the license agreement, and click Next to display the Select Destination Location dialog. Alternatively, if you do not agree to the terms, click Cancel to stop the installation.
Step 4  You must choose a different folder from the one in which you installed Connector. Enter a new path in the Save files in folder box, or click **Browse** and navigate to the required folder, then click **Next** to display the Welcome dialog.
Step 5  
Click **Configure a Connector**, and click **Next** to display the Startup Settings dialog.
Step 6  From the Charset pull-down list, choose a character encoding set. The default is ISO-8859-1. Choose UTF-8 if your user or group names contain UTF-8 symbols.

Step 7  Click Next to display the Connector Type dialog.
Step 8 Click Enterprise Connector, and click Next to display the Authentication Configuration dialog.
Step 9  You can use IP-based or key-based authentication. IP-based authentication requires a static IP address. To use IP-based authentication, click **I authenticate with my static IP**. Alternatively, click **Enter your authentication key here** and enter a company or group authentication key. For details of how to generate a key, refer to the Cisco ScanCenter Administrator Guide.

Step 10  Click **Next** to display the Enterprise Gateway Settings dialog.
Step 11  Click Microsoft TMG Server.
Step 12  Click Next to display the Microsoft TMG Settings dialog.
Step 13  Click **TMG will run on this computer**.

Step 14  Click the folder button and navigate to the folder where the TMG server is installed.

Step 15  Click **Next** to begin the configuration.
If the Microsoft Firewall service is running, you are prompted to stop the service. The service is restarted when the configuration is complete. Click **OK**.

When the configuration tasks have completed successfully, the following dialog is displayed:
Step 16  Click Finish to close the wizard.

**Post-Installation Forefront TMG Configuration**

You can verify that the Connector plug-in has been installed in Forefront TMG by making sure you can see a Connector Plugin entry under the Web Filters tab.

To enable the plug-in, you must edit the hosts file (typically C:\WINDOWS\system32\drivers\etc\hosts) and add the following entry:

```
127.0.0.1 connector
```
After Connector is installed, configure Forefront TMG:

**Step 1** Ensure that you have assigned your organization's Domain Name to the TMG server's internal network object.

**Step 2** Create an Access rule to allow the All Authenticated Users user set access to the Internet via FTP, HTTP, and HTTPS. Ensure that no other user sets are selected.

**Step 3** Create a Web Chaining rule with the **Redirect them to a specified upstream server** action.

**Step 4** Click **Settings**. In the Upstream Server Setting dialog, enter the Cisco Cloud Web Security primary proxy IP address from your provisioning email in the Server box.

**Step 5** Enter **8080** in the Port and SSL Port boxes.

**Step 6** Ensure the **Automatically poll upstream server for the configuration** and **Use this account** check boxes are cleared.

**Step 7** In the Backup route menu, click **Upstream proxy server**.

**Step 8** Click **Settings**. In the Upstream Server Setting dialog, enter the Cisco Cloud Web Security secondary proxy IP address from your provisioning email in the Server box.

**Step 9** Ensure the **Automatically poll upstream server for the configuration** and **Use this account** check boxes are cleared.

**Step 10** Apply your changes to Forefront TMG.

**Enabling Persistent ICAP Mode**

Creating a persistent connection to the ICAP server can improve performance in some circumstances.

Persistence ICAP mode is switched off by default. It can be enabled by adding the appropriate arguments to the Connector agent.properties file and the TMG plug-in agent.properties file.

In the Connector file, add:

```
icap.connection.pool=true
```

In the TMG plug-in, add:

```
persistentICap=true
```

The ICAP persistence for the plug-in requires additional parameters to control its operation:

```
minThreads=50
maxThreads=100
maxIdleTime=320
minIdleConnections=10
readTimeout=10
```

**Note** The above values are for a generic system, so you may need to use different values. Contact customer support for further information on choosing appropriate values.

**Bypassing Cisco Cloud Web Security**

In some cases, you may need to bypass Cisco Cloud Web Security for particular web sites or IP addresses. For example, a web site or web application located on your Intranet. In this case, your users need to connect directly, because Cisco Cloud Web Security cannot access anything within your Intranet.
To add an exception:

**Step 1** Create a Web Chaining rule for System Policy Allowed Sites.

**Step 2** Edit the System Policy Allowed Sites properties to include the web sites for which you want to bypass Cisco Cloud Web Security.

**Step 3** Set the Request Action to ‘Retrieve requests directly from the specified destination.’

**Step 4** Ensure the rule is applied before the Last Default rule.

**Step 5** Apply your changes to TMG. You can add additional web sites by editing the rule.

---

**Upgrading Connector**

To upgrade Connector, you must remove the currently installed version before you install the new version. Before removing the existing version, you should make a backup of the agent.properties file as this contains your settings. After you have installed the new version of Connector, replace the new version of the file with your backup.

To support reliable cipher algorithms, you should update JRE with Java Cryptography Extension (JCE) unlimited strength jurisdiction policy files. Note that there are different files for different versions of Java.

You should check whether the system DNS cache honors TTL of tower DNS records.

**Removing Connector**

To remove Connector from a server:

**Step 1** In the folder where you installed Connector, double-click the batch file to run the configuration wizard.
Step 2  Click **Uninstall a Connector** and click **Next**. When Connector has been removed successfully, the following dialog is displayed:
Step 3

Click Finish to close the wizard.

What to Do Next

It may be necessary to stop Forefront TMG in order to manually delete the folder where Connector was installed.
Configuring Connector

- Web Browser Configuration, page 71
- Authentication, page 74
- Host Exceptions, page 75
- Acceptable Use Policy, page 77
- SSL Tunneling, page 78
- Cipher Suites for TLS/SSL, page 78
- LDAP Servers, page 78
- Groups, page 81
- Configurable Domain Name Mapping, page 83

Web Browser Configuration

When Connector is installed and running, you must configure your web browser proxy settings to point to the Connector. For example, in Microsoft Internet Explorer, this is configured in the Local Area Network (LAN) Settings dialog.
In this example, the web browser is configured to look for a Proxy Auto-Config (PAC) file on a local web server and use that to configure the proxy settings automatically. In the event that the PAC file cannot be found, the web browser falls back to the local settings. The Address box must contain the IP address or DNS name of the server where Connector is installed. The Port box must contain 8080 (the default port on which Connector listens for HTTP requests).

In larger organizations, the most effective way to implement this change is either by using a network Logon Script or through Group Policy Objects (GPO) in Active Directory.

The following sections are intended to help you choose the method that best suits your requirements.

**Manual Configuration**

This method allows the use of a single proxy defined in the web browser connection settings. It is the simplest method and usually the more reliable choice.

The advantages are:

- It is simple to configure. All that is required is the location of the proxy and its relevant port.
- It is easy to enable the user to go directly to specified sites instead of using the proxy.
- In most situations, it is the more secure method.

The disadvantages are:

- It lacks flexibility. Only one proxy can be specified, so it is not possible to specify a failover proxy.
- The proxy setting must be applied to each machine. With Internet Explorer, this can be pushed out using Group Policies. However, with web browsers such as Firefox, the setting would have to be amended manually for each web browser.

**Proxy Auto-Config File**

This is likely to be the preferred method in most cases. The location of the PAC file must still be set in each web browser, either manually or by group policy. However, the PAC file allows greater control and flexibility limited only by the author’s ability to code the file in JavaScript and the infrastructure available.
The advantages are:

- The potential to implement failover proxies, load balancing, fault tolerance, and so on.
- Scalability. The PAC file can be as complex as the requirements that need to be met.

The disadvantage is:

- Basic understanding of JavaScript programming may be necessary to create or amend PAC file scripts to meet requirements.

For more information, see Proxy Auto-Config File, on page 103.

**Web Proxy Auto-Discovery Protocol**

The Web Proxy Auto-Discovery (WPAD) protocol is a method used by web browsers to locate a PAC file. The advantages are:

- WPAD has all the advantages of using a PAC file.
- It requires the least amount of user and administrator intervention to set up each user.

The disadvantages are:

- It requires that explicit requirements are met before it can function correctly.
- The system serving the PAC file must have a high uptime level.
- There are inherent security issues.

For more information, see Web Proxy Auto-Discovery Protocol, on page 109.

**Internal DNS Resolution**

Connector can use internal, Java-based DNS resolution to handle primary, secondary, and tertiary proxies only. All other names are resolved by the underlying DNS resolver provided by the OS.

Connector’s internal DNS resolver has a failover mechanism: if Connector is unable to reach any of the DNS servers specified in the proxyDns.nameServers property of the agent.properties file, it uses the system-wide DNS resolver provided by the OS. Likewise, if internal, Java-based DNS resolution is not configured in the agent.properties file, the system-wide DNS resolution is used.

Properties in the agent.properties file used to configure this feature include:

- `proxyDns.nameServers`
- `proxyDns.default.ttl`
- `proxyDns.default.negative.ttl`
- `proxyDns.timeout`
- `proxyDns.failover.numberOfRetriesForResource`
- `proxyDns.tcp`
Upstream Network Interface Selection

On multi-homed servers with Connector installed in workgroup mode, the server automatically selects the network interface that is used to connect to the Cloud Web Security proxy, another proxy, or directly to the origin server. A specific network interface can be configured using the upstreamNicAddr property in the agent.properties file. If you want the server to select the interface, do not set a property for upstreamNicAddr.

Connector Host and Client on the Same Computer

This method is used when Connector is installed on a portable computer for individual system use. The normal system requirements do not apply in this instance because Connector is only processing the requests of a single user. You can use the client version of any of the supported server operating system. If you wish to use a different client operating system, contact Support.

Note

You must use the name localhost and port 8080 for the proxy server settings.

Authentication

NTLM Pass-Through

Connector supports websites that require NTLM authentication. This feature is referred to as NTLM pass-through because messages pass from the client, through Connector, and to the website. This can be a direct connection to a website that has been configured as a host exception, or a connection that goes through Cloud Web Security.

NTLM pass-through requires the keepalive.enable property to be set to TRUE to enable persistent connections. If no value is set, the property is set to TRUE by default. NTLM pass-through is also supported when Connector is used in conjunction with the Cisco Web Security Appliance and other third-party proxy servers.

Preemptive Basic Authentication

Clients that do not support NTLM authentication may attempt to authenticate using Preemptive Basic Authentication (PBA). This does not wait for Connector to request authentication. Instead, clients that use PBA send the credentials with each HTTP request. When Connector is configured for NTLM authentication, it can also accept PBA by setting the value of the acceptPreemptiveBasicAuth property to TRUE in the agent.properties file.

User Principal Name

During the authentication process, some clients may provide the User Principal Name (UPN) rather than the sAMAccountName. By default, Connector expects only the sAMAccountName. It can be configured to also
accept a UPN by setting the acceptUPN property to TRUE in the agent.properties file. Connector can be configured to accept either the sAMAccountName or the UPN.

---

**Note**

If there are two users with the same sAMAccount name in different domains of a multi-domain forest, you must enable dynamic mapping to allow Connector to correctly identify both users by setting grouplookup.process to 3 and enabling TCP port 389 on the domain controller.

---

### Host Exceptions

With Windows, host exceptions can be configured using the configuration wizard. See Adding Host Exceptions, on page 35. It is also possible to edit or add exceptions by editing the agent.properties file using a text editor. With Linux, this is the only method. See Configuring Connector, on page 13.

Host exceptions are added using the following properties:

- `<exception name>-exception_pattern=<host patterns>` (mandatory)
- `<exception name>-primaryProxy=<primary proxy>` (mandatory)
- `<exception name>-primaryProxyPort=<primary proxy port>` (only applicable if primaryProxy is not DIRECT)
- `<exception name>-primary_allowed=<allowed port list for primary>` (only applicable if primaryProxy is DIRECT)
- `<exception name>-secondaryProxy=<secondary proxy>`
- `<exception name>-secondaryProxyPort=<secondary proxy port>` (only applicable if secondaryProxy is not DIRECT)
- `<exception name>-secondary_allowed=<allowed port list for secondary>` (only applicable if secondaryProxy is DIRECT)
- `<exception name>-tertiaryProxy=<tertiary proxy>`
- `<exception name>-tertiaryProxyPort=<tertiary proxy port>` (only applicable if tertiaryProxy is not DIRECT)
- `<exception name>-tertiary_allowed=<allowed port list for tertiary>` (only applicable if tertiaryProxy is DIRECT)
- `<exception name>-keepalive.enable=<boolean>` (use a persistent connection)
- `<exception name>-tunnel=<boolean>` (make a connection through a tunnel)

Local IP address ranges as defined in RFC1918 are by default added to the list as exceptions:

RFC1918-exception_pattern=10.0.0.0/8,172.16.0.0/12,192.168.0.0/16

RFC1918-primaryProxy=DIRECT

### Example Host Exceptions

To configure an exception with a single proxy on a specific IP address and port for a range of domains use the following:

- `<exception name>-exception_pattern=*.<domain1>, *.<domain2>`
- `<exception name>-primaryProxy=<IP>`
- `<exception name>-primaryProxyPort=<port>`

For example:

example-exception_pattern=*.example.com, *.example.net
example-primaryProxy=192.168.32.122
example-primaryProxyPort=8081
To configure an exception with a direct connection for a range of domains on port 8080 (the default) use the following:

```
<exception name>-exception_pattern=*.<domain1>, *.<domain2>
<exception name>-primaryProxy=DIRECT
```

For example:

```
example-exception_pattern=*.example.com, *.example.net
example-primaryProxy=DIRECT
```

**Note**
The default value of `<exception name>-keepalive.enable` is TRUE, even if the global `keepalive.enable` property is set to FALSE.

To configure an exception with a direct connection for all hosts on a given domain using the default port use the following:

```
<exception name>-exception_pattern=*.<domain>
<exception name>-primaryProxy=DIRECT
```

For example:

```
example-exception_pattern=*.example.com
example-primaryProxy=DIRECT
```

To enable additional ports, use the following:

```
<exception name>-exception_pattern=*.<domain>
<exception name>-primaryProxy=DIRECT
<exception name>-primary_allowed=<port1>, <port2>
```

For example:

```
example-exception_pattern=*.example.com
example-primaryProxy=DIRECT
example-primary_allowed=443, 8443, 1245
```

**Note**
When specifying custom ports, you must explicitly specify the default ports. Ports 443 and 8443 are commonly used for HTTPS traffic. These common ports are allowed by default if no custom allowed ports are specified.

---

**Persistent Connections**

Persistent connections were introduced in version 1.1 of the HTTP protocol to improve performance by removing the need to reopen closed connections. Connector enables persistent connections by default. This enables Connector to reuse connections to upstream servers, and clients to reuse connections to Connector. If required, persistent connections can be switched off for exceptions as follows:

```
<exception name>-keepalive.enable=FALSE
```

**Tunnels**

You may want to use a tunnel if Connector has problems with the HTTP response message or the body of an HTTP request message. After the initial connection is established, a direct tunnel is created over that connection.
between the client and the server, enabling messages to pass unmodified between them. Tunnels can be enabled for exceptions as follows:
<exception name>-tunnel=TRUE

⚠️ Caution
If <exception name>-keepalive.enable and <exception name>-tunnel are both set to TRUE, requests could be sent to the wrong host.

#### Acceptable Use Policy

To show an AUP page to your users on a daily or weekly basis with Connector in workgroup mode:

**Step 1**
Edit the agent.properties file and add the line:
```
aup.enable=true
```

**Step 2**
Restart Connector.

**Step 3**
In ScanCenter, navigate to the Global Settings page and ensure the Acceptable Use Policy pane is shown. If it is not shown, contact support to have this enabled for your account.

**Step 4**
Select the Enable AUP for all users check box.

**Step 5**
Select the Include standard HTML page templates for AUP page if you want to include the default image and text on the AUP page.

**Step 6**
Click Daily or Weekly to set how often the page is displayed.

**Step 7**
Edit the HTML in the box.

**Step 8**
Click Save to save your changes.

A sample page is provided as a template for you to use if your organization does not currently have an AUP. However, we recommend you seek professional advice in creating your own AUP. Care should be taken to include references to the latest Web 2.0 technologies and you should lock down your portable computers so that they can only use the Internet through Cloud Web Security (both internally and externally). Any attempt to circumvent this should be strictly prohibited in the AUP. All AUP pages have an "I Agree" button at the bottom of the page for your users to click. You should include this in your AUP and state that by clicking "I Agree" the user agrees to abide by your organization's AUP.

When the Include standard HTML page templates for AUP page check box is cleared you can specify the full content of the page, from the opening `<html>` tag to the closing `/html>` tag. Any images or CSS (cascading style sheets) must be stored at a resolvable location. Normally, you must host your own images and CSS. When you have saved your changes, you can view the AUP page by clicking the Preview button.

**Note**
The AUP page relies on the quota functionality of Connector. If Connector is reset, the count also resets to zero and the AUP page is displayed again to users.
SSL Tunneling

You can send all web traffic to Cloud Web Security using an SSL-based tunnel. Note that not all Cloud Web Security proxies support this functionality, and you should work with the support team to make sure you are provisioned on a proxy that supports this functionality.

---

Note

Enabling the SSL Tunneling feature puts additional load on the server running Connector. It also adds a small amount of extra latency into the link. Your organization is responsible for the use of this feature. Cisco does not condone its use to bypass country-based firewall blocking.

---

Cipher Suites for TLS/SSL

A cipher suite is a named combination of authentication, encryption, message authentication code (MAC), and key exchange algorithms used to negotiate the security settings for a network connection using the Transport Layer Security (TLS) / Secure Sockets Layer (SSL) network protocol.

When a TLS/SSL connection is established, the client sends a cipher suite list, which is a list of the cipher suites that it supports in order of preference. The server replies with the cipher suite that it has selected from the list.

- The allowedTlsProtocols property lists the allowed TLS protocols in order of preference.
- The allowedTlsCipherSuitesList property contains the location of the file listing the allowed TLS cipher suites.
- The ciphers.properties file lists the allowed cipher suites in order of preference. For example:

```
# This configuration file lists the TLS cipher suites allowed for use in TLS connections.
# Cipher suites are arranged in order of preference.
# The default order is based on Java preference order and defined by reliability and productivity.
# You can reorder the list of cipher suites.
# You can add a new cipher suite.
# You can comment/disable cipher suites.
# Any cipher suite listed below will be used if current JRE supports such a cipher suite.
# Otherwise, it will be ignored with an appropriate warning message in the log file.
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
TLS_RSA_WITH_AES_256_CBC_SHA256
#TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384
#TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384
#TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384
```

LDAP Servers

Connector supports the use of multiple LDAP servers, including generic LDAP servers such as Lotus Domino. By default, Connector uses Active Directory LDAP.
Generic LDAP

Connector supports basic authentication lookups against generic LDAP servers. You may have to change the default settings in your LDAP server to allow for this.

In order to perform generic LDAP lookups, enable basic authentication rather than NTLM authentication in the `agent.properties` file:

```properties
useNtlm=false
useBasic=true
auth.realm=MyRealm
passwordRequired=true
```

You can also change the name of the realm that appears in the basic authentication dialog by changing the `auth.realm` value.

After selecting basic authentication, the LDAP server must be configured. By default, the LDAP type is Active Directory. To use generic LDAP, use the following:

```properties
ldap.type=generic
useLdap=true
useNtlm=false
useBasic=true
providerUrl=ldap://127.0.0.1:3268
securityPrincipal=cn=proxyagent,cn=users,dc=company,dc=com
ldap.base.dn=ou=People,dc=example,dc=com
ldap.user.attr=uid
ldap.group.attr=ou
ldap.accountdisabled.attribute=
ldap.group.attr.string.parse=
```

- The `ldap.base.dn` property specifies the base DN in the LDAP tree where the query starts.
- The `ldap.user.attr` is the name of the user attribute in the LDAP server configuration.
- The `ldap.group.attr` is the name of the group attribute in the LDAP server configuration.
- The `ldap.accountdisabled.attribute`, if it has a non-empty value, represents the name of the attribute that flags if the user is allowed to browse. If a user is marked as 'disabled' in the LDAP server, then that user is not allowed to browse, even if they provide the correct password and user account when prompted with the basic authentication dialog.
- The `ldap.group.attr.string.parse` property is the name of the attribute for parsing out the group name from a LDAP query response. For example, if the response to the group LDAP query is `ou=mygroup, o=mycompany, l=mylocation`, then by specifying `ldap.group.attr.string.parse=ou` the group name is parsed out from the LDAP response string, which in this case is `mygroup`.

Novell

The Connector LDAP settings for Novell LDAP are:

```properties
useNtlm=false
useBasic=true
auth.realm=<realm>
ldap.type=generic
useLdap=true
providerUrl=ldap://<IP address>:389
securityPrincipal=cn=<admin user name>,o=<context path>
securityCredentials=<admin password>
ldap.base.dn=o=<context path>
ldap.user.attr=cn
```

Lotus Domino

The Connector LDAP settings for Lotus Domino are:

```java
useNtlm=false
useBasic=true
auth.realm=<realm>
ldap.type=generic
useLdap=true
providerUrl=/<IP address>:389
securityPrincipal=cn=<admin user name>,o=<organization>
securityCredentials=<admin password>
ldap.base.dn=o=<organization>
ldap.user.attr=cn
ldap.group.attr=dominoaccessgroups
ldap.group.attr.string.parse=cn
```

Secondary LDAP Server

Connector supports secondary LDAP servers in case of primary failure. For backward compatibility, primary settings can also be specified without 'primary' suffix. For example:

```java
providerUrl.primary=ldap://192.168.10.251:3268
securityPrincipal.primary=cn=proxyagent,cn=users,dc=UK,dc=mycompany,dc=com
securityCredentials.primary=abc
providerUrl.secondary=ldap://192.168.0.251:3268
securityPrincipal.secondary=cn=proxyagent,cn=users,dc=UK,dc=mycompany,dc=com
securityCredentials.secondary=abc
Secondary settings can also be specified for a particular domain:
providerUrl.primary.uk=ldap://192.168.10.251:3268
securityPrincipal.primary.uk=cn=proxyagent,cn=users,dc=UK,dc=mycompany,dc=com
securityCredentials.primary.uk=abc
providerUrl.secondary.uk=ldap://192.168.0.251:3268
securityPrincipal.secondary.uk=cn=proxyagent,cn=users,dc=UK,dc=mycompany,dc=com
securityCredentials.secondary.uk=abc
```

Multiple LDAP Servers and Domains

Connector can support multiple LDAP servers and domains. Default LDAP must be switched off, but otherwise, the initial configuration is similar to that for a single LDAP server.

```java
useLdap=false
providerUrl=ldap://192.168.0.251:3268
securityPrincipal=cn=proxyagent,cn=users,dc=UK,dc=company,dc=com
securityCredentials=mBxm8shsZArd1ds3dbw_-DsSBrGK5x
ldapRefreshTimeout=3600000
```

Connector has the ability to specify a separate LDAP server for specific domains. For example, given an authorization user name obtained from the NTLM challenge as "UK\somebody" you can set up a specific LDAP server that is queried to get the group details for this user. For example:

```java
useLdap.UK=true
providerUrl.UK=ldap://127.0.0.1:3268
securityPrincipal.UK=cn=proxyagent,cn=users,dc=company,dc=com
```
The `ldapRefreshTimeout` property is global. It is configured for all the configured LDAP servers.

To configure specific LDAP servers for querying groups for certain domains, you need to use the domain name as a suffix for the LDAP properties, as in the previous example.

The `useLdap.<domain>` properties are always mandatory for all defined LDAP servers. The name of the domain is case sensitive.

If any of the other properties for the extending LDAP servers is missing, the property is inherited from the default LDAP setting.

Unless a default LDAP server configuration is defined, Connector uses default values. The default configuration acts as fallback for domains for which there is no explicit configuration. For example, to find the domains for the user `WinNT://SOMEDOMAIN\someuser` and where there is no configuration for `SOMEDOMAIN` domain, the default configuration is used.

It is also possible to assign multiple domain names to a domain configuration. You can do this with the domains property, for example:

```properties
useLdap.UK=true
providerUrl.UK=ldap://127.0.0.1:3268
securityPrincipal.UK=cn=proxyagent, cn=users, dc=company, dc=com
domains.UK=DOMA, DOMB, DOMC
securityCredentials.UK="?Y*FH
```

In this example, the UK domain configuration is assigned to the domains DOMA, DOMB, and DOMC, but not UK. To include the UK domain name for this configuration, you would need to define it in the list of the acceptable domains:

```properties
domains.UK=DOMA, DOMB, DOMC, UK
```

### Groups

Connector enables you to manage connections based on group membership. You can enable multiple authentication keys by group, exclude groups from Web filtering, and set the depth of nested groups.

**Note**

Group names such as `WinNT://UK\dev` must have the backslash escaped in the agent.properties file, such as `WinNT://UK\dev`.

### Multiple Authentication Keys

Multiple authentication keys can be specified in the agent.properties file. Authentication keys are mapped to the groups the user belongs to. For example:

```properties
licence.1-authkey1
licence.1.groups=a,b,c
domains.2-authkey2
domains.2.groups=d,e
```
In the example, if a user belongs to group a, b, or c, the licence.1 authentication key is used. If the user belongs to group d or e, the licence.2 authentication key is used. If the user does not belong to any of these groups, the default authentication key is used.

The groups can also contain trailing wildcards. For example:

```
licence.1.groups=WinNT://UK*
```

This would match all the groups that start with WinNT://UK.

### Excluding Groups

Many organizations use a large number of directory groups for different internal functionality. Adding all these groups into the headers would create a large overhead on each request. To avoid this, Connector can exclude any groups that are not relevant to web filtering.

Groups to be excluded are specified in the agent.properties file. You can either have global group exclusions which apply across all directories or specific exclusions on a directory basis.

Any combination of filters is permitted. If no filters are defined or if both are empty, there is no group filtering at all.

#### Global Group Exclusions

This exclusion applies to all groups determined by both LDAP and Domain Controller querying. The property for this filter is called groupInclude. If it is absent or empty, there is no global group filtering at this level. Group names are case insensitive and must be comma separated. For example:

```
groupInclude=Winnt://UK\Dev, Winnt://UK\others
```

Domain names and group names can include wildcards. For example:

```
groupInclude=Winnt://UK\group*, Winnt://UK\Dev, Winnt://domain\group*
```

#### LDAP Group Exclusions

These exclusions apply per LDAP setting only. The default LDAP configuration cannot have group filters (use the global group filters in this case). If it is absent or empty, there is no global group filtering at this level. For example:

```
useLdap.UK=true
providerUrl.UK=ldap://127.0.0.1:3268
securityPrincipal.UK=cn=proxyagent,cn=users,dc=UK,dc=domain, dc=com
securityCredentials.UK=mBxmSArld1d7s3dbw_-DsSBrGK5x
groupInclude.UK=WinNT://UK\dev,Winnt://UK\test
```

#### Nested Groups

Connector supports nested groups. By default, the depth for the nested group hierarchy is five. Nested groups can be configured by adding the following properties to the agent.properties file:

- `groupslookuprecursive.depth`. The depth for the nested groups, the default value is 1. Nesting can be switched on by setting the value to 2 or higher.
• groupslookup.recursive.exclude. A comma separated list of groups which should be excluded from nesting.

This groupslookup.recursive.exclude property can also be set for the domain. For example:
groupslookup.recursive.exclude.UK

This contains the exception groups for the UK domain.

Configurable Domain Name Mapping

If there are two users with the same sAMAccount name in different domains of a multi-domain forest, Connector may not always identify both users correctly. In this case, you can map the domain name provided during the NTLM authentication process to the sequence of DC values that appear in the distinguished name.

There are two modes of domain name mapping: static (default) and dynamic.

To enable static domain name mapping, in the agent.properties file, set grouplookup.process to 2 (default) and manually specify static maps for each domain. For example:

domainMap.GLOBEXPML=DC=uk,DC=global,DC=example,DC=com

To enable dynamic domain name mapping, in the agent.properties file, set grouplookup.process to 3 and ensure that TCP port 389 is up and reachable on a Domain Controller.

The dynamic mode supports running simultaneously with static maps in which the static maps have priority over the dynamic maps.
Agent Properties

- Overview, page 85
- Agent Properties, page 85

Overview

The agent.properties file contains the configuration settings for Connector. Typically, properties containing lists do not support the uses of spaces between separators.

A comment line containing an incremental build number is automatically added to the agent.properties file and log file. For example, # Build Number: 175. This corresponding build number may be helpful for tracking and troubleshooting.

Caution

Before changing the settings of the agent.properties file you should discuss your requirements with customer support. In the worst case, certain settings could lead to Connector effectively blocking all traffic.

Agent Properties

Table 1: Connector Settings in the Agent Properties File

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
<th>Default</th>
<th>Alternate</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;exception name&gt;-exception_pattern</td>
<td>See Host Exceptions, on page 75.</td>
<td></td>
<td>&lt;pattern1&gt; [,&lt;pattern2&gt;...]</td>
</tr>
<tr>
<td>&lt;exception name&gt;-keepalive.enable</td>
<td>See Host Exceptions, on page 75.</td>
<td>TRUE</td>
<td>FALSE</td>
</tr>
<tr>
<td>&lt;exception name&gt;-tunnel</td>
<td>See Host Exceptions, on page 75.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>&lt;exception name&gt;-primaryProxy</td>
<td>See Host Exceptions, on page 75.</td>
<td></td>
<td>&lt;IP address or host name&gt;</td>
</tr>
<tr>
<td>&lt;exception name&gt;-primaryProxyPort</td>
<td>See Host Exceptions, on page 75.</td>
<td></td>
<td>&lt;port&gt;</td>
</tr>
<tr>
<td>&lt;exception name&gt;-secondaryProxy</td>
<td>See Host Exceptions, on page 75.</td>
<td></td>
<td>&lt;IP address or host name&gt;</td>
</tr>
<tr>
<td>&lt;exception name&gt;-secondaryProxyPort</td>
<td>See Host Exceptions, on page 75.</td>
<td></td>
<td>&lt;port&gt;</td>
</tr>
<tr>
<td>&lt;exception name&gt;-tertiaryProxy</td>
<td>See Host Exceptions, on page 75.</td>
<td></td>
<td>&lt;IP address or host name&gt;</td>
</tr>
<tr>
<td>&lt;exception name&gt;-tertiaryProxyPort</td>
<td>See Host Exceptions, on page 75.</td>
<td></td>
<td>&lt;port&gt;</td>
</tr>
<tr>
<td>acceptPreemptiveBasicAuth</td>
<td>Enable preemptive BASIC authentication in addition to NTLM when useNtlm is TRUE.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>acceptUPN</td>
<td>When useNtlm is TRUE accepts a user principle name (UPN) or a sAMAccountName. By default the user name provided during NTLM authentication must be a sAMAccountName.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>addTowerFqdnHeader</td>
<td>In release 3.0.3.0 and newer, Connector sends the DNS-FQDN field with the FQDNs of towers used. The default setting of true allows Connector to send the DNS-FQDN data. For troubleshooting purposes, this setting can be set to false, so that the DNS-FQDN field is not included in the X-ScanSafe-Data header sent to towers.</td>
<td>TRUE</td>
<td>FALSE</td>
</tr>
<tr>
<td>allowedTlsCipherSuitesList</td>
<td>Location of the file containing the list of allowed TLS cipher suites.</td>
<td>./etc/ciphers.properties</td>
<td>&lt;relative path from location of agent.properties file&gt;</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
<td>--------------------</td>
<td>------------</td>
</tr>
<tr>
<td>allowedTlsProtocols</td>
<td>Comma separated list of allowed TLS protocols in order of preference. For example, in TLSv1.2,TLSv1.1,TLSv1, the first protocol, TLSv1.2, is used by default. If this property is empty, Connector will fail to start.</td>
<td>TLSv1.2, TLSv1.1</td>
<td>&lt;protocol&gt;</td>
</tr>
<tr>
<td>aup.enable</td>
<td>Enable Acceptable Use Policy support for Connector in standalone mode. This is not supported in enterprise mode.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>auth.realm</td>
<td>Name of the realm that appears in the basic authentication dialog.</td>
<td></td>
<td>&lt;realm&gt;</td>
</tr>
<tr>
<td>backlog.size</td>
<td>Maximum number of connections to queue.</td>
<td>100 (Windows), 900 (Linux)</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>brand.file</td>
<td>File that applies any branding text.</td>
<td>branding.properties</td>
<td>&lt;filename&gt;</td>
</tr>
<tr>
<td>connectorId</td>
<td>Unique string (up to 32 characters in length) used to identify the Connector instance for reporting purposes. Can contain only alphanumeric characters, hyphens (-), and underscores (_).</td>
<td></td>
<td>&lt;string&gt;</td>
</tr>
<tr>
<td>defaultUpstreamPort</td>
<td>Value used when upstream ports for primary, secondary, or tertiary upstream proxies are not specified. For example, if secondaryProxy is specified and secondaryProxyPort is not, the defaultUpstreamPort value will be used.</td>
<td>8080</td>
<td>&lt;port&gt;</td>
</tr>
<tr>
<td>domains.&lt;domain&gt;</td>
<td>Comma separated list of domains to be grouped under a single domain for LDAP queries. This will override individual domain settings.</td>
<td></td>
<td>&lt;domain&gt;</td>
</tr>
<tr>
<td>domainMap.&lt;domain&gt;</td>
<td>Comma separated list of domain controllers for static mapping.</td>
<td></td>
<td>DC=&lt;domain&gt;</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------</td>
<td>-----------</td>
</tr>
<tr>
<td>elb.buckets</td>
<td>Specifies how many upstream servers Connector should do load balancing to.</td>
<td>1</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>elb.enable</td>
<td>Used to enable enterprise load balancing.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>elb.mode</td>
<td>Sets the load balancing policy.</td>
<td>client-ip</td>
<td>host</td>
</tr>
<tr>
<td>encryptHeaders</td>
<td>Sets whether or not Connector encrypts headers added to a request. Do not change this setting unless explicitly instructed to do so by a support engineer.</td>
<td>TRUE</td>
<td>FALSE</td>
</tr>
<tr>
<td>encryptionVersion</td>
<td>Sets the header encoding: 0 - hex, 1 - base-64 encoded and gzipped (smallest but increases CPU load), or 2 - base-64 (larger than 1 but faster)</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>groupInclude</td>
<td>Comma separated list of groups to be sent to Cloud Web Security. All other groups (which are not relevant to Web filtering) are excluded. Note the double \ and /. The domain and group only are case insensitive. Wildcards are supported.</td>
<td>all groups</td>
<td>WinNT://&lt;domain&gt;\&lt;group&gt; WinNT://&lt;domain*&gt;\&lt;group*&gt;</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------</td>
<td>-----------</td>
</tr>
<tr>
<td>groupLookupNoDomainMatch</td>
<td>Specifies how Connector responds when it recognizes that a non-domain user and a domain user with matching user names are not members of the same domain. When set to UseFirst, Connector assumes the two users are the same person, and associates the local non-domain user with the group memberships of the domain user. When set to SkipGroups, Connector assumes the two users are different, and does not associate the local non-domain user with any group memberships. In the latter case, the Tower applies the default no group membership policy to the HTTP request.</td>
<td>UseFirst</td>
<td>SkipGroups</td>
</tr>
<tr>
<td>groupLookup.process</td>
<td>Determines the group mapping process. To enable static domain name mapping, set groupLookup.process to 2 and manually specify static maps for each domain. To enable dynamic domain name mapping, set groupLookup.process to 3 and ensure that TCP port 389 is up and reachable on a Domain Controller. Customers that currently use a setting of 2 should consult customer support before moving to a setting of 3. See Configurable Domain Name Mapping, page 1-14.</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>groupslookup.recursive.depth</td>
<td>Depth for nested groups. A setting of 1 switches off support for nested groups.</td>
<td>1</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>groupslookup.recursive.exclude</td>
<td>Comma separated list of exception groups which should not be included in nesting.</td>
<td>no groups</td>
<td>WinNT://&lt;domain&gt;&lt;group&gt;</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>---------</td>
<td>-----------</td>
</tr>
<tr>
<td><code>http.failover.alivePoll</code></td>
<td>Whether to check if the upstream Cloud Web Security proxy server is available.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td><code>http.failover.alivePollDelaySec</code></td>
<td>Delay in seconds between checks.</td>
<td>30</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td><code>http.failover.aliveRepeatsToWhiteList</code></td>
<td>Number of successful requests before removal from the blacklist.</td>
<td>1</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td><code>http.failover.failPollDelaySec</code></td>
<td>How often in seconds to poll blacklisted proxy servers.</td>
<td>3</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td><code>http.failover.failRepeatsToBlacklist</code></td>
<td>Number of failures before adding to blacklist.</td>
<td>5</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td><code>http.failover.numberOfRetriesForResource</code></td>
<td>Number of retries to count as failure.</td>
<td>2</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td><code>httpAddress</code></td>
<td>Interface to bind to for HTTP. In workgroup mode, limits Connector to a single interface for use with multi-homed servers.</td>
<td>accept connections on any interface</td>
<td>&lt;IP address&gt;</td>
</tr>
<tr>
<td><code>httpPort</code></td>
<td>Port which Connector listens to for HTTP traffic.</td>
<td>8080</td>
<td>&lt;port&gt;</td>
</tr>
<tr>
<td><code>icap.generate.random.istag</code></td>
<td>Enables Connector to respond with random ISTags required by some gateways.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td><code>icapAddress</code></td>
<td>Interface to bind to for ICAP. In workgroup mode, limits Connector to a single interface for use with multi-homed servers.</td>
<td>accept connections on any interface</td>
<td>&lt;IP address&gt;</td>
</tr>
<tr>
<td><code>icapPort</code></td>
<td>Port on which Connector should listen for ICAP traffic.</td>
<td>1344</td>
<td>&lt;port&gt;</td>
</tr>
<tr>
<td><code>install.mode</code></td>
<td>Sets workgroup or enterprise mode.</td>
<td>enterprise.install workgroup.install</td>
<td></td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------</td>
<td>-----------</td>
</tr>
<tr>
<td>keepalive.enable</td>
<td>Enabling keep-alive creates persistent connections. This is a requirement for NTLM pass-through.</td>
<td>TRUE</td>
<td>FALSE</td>
</tr>
<tr>
<td>ldap.accountdisabled.attribute</td>
<td>Where a value is specified, the name of the attribute that flags if the user is allowed to browse. A user with a 'disabled' account in the LDAP server is not allowed to browse, even if the correct user name and password are provided at the basic authentication dialog.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ldap.base.dn</td>
<td>Base DN in the LDAP tree where the query starts.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ldap.connect.timeout</td>
<td>Number of milliseconds before connect time-out.</td>
<td>5000</td>
<td></td>
</tr>
<tr>
<td>ldap.failover.alivePoll</td>
<td>When set to TRUE, the LDAP Resource Manager polls resources to determine if they are available.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>ldap.failover.alivePollDelay</td>
<td>Delay in seconds between polling available LDAP resources.</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>ldap.failover.aliveRepeatsToWhitelist</td>
<td>Number of successful repeat attempts to connect to an LDAP server with its status set to unavailable before its status is changed to available.</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>ldap.failover.failPollDelay</td>
<td>Delay between attempts to connect to LDAP servers that have had their status changed to unavailable.</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>ldap.failover.failRepeatsToBlacklist</td>
<td>Number of failures before the LDAP server's status is changed to unavailable.</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>ldap.failover.numberOfRetriesForResource</td>
<td>Number of retries to count as failure. Applied to both the primary and secondary LDAP server.</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>ldap.group.attr</td>
<td>Name of the group attribute in the LDAP server configuration.</td>
<td>ou</td>
<td></td>
</tr>
<tr>
<td>ldap.group.attr.string.parse</td>
<td>Name of the attribute for parsing out the group name from an LDAP query response. For example, if the response to the group query is ou=mygroup, o=mycompany, l=location then by setting the ldap.group.attr.string.parse to ou you would derive the group name mygroup.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ldap.read.timeout</td>
<td>Number of milliseconds before read time-out.</td>
<td>5000</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>ldap.type</td>
<td>Type of LDAP in use, either Active Directory or generic.</td>
<td>ad</td>
<td>generic</td>
</tr>
<tr>
<td>ldap.user.attr</td>
<td>Name of the user attribute in the LDAP server configuration.</td>
<td>uid</td>
<td></td>
</tr>
<tr>
<td>ldapRefreshTimeout</td>
<td>Amount of time in milliseconds that Connector should remember a user's group details before querying the LDAP/Active Directory server again. This can greatly reduce the number of requests made via LDAP and increase the speed at which Connector services requests.</td>
<td>3600000</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>licence</td>
<td>Company, Group, or User authentication key generated in the portal and used to identify computers where the egress IP has a dynamically assigned IP address.</td>
<td></td>
<td>&lt;authentication key&gt;</td>
</tr>
<tr>
<td>local.response.html.file</td>
<td>HTTP error 503 page.</td>
<td>etc/localresponse.html</td>
<td>&lt;relative path from location of agent.properties file&gt;</td>
</tr>
<tr>
<td>lowercase.user</td>
<td>Make user names lowercase.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>ntlm.authenticate</td>
<td>Enables validation of credentials provided by the user's web browser.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------------</td>
<td>---------------</td>
</tr>
<tr>
<td>ntlm.cache.timeout</td>
<td>When ntlm.authenticate is TRUE, sets the period in milliseconds after which cached credentials expire. The default period is one day.</td>
<td>86400000</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>ntlm.dc.primary</td>
<td>Address of the primary Windows Domain Controller. This must be specified if ntlm.authenticate or ntlm.lookup.groups are set to true.</td>
<td></td>
<td>&lt;IP address or host name&gt;</td>
</tr>
<tr>
<td>ntlm.dc.secondary</td>
<td>Address of the secondary Windows Domain Controller.</td>
<td></td>
<td>&lt;IP address or host name&gt;</td>
</tr>
<tr>
<td>ntlm.dc.tertiary</td>
<td>Address of the tertiary Windows Domain Controller.</td>
<td></td>
<td>&lt;IP address or host name&gt;</td>
</tr>
<tr>
<td>ntlm.enable.passive.v2</td>
<td>When ntlm.authenticate is FALSE, enables passive NTLMv2 authentication.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>ntlm.icap.auth.password</td>
<td>Password that Connector uses when authenticating with an Active Directory/NT4 domain. Used only in ICAP mode.</td>
<td></td>
<td>&lt;password&gt;</td>
</tr>
<tr>
<td>ntlm.icap.auth.user</td>
<td>User name Connector uses to identify itself to an Active Directory/NT4 domain. Used only in ICAP mode. Note the double \ and /. The domain and group only are case insensitive.</td>
<td></td>
<td>WinNT:/&lt;domain&gt;&lt;user name&gt;</td>
</tr>
<tr>
<td>ntlm.lookup.groups</td>
<td>Enables group lookups via NTLM using the Domain Controller. Overrides the LDAP.lookup.groups setting when TRUE.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>ntlm.preauth.domain</td>
<td>Domain controller used for SMB signing. The ntlm.preauth settings are required when using Windows Server 2003 or later.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ntlm.preauth.username</td>
<td>User name of a normal user of the domain controller.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>-------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------</td>
<td>------------</td>
</tr>
<tr>
<td>ntlm.preauth.password</td>
<td>Password of the user of the domain controller.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ntlm.timeout</td>
<td>Number of milliseconds before time-out while waiting for a server response.</td>
<td>10000</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>ntlmIpExceptions</td>
<td>Comma separated list of IP addresses (not hostnames) of computers you wish to exclude from NTLM authentication requests. Supports Classless Inter-Domain Routing (CIDR) notation.</td>
<td></td>
<td>&lt;IP address&gt;</td>
</tr>
<tr>
<td>onErrorLdapRefreshTimeout</td>
<td>Length of time in milliseconds that Connector should wait before attempting to update the user’s group details following an LDAP or Active Directory server failure. A smaller value enables the Connector to update the user’s group details more quickly after the server is back online, but the smaller value will also have a negative impact on performance while the server is unavailable.</td>
<td>300000</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>parse.x.forwarded.for</td>
<td>When TRUE, the IP address in the X-Forwarded-For header will be available for reporting and for determining IP address based policy (if any). This is required when there is a proxy between the client and Connector.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>passwordRequired</td>
<td>Prevent users from authenticating with blank passwords when useBasic is TRUE.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>policyAllowBrowsingWhenDcIsDown</td>
<td>When set to TRUE enables a user to connect to the Internet when NTLM authentication is enabled if the domain controller is unavailable.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------</td>
<td>-----------</td>
</tr>
<tr>
<td>pool.keep.alive.time</td>
<td>Time in seconds after which idle threads are removed from the thread pool. The default is one hour.</td>
<td>3600</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>pool.max.size</td>
<td>Maximum number of threads. Value can be increased depending on the number of simultaneous connections you want to serve. Recommended maximum values are 2500 for Windows and 2800 for Linux. For more information, see the Release Notes for Cisco CWS Connector.</td>
<td>1500</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>pool.prestart.corethreads</td>
<td>Create threads on startup.</td>
<td>TRUE</td>
<td>FALSE</td>
</tr>
<tr>
<td>pool.queue.size</td>
<td>Maximum number of HTTP requests stored in the input queue.</td>
<td>50</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>pool.start.size</td>
<td>Minimum number of threads created on startup.</td>
<td>250</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>primaryProxy</td>
<td>Primary Cloud Web Security proxy included in your provisioning email. Set this value to DIRECT to bypass the proxy and go direct to the origin server.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>primaryProxyPort</td>
<td>Primary Cloud Web Security proxy port included in your provisioning email.</td>
<td></td>
<td>&lt;port&gt;</td>
</tr>
<tr>
<td>primaryProxyType</td>
<td>Sets whether SSL tunneling is enabled for the primary proxy.</td>
<td>PLAIN</td>
<td>SSL</td>
</tr>
<tr>
<td>primaryProxyValidateCertificate</td>
<td>If set to true, Connector validates whether primary proxy certificate has been signed by trusted root certificate.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>providerUrl.primary[.&lt;domain&gt;]</td>
<td>Primary LDAP/Active Directory server queried by Connector. The primary part of the property is optional. Optionally a domain may be included.</td>
<td>ldap://&lt;IP address or host name&gt;:3268</td>
<td></td>
</tr>
<tr>
<td>providerUrl.secondary[.&lt;domain&gt;]</td>
<td>Secondary LDAP/Active Directory server queried by Connector. Optionally a domain may be included.</td>
<td>ldap://&lt;IP address or host name&gt;:3268</td>
<td></td>
</tr>
<tr>
<td>providerUrl.tertiary[.&lt;domain&gt;]</td>
<td>Tertiary LDAP/Active Directory server queried by Connector. Optionally a domain may be included.</td>
<td>ldap://&lt;IP address or host name&gt;:3268</td>
<td></td>
</tr>
<tr>
<td>proxyConnHdrOption</td>
<td>Sets the header type to send to the client in response to messages that are associated with proxy authentication: 1=proxy-connection, 2=connection, 3=proxy-connection and connection. Do not change this setting unless explicitly instructed to do so by a support engineer.</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>proxyDns.default.negative.ttl</td>
<td>Timeout value in seconds before a next attempt to resolve a proxy IP address, in case the previous resolution attempt failed. Positive number from 1 to 3600.</td>
<td>10</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>proxyDns.default.ttl</td>
<td>Time-to-live (TTL) value in seconds for internal Java-based DNS resolution. Also used as default TTL value for DNS records that were resolved using DnsJava library. Positive number from 1 to 3600.</td>
<td>30</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>proxyDns.failover.numberOfRetriesForResource</td>
<td>Number of additional times the failover DNS resource tries to resolve IP address. Positive number from 1 to 5.</td>
<td>1</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>proxyDns.nameServers</td>
<td>In release 3.0.4.0 and newer, Connector can use internal Java-based DNS resolution. This property contains a comma separated list of DNS servers used for tower name resolution with the primary, secondary, and tertiary proxies only. Accepts up to five IPv4 addresses. If any other format is used, Connector stops and logs an error message in the connector.log file. System-wide DNS servers used if internal DNS resolution is not configured.</td>
<td>&lt;IP address&gt;</td>
<td></td>
</tr>
<tr>
<td>proxyDns.tcp</td>
<td>Whether to use TCP to connect to the DNS server. When set to true, TCP is used. When set to false, UDP is used.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>proxyDns.timeout</td>
<td>UDP/TCP connection timeout in milliseconds. Failover manager should check its timeout and ensure it is greater than the connection timeout. Positive number from 100 to 60000.</td>
<td>3000</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>publicKeyFile</td>
<td>Location of the public key used to encrypt headers. Do not change this setting unless explicitly instructed to do so by a support engineer.</td>
<td>etc/publicKey.txt</td>
<td>&lt;relative path from location of agent.properties file&gt;</td>
</tr>
<tr>
<td>read.timeout.downstream</td>
<td>Number of milliseconds before downstream read time-out.</td>
<td>10000</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>read.timeout.upstream</td>
<td>Number of milliseconds before upstream read time-out.</td>
<td>182000</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>secondaryProxy</td>
<td>Secondary Cloud Web Security proxy included in your provisioning email. Set this value to DIRECT to bypass the proxy and go direct to the origin server.</td>
<td>DIRECT</td>
<td></td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------</td>
<td>-----------</td>
</tr>
<tr>
<td>secondaryProxyPort</td>
<td>Secondary Cloud Web Security proxy port included in your provisioning email</td>
<td></td>
<td>&lt;port&gt;</td>
</tr>
<tr>
<td>secondaryProxyType</td>
<td>Sets whether SSL tunneling is enabled for the secondary proxy.</td>
<td>PLAIN</td>
<td>SSL</td>
</tr>
<tr>
<td>secondaryProxyValidateCertificate</td>
<td>If set to true, Connector validates whether secondary proxy certificate has been signed by trusted root certificate.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>securityAuthentication[,primary]</td>
<td>LDAP security strength. The .primary part of the property is optional.</td>
<td>none</td>
<td>simplestrong</td>
</tr>
<tr>
<td>securityAuthentication.secondary</td>
<td>LDAP security strength.</td>
<td>none</td>
<td>simplestrong</td>
</tr>
<tr>
<td>securityCredentials[,primary[,&lt;domain&gt;]]</td>
<td>Password for the primary account Connector uses when authenticating with an LDAP/Active Directory server. The .primary part of the property is optional. Optionally a domain may be included.</td>
<td></td>
<td>&lt;password&gt;</td>
</tr>
<tr>
<td>securityCredentials.secondary[,&lt;domain&gt;]</td>
<td>Password for the secondary account Connector uses when authenticating with an LDAP/Active Directory server. Optionally a domain may be included.</td>
<td></td>
<td>&lt;password&gt;</td>
</tr>
<tr>
<td>securityCredentials.tertiary[,&lt;domain&gt;]</td>
<td>Password for the tertiary account Connector uses when authenticating with an LDAP/Active Directory server. Optionally a domain may be included.</td>
<td></td>
<td>&lt;password&gt;</td>
</tr>
<tr>
<td>securityPrincipal[,primary[,&lt;domain&gt;]]</td>
<td>Primary user name Connector uses to identify itself to an LDAP/Active Directory server. The .primary part of the property is optional. Optionally a domain may be included.</td>
<td></td>
<td>cc=&lt;user name&gt;, cn=users, dc=&lt;company&gt;,dc=com</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------------------</td>
<td>------------------------------------</td>
</tr>
<tr>
<td>securityPrincipal. secondary[.&lt;domain&gt;]</td>
<td>Secondary user name Connector uses to identify itself to an LDAP/Active Directory server. Optionally a domain may be included.</td>
<td>cc=&lt;username&gt;, cn=users, dc=&lt;company&gt;,dc=com</td>
<td></td>
</tr>
<tr>
<td>securityPrincipal. tertiary[.&lt;domain&gt;]</td>
<td>Tertiary user name Connector uses to identify itself to an LDAP/Active Directory server. Optionally a domain may be included.</td>
<td>cc=&lt;username&gt;, cn=users, dc=&lt;company&gt;,dc=com</td>
<td></td>
</tr>
<tr>
<td>skip.wmp.authentication</td>
<td>Skip NTLM authentication for Windows Media Player.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>skipUnsupportedAuth</td>
<td>When TRUE, the authentication process is skipped when the client attempts an unsupported authentication protocol.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>sslTunnelTimeout</td>
<td>Number of milliseconds for which Connector should keep SSL tunnel requests open.</td>
<td>60000</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>support100Continue</td>
<td>When TRUE, the 100 (Continue) response code is supported. Do not change this setting unless explicitly instructed to do so by a support engineer.</td>
<td>TRUE</td>
<td>FALSE</td>
</tr>
<tr>
<td>SupportExpectHdr</td>
<td>When TRUE, the 'Expect' header is supported when the header value is 100-continue. Do not change this setting unless explicitly instructed to do so by a support engineer.</td>
<td>TRUE</td>
<td>FALSE</td>
</tr>
<tr>
<td>system.telemetry</td>
<td>Comma-separated list of system properties to include in the XSD header when upload.stasis is TRUE.</td>
<td>os.name, os.version</td>
<td>&lt;system property&gt;[, &lt;system property&gt;]</td>
</tr>
<tr>
<td>tertiaryProxy</td>
<td>Tertiary Cloud Web Security proxy included in your provisioning email. Set this value to DIRECT to bypass the proxy and go direct to the origin server.</td>
<td>&lt;IP address or host name&gt;</td>
<td>DIRECT</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------</td>
<td>-----------</td>
</tr>
<tr>
<td>tertiaryProxyPort</td>
<td>Tertiary Cloud Web Security proxy port included in your provisioning email.</td>
<td></td>
<td>&lt;port&gt;</td>
</tr>
<tr>
<td>tertiaryProxyType</td>
<td>Sets whether SSL tunneling is enabled for the tertiary proxy.</td>
<td>PLAIN</td>
<td>SSL</td>
</tr>
<tr>
<td>tertiaryProxyValidateCertificate</td>
<td>If set to true, Connector validates whether tertiary proxy certificate has been signed by trusted root certificate.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>upload.stats</td>
<td>Whether to send statistics (connection time, total connections, running time, thread count, OS name, and OS version) to Cloud Web Security.</td>
<td>TRUE</td>
<td>FALSE</td>
</tr>
<tr>
<td>upstream.connect.timeout</td>
<td>Number of milliseconds before upstream connection time-out.</td>
<td>3000</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>upstream.ssl.handshake.timeout</td>
<td>Number of milliseconds before upstream SSL proxy handshake time-out.</td>
<td>60000</td>
<td>&lt;number&gt;</td>
</tr>
<tr>
<td>upstreamNicAddress</td>
<td>On multi-homed servers with the Connector installed in workgroup mode, limits Connector to using only the specified interface when connecting to the proxy, another proxy or directly to the origin server. Leave this property undefined to enable the server to select the interface.</td>
<td></td>
<td>&lt;IP address&gt;</td>
</tr>
<tr>
<td>useBasic</td>
<td>Whether or not to use basic authentication.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>useHttp</td>
<td>Tells Connector whether or not to run in workgroup mode. It enables Connector to act as a simple Web proxy server, listening to all user web requests. If set to true, useIcap must be set to false.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
<td>Default</td>
<td>Alternate</td>
</tr>
<tr>
<td>------------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------</td>
<td>-----------</td>
</tr>
<tr>
<td>useIcap</td>
<td>Whether or not to listen for Web requests using ICAP. Used with ISA Server and ICAP compatible gateways. If set to true, useHttp must be set to false.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>useISA2000</td>
<td>Specifies if ISA 2000 Server is in use.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>useISA2004</td>
<td>Specifies if ISA Server 2004/2006 is in use.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>useLdap</td>
<td>Whether or not Connector should use LDAP to query Active Directory for the groups of which the user is a member.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>UseLdapResourceManager</td>
<td>LDAP Resource Manager, handles failovers from the primary LDAP server to the secondary LDAP server. You must not modify this value unless instructed to do so by customer support.</td>
<td>TRUE</td>
<td>FALSE</td>
</tr>
<tr>
<td>useNtlm</td>
<td>Enables Connector to collect users' internal IP addresses and user names using the NTLM authentication protocol. In most cases this authentication is transparent to the user.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>user.agent.skip.authentication</td>
<td>Enables user agent string matching.</td>
<td>FALSE</td>
<td>TRUE</td>
</tr>
<tr>
<td>user.agent.skip.authentication.regexp</td>
<td>When user.agent.skip.authentication is TRUE, skip authentication for user agent strings matching a regular expression, for example ( Chrome</td>
<td>Safari )\d). Note that if this is left blank when user.agent.skip.authentication is TRUE, authentication will be effectively switched off for all sites.</td>
<td>&lt;regular expression&gt;</td>
</tr>
</tbody>
</table>
Proxy Auto-Config File

Overview

Proxy Auto-Configuration (PAC) is a method used by web browsers to select a proxy for a given URL. The method for choosing a proxy is written as a JavaScript function contained in a PAC file. This file can be hosted locally or on a network. Web browsers can be configured to use the file either manually or, in Microsoft Windows environments, automatically using Group Policy Objects. This chapter explains the basics of using PAC files.

Operation

A PAC file is referenced each time a new URL is loaded. The host (cnn.com), the URL (cnn.com/images/logo.jpg), and other information (such as the local machine IP address) can be evaluated and rules based on this information used to determine whether to send the traffic through a proxy or directly to the Internet.

The following example compares the URL requested by the user with the URL ipcheckit.com/data/. If the URLs match, the PAC file instructs the browser to send the request directly to the Internet. This can be used if you need to except a section of a web site from going through Cloud Web Security. If the user had requested only ipcheckit.com, this rule would not apply.

```javascript
if (shExpMatch(url,"ipcheckit.com/data/*"))
    return "DIRECT";
```

In the following example, the local IP address of the machine making a web request is evaluated. If the IP address falls within the IP address range 10.10.1.*, the PAC file sends the request to proxy182.scansafe.net.
If this proxy is unavailable, it fails over to proxy137.scansafe.net. This can be used if you have different office locations using different IP address ranges with a Cloud Web Security proxy or Connector specific to each location.

```javascript
if (isInNet(myIpAddress(), "10.10.1.0", "255.255.255.0")) return "PROXY proxy182.scansafe.net:8080; PROXY proxy137.scansafe.net:8080";
```

Although a PAC file can have any name, normally it should be called proxy.pac.

**Deployment**

There are three ways to deploy a PAC file:

- **Local PAC.** In some cases, it may be appropriate to host the file on the local machine. This can be useful if the machine is likely to leave the network and doesn't have Anywhere+ installed. Rules can be specified in the PAC file to allow direct Internet access when off-network.

- **Share PAC.** The file can be hosted on a Windows share, assuming that the share is accessible to the machine and that the correct permissions have been applied. If the location of the PAC file is password protected, this may prevent the web browser from downloading the file.

- **Hosted PAC.** Hosting the file on a web server is the most popular and widely supported method. The only requirement is that the file be served by the web server with a specific MIME type (application/x-ns-proxy-autoconfig).

**Examples**

Direct all traffic through the first proxy. If it is unreachable, use the second proxy. If both are unavailable, go directly:

```javascript
function FindProxyForURL(url, host) {
    return "PROXY proxy1.my.com:8080; PROXY proxy2.my.com:8080; DIRECT";
}
```

Direct HTTP traffic as in the first example, but send all HTTPS traffic directly:

```javascript
function FindProxyForURL(url, host) {
    if (url.substring(0,6)="https") return "DIRECT"; else return "PROXY proxy1.my.com:8080; PROXY proxy2.my.com:8080; DIRECT";
}
```

Direct all traffic as in the first example, but send traffic for a given domain directly:

```javascript
function FindProxyForURL(url, host) {
    if (host=="my.com") return "DIRECT"; else return "PROXY proxy1.my.com:8080; PROXY proxy2.my.com:8080; DIRECT";
}
```

If the client computer is on the specified internal network, go through the proxy. Otherwise, go directly:

```javascript
function FindProxyForURL(url, host) {
    if (isInNet(myIpAddress(), "192.168.1.0", "255.255.255.0")) return "PROXY proxy1.my.com:8080; PROXY proxy2.my.com:8080; DIRECT";
    else return "DIRECT";
}
```
Example PAC file:

```javascript
function FindProxyForURL(url, host) {
    // Web sites you wish to go to direct and not through Cloud Web Security. This list would include internally hosted Web sites, intranets, and so on
    if (shExpMatch(url,"*.somecompany.co.uk*")) ||
        shExpMatch(url,"*.example.com*") ||
        shExpMatch(url,"*.antherexample.com" ))
    { return "DIRECT"; }

    // Internal IP address ranges that you need to be able to go to directly
    else if(isInNet(host, "xxx.xxx.xxx.xxx", "255.255.0.0") ||
            isInNet(host, "xxx.xxx.xxx.xxx", "255.255.0.0") ||
            isInNet (host, "xxx.xxx.xxx.xxx", "255.255.0.0"))
    { return "DIRECT"; }

    // Send all other HTTP HTTPS and FTP traffic to Web Services
    else { return "PROXY proxy.example1.com:8080"; } }
```

**Configuring**

**With Firefox:**

- In the Tools menu, click Options.
- Click the Network tab, and then click Settings.
- Click Automatic Proxy Configuration URL.
- Enter the URL of the PAC file in the box, and then click OK to save the settings.

**With Internet Explorer:**

- In the Tools menu, click Internet Options.
- Click the Connections tab, and then click LAN settings.
- Select “Use automatic configuration script.”
- Enter the URL of the PAC file in the box, and then click OK to save the settings.

**With Opera:**

- In the Tools menu, click Preferences.
- Click the Advanced tab, and then click Network in the left panel.
- Click Proxy Servers, and select “use automatic proxy configuration.”
- Enter the URL of the PAC file in the box, and then click OK to save the settings.

**With Safari for Windows:**

- In the Edit menu, click Preferences.
Host on a Network Share

It is possible to host a PAC file on a network share by using a VBScript to copy it to the local machine. This can be integrated with Windows logon scripting.

**Step 1**
Set a shared directory on a file server that everyone has access to.

**Step 2**
Create the proxy.pac file in the shared directory.

**Step 3**
Create a script.vbs file to copy the proxy.pac file from the network share to the local machine. For example:

```vbscript
Const OverwriteExisting = True
Set objFSO = CreateObject("Scripting.FileSystemObject")
Set objName= CreateObject("wscript.network")
objFSO.CopyFile "\\server_name\share_name\proxy.pac", "C:\proxy.pac",
OverwriteExisting
```

**Note**
Logon scripts run with the same permissions as the logged-on user and may not have write permission for the root of C:\. Ensure the VBScript copies the PAC file to a location where the user has write permission. However, the PAC file should be write-protected to prevent users from changing it.

**Step 4**
Open the Active Directory Users and Computers control panel.

**Step 5**
View the properties of the OU or Domain for which you want to apply the Group Policy.

**Step 6**
Edit the Group Policy.

**Step 7**
In the User Configuration area, expand Windows Settings and click Scripts (Logon/Logoff).

**Step 8**
Add a Logon Script.

**Step 9**
Browse to find the script.vbs file you created earlier, and then click OK.

**What to Do Next**

**Table 2: Local PAC URL Syntax**

<table>
<thead>
<tr>
<th>Browser</th>
<th>Windows XP</th>
<th>Windows 7 / Vista</th>
<th>MacOS X</th>
<th>GNU/Linux</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet Explorer</td>
<td>file://c:\data\proxy.pac</td>
<td>file://c:\data\proxy.pac</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Firefox</td>
<td>file:///c:/data/ proxy.pac</td>
<td>file:///c:/data/ proxy.pac</td>
<td>file://localhost/data/ proxy.pac</td>
<td>file:///data/ proxy.pac</td>
</tr>
<tr>
<td>Safari</td>
<td>Uses Internet Explorer settings</td>
<td>Uses Internet Explorer settings</td>
<td>file://localhost/ data/proxy.pac</td>
<td></td>
</tr>
</tbody>
</table>
### Proxy Auto-Config File

#### Hosting on a Network Share

<table>
<thead>
<tr>
<th>Browser</th>
<th>Windows XP</th>
<th>Windows 7 / Vista</th>
<th>MacOS X</th>
<th>GNU/Linux</th>
</tr>
</thead>
<tbody>
<tr>
<td>Opera</td>
<td>c:\data\proxy.pac</td>
<td>c:\data\proxy.pac</td>
<td>file://localhost/data/proxy.pac</td>
<td>file:///data/proxy.pac</td>
</tr>
</tbody>
</table>

**Table 3: Share PAC URL Syntax**

<table>
<thead>
<tr>
<th>Browser</th>
<th>Windows XP</th>
<th>Windows 7 / Vista</th>
<th>MacOS X</th>
<th>GNU/Linux</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet Explorer</td>
<td>file://\10.10.1.2\data\proxy.pac</td>
<td>file://\10.10.1.2\data\proxy.pac</td>
<td>file://localhost/Volumes/data/proxy.pac</td>
<td>file:///mnt/server/data/proxy.pac</td>
</tr>
<tr>
<td>Firefox</td>
<td>file://\10.10.1.2\data\proxy.pac</td>
<td>file:///\10.10.1.2\data\proxy.pac</td>
<td>file://localhost/Volumes/data/proxy.pac</td>
<td>file:///mnt/server/data/proxy.pac</td>
</tr>
<tr>
<td>Safari</td>
<td>Uses Internet Explorer settings</td>
<td>Uses Internet Explorer settings</td>
<td>file://localhost/Volumes/data/proxy.pac</td>
<td>file:///mnt/server/data/proxy.pac</td>
</tr>
<tr>
<td>Opera</td>
<td>\10.10.1.2\data\proxy.pac</td>
<td>\10.10.1.2\data\proxy.pac</td>
<td>file://localhost/Volumes/data/proxy.pac</td>
<td>file:///mnt/server/data/proxy.pac</td>
</tr>
</tbody>
</table>
Web Proxy Auto-Discovery Protocol

Overview

The Web Proxy Auto-Discovery (WPAD) protocol is a method used by web browsers to automatically locate a Proxy Auto-Config (PAC) file. The WPAD protocol uses DHCP and DNS systems and requires minimal configuration of the web browser. In most cases, all that is required is selecting a check box. WPAD is not an official Internet standard, but it is widely supported by modern web browsers. See PAC file Operation, on page 103.

Operation

WPAD can use DNS or DHCP to locate a PAC file. DHCP detection involves the URL being pushed to the end-user in the DHCP assignment, while DNS detection is based on an educated guess using known information about the DNS system.

A web browser must be instructed to use WPAD. In most web browsers, this is as achieved by selecting a check box or button. The feature is most commonly known and labeled as Auto-Detect. A web browser that supports both methods checks the DHCP assignment first, before attempting the DNS method.

The PAC file must have the file name wpad.dat for the DNS method to function.

When using both WPAD methods, the file must be served by the web server with the 'application/x-ns-proxy-autoconfig' MIME type.

If the web browser is unable to load a PAC file using the DHCP or DNS methods, it allows direct Internet access.
WPAD Using DHCP

A DHCP server must be configured to serve an additional setting in an IP address assignment; option 252. This option specifies the exact location of the PAC file. The file name does not need to follow any specific naming convention. However, if WPAD DNS is also to be used, the file must have the file name wpad.dat.

A web browser implementing this method sends the DHCP server a DHCPINFORM query. The DHCP server returns the expected IP settings, along with the 252 option, which defines the location of the PAC file. The web browser then downloads this PAC file from the URL provided.

WPAD Using DNS

The DNS method differs in that it guesses the location of a PAC file. On Windows, this is based on the domain the machine is joined to, while on Linux and Mac OS X this is based on the Search Domain(s) configured in the network settings.

When attempting the WPAD DNS method, the web browser prefixes the domain with wpad and attempts to download the file wpad.dat. For example, wpad.domain.com/wpad.dat.

In the following example, a Windows machine is joined to the domain uk.scansafe.com, and a PAC file with the file name wpad.dat is hosted on wpad.scansafe.com:

- After checking the network settings, the web browser identifies the host machine as being part of the domain uk.scansafe.com.
- The web browser attempts to resolve wpad.scansafe.com and fails.
- The web browser attempts and succeeds in resolving wpad.scansafe.com.
- The web browser attempts to download the PAC file wpad.scansafe.com/wpad.dat.

Manual Browser Configuration for Windows Clients

You may need to restart your web browser for changes to take effect.

- In Internet Explorer, select the Automatically detect settings check box in the Local Area Network (LAN) Settings dialog.
- In Firefox, click Auto-detect proxy settings for this network in the Connection Settings dialog.
- In Opera, open the Preferences dialog and click the Advanced tab. In the left menu, click Network and then Proxy Servers. Select the Use automatic proxy configuration check box, and enter the WPAD URL in the box. Ensure the other check boxes are cleared, and click OK.
- Safari for Windows uses Internet Explorer settings.

Deploying WPAD with Windows Server

Deploying WPAD on a Windows server enables you to centrally configure Internet Explorer users who are joined to a domain. It also makes it easy to configure the web browsers of users who are not members of a domain.
Beforehand, the following should be installed and configured on Windows Server:

- Internet Information Services (IIS)
- DHCP server
- DNS server
- Active Directory

Active Directory is not a functional requirement of WPAD, but it is recommended in order to simplify deployment.

Currently, only Internet Explorer offers complete support for the DHCP method. Therefore, the DNS method is essential for support with alternate web browsers.

Test your PAC file before renaming it wpad.dat and uploading it to the web site that will serve the file.

**Configure Internet Information Services**

Some web browsers cannot read a PAC file served with an incorrect MIME type, so you should configure IIS to use 'application/x-ns-proxy-autoconfig' for the '.dat' extension. When you have made the change, restart IIS.

When the entry for WPAD is created and activated, all users of the relevant DHCP scope receive the wpad.dat location, ready to be used by a user’s web browser.

**Create an Option 252 Entry in DHCP**

To automatically configure proxy settings:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Open the DHCP control panel.</td>
</tr>
<tr>
<td>Step 2</td>
<td>In the console tree, right-click DHCP server, click Set Predefined Options, and click Add.</td>
</tr>
<tr>
<td>Step 3</td>
<td>In the Name box, enter WPAD.</td>
</tr>
<tr>
<td>Step 4</td>
<td>In the Data type box, enter String.</td>
</tr>
<tr>
<td>Step 5</td>
<td>Clear the Array check box.</td>
</tr>
<tr>
<td>Step 6</td>
<td>In the Code box, enter 252.</td>
</tr>
<tr>
<td>Step 7</td>
<td>In the Description box, enter http://&lt;url&gt;[:&lt;port&gt;]/wpad.dat, and click OK.</td>
</tr>
</tbody>
</table>

**What to Do Next**

To confirm Option 252 is selected, right-click Server Options, and click Configure Options.
Enable Option 252 for a DHCP Scope

To configure Option 252 for a DHCP scope:

**Step 1** Open the DHCP control panel.

**Step 2** Right-click Scope Options, click Configure Options, and click Advanced.

**Step 3** In Vendor Class, click Standard Options.

**Step 4** In Available Options, click 252 Proxy Autodiscovery, and click OK.

Active Directory and Group Policy Objects

One of the benefits of WPAD is that it greatly reduces the amount of work it takes to configure a web browser for use with a PAC file/proxy.

Using Active Directory (AD) and Group Policy Objects (GPO), you can configure Internet Explorer settings automatically. A third-party tool called FirefoxADM is available for Firefox which allows configuration using GPO.