



Objects

Objects are reusable containers that define criteria that you want to use in policies or other settings. For example, network objects define host and subnet addresses.

Objects let you define criteria so that you can easily reuse the same criteria in different policies. When you update an object, all policies that use the object are automatically updated.

- [Object Types, on page 1](#)
- [Managing Objects, on page 3](#)

Object Types

You can create the following types of object. In most cases, if a policy or setting allows an object, you must use an object.

Object Type	Main Use	Description
AnyConnect Client Profile	Remote access VPN.	AnyConnect client profiles are downloaded to clients along with the AnyConnect client software. These profiles define many client-related options, such as auto connect on startup and auto reconnect, and whether the end user is allowed to change the option from the AnyConnect client preferences and advanced settings. See Configure and Upload Client Profiles .
Application Filter	Access control rules.	An application filter object defines the applications used in an IP connection, or a filter that defines applications by type, category, tag, risk, or business relevance. You can use these objects in policies to control traffic instead of using port specifications. See Configuring Application Filter Objects, on page 7 .
Certificates	Identity policies. Remote access VPN.	Digital certificates provide digital identification for authentication. Certificates are used for SSL (Secure Socket Layer), TLS (Transport Layer Security), and DTLS (Datagram TLS) connections, such as HTTPS and LDAPS. See Configuring Certificates, on page 16 .

Object Type	Main Use	Description
Geolocation	Security policies.	A geolocation object defines countries and continents that host the device that is the source or destination of traffic. You can use these objects in policies to control traffic instead of using IP addresses. See Configuring Geolocation Objects, on page 10 .
Identity Realm	Identity policies. Remote access VPN.	An identity realm is a directory server plus other attributes required to provide authentication services. The directory server contains information about the users and user groups who are allowed access to your network. See Configuring AD Identity Realms, on page 11 .
IKE Policy	VPN.	Internet Key Exchange (IKE) Policy objects define the IKE proposal used to authenticate IPsec peers, negotiate and distribute IPsec encryption keys, and automatically establish IPsec security associations (SAs). There are separate objects for IKEv1 and IKEv2. See Configuring the Global IKE Policy .
IPsec Proposal	VPN.	IPsec Proposal objects configure the IPsec proposal used during IKE Phase 2 negotiations. The IPsec proposal defines the combination of security protocols and algorithms that secure traffic in an IPsec tunnel. There are separate objects for IKEv1 and IKEv2. See Configuring IPsec Proposals .
Network	Security policies and a wide variety of device settings.	Network groups and network objects (collectively referred to as network objects) define the addresses of hosts or networks. See Configuring Network Objects and Groups, on page 3 .
Port	Security policies.	Port groups and port objects (collectively referred to as port objects) define the protocols, ports, or ICMP services for traffic. See Configuring Port Objects and Groups, on page 4 .
Security Zone	Security policies.	A security zone is a grouping of interfaces. Zones divide the network into segments to help you manage and classify traffic. See Configuring Security Zones, on page 5 .
Syslog Servers	Access control rules. Diagnostic logging.	A syslog server object identifies a server that can receive connection-oriented or diagnostic system log (syslog) messages. See Configuring Syslog Servers, on page 11 .
URL	Access control rules.	URL objects and groups (collectively referred to as URL objects) define the URL or IP addresses of web requests. See Configuring URL Objects and Groups, on page 8 .

Managing Objects

You can configure objects directly through the Objects page, or you can configure them while editing policies. Either method yields the same results, a new or updated object, so use the technique that suits your needs at the time.

The following procedure explains how you can create and manage your objects directly through the Objects page.



Note When you edit a policy or setting, if a property requires an object, you are shown a list of the ones that are already defined, and you select the appropriate object. If the desired object does not yet exist, simply click the **Create New Object** link shown in the list.

Procedure

Step 1 Select Objects.

The Objects page has a table of contents listing the available types of objects. When you select an object type, you see a list of existing objects, and you can create new ones from here. You can also see the object contents and type.

Step 2 Select the object type from the table of contents and do any of the following:

- To create an object, click the + button. The content of the objects differ based on type; see the configuration topic for each object type for specific information.
- To create a group object, click the **Add Group** (📁) button. Group objects include more than one item.
- To edit an object, click the edit icon (🔗) for the object. You cannot edit the contents of a pre-defined object.
- To delete an object, click the delete icon (🗑️) for the object. You cannot delete an object if it is currently being used in a policy or another object, or if it is a pre-defined object.

Configuring Network Objects and Groups

Use network group and network objects (collectively referred to as network objects) to define the addresses of hosts or networks. You can then use the objects in security policies for purposes of defining traffic matching criteria, or in settings to define the addresses of servers or other resources.

A network object defines a single host or network address, whereas a network group object can define more than one address.

The following procedure explains how you can create and edit objects directly through the Objects page. You can also create network objects while editing an address property by clicking the **Create New Network** link shown in the object list.

Procedure

Step 1 Select **Objects**, then select **Network** from the table of contents.

Step 2 Do one of the following:

- To create an object, click the + button.
- To create a group, click the **Add Group** (📁) button.
- To edit an object or group, click the edit icon (✎) for the object.

To delete an unreferenced object, click the trash can icon (🗑️) for the object.

Step 3 Enter a Name for the object and optionally, a description, and define the object contents.

Network Objects

Select the object **Type** and configure the contents:

- **Network**—Enter a network address using one of the following formats:
 - IPv4 network including subnet mask, for example, 10.100.10.0/24 or 10.100.10.0/255.255.255.0.
 - IPv6 network including prefix, for example, 2001:DB8:0:CD30::/60.
- **Host**—Enter a host IP address using one of the following formats:
 - IPv4 host address, for example, 10.100.10.10.
 - IPv6 host address, for example, 2001:DB8::0DB8:800:200C:417A or 2001:DB8:0:0:0DB8:800:200C:417A.

Network Groups

Click the + button to select network objects to add to the group. You can also create new objects.

Step 4 Click **OK** to save your changes.

Configuring Port Objects and Groups

Use port group and port objects (collectively referred to as port objects) to define the protocols, ports, or ICMP services for traffic. You can then use the objects in security policies for purposes of defining traffic matching criteria, for example, to use access rules to allow traffic to specific TCP ports.

A port object defines a single protocol, TCP/UDP port or port range, or ICMP service, whereas a port group object can define more than one service.

The system includes several pre-defined objects for common services. You can use these objects in your policies. However, you cannot edit or delete system-defined objects.





Note When creating port group objects, ensure that the combination of objects makes sense. For example, you cannot have a mixture of protocols in an object if you use it to specify both source and destination ports in an access rule. Exercise care when editing an object that is already being used, or you could invalid (and disable) policies that use the object.


The following procedure explains how you can create and edit objects directly through the Objects page. You can also create port objects while editing a service property by clicking the **Create New Port** link shown in the object list.

Procedure

Step 1 Select **Objects**, then select **Ports** from the table of contents.

Step 2 Do one of the following:

- To create an object, click the + button.
- To create a group, click the **Add Group** () button.
- To edit an object or group, click the edit icon () for the object.

To delete an unreferenced object, click the trash can icon () for the object.

Step 3 Enter a name for the object and optionally, a description, and define the object contents.

Port Objects

Select the **Protocol**, then configure the protocol as follows:

- **TCP, UDP**—Enter the single port or port range number, for example, 80 (for HTTP) or 1-65535 (to cover all ports).
- **ICMP, IPv6-ICMP**—Select the **ICMP Type** and optionally, the **Code**. Select **Any** for the type to apply to all ICMP messages. For information on the types and codes, see the following pages:
 - ICMP—<http://www.iana.org/assignments/icmp-parameters/icmp-parameters.xml>
 - ICMPv6—<http://www.iana.org/assignments/icmpv6-parameters/icmpv6-parameters.xml>
- **Other**—Select the desired protocol.

Port Groups

Click the + button to select port objects to add to the group. You can also create new objects.

Step 4 Click **OK** to save your changes.

Configuring Security Zones

A security zone is a grouping of interfaces. Zones divide the network into segments to help you manage and classify traffic. You can define multiple zones, but a given interface can be in one zone only.

The system creates the following zones during initial configuration. You can edit these zones to add or remove interfaces, or you can delete the zones if you no longer use them.

- **inside_zone**—Includes the inside interface. If the inside interface is a bridge group, this zone includes all the bridge group member interfaces instead of the inside Bridge Virtual Interface (BVI). This zone is intended to represent internal networks.
- **outside_zone**—Includes the outside interface. This zone is intended to represent networks external to your control, such as the Internet.

Typically, you would group interfaces by the role they play in your network. For example, you would place the interface that connects to the Internet in the **outside_zone** security zone, and all of the interfaces for your internal networks in the **inside_zone** security zone. Then, you could apply access control rules to traffic coming from the outside zone and going to the inside zone.


Before creating zones, consider the access rules and other policies you want to apply to your networks. For example, you do not need to put all internal interfaces into the same zone. If you have 4 internal networks, and you want to treat one differently than the other three, you can create two zones rather than one. If you have an interface that should allow outside access to a public web server, you might want to use a separate zone for the interface.


The following procedure explains how you can create and edit objects directly through the Objects page. You can also create security zones while editing a security zone property by clicking the **Create New Security Zone** link shown in the object list.

Procedure

Step 1 Select **Objects**, then select **Security Zones** from the table of contents.

Step 2 Do one of the following:

- To create an object, click the + button.
- To edit an object, click the edit icon () for the object.

To delete an unreferenced object, click the trash can icon () for the object.

Step 3 Enter a Name for the object and optionally, a description.

Step 4 In the **Interfaces** list, click + and select the interfaces to add to the zone.

The list shows all named interfaces that are not currently in a zone. You must configure an interface and give it a name before you can add it to a zone.

If all named interfaces are already in zones, the list is empty. If you are trying to move an interface to a different zone, you must first remove it from its current zone.

Note You cannot add a bridge group interface (BVI) to a zone. Instead, add the member interfaces. You can put the members into different zones.

Step 5 Click **OK** to save your changes.

Configuring Application Filter Objects

An application filter object defines the applications used in an IP connection, or a filter that defines applications by type, category, tag, risk, or business relevance. You can use these objects in policies to control traffic instead of using port specifications.

Although you can specify individual applications, application filters simplify policy creation and administration. For example, you could create an access control rule that identifies and blocks all high risk, low business relevance applications. If a user attempts to use one of those applications, the session is blocked.

You can select applications and application filters directly in a policy without using application filter objects. However, an object is convenient if you want to create several policies for the same group of applications or filters. The system includes several pre-defined application filters, which you cannot edit or delete.




Note Cisco frequently updates and adds additional application detectors via system and vulnerability database (VDB) updates. Thus, a rule blocking high risk applications can automatically apply to new applications without you having to update the rule manually.


The following procedure explains how you can create and edit objects directly through the Objects page. You can also create application filter objects while editing an access control rule by clicking the **Save As Filter** link after adding application criteria to the Applications tab.

Procedure

Step 1 Select **Objects**, then select **Application Filters** from the table of contents.

Step 2 Do one of the following:

- To create an object, click the + button.
- To edit an object, click the edit icon () for the object.

To delete an unreferenced object, click the trash can icon () for the object.

Step 3 Enter a Name for the object and optionally, a description.

Step 4 In the **Applications** list, click **Add +** and select the applications and filters to add to the object.

The initial list shows applications in a continually scrolling list. Click **Advanced Filter** to see the filter options and to get an easier view for selecting applications. Click **Add** when you have made your selections. You can repeat the process to add additional applications or filters.

Note Multiple selections within a single filter criteria have an OR relationship. For example, Risk is High OR Very High. The relationship between filters is AND, so Risk is High OR Very High, AND Business Relevance is Low OR Very Low. As you select filters, the list of applications in the display updates to show only those that meet the criteria. You can use these filters to help you find applications that you want to add individually, or to verify that you are selecting the desired filters to add to the rule.

Risks

The likelihood that the application is used for purposes that might be against your organization's security policy, from very low to very high.

Business Relevance

The likelihood that the application is used within the context of your organization's business operations, as opposed to recreationally, from very low to very high.

Types

The type of application:

- **Application Protocol**—Application protocols such as HTTP and SSH, which represent communications between hosts.
- **Client Protocol**—Clients such as web browsers and email clients, which represent software running on the host.
- **Web Application**—Web applications such as MPEG video and Facebook, which represent the content or requested URL for HTTP traffic.

Categories

A general classification for the application that describes its most essential function.

Tags

Additional information about the application, similar to category.

For encrypted traffic, the system can identify and filter traffic using only the applications tagged **SSL Protocol**. Applications without this tag can only be detected in unencrypted or decrypted traffic. Also, the system assigns the **decrypted traffic** tag to applications that the system can detect in decrypted traffic only, not encrypted or unencrypted.

Applications List (bottom of the display)

This list updates as you select filters from the options above the list, so you can see the applications that currently match the filter. Use this list to verify that your filter is targeting the desired applications when you intend to add filter criteria to the rule. If your intention is to add specific applications, select them from this list.

Step 5 Click **OK** to save your changes.

Configuring URL Objects and Groups

Use URL objects and groups (collectively referred to as URL objects) to define the URL or IP addresses of web requests. You can use these objects to implement manual URL filtering in access control policies.

A URL object defines a single URL or IP address, whereas a URL group object can define more than one URL or address.

When creating URL objects, keep the following points in mind:

- If you do not include a path (that is, there is no / character in the URL), the match is based on the server's hostname only. The hostname is considered a match if it comes after the `://` separator, or after any dot in

the hostname. For example, ign.com matches ign.com and www.ign.com, but it does not match verisign.com.

- If you include one or more / character, the entire URL string is used for a substring match, including the server name, path, and any query parameters. However, we recommend that you do not use manual URL filtering to block or allow individual web pages or parts of sites, as servers can be reorganized and pages moved to new paths. Substring matching can also lead to unexpected matches, where the string you include in the URL object also matches paths on unintended servers or strings within query parameters.
- The system disregards the encryption protocol (HTTP vs HTTPS). In other words, if you block a website, both HTTP and HTTPS traffic to that website is blocked, unless you use an application condition to target a specific protocol. When creating a URL object, you do not need to specify the protocol when creating an object. For example, use example.com rather than http://example.com.
- If you plan to use a URL object to match HTTPS traffic in an access control rule, create the object using the subject common name in the public key certificate used to encrypt the traffic. Also, the system disregards subdomains within the subject common name, so do not include subdomain information. For example, use example.com rather than www.example.com.

However, please understand that the subject common name in the certificate might be completely unrelated to a web site's domain name. For example, the subject common name in the certificate for youtube.com is *.google.com (this of course might change at any time). You will get more consistent results if you use the SSL Decryption policy to decrypt HTTPS traffic so that URL filtering rules work on decrypted traffic.





Note URL objects will not match HTTPS traffic if the browser resumes a TLS session because the certificate information is no longer available. Thus, even if you carefully configure the URL object, you might get inconsistent results for HTTPS connections.


The following procedure explains how you can create and edit objects directly through the Objects page. You can also create URL objects while editing a URL property by clicking the **Create New URL** link shown in the object list.

Procedure

Step 1 Select **Objects**, then select **URL** from the table of contents.

Step 2 Do one of the following:

- To create an object, click the + button.
- To create a group, click the **Add Group** () button.
- To edit an object or group, click the edit icon () for the object.

To delete an unreferenced object, click the trash can icon () for the object.

Step 3 Enter a Name for the object and optionally, a description.

Step 4 Define the object contents.

URL Objects

Enter a URL or IP address in the **URL** box. You cannot use wildcards in the URL.

URL Groups

Click the + button to select URL objects to add to the group. You can also create new objects.

Step 5 Click **OK** to save your changes.

Configuring Geolocation Objects

A geolocation object defines countries and continents that host the device that is the source or destination of traffic. You can use these objects in policies to control traffic instead of using IP addresses. For example, using geographical location, you could easily restrict access to a particular country without needing to know all of the potential IP addresses used there.

You can typically select geographical locations directly in a policy without using geolocation objects. However, an object is convenient if you want to create several policies for the same group of countries and continents.




Note To ensure that you are using up-to-date geographical location data to filter your traffic, Cisco strongly recommends that you regularly update the geolocation database (GeoDB).


The following procedure explains how you can create and edit objects directly through the Objects page. You can also create geolocation objects while editing a network property by clicking the **Create New Geolocation** link shown in the object list.

Procedure

Step 1 Select **Objects**, then select **Geolocation** from the table of contents.

Step 2 Do one of the following:

- To create an object, click the + button.
- To edit an object, click the edit icon () for the object.

To delete an unreferenced object, click the trash can icon () for the object.

Step 3 Enter a Name for the object and optionally, a description.

Step 4 In the **Continents/Countries** list, click **Add +** and select the continents and countries to add to the object. Selecting a continent selects all countries within the continent.

Step 5 Click **OK** to save your changes.

Configuring Syslog Servers

A syslog server object identifies a server that can receive connection-oriented or diagnostic system log (syslog) messages. If you have a syslog server set up for log collection and analysis, create objects to define them and use the objects in the related policies.

You can send the following types of events to the syslog server:


- Connection events. Configure the syslog server object on the following types of policy: access control rules and default action.
- Diagnostic events. See [Configuring Diagnostic Logging](#).


The following procedure explains how you can create and edit objects directly through the Objects page. You can also create syslog server objects while editing a syslog server property by clicking the **Add Syslog Server** link shown in the object list.

Procedure

Step 1 Select **Objects**, then select **Syslog Servers** from the table of contents.

Step 2 Do one of the following:

- To create an object, click the + button.
- To edit an object, click the edit icon () for the object.

To delete an unreferenced object, click the trash can icon () for the object.

Step 3 Configure the syslog server properties:

- **Device Interface**—Select the interface through which the syslog server is reached. If the server is accessible through a bridge group member interface, select the bridge group interface (BVI) instead.
- **IP Address**—Enter the IP address of the syslog server.
- **Port**—Enter the UDP port that the server uses for receiving syslog messages. The default is 514. If you change the default, the port must be in the range 1025 to 65535.

Step 4 Click **OK** to save your changes.

Configuring AD Identity Realms

An identity realm is a directory server plus other attributes required to provide authentication services. The directory server contains information about the users and user groups who are allowed access to your network.

For Active Directory, a realm is equivalent to an Active Directory domain.

Realms are used in the following policies:

- Identity—The realm provides user identity and group membership information, which you can then use in access control rules. The system downloads updated information about all users and groups every day in the last hour of the day (UTC). The directory server must be reachable from the management interface.

- Remote access VPN—The realm provides authentication services, which determine whether a connection is allowed. The directory server must be reachable from the RA VPN outside interface.

Work with your directory administrator to get the values required to configure the directory server properties.



Note If the directory server is not on an attached network or available through the default route, create a static route for the server. Select **Device > Routing > View Configuration** to create static routes.

The following procedure explains how you can create and edit objects directly through the Objects page. You can also create identity realm objects while editing a realm property by clicking the **Create New Identity Realm** link shown in the object list.


Before you begin

Ensure that time settings are consistent among the directory servers, Firepower Threat Defense device, and clients. A time shift among these devices can prevent successful user authentication. "Consistent" means that you can use different time zones, but the time should be the same relative to those zones; for example, 10 AM PST = 1 PM EST.

Procedure

Step 1 Select **Objects**, then select **Identity Realm** from the table of contents.

Step 2 Do one of the following:

- To create a realm, click the + button. You can create at most one realm.
- To edit a realm, click the edit icon () for the realm.

You cannot delete a realm once you create it. To stop using the realm, disable the configured feature that uses it.

Step 3 Configure the basic realm properties.

- **Name**—A name for the directory realm.
- **Type**—The type of directory server. Active Directory is the only supported type, and you cannot change this field.
- **Directory Username, Directory Password**—The distinguished username and password for a user with appropriate rights to the user information you want to retrieve. For Active Directory, the user does not need elevated privileges. You can specify any user in the domain. The username must be fully qualified; for example, Administrator@example.com (not simply Administrator).

Note The system generates ldap-login-dn and ldap-login-password from this information. For example, Administrator@example.com is translated as cn=administrator,cn=users,dc=example,dc=com. Note that cn=users is always part of this translation, so you must configure the user you specify here under the common name "users" folder.

- **Base DN**—The directory tree for searching or querying user and group information, that is, the common parent for users and groups. For example, `cn=users,dc=example,dc=com`. For information on finding the base DN, see [Determining the Directory Base DN, on page 15](#).
- **AD Primary Domain**— The fully qualified Active Directory domain name that the device should join. For example, `example.com`.

Step 4 Configure the directory server properties.

- **Hostname/IP Address**—The hostname or IP address of the directory server. If you use an encrypted connection to the server, you must enter the fully-qualified domain name, not the IP address.
- **Port**—The port number used for communications with the server. The default is 389. Use port 636 if you select LDAPS as the encryption method.
- **Encryption**—To use an encrypted connection for downloading user and group information, select the desired method, **STARTTLS** or **LDAPS**. The default is **None**, which means that user and group information is downloaded in clear text.
 - **STARTTLS** negotiates the encryption method, and uses the strongest method supported by the directory server. Use port 389. This option is not supported if you use the realm for remote access VPN.
 - **LDAPS** requires LDAP over SSL. Use port 636.
- **Trusted CA Certificate**—If you select an encryption method, upload a Certificate Authority (CA) certificate to enable a trusted connection between the system and the directory server. If you are using a certificate to authenticate, the name of the server in the certificate must match the server Hostname / IP Address. For example, if you use 10.10.10.250 as the IP address but `ad.example.com` in the certificate, the connection fails.

Step 5 Click the **Test** button to verify the system can contact the server.

The system uses separate processes and interfaces to access the server, so you might get errors indicating that the connection works for one type of use but not another, for example, available for Identity policies but not for remote access VPN. If the server cannot be reached, verify that you have the right IP address and host name, that the DNS server has an entry for the hostname, and so forth. You might need to configure a static route for the server. For more information, see [Troubleshooting Directory Server Connections, on page 13](#).

Step 6 Click **OK**.

Troubleshooting Directory Server Connections

The system uses different processes to communicate with your directory server depending on the feature. Thus, a connection for identity policies might work, whereas one for remote access VPN fails.

These processes use different interfaces to communicate with the directory server. You must ensure connectivity from these interfaces.

- Management interface, for: identity policies.
- Data interface, for: remote access VPN (outside interface).

When you configure the identity realm, use the **Test** button to verify that the connection can work. Failure messages should indicate the feature that is having connection problems. The following are the general issues you might encounter, based on authentication attributes and routing/interface configuration.

Directory user authentication issues.

If the problem is that the system could not log into the directory server because of the username or password, ensure that the name and password are correct and valid on the directory server. For Active Directory, the user does not need elevated privileges. You can specify any user in the domain. The username must be fully qualified; for example, Administrator@example.com (not simply Administrator).

Also, the system generates ldap-login-dn and ldap-login-password from the username and password information. For example, Administrator@example.com is translated as cn=admin, cn=users, dc=example, dc=com. Note that cn=users is always part of this translation, so you must configure the user you specify here under the common name “users” folder.

The directory server is accessible through a data interface.

If the directory server is on a network that is either directly connected to a data interface (such as a GigabitEthernet interface), or routeable from a directly-connected network, you must ensure that there is a route between the virtual management interface and the directory server.

- Using **data-interfaces** as the management gateway should make routing successful.
- If you have an explicit gateway on the management interface, that gateway router needs to have a route to the directory server.
- You do not need to configure an IP address on the **diagnostic** interface, which is the physical interface used by the virtual management interface. However, if you do configure an address, do not also configure a static route (such as a default route) that would redirect traffic to the directory server to the diagnostic interface.
- If there is a router between the directly-connected network and the network that hosts the directory server, configure a static route for the directory server (**Device > Routing**).
- Verify that the data interface has the correct IP address and subnet mask.

The directory server is accessible through the Management physical interface.

If the directory server is on the network that is either directly connected to the Management physical interface (such as Management0/0) or routeable from that network, you must do the following:

- Configure an IPv4 address for the Management interface (with the logical name **diagnostic**) on **Device > Interfaces**. The IP address must be on the same subnet as the virtual management address (**Device > System Settings > Management Interface**).
- If there is a router between the directory server and the Management interface, configure a route for the directory server on **Device > Routing** for the **diagnostic** interface.
- Verify that the diagnostic and management interfaces have the correct IP address and subnet mask.

The directory server is on an external network.

If the directory server is on a network on the other side of the outside (uplink) interface, you might need to configure a site-to-site VPN connection. For the detailed procedure, see [How to Use a Directory Server on an Outside Network with Remote Access VPN](#).

Limitations on Number of Users

Firepower Device Manager can download information on up to 2000 users from the directory server.

If your directory server includes more than 2000 user accounts, you will not see all possible names when selecting users in an access rule or when viewing user-based dashboard information. You can write rules on only those names that were downloaded.

The limit also applies to the names associated with groups. If a group has more than 2000 members, only the 2000 names that were downloaded can be matched against the group membership.

If you have more than 2000 users, consider using Firepower Management Center (the remote manager) instead of Firepower Device Manager. Firepower Management Center supports significantly more users.

Supported Directory Servers

You can use Microsoft Active Directory (AD) on Windows Server 2008 and 2012.

Note the following about your server configuration:

- If you want to perform user control on user groups or on users within groups, you must configure user groups on the directory server. The system cannot perform user group control if the server organizes the users in basic object hierarchy.
- The directory server must use the field names listed in the following table in order for the system to retrieve user metadata from the servers for that field.

Metadata	Active Directory Field
LDAP user name	samaccountname
first name	givenname
last name	sn
email address	mail userprincipalname (if mail has no value)
department	department distinguishedname (if department has no value)
telephone number	telephonenumber

Determining the Directory Base DN

When you configure directory properties, you need to specify the common base distinguished name (DN) for users and groups. The base is defined in your directory server, and differs from network to network. You must enter the correct bases for identity policies to work. If the base is wrong, the system cannot determine user or group names, and thus identity-based policies will be inoperable.



Tip To get the correct bases, consult the administrator who is responsible for the directory servers.

For active directory, you can determine the correct bases by logging into the Active Directory server as domain administrator, and using the **dsquery** command at a command prompt as follows to determine the bases:

User search base

Enter the **dsquery user** command with a known username (partial or complete) to determine the base distinguished name. For example, the following command uses the partial name “John*” to return information for all users that start with “John.”

```
C:\Users\Administrator>dsquery user -name "John*"
"CN=John Doe,CN=Users,DC=csc-lab,DC=example,DC=com"
```

The base DN would be “DC=csc-lab,DC=example,DC=com.”

Group search base

Enter the **dsquery group** command with a known group name to determine the base distinguished name. For example, the following command uses the group name Employees to return the distinguished name:

```
C:\>dsquery group -name "Employees"
"CN=Employees,CN=Users,DC=csc-lab,DC=example,DC=com"
```

The group base DN would be “DC=csc-lab,DC=example,DC=com.”

You can also use the ADSI Edit program to browse the Active Directory structure (**Start > Run > adsiedit.msc**). In ADSI Edit, right click any object, such as an organizational unit (OU), group, or user, and choose **Properties** to view the distinguished name. You can then copy the string of DC values as the base.

To verify that you have the correct base:

1. Click the Test Connection button in the directory properties to verify connectivity. Resolve any problems, and save the directory properties.
2. Commit changes to the device.
3. Create an access rule, select the **Users** tab, and try to add known user and group names from the directory. You should see auto-complete suggestions as you type for matching users and groups in the realm that contains the directory. If these suggestions appear in a drop-down list, then the system was able to query the directory successfully. If you see no suggestions, and you are certain the string you typed should appear in a user or group name, you need to correct the corresponding search base.

Configuring Certificates

Digital certificates provide digital identification for authentication. A digital certificate includes information that identifies a device or user, such as the name, serial number, company, department, or IP address. A digital certificate also includes a copy of the public key for the user or device. Certificates are used for SSL (Secure Socket Layer), TLS (Transport Layer Security), and DTLS (Datagram TLS) connections, such as HTTPS and LDAPS.



Certificate Authorities (CAs) are trusted authorities that “sign” certificates to verify their authenticity, thereby guaranteeing the identity of the device or user. CAs issue digital certificates in the context of a PKI, which uses public-key or private-key encryption to ensure security. A CA can be a trusted third party, such as VeriSign, or a private (in-house) CA that you establish within your organization. CAs are responsible for

managing certificate requests and issuing digital certificates. For more information, see [Public Key Cryptography, on page 17](#).

FTD supports X509 certificates in PEM or DER format. Use OpenSSL to generate certificates if needed, or obtain them from a trusted Certificate Authority.

The following procedure explains how you can create and edit objects directly through the Objects page. You can also create certificate objects while editing a certificate property by clicking the **Create New Certificate** link shown in the object list.

Procedure

- Step 1** Select **Objects**, then select **Certificates** from the table of contents.
- The system comes with a pre-defined certificate, DefaultInternalCertificate. You can edit and replace the certificate or use it as is.
- Step 2** Do one of the following:
- To create a new certificate object, use the command for the type of certificate. For information on which type is used for each feature, see [Certificate Types Used by Feature, on page 18](#).
 - Click + > **Add Internal Certificate**.
 - Click + > **Add Trusted CA Certificate**.
 - To edit an object, click the edit icon () for the object.
- To delete an unreferenced object, click the trash can icon () for the object.
- Step 3** Configure the certificate properties.
- [Internal Certificate Properties, on page 18](#)
 - [Trusted CA Certificate Properties, on page 20](#)
- Step 4** Click **OK**.
-

Public Key Cryptography

In public key cryptography, such as the RSA encryption system, each user has a key pair containing both a public and a private key. The keys act as complements, and anything encrypted with one of the keys can be decrypted with the other.

In simple terms, a signature is formed when data is encrypted with a private key. The signature is attached to the data and sent to the receiver. The receiver applies the public key of the sender to the data. If the signature sent with the data matches the result of applying the public key to the data, the validity of the message is established.

This process relies on the receiver having a copy of the public key of the sender and a high degree of certainty that this key belongs to the sender, not to someone pretending to be the sender.

Obtaining the public key of a sender is normally handled externally or through an operation performed at installation. For example, most web browsers are configured with the root certificates of several CAs by default.

You can learn more about digital certificates and public key cryptography through [openssl.org](https://www.openssl.org), Wikipedia, or other sources. Having a firm understanding of SSL/TLS cryptography will help you establish secure connections to your device.

Certificate Types Used by Feature

You need to create the right type of certificate for each feature. The following features require certificates.

Identity Policies (Captive Portal)—Internal Certificate

(Optional.) Captive portal is used in identity policies. Users must accept this certificate when authenticating to the device for purposes of identifying themselves and getting their IP address associated with their usernames. If you do not supply a certificate, the device uses an automatically generated certificate.

Identity Realms (Identity Policies and Remote Access VPN)—Trusted CA Certificate

(Optional.) If you use an encrypted connection for your directory server, the certificate must be accepted to perform authentication with the directory server. Users must authenticate when prompted by identity and remote access VPN policies. A certificate is not needed if you do not use encryption for the directory server.

Remote Access VPN—Internal Certificate

(Required.) The internal certificate is for the outside interface, which establishes the device identity for AnyConnect clients when they make a connection to the device. Clients must accept this certificate.

Internal Certificate Properties

Internal identity certificates are certificates for specific systems or hosts. You can generate these yourself using the OpenSSL toolkit or get them from a Certificate Authority. For an example of how to generate a certificate, see [Example: Generating an Internal Certificate using OpenSSL, on page 19](#).

The system comes with a pre-defined internal certificate, `NGFW_DefaultInternalCertificate`. You can use this default certificate or create your own.

FTD uses internal certificates for the following features:

- Captive portal, which is used in identity policies. Users must accept this certificate when authenticating to the device for purposes of identifying themselves and getting their IP address associated with their usernames.
- Remote access VPN. The internal certificate is for the outside interface, which establishes the device identity for AnyConnect clients when they make a connection to the FTD device. Clients must accept this certificate.

Configure the following properties:

Name

Enter a name for the certificate. The name is used in the configuration as an object name only, it does not become part of the certificate itself.

Server Certificate

Click **Upload Certificate** (or **Replace Certificate** when editing) and select the server identity certificate file (for example, *.cert). Alternatively, paste in the server identity certificate. The certificate must be an X509 certificate in PEM or DER format.

The certificate you paste must include the BEGIN CERTIFICATE and END CERTIFICATE lines. For example:

```
-----BEGIN CERTIFICATE-----
MIICMTCCAzoCCQDdUV3NGK/cUjANBgkqhkiG9w0BAQsFADBdMQswCQYDVQQGEwJV
UzETMBEGA1UECAwKU29tZS1TdGF0ZTEhMB8GA1UECgwYSW50ZXJuZXQvV2lkZ210
(...5 lines removed...)
shGJDRerYJQqilhHZrYTWZAYTrD7NQP HutK+ZiJng67cPgnNDuXEn55UwMOQoHBp
HMUwmhiGZLzJM8BpX2Js2yQ3ms30pr8rO+gPCPMCAWEAATANBgkqhkiG9w0BAQsF
AAOBgQCB02CebA6YjJCGr2CJZrQSeUwSveRBpmOuoqm98o2Z+5gJM5CkqgfXwCUn
RV7LrfQGFYd76V/5uor4Wx2ZCjy6+zuQEm4ZxWNSZpA9UBixFXJCS9MBO4qkG5D
v1k3WYJfcgyJ10h4E4b0W2xiixBU+xoOTLRATnbKY36EWAG5cw==
-----END CERTIFICATE-----
```

Certificate Key

Click **Upload Key** (or **Replace Key** when editing) and select the server identity certificate file (for example, *.key). Alternatively, paste in the key for the server certificate. The key cannot be encrypted.

For example:

```
-----BEGIN RSA PRIVATE KEY-----
MIICXQIBAAKBgQC1SulBknrMjzw/5FZ9YgdMLDUGJlbyGkjN7mVrkjyLQx2TYsem
r8iTikB6iyTKbuS4iPeyEYkNF5FglCqKWEdmthNZkBhOsPs1A8e60r5mImeDrtw+
Cc005cSfnlTAw5CgcGkcxTCaGIzMXMkzwG1fYmzbJDeazfSmvys76A8I8wIDAQAB
AoGAUVDgEX8vXE0m9cOubPZ54pZo64KW/OJzUKP0TwxdlqGw/h39XFpkEXiIgmDL
(...5 lines removed...)
DSWzvekRDH83dmP66+MIbWePhbhty+D10xbiuVuHV0/ZhxOhCG8tig3R8QJBAJmj
fId05+1dNI4tGbWv6hHh/H/dTP2ST1Z3jERMzd29fjIRuJ9jpFC21IDjvs8YGeAe
0YHkFSOULJn8/jOCf6kCQQDIJiHfGF/31Dk/8/5MGrG+3zau6oKXiuv6db8Rh+71
MUOx09tvbBUy9REJq1YJWTKpeKD+E0QL+FX0bqvz4tHA
-----END RSA PRIVATE KEY-----
```

Example: Generating an Internal Certificate using OpenSSL

The following example uses OpenSSL commands to generate an internal server certificate. You can obtain OpenSSL from opnssl.org. Consult OpenSSL documentation for specific information. The commands used in this example might change, and you might have other options available that you might want to use.

This procedure is meant to give you an idea of how to obtain a certificate to upload to FTD.



Note The OpenSSL commands shown here are examples only. Adjust the parameters to fit your security requirements.

Procedure

Step 1 Generate a key.

```
openssl genrsa -out server.key 4096
```

Step 2 Generate a certificate signing request (CSR).

```
openssl req -new -key server.key -out server.csr
```

Step 3 Generate a self-signed certificate with the key and CSR.

```
openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt
```

Because Firepower Device Manager does not support encrypted keys, try to skip the challenge password by just pressing return when generating a self signed certificate.

Step 4 Upload the files into the appropriate fields when creating an internal certificate object in Firepower Device Manager.

You can also copy/paste the file contents. The sample commands create the following files:

- server.crt—Upload or paste the contents into the Server Certificate field.
- server.key—Upload or paste the contents into the Certificate Key field. If you provided a password when generating the key, you can decrypt it using the following command. The output is sent to stdout, where you can copy it.

```
openssl rsa -in server.key -check
```

Trusted CA Certificate Properties

A trusted Certificate Authority (CA) certificate is used to sign other certificates. It is self-signed and called a root certificate. A certificate that is issued by another CA certificate is called a subordinate certificate.

FTD uses Trusted CA certificates for directory realms if you use an encrypted connection for the directory server.

Obtain a trusted CA certificate from an external Certificate Authority, or create one using your own internal CA, for example, with OpenSSL tools.

Configure the following properties:

Name

Enter a name for the certificate. The name is used in the configuration as an object name only, it does not become part of the certificate itself.

Server Certificate

Click **Upload Certificate** (or **Replace Certificate** when editing) and select the trusted CA certificate file (for example *.pem). Alternatively, paste in the trusted CA certificate. The name of the server in the certificate must match the server Hostname / IP Address. For example, if you use 10.10.10.250 as the IP address but ad.example.com in the certificate, the connection fails.

The certificate you paste must include the BEGIN CERTIFICATE and END CERTIFICATE lines. For example:

```
-----BEGIN CERTIFICATE-----
MIIFgTCCA2mgAwIBAgIJANvdcLnabFGYMA0GCSqGSIb3DQEBCwUAMFcx CzA JBgNV
BAYTA1VTMQswCQYDVQQIDAJUWDEPMA0GA1UEBwwGYXVzdGluMRQwEgYDVQQKDAsx
OTIuMTY4LjEuMTEUMBIGA1UEAwLMTKyLjE2OC4xLjEwHhcNMTYxMDI3MjIzNDE3
WhcNMTCxMDI3MjIzNDE3WjBXMQswCQYDVQQGEwJVUzELMAkGA1UECAwCVFgx DzAN
BgNVBACMBmF1c3RpbjEUMBIGA1UECgwLTKyLjE2OC4xLjEwExFDASBgNVBAMMCzE5
Mi4xNjguMS4xMIIICIjANBgkqhkiG9w0BAQEFAAOCAg8AMIICGKCAgEA5NceYwtP
ES6Ve+S9z7WLKGX5JlF58AvH82GpkOQdrinxn3FZeWLQapTpJZt/vgtAI2FZIK31h
(...20 lines removed...)
hbr6HOgKlOwXbRvOdkstzTEzVUqbgxt5Lwupg3b2ebQhWJz4BZvMsZX9etveEXDh
PY184V3yeSeYjbSCF5rP71fObG9Iu6+u4EfHp/NQv9s9dN5PMffXKieqpuN20Ojv
2b1sfOydf4GMUKLBUMkhQnip6+3W
-----END CERTIFICATE-----
```

