O

- object-group, on page 2
- object-group-search, on page 7
- object network, on page 10
- object network-service, on page 12
- object service, on page 14
- ocsp disable-nonce, on page 16
- ocsp interface, on page 18
- ocsp url, on page 20
- onscreen-keyboard(Deprecated), on page 22
- ospf authentication, on page 23
- ospf authentication-key, on page 25
- ospf cost, on page 27
- ospf database-filter, on page 29
- ospf dead-interval, on page 30
- ospf hello-interval, on page 32
- ospf message-digest-key, on page 33
- ospf mtu-ignore, on page 35
- ospf network point-to-point non-broadcast, on page 36
- ospf priority, on page 38
- ospf retransmit-interval, on page 39
- ospf transmit-delay, on page 40
- otp expiration, on page 41
- output console, on page 43
- output file, on page 44
- output none, on page 46
- outstanding (Deprecated), on page 47
- override-account-disable (Deprecated), on page 49
- override-svc-download, on page 51

object-group

To define object groups that you can use to optimize your configuration, use the **object-group** command in global configuration mode. Use the **no** form of this command to remove object groups from the configuration.

object-group { **protocol | network | icmp-type | security | user | network-service** } *grp_name*
object-group service *grp_name* [**tcp | udp | tcp-udp**]

Syntax Description

grp_name	Identifies the object group (one to 64 characters) and can be any combination of letters, digits, and the "_", "-", "." characters.
icmp-type	(Not recommended, use **service** instead.) Defines a group of ICMP types such as echo and echo-reply. After entering the **object-group icmp-type** command, use the **icmp-object** and the **group-object** commands to add ICMP objects.
network	Defines a group of hosts or subnet IP addresses. After entering the **object-group network** command, use the **network-object** and the **group-object** commands to add network objects. You can create a group with a mix of IPv4 and IPv6 addresses. **Note** You cannot use a mixed object group for NAT.
network-service	Defines a group of subnets or domain names with optional service specifications. After entering this command, use the **network-service-member** command to add network-service objects, or the **domain** and **subnet** commands to add members directly.
protocol	(Not recommended, use **service** instead.) Defines a group of protocols such as TCP and UDP. After entering the **object-group protocol** command, use the **protocol-object** and the **group-object** commands to add protocol objects.
security	Defines a security group object for use with Cisco TrustSec. After entering the **object-group protocol** command, use the **security-group** and the **group-object** commands to add security group objects.
service **[tcp | udp | tcp-udp**	Defines a service based on protocol, ICMP types, and TCP/UDP/SCTP ports. To define a mixed group of services, or SCTP ports, do not specify the protocol type for the object-group. After entering the **object-group service** command, add service objects to the service group with the **service-object** and the **group-object** commands. This is the preferred method, even if the object is meant to include only lists of TCP or UDP (or both) ports. Using the **tcp**, **udp**, and **tcp-udp** keywords directly on the object-group service command is not recommended. Instead, leave these keywords off the command and configure TCP and UDP ports on the **service-object** command. If you do include one of these keywords, use the **port-object** and the **group-object** commands to add port groups.
user	Defines users and user groups that you can use to control access with the identity firewall. After entering the **object-group protocol** command, use the **user**, **user-group**, and the **group-object** commands to add user and user group objects.

Command Default

No default behavior or values.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Global Configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
8.4(2)	Support for the **user** keyword was added to support identity firewall.
9.0(1)	You can now create network object groups that can support a mix of both IPv4 and IPv6 addresses. Support for the **security** keyword was added to support Cisco TrustSec.
9.14	The **icmp-type** keyword was deprecated. Use the **service** keyword, and specify **service icmp** in the object instead.
9.17(1)	The **network-service** keyword was added.

Usage Guidelines

Objects such as hosts or services can be grouped, and then you can use the object group in features such as ACLs (**access-list**) and NAT (**nat**). This example shows the use of a network object group in an ACL:

```
ciscoasa(config)# access-list access_list_name extended permit tcp any object-group NWgroup1
```

You can group commands hierarchically; an object group can be a member of another object group.

Examples

The following example shows how to use the **object-group network** command to create a network object group:

```
ciscoasa(config)# object-group network sjc_eng_ftp_servers
ciscoasa(config-network-object-group)# network-object host sjc.eng.ftp.servcers
ciscoasa(config-network-object-group)# network-object host 172.23.56.194
ciscoasa(config-network-object-group)# network-object 192.1.1.0 255.255.255.224
ciscoasa(config-network-object-group)# exit
```

The following example shows how to use the **object-group network** command to create a network object group that includes an existing object-group:

```
ciscoasa(config)# object-group network sjc_ftp_servers
ciscoasa(config-network-object-group)# network-object host sjc.ftp.servers

ciscoasa(config-network-object-group)# network-object host 172.23.56.195
ciscoasa(config-network-object-group)# network-object 193.1.1.0 255.255.255.224

ciscoasa(config-network-object-group)# group-object sjc_eng_ftp_servers
ciscoasa(config-network-object-group)# exit
```

The following example shows how to use the **group-object** mode to create a new object group that consists of previously defined objects, and then how to use these objects in an ACL:

```
ciscoasa(config)# object-group network host_grp_1
ciscoasa(config-network-object-group)# network-object host 192.168.1.1
ciscoasa(config-network-object-group)# network-object host 192.168.1.2
ciscoasa(config-network-object-group)# exit
ciscoasa(config)# object-group network host_grp_2
ciscoasa(config-network-object-group)# network-object host 172.23.56.1
ciscoasa(config-network-object-group)# network-object host 172.23.56.2
ciscoasa(config-network-object-group)# exit
ciscoasa(config)# object-group network all_hosts
ciscoasa(config-network-object-group)# group-object host_grp_1
ciscoasa(config-network-object-group)# group-object host_grp_2
ciscoasa(config-network-object-group)# exit
ciscoasa(config)# access-list grp_1 permit tcp object-group host_grp_1 any eq ftp
ciscoasa(config)#access-list grp_2 permit tcp object-group host_grp_2 any eq smtp
ciscoasa(config)#access-list all permit tcp object-group all_hosts any eq www
```

Without the **group-object** command, you need to define the *all_hosts* group to include all the IP addresses that have already been defined in *host_grp_1* and *host_grp_2*. With the **group-object** command, the duplicated definitions of the hosts are eliminated.

The following example shows how to add both TCP and UDP services to a service object group:

```
ciscoasa(config)# object-group service CommonApps
ciscoasa(config-service-object-group)# service-object tcp destination eq ftp
ciscoasa(config-service-object-group)# service-object tcp-udp destination eq www
ciscoasa(config-service-object-group)# service-object tcp destination eq h323
ciscoasa(config-service-object-group)# service-object tcp destination eq https
ciscoasa(config-service-object-group)# service-object udp destination eq ntp
```

The following example shows how to add multiple service objects to a service object group:

```
ciscoasa(config)# object-group service SSH
ciscoasa(config-service-object)# service tcp destination eq ssh
ciscoasa(config)# object-group service EIGRP
ciscoasa(config-service-object)# service eigrp
ciscoasa(config)# object-group service HTTPS
ciscoasa(config-service-object)# service tcp source range 0 1024 destination eq https
ciscoasa(config)# object-group service Group1
ciscoasa(config-service-object-group)# group-object SSH
ciscoasa(config-service-object-group)# group-object EIGRP
ciscoasa(config-service-object-group)# group-object HTTPS
```

The following example shows how to add a mix of protocol, port, and ICMP specifications in a service object group:

```
ciscoasa(config)# object-group service mixed
ciscoasa(config-service-object-group)# service-object tcp destination eq ftp
ciscoasa(config-service-object-group)# service-object tcp-udp destination eq www
ciscoasa(config-service-object-group)# service-object ipsec
ciscoasa(config-service-object-group)# service-object tcp destination eq domain
ciscoasa(config-service-object-group)# service-object icmp echo
```

The following example shows how to use the **service-object** subcommand, which is useful for grouping TCP and UDP services:

```
ciscoasa(config)# object-group network remote
ciscoasa(config-network-object-group)# network-object host kqk.suu.dri.ixx
ciscoasa(config-network-object-group)# network-object host kqk.suu.pyl.gnl
ciscoasa(config)# object-group network locals
ciscoasa(config-network-object-group)# network-object host 209.165.200.225
ciscoasa(config-network-object-group)# network-object host 209.165.200.230
ciscoasa(config-network-object-group)# network-object host 209.165.200.235
ciscoasa(config-network-object-group)# network-object host 209.165.200.240
ciscoasa(config)# object-group service usr_svc
ciscoasa(config-service-object-group)# service-object tcp destination eq www
ciscoasa(config-service-object-group)# service-object tcp destination eq https
ciscoasa(config-service-object-group)# service-object tcp destination eq pop3
ciscoasa(config-service-object-group)# service-object udp destination eq ntp
ciscoasa(config-service-object-group)# service-object udp destination eq domain
ciscoasa(config)# access-list acl extended permit object-group usr_svc object-group locals
 object-group remote
```

The following example shows how to use the **object-group user** command to create user group objects:

```
ciscoasa(config)# object-group user sampleuser1-group
ciscoasa(config-object-group user)# description group members of sampleuser1-group
ciscoasa(config-object-group user)# user-group EXAMPLE\\group.sampleusers-all
ciscoasa(config-object-group user)# user EXAMPLE\user2
ciscoasa(config-object-group user)# exit
ciscoasa(config)# object-group user sampleuser2-group
ciscoasa(config-object-group user)# description group members of sampleuser2-group
ciscoasa(config-object-group user)# group-object sampleuser1-group
ciscoasa(config-object-group user)# user-group EXAMPLE\\group.sampleusers-marketing
ciscoasa(config-object-group user)# user EXAMPLE\user3
```

(Not recommended, use service objects instead.) The following example shows how to use the **object-group icmp-type** mode to create a ICMP object group:

```
ciscoasa(config)# object-group icmp-type icmp-allowed
ciscoasa(config-icmp-object-group)# icmp-object echo
ciscoasa(config-icmp-object-group)# icmp-object time-exceeded
ciscoasa(config-icmp-object-group)# exit
```

(Not recommended, use service objects instead.) The following example shows how to use the **object-group protocol** mode to create a protocol object group:

```
ciscoasa(config)# object-group protocol proto_grp_1
ciscoasa(config-protocol-object-group)# protocol-object udp
ciscoasa(config-protocol-object-group)# protocol-object ipsec
ciscoasa(config-protocol-object-group)# exit
ciscoasa(config)# object-group protocol proto_grp_2
ciscoasa(config-protocol-object-group)# protocol-object tcp
ciscoasa(config-protocol-object-group)# group-object proto_grp_1
ciscoasa(config-protocol-object-group)# exit
```

(Not recommended, leave off the **tcp** keyword and define the port with the **service-object** command instead.) The following example shows how to use the **object-group service** mode to create a TCP port object group:

```
ciscoasa(config)# object-group service eng_service tcp
ciscoasa(config-service-object-group)# group-object eng_www_service
ciscoasa(config-service-object-group)# port-object eq ftp
ciscoasa(config-service-object-group)# port-object range 2000 2005
ciscoasa(config-service-object-group)# exit
```

The following examples show how to use object groups to simplify the access list configuration. This grouping enables the access list to be configured in 1 line instead of 24 lines, which would be needed if no grouping is used.

```
ciscoasa(config)# object-group network remote
ciscoasa(config-network-object-group)# network-object host 10.1.1.15
ciscoasa(config-network-object-group)# network-object host 10.1.1.16
ciscoasa(config)# object-group network locals
ciscoasa(config-network-object-group)# network-object host
209.165.200.225
ciscoasa(config-network-object-group)# network-object host
209.165.200.230
ciscoasa(config-network-object-group)# network-object host
209.165.200.235
ciscoasa(config-network-object-group)# network-object host
209.165.200.240
ciscoasa(config)# object-group service eng_svc tcp
ciscoasa(config-service-object-group)# port-object eq www
ciscoasa(config-service-object-group)# port-object eq smtp
ciscoasa(config-service-object-group)# port-object range 25000 25100
ciscoasa(config)# access-list acl extended permit tcp object-group remote object-group
locals object-group eng_svc
```

Note The **show running-config access-list** command displays the access list as configured with the object group names. The **show access-list** command displays this information plus the access list entries that use groups expanded out into individual entries without their object groupings.

The following example configures a set of SaaS applications using previously-defined network-service objects.

```
object-group network-service SaaS_Applications
    description This group includes relevant 'Software as a Service' applications
    network-service-member "outlook 365"
    network-service-member webex
    network-service-member box
```

Related Commands

Command	Description
clear configure object-group	Removes all the object group commands from the configuration.
group-object	Adds network object groups.
network-object	Adds a network object to a network object group.
port-object	Adds a port object to a service object group.
security-group	Adds a security group to a security group object group.
show running-config object-group	Displays the current object groups.
user	Adds a username to a user group object.
user-group	Adds a user group name to a user group object.

object-group-search

To enable ACL optimization, use the **object-group-search** command in global configuration mode. Use the **no** form of this command to disable ACL optimization.

object-group-search { **access-control | threshold** }
no object-group-search { **access-control | threshold** }

Syntax Description

access-control	Enables object group search for access control rules.
threshold	Enables a maximum threshold for object group search processing. See the usage notes for detailed information.

Command Default

(Pre-9.18) Object group search is disabled by default. The threshold is also disabled by default.

Starting with 9.18, object group search is enabled by default for access control for new deployments.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Global Configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
8.3(1)	This command was added.
9.12(1)	The **threshold** keyword was added. The keyword was also added in interim releases of 9.8, 9.9, and 9.10.
9.18(1)	The default for access control was changed to enabled for new deployments. You must enable it on upgrades if it was not previously enabled.

Usage Guidelines

The **object-group-search** command optimizes all ACLs in the inbound direction.

You can reduce the memory required to search access rules by enabling object group search, but this is at the expense rule of lookup performance and increased CPU utilization. When enabled, object group search does not expand ACLs that use network or service objects in the ASP table, but instead searches access rules for matches based on those group definitions. You will see this in the **show access-list** output.

Object group search is subject to a threshold. For each connection, both the source and destination IP addresses are matched against network objects. If the number of objects matched by the source address times the number matched by the destination address exceeds 10,000, the connection is dropped. This check is to prevent performance degradation. Configure your rules to prevent an excessive number of matches. Avoid the creation of duplicate objects to prevent reaching the threshold.

Starting in release 9.12(1), and in interim releases back to 9.8(x), this threshold is disabled by default. Use the **show running-config all object-group-search** command to determine whether the threshold option is configured, and if so, the current setting.

When you enable object group search, and you have a significant number of features enabled, a large number of active connections, and large ACLs for your access groups, there will be a connection drop during the operation and a performance drop while establishing new connections. These drops can happen even if you enable transactional commit (**asp rule-engine transactional-commit access-group**).

Note Object group search works with network and service objects only. It does not work with security group or user objects. Do not enable this feature if the ACLs include security groups. The result can be inactive ACLs or other unexpected behavior.

Examples

The following example shows how to use the **object-group-search** command to enable ACL optimization:

```
ciscoasa(config)# object-group-search access-control
```

The following is sample output from the **show access-list** command when **object-group-search** is not enabled:

```
ciscoasa# show access-list KH-BLK-Tunnel
access-list KH-BLK-Tunnel; 9 elements
access-list KH-BLK-Tunnel line 1 extended permit ip object-group KH-LAN object-group BLK-LAN
 0x724c956b
   access-list KH-BLK-Tunnel line 1 extended permit ip 192.168.97.0 255.255.255.0 192.168.4.0
 255.255.255.0 (hitcnt=10) 0x30fe29a6
   access-list KH-BLK-Tunnel line 1 extended permit ip 13.13.13.0 255.255.255.0 192.168.4.0
 255.255.255.0 (hitcnt=4) 0xc6ef2338
   access-list KH-BLK-Tunnel line 1 extended permit ip 192.168.97.0 255.255.255.0 14.14.14.0
 255.255.255.0 (hitcnt=2) 0xce8596ec
   access-list KH-BLK-Tunnel line 1 extended permit ip 13.13.13.0 255.255.255.0 14.14.14.0
 255.255.255.0 (hitcnt=0) 0x9a2f1c4d
access-list KH-BLK-Tunnel line 2 extended permit ospf interface pppoe1 host 87.139.87.200
(hitcnt=0) 0xb62d5832
access-list KH-BLK-Tunnel line 3 extended permit ip interface pppoe1 any (hitcnt=0) 0xa2c9ed34
access-list KH-BLK-Tunnel line 4 extended permit ip host 1.1.1.1 any (hitcnt=0) 0xd06f7e6b
access-list KH-BLK-Tunnel line 5 extended deny ip 1.1.0.0 255.255.0.0 any (hitcnt=0)
0x9d979934
access-list KH-BLK-Tunnel line 6 extended permit ip 1.1.1.0 255.255.255.0 any (hitcnt=0)
0xa52a0761
```

The following is sample output from the **show access-list** command when **object-group-search** is enabled:

```
ciscoasa# show access-list KH-BLK-Tunnel
access-list KH-BLK-Tunnel; 6 elements
access-list KH-BLK-Tunnel line 1 extended permit ip object-group KH-LAN(1) object-group
BLK-LAN(2)(hitcount=16) 0x724c956b
access-list KH-BLK-Tunnel line 2 extended permit ospf interface pppoe1 host 87.139.87.200
(hitcnt=0) 0xb62d5832
access-list KH-BLK-Tunnel line 3 extended permit ip interface pppoe1 any (hitcnt=0) 0xa2c9ed34
access-list KH-BLK-Tunnel line 4 extended permit ip host 1.1.1.1 any (hitcnt=0) 0xd06f7e6b
access-list KH-BLK-Tunnel line 5 extended deny ip 1.1.0.0 255.255.0.0 any (hitcnt=0)
0x9d979934
```

```
access-list KH-BLK-Tunnel line 6 extended permit ip 1.1.1.0 255.255.255.0 any (hitcnt=0)
0xa52a0761
```

Related Commands

Command	Description
clear config object-group search	Clears the object-group-search configuration.
show object-group	Shows the hit count if the object group is of the network object-group type.
show running-config object-group	Displays the current object groups.
show running-config object-group-search	Show the object-group-search configuration in the running configuration.

object network

To configure a named network object, use the **object network** command in global configuration mode. Use the **no** form of this command to remove the object from the configuration.

object network *name* [**rename** *new_obj_name*]
no object network *name*

Syntax Description

name	Specifies the name of the network object. The name can be from 1 to 64 characters in length, consisting of letters, numbers, and the following special characters: underscore, hyphen, comma, forward slash, and period. Objects and object groups share the same name space.
rename *new_obj_name*	(Optional) Renames the object to the new object name.

Command Default

No default behavior or values.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Global Configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
8.3(1)	This command was added.
8.4(2)	Support for fully-qualified domain names (FQDN) was added. See the **fqdn** command.

Usage Guidelines

The network object can contain a host, a network, a range IP addresses (IPv4 or IPv6), or an FQDN. After you enter the command, use the **host**, **fqdn**, **subnet**, or **range** command to add one address to the object.

You can also enable NAT rules on this network object using the **nat** command. You can only define a single NAT rule for a given object; if you want to configure multiple NAT rules, you need to create multiple objects that specify the same IP address, for example, **object network obj-10.10.10.1-01**, **object network obj-10.10.10.1-02**, and so on.

If you configure an existing network object with a different IP address, the new configuration will replace the existing configuration.

Examples

The following example shows how to create a network object:

```
ciscoasa (config)# object network OBJECT1
ciscoasa (config-network-object)# host 10.1.1.1
```

Related Commands

Command	Description
clear configure object	Clears all objects created.
description	Adds a description to the network object.
fqdn	Specifies a fully-qualified domain name network object.
host	Specifies a host network object.
nat	Enables NAT for the network object.
object-group network	Creates a network object group.
range	Specifies a range of addresses for the network object.
show running-config object network	Shows the network object configuration.
subnet	Specifies a subnet network object.

object network-service

To configure a named network-service object, use the **object network-service** command in global configuration mode. Use the **no** form of this command to remove the object from the configuration.

object network-service *name* [**dynamic**]
no object network-service *name*

Syntax Description

dynamic	(Optional.) The **dynamic** keyword means that the object will not be saved to the running configuration, it will be shown in the **show object** output only. The **dynamic** keyword is primarily for use by external device managers.
name	The name can be up to 128 characters, and can include spaces. If you include spaces, you must enclose the name in double quotation marks.

Command Default

No default behavior or values.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Global Configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
9.17(2)	This command was added.

Usage Guidelines

A network-service object defines a single application. It defines the application location either by subnet specification or more commonly, DNS domain name. Optionally, you can include protocol and port to narrow the scope of the application.

You can use these objects in network-service group objects only; you cannot directly use a network-service object in an access control list entry (ACE).

Add one or more application locations and optional services to the object using one of the following commands. Use the **no** form of the command to remove the location. You can enter these commands multiple times.

- **domain** *domain_name* [*service*]—The DNS name, up to 253 characters. This can be fully-qualified (such as www.example.com) or partial (such as example.com), in which case the object matches all subdomains, that is, servers with the partial name (such as www.example.com, www1.example.com, long.server.name.example.com, and so forth). Connections will be matched against the longest name if an exact match is available. The domain name can resolve to multiple IP addresses.

- **subnet** {*IPv4_address IPv4_mask* | *IPv6_address/IPv6_prefix*} [*service*]—The address of a network. For IPv4 subnets, include the mask after a space, for example, 10.0.0.0 255.0.0.0. For IPv6, include the address and prefix as a single unit (no spaces), such as 2001:DB8:0:CD30::/60.

The service specification for these commands is the same. Specify the service only if you want to limit the scope of the connections matched. By default, any connection to the resolved IP addresses matches the object.

protocol [*operator port*]

where:

- *protocol* is the protocol used in the connection, such as tcp, udp, ip, and so forth. Use ? to see the list of protocols.
- (TCP/UDP only.) *operator* is one of the following:
  - **eq** equals the port number specified.
  - **lt** means any port less than the specified port number.
  - **gt** means any port greater than the specified port number.
  - **range** means any port between the two ports specified.
- (TCP/UDP only.) *port* is the port number, 1-65535 or a mnemonic, such as www. Use ? to see the mnemonics. For ranges, you must specify two ports, with the first port being a lower number than the second port.

Example

Following is an example of a network-service object.

```
object network-service outlook365
    description This defines Microsoft office365 'outlook' application.
  domain outlook.office.com tcp eq 443
object network-service webex
  domain webex.com tcp eq 443
object network-service partner
  subnet 10.34.56.0 255.255.255.0 ip
```

Related Commands

Command	Description
app-id	Specifies an application ID for the object.
clear object	Clears hit counts for network-service objects.
description	Adds a description to the object.
domain	Specifies domain name for the object.
object-group network-service	Creates a network-service object group.
show object	Shows the network-service objects.
subnet	Specifies a subnet for the object.

object service

To configure a service object that is automatically reflected in all configurations in which the object is used, use the **object service** command in global configuration mode. Use the **no** form of this command to remove the object.

object service *name* [**rename** *new_obj_name*]
no object service *object name* [**rename** *new_obj_name*]

Syntax Description

name	Specifies the name of the service object. The name can be from 1 to 64 characters in length, consisting of letters, numbers, and the following special characters: underscore, hyphen, comma, and period. The object name must start with a letter.
rename *new_obj_name*	(Optional) Renames the object to the new object name.

Command Default

No default behavior or values.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Global Configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
8.3(1)	This command was added.

Usage Guidelines

The service object can contain a protocol, ICMP, ICMPv6, or TCP /UDP/SCTP port or port ranges. After you enter the command, use the **service** command to add one service specification to the object.

If you configure an existing service object with a different protocol and port (or ports), the new configuration replaces the existing protocol and port (or ports) with the new ones.

Examples

The following example shows how to create a service object:

```
ciscoasa(config)# object service SERVOBJECT1
ciscoasa(config-service-object)# service tcp source eq www destination eq ssh
```

Related Commands

Command	Description
clear configure object	Clears all objects created.

Command	Description
service	Configures the protocol and port for the service object.

ocsp disable-nonce

To disable the nonce extension, use the ocsp disable-nonce command in crypto ca trustpoint configuration mode. To re-enable the nonce extension, use the **no** form of this command.

ocsp disable-nonce
no ocsp disable-nonce

Syntax Description

This command has no arguments or keywords.

Command Default

By default, OCSP requests include a nonce extension.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Crypto ca trustpoint configuration	• Yes	• Yes	• Yes	• Yes	• Yes

Command History

Release	Modification
7.2(1)	This command was added.

Usage Guidelines

When you use this command, the OCSP request does not include the OCSP nonce extension, and the ASA does not check it. By default, OCSP requests include a nonce extension, which cryptographically binds requests with responses to avoid replay attacks. However, some OCSP servers use pre-generated responses that do not contain this matching nonce extension. To use OCSP with these servers, you must disable the nonce extension.

Examples

The following example shows how to disable the nonce extension for a trustpoint called newtrust.

```
ciscoasa(config)# crypto ca trustpoint
 newtrust
ciscoasa(config-ca-trustpoint)# ocsp disable-nonce
ciscoasa(config-ca-trustpoint)#
```

Related Commands

Command	Description
crypto ca trustpoint	Enters crypto ca trustpoint configuration mode. Use this command in global configuration mode.
match certificate	Configures an OCSP override rule.
ocsp interface*nameif*	Specifies the interface that can be used in OCSP revocation check.

Command	Description
ocsp url	Specifies the OCSP server to use to check all certificates associated with a trustpoint.
revocation-check	Specifies the method(s) to use for revocation checking, and the order in which to try them.

ocsp interface

To configure the source interface for ASA to reach OCSP, use **interface** *nameif* command in the crypto ca trustpool configuration mode. To remove the interface from the configuration, use the **no** form of this command.

ocsp interface *nameif*
no ocsp interface *nameif*

Syntax Description

interface *nameif*	Specifies the interface that the ASA uses to reach the OCSP server.

Command Default

No defaults for this command.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Crypto ca trustpoint configuration.	• Yes	• Yes	• Yes	• Yes	• Yes

Command History

Release	Modification
9.5(1)	This command was added.

Usage Guidelines

By default, OCSP uses the global routing table that does not include management interface entries. If OCSP is behind a management interface, the OCSP revocation check does not succeed. When you use this command, the OCSP revocation check can be configured to use the interfaces, including management interface as required.

Examples

The following example shows how to configure the source interface for OCSP.

```
ciscoasa(config)# crypto ca trustpoint TP
ciscoasa(config-ca-trustpoint)# ocsp ?

crypto-ca-trustpoint mode commands/options:
  disable-nonce  Disable OCSP Nonce Extension
  interface      Configure Source interface
  url            OCSP server URL
ciscoasa(config-ca-trustpoint)# ocsp  interface
ciscoasa(config-ca-trustpoint)# ocsp interface ?

crypto-ca-trustpoint mode commands/options:
Current available interface(s):
  inside   Name of interface GigabitEthernet0/0.100
  inside1  Name of interface GigabitEthernet0/0.41
  mgmt     Name of interface Management0/0
  outside  Name of interface GigabitEthernet0/0.51
```

```
ciscoasa(config-ca-trustpoint)# ocsp interface mgmt
ciscoasa(config-ca-trustpoint)# ocsp interface mgmt ?

crypto-ca-trustpoint mode commands/options:
  disable-nonce  Disable OCSP Nonce Extension
  url            OCSP server URL
ciscoasa(config-ca-trustpoint)# ocsp interface mgmt  url
ciscoasa(config-ca-trustpoint)# ocsp interface mgmt url  ?

crypto-ca-trustpoint mode commands/options:
  LINE < 500 char  URL
ciscoasa(config-ca-trustpoint)# ocsp interface mgmt url http://lal-bagh:8888
```

Related Commands

Command	Description
ocsp url	Specifies the OCSP server to use to check all certificates associated with a trustpoint.
ocsp disable-nonce	Disables the nonce extension of the OCSP request.
revocation-check	Specifies the methods to use for revocation checking, and the order in which to try them.

ocsp url

To configure an OCSP server for the ASA to use to check all certificates associated with a trustpoint rather than the server specified in the AIA extension of the client certificate, use the **ocsp url** command in crypto ca trustpoint configuration mode. To remove the server from the configuration, use the **no** form of this command.

ocsp url *URL*
no ocsp url

Syntax Description

URL	Specifies the HTTP URL for the OCSP server. **Note** ASA supports both IPv4 and IPv6 OCSP URLs. Enclose IPv6 addresses in square brackets, for example: *http://[0:0:0:0:0.18:0a01:7c16]*.

Command Default

No default behavior or values.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Crypto ca trustpoint configuration	• Yes	• Yes	• Yes	• Yes	• Yes

Command History

Release	Modification
7.2(1)	This command was added.
9.20(1)	Support for IPv6 OCSP URL was added.

Usage Guidelines

The ASA supports only HTTP URLs, and you can specify only one URL per trustpoint.

The ASA provides three ways to define an OCSP server URL, and it attempts to use OCSP servers according to how you define them, in the following order:

- An OCSP server you set using **match certificate** command.
- An OCSP server you set using the **ocsp url** command.
- The OCSP server in the AIA field of the client certificate.

If you do not configure an OCSP URL via the **match certificate** command or the **ocsp url** command, the ASA uses the OCSP server in the AIA extension of the client certificate. If the certificate does not have an AIA extension, revocation status checking fails.

Examples

The following example shows how to configure an OCSP server with the URL http://10.1.124.22.

```
ciscoasa(config)# crypto ca trustpoint
 newtrust
ciscoasa(config-ca-trustpoint)# ocsp url http://10.1.124.22
ciscoasa(config-ca-trustpoint)#
```

The following example shows how to configure OCSP with the IPv6 URL:

```
ciscoasa(config)# crypto ca trustpoint
 newtrust
ciscoasa(config-ca-trustpoint)# ocsp url  http://[0:0:0:0:0:ffff:0a01:7c16]
ciscoasa(config-ca-trustpoint)#
```

Related Commands

Command	Description
crypto ca trustpoint	Enters crypto ca trustpoint configuration mode. Use this command in global configuration mode.
match certificate	Configures an OCSP override rule,
ocsp disable-nonce	Disables the nonce extension of the OCSP request.
ocsp interface*nameif*	Specifies the interface that can be used in OCSP revocation check.
revocation-check	Specifies the method(s) to use for revocation checking, and the order in which to try them.

onscreen-keyboard(Deprecated)

To insert an onscreen keyboard into the logon pane or all panes with a login/password requirement, use the **onscreen-keyboard** command in webvpn mode. To remove a previously configured onscreen keyboard, use the **no** version of the command.

onscreen-keyboard { **logon** | **all** }
no onscreen-keyboard [**logon** | **all**]

Syntax Description

logon	Inserts the onscreen keyboard for the logon pane.
all	Inserts the onscreen keyboard for the logon pane, and for all other panes with a login/password requirement.

Command Default

No onscreen keyboard.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Webvpn configuration mode	• Yes	—	• Yes	—	—

Command History

Release	Modification
8.0(2)	This command was added.
9.17(1)	This command was deprecated due to support removal for web VPN.

Usage Guidelines

The onscreen keyboard lets you enter user credentials without keystrokes.

Examples

The following example shows how to enable the onscreen keyboard for the logon page:

```
ciscoasa
(config)#
webvpn
ciscoasa
(config-webvpn)#
 onscreen-keyboard logon
ciscoasa(config-webvpn)#
```

Related Commands

Command	Description
webvpn	Enters webvpn mode, which lets you configure attributes for clientless SSLVPN connections.

ospf authentication

To enable the use of OSPF authentication, use the **ospf authentication** command in interface configuration mode. To restore the default authentication stance, use the **no** form of this command.

ospf authentication { **key-chain key-chain-name** | **message-digest** | **null** }
no ospf authentication

Syntax Description

key-chain *key-chain-name*	(Optional) Specifies a key chain to use for authentication. The key-name argument can be a maximum of 63 alphanumeric characters.
message-digest	(Optional) Specifies to use OSPF message digest authentication.
null	(Optional) Specifies to not use OSPF authentication.

Command Default

By default, OSPF authentication is not enabled.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Interface configuration	• Yes	—	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.0(1)	Support for multiple context mode was added.
9.12(1)	Key chain feature was added to support rotating keys for OSPF authentication.

Usage Guidelines

Before using the **ospf authentication** command, configure a password for the interface using the **ospf authentication-key** command. If you use the **message-digest** keyword, configure the message-digest key for the interface with the **ospf message-digest-key** command.

For backward compatibility, authentication type for an area is still supported. If the authentication type is not specified for an interface, the authentication type for the area will be used (the area default is null authentication).

When this command is used without any options, simple password authentication is enabled.

Examples

The following example shows how to enable simple password authentication for OSPF on the selected interface:

```
ciscoasa(config-if)# ospf authentication
ciscoasa(config-if)#
```

The following example shows how to enable key-chain password authentication for OSPF on the selected interface:

```
ciscoasa(config)# interface gigabitEthernet 0/0
ciscoasa(config-if)# ospf authentication key-chain CHAIN-INT-OSPFKEYS
```

Related Commands

Command	Description
ospf **authentication-key**	Specifies the password used by neighboring routing devices.
ospf message-digest-key	Enables MD5 authentication and specifies the MD5 key.

ospf authentication-key

To specify the password used by neighboring routing devices, use the **ospf authentication-key** command in interface configuration mode. To remove the password, use the **no** form of this command.

ospf authentication-key [**0** | **8** } *password*
no ospf authentication-key

Syntax Description

0	Specifies an unencrypted password will follow
8	Specifies an encrypted password will follow.
password	Assigns an OSPF authentication password for use by neighboring routing devices. The password must be less than 9 characters. You can include blank space between two characters. Spaces at the beginning or end of the password are ignored.

Command Default

No default behavior or values.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Interface configuration	• Yes	—	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.0(1)	Support for multiple context mode was added.

Usage Guidelines

The password created by this command is used as a key that is inserted directly into the OSPF header when routing protocol packets are originated. A separate password can be assigned to each network on a per-interface basis. All neighboring routers on the same network must have the same password to be able to exchange OSPF information.

Examples

The following example shows how to specify a password for OSPF authentication:

```
ciscoasa(config-if)# ospf authentication-key 8
yWIvi0qJAnGK5MRWQzrhIohkGP1wKb
```

Related Commands

Command	Description
area authentication	Enables OSPF authentication for the specified area.
ospf authentication	Enables the use of OSPF authentication.

ospf cost

To specify the cost of sending a packet through the interface, use the **ospf cost** command in interface configuration mode. To reset the interface cost to the default value, use the **no** form of this command.

ospf cost *interface_cost*
no ospf cost

Syntax Description

interface_cost	The cost (a link-state metric) of sending a packet through an interface. This is an unsigned integer value from 0 to 65535. 0 represents a network that is directly connected to the interface, and the higher the interface bandwidth, the lower the associated cost to send packets across that interface. In other words, a large cost value represents a low bandwidth interface and a small cost value represents a high bandwidth interface. The OSPF interface default cost on the ASA is 10. This default differs from Cisco IOS software, where the default cost is 1 for Fast Ethernet and Gigabit Ethernet and 10 for 10BaseT. This is important to take into account if you are using ECMP in your network.

Command Default

The default *interface_cost* is 10.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Interface configuration	• Yes	—	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.0(1)	Support for multiple context mode was added.

Usage Guidelines

The **ospf cost** command lets you explicitly specify the cost of sending a packet on an interface. The *interface_cost* parameter is an unsigned integer value from 0 to 65535.

The **no ospf cost** command allows you to reset the path cost to the default value.

Examples

The following example show how to specify the cost of sending a packet on the selected interface:

```
ciscoasa(config-if)# ospf cost 4
```

Related Commands

Command	Description
show running-config interface	Displays the configuration of the specified interface.

ospf database-filter

To filter out all outgoing LSAs to an OSPF interface during synchronization and flooding, use the **ospf database-filter** command in interface configuration mode. To restore the LSAs, use the **no** form of this command.

ospf database-filter all out
no ospf database-filter all out

Syntax Description

all out	Filters all outgoing LSAs to an OSPF interface.

Command Default

No default behavior or values.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Interface configuration	• Yes	—	• Yes	—	—

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

The **ospf database-filter** command filters outgoing LSAs to an OSPF interface. The **no ospf database-filter all out** command restores the forwarding of LSAs to the interface.

Examples

The following example shows how to use the **ospf database-filter** command to filter outgoing LSAs:

```
ciscoasa(config-if)# ospf database-filter all out
```

Related Commands

Command	Description
show interface	Displays interface status information.

ospf dead-interval

To specify the interval before neighbors declare a router down, use the **ospf dead-interval** command in interface configuration mode. To restore the default value, use the **no** form of this command.

ospf dead-interval { *seconds* **minimal** | **hello-multiplier** *multiplier* }
no ospf dead-interval

Syntax Description

seconds	The length of time during which no hello packets are seen. The default for *seconds* is four times the interval set by the **ospf hello-interval** command (which ranges from 1 to 65535).
minimal	Sets the dead interval to 1 second. Using this keyword requires that the hello-multiplier keyword and multiplier argument are also configured.
hello-multiplier multiplier	Integer value in the range from 3 to 20, representing the number of hello packets sent during 1 second.

Command Default

The default value for *seconds* is four times the interval set by the **ospf hello-interval** command.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Interface configuration	• Yes	—	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.0(1)	Support for multiple context mode was added.
9.2(1)	Support for Fast Hello packets was added.

Usage Guidelines

The ospf dead-interval command lets you set the dead interval before neighbors to declare the router down (the length of time during which no hello packets are seen). The *seconds* argument specifies the dead interval and must be the same for all nodes on the network. The default for *seconds* is four times the interval set by the **ospf hello-interval** command from 1 to 65535.

The **no** ospf dead-interval command restores the default interval value.

The dead interval is advertised in OSPF hello packets. This value must be the same for all networking devices on a specific network.

Specifying a smaller dead interval (seconds) will give faster detection of a neighbor being down and improve convergence, but might cause more routing instability.

OSPF Support for Fast Hello Packets

By specifying the minimal and hello-multiplier keywords with a multiplier argument, you are enabling OSPF fast hello packets. The minimal keyword sets the dead interval to 1 second, and the hello-multiplier value sets the number of hello packets sent during that 1 second, thus providing subsecond or "fast" hello packets.

When fast hello packets are configured on the interface, the hello interval advertised in the hello packets that are sent out this interface is set to 0. The hello interval in the hello packets received over this interface is ignored.

The dead interval must be consistent on a segment, whether it is set to 1 second (for fast hello packets) or set to any other value. The hello multiplier need not be the same for the entire segment as long as at least one hello packet is sent within the dead interval.

Use the show ospf interface command to verify the dead interval and fast hello interval.

Examples

In the following example, OSPF Support for Fast Hello Packets is enabled by specifying the minimal keyword and the hello-multiplier keyword and value. Because the multiplier is set to 5, five hello packets will be sent every second.

```
ciscoasa(config-if)# ospf dead-interval minimal hello-multiplier 5
```

Related Commands

Command	Description
ospf hello-interval	Specifies the interval between hello packets sent on an interface.
show ospf interface	Displays OSPF-related interface information.

ospf hello-interval

To specify the interval between hello packets sent on an interface, use the **ospf hello-interval** command in interface configuration mode. To return the hello interval to the default value, use the **no** form of this command.

ospf hello-interval *seconds*
no ospf hello-interval

Syntax Description

seconds	Specifies the interval between hello packets that are sent on the interface; valid values are from 1 to 65535 seconds.

Command Default

The default value for **hello-interval** *seconds* is 10 seconds.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Interface configuration	• Yes	—	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.0(1)	Support for multiple context mode was added.

Usage Guidelines

This value is advertised in the hello packets. The smaller the hello interval, the faster topological changes will be detected, but more routing traffic will ensue. This value must be the same for all routers and access servers on a specific network.

Examples

The following example sets the OSPF hello interval to 5 seconds:

```
ciscoasa(config-if)# ospf hello-interval 5
```

Related Commands

Command	Description
ospf dead-interval	Specifies the interval before neighbors declare a router down.
show ospf interface	Displays OSPF-related interface information.

ospf message-digest-key

To enable OSPF MD5 authentication, use the **ospf message-digest-key** command in interface configuration mode. To remove an MD5 key, use the **no** form of this command.

ospf message-digest-key *key-id* **md5** [**0** | **8**] *key*
no ospf message-digest-key

Syntax Description

key-id	Enables MD5 authentication and specifies the numerical authentication key ID number; valid values are from 1 to 255.
md5 *key*	Alphanumeric password of up to 16 bytes. You can include spaces between key characters. Spaces at the beginning or end of the key are ignored. MD5 authentication verifies the integrity of the communication, authenticates the origin, and checks for timeliness.
0	Specifies an unencrypted password will follow
8	Specifies an encrypted password will follow.

Command Default

No default behavior or values.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Interface configuration	• Yes	—	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.0(1)	Support for multiple context mode was added.

Usage Guidelines

The ospf message-digest-key command lets you enable MD5 authentication. The **no** form of the command let you remove an old MD5 key. *key_id* is a numerical identifier from 1 to 255 for the authentication key. *key* is an alphanumeric password of up to 16 bytes. MD5 verifies the integrity of the communication, authenticates the origin, and checks for timeliness.

Examples

The following example shows how to specify an MD5 key for OSPF authentication:

```
ciscoasa(config-if)# ospf message-digest-key 3 md5 8
yWIvi0qJAnGK5MRWQzrhIohkGP1wKb
```

Related Commands

Command	Description
area authentication	Enables OSPF area authentication.
ospf authentication	Enables the use of OSPF authentication.

ospf mtu-ignore

To disable OSPF maximum transmission unit (MTU) mismatch detection on receiving database packets, use the **ospf mtu-ignore** command in interface configuration mode. To restore MTU mismatch detection, use the **no** form of this command.

ospf mtu-ignore
no ospf mtu-ignore

Syntax Description

This command has no arguments or keywords.

Command Default

By default, **ospf mtu-ignore** is enabled.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Interface configuration	• Yes	—	• Yes	—	—

Command History

Release	Modification
7.0(1)	This command was added.

Usage Guidelines

OSPF checks whether neighbors are using the same MTU on a common interface. This check is performed when neighbors exchange Database Descriptor (DBD) packets. If the receiving MTU in the DBD packet is higher than the IP MTU configured on the incoming interface, OSPF adjacency will not be established.The **ospf mtu-ignore** command disables OSPF MTU mismatch detection on receiving DBD packets. It is enabled by default.

Examples

The following example shows how to disable the **ospf mtu-ignore** command:

```
ciscoasa(config-if)# ospf mtu-ignore
```

Related Commands

Command	Description
show interface	Displays interface status information.

ospf network point-to-point non-broadcast

To configure the OSPF interface as a point-to-point, non-broadcast network, use the **ospf network point-to-point non-broadcast** command in interface configuration mode. To remove this command from the configuration, use the **no** form of this command.

ospf network point-to-point non-broadcast
no ospf network point-to-point non-broadcast

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Interface configuration	• Yes	—	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.0(1)	Support for multiple context mode was added.

Usage Guidelines

The **ospf network point-to-point non-broadcast** command lets you to transmit OSPF routes over VPN tunnels.

When the interface is specified as point-to-point, the OSPF neighbors have to be manually configured; dynamic discovery is not possible. To manually configure OSPF neighbors, use the **neighbor** command in router configuration mode.

When an interface is configured as point-to-point, the following restrictions apply:

- > You can define only one neighbor for the interface.
- You need to define a static route pointing to the crypto endpoint.
- The interface cannot form adjacencies unless neighbors are configured explicitly.
- If OSPF over the tunnel is running on the interface, regular OSPF with an upstream router cannot be run on the same interface.
- You should bind the crypto-map to the interface before specifying the OSPF neighbor to ensure that the OSPF updates are passed through the VPN tunnel. If you bind the crypto-map to the interface after specifying the OSPF neighbor, use the **clear local-host all** command to clear OSPF connections so the OSPF adjacencies can be established over the VPN tunnel.

Examples

The following example shows how to configure the selected interface as a point-to-point, non-broadcast interface:

```
ciscoasa(config-if)# ospf network point-to-point non-broadcast
ciscoasa(config-if)#
```

Related Commands

Command	Description
neighbor	Specifies manually configured OSPF neighbors.
show interface	Displays interface status information.

ospf priority

To change the OSPF router priority, use the **ospf priority** command in interface configuration mode. To restore the default priority, use the **no** form of this command.

ospf priority *number*
no ospf priority [*number*]

Syntax Description

number	Specifies the priority of the router; valid values are from 0 to 255.

Command Default

The default value for *number* is 1.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Interface configuration	• Yes	—	• Yes	• Yes	—

Usage Guidelines

When two routers attached to a network both attempt to become the designated router, the one with the higher router priority takes precedence. If there is a tie, the router with the higher router ID takes precedence. A router with a router priority set to zero is ineligible to become the designated router or backup designated router. Router priority is configured only for interfaces to multiaccess networks (in other words, not to point-to-point networks).

In multiple context mode, for shared interfaces, specify 0 to ensure the device does not become the designated router. OSPFv2 instances cannot form adjacencies with each other across shared interfaces.

Examples

The following example shows how to change the OSPF priority on the selected interface:

```
ciscoasa(config-if)# ospf priority 4
ciscoasa(config-if)#
```

Related Commands

Command	Description
show ospf interface	Displays OSPF-related interface information.

ospf retransmit-interval

To specify the time between LSA retransmissions for adjacencies belonging to the interface, use the **ospf retransmit-interval** command in interface configuration mode. To restore the default value, use the **no** form of this command.

ospf retransmit-interval [*seconds*]
no ospf retransmit-interval [*seconds*]

Syntax Description

seconds	Specifies the time between LSA retransmissions for adjacent routers belonging to the interface; valid values are from 1 to 65535 seconds.

Command Default

The default value of **retransmit-interval** *seconds* is 5 seconds.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Interface configuration	• Yes	—	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.0(1)	Support for multiple context mode was added.

Usage Guidelines

When a router sends an LSA to its neighbor, it keeps the LSA until it receives the acknowledgment message. If the router receives no acknowledgment, it will re-send the LSA.

The setting of this parameter should be conservative, or needless retransmission will result. The value should be larger for serial lines and virtual links.

Examples

The following example shows how to change the retransmit interval for LSAs:

```
ciscoasa(config-if)# ospf retransmit-interval 15
ciscoasa(config-if)#
```

Related Commands

Command	Description
show ospf interface	Displays OSPF-related interface information.

ospf transmit-delay

To set the estimated time required to send a link-state update packet on the interface, use the **ospf transmit-delay** command in interface configuration mode. To restore the default value, use the **no** form of this command.

ospf transmit-delay [*seconds*]
no ospf transmit-delay [*seconds*]

Syntax Description

seconds	Sets the estimated time required to send a link-state update packet on the interface. The default value is 1 second with a range from 1 to 65535 seconds.

Command Default

The default value of *seconds* is 1 second.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Interface configuration	• Yes	—	• Yes	• Yes	—

Command History

Release	Modification
7.0(1)	This command was added.
9.0(1)	Support for multiple context mode was added.

Usage Guidelines

LSAs in the update packet must have their ages incremented by the amount specified in the *seconds* argument before transmission. The value assigned should take into account the transmission and propagation delays for the interface.

If the delay is not added before transmission over a link, the time in which the LSA propagates over the link is not considered. This setting has more significance on very low-speed links.

Examples

The following example sets the transmit delay to 3 seconds for the selected interface:

```
ciscoasa(config-if)# ospf restransmit-delay 3
ciscoasa(config-if)#
```

Related Commands

Command	Description
show ospf interface	Displays OSPF-related interface information.

otp expiration

To specify the duration in hours that an issued One-Time Password (OTP) for the local Certificate Authority (CA) enrollment page is valid, use the **otp expiration** command in ca server configuration mode. To reset the duration to the default number of hours, use the **no** form of this command.

otp expiration *timeout*
no otp expiration

Syntax Description

timeout	Specifies the time in hours users have to enroll for a certificate from the local CA before the OTP for the enrollment page expires. Valid values range from 1 to 720 hours (30 days).

Command Default

By default, a OTP expiration for certificate enrollment is 72 hours (3 days).

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Ca server configuration	• Yes	—	• Yes	—	—

Command History

Release	Modification
8.0(2)	This command was added.

Usage Guidelines

The OTP expiration period specifies the number of hours that a user has to log in to the enrollment page of the CA server. After the user logs in and enrolls for a certificate, the time period specified by the **enrollment retrieval** command starts.

Note The user OTP for enrolling for a certificate with the enrollment interface page is also used as the password to unlock the PKCS12 file containing the issued certificate and keypair for that user.

Examples

The following example specifies that the OTP for the enrollment page applies for 24 hours:

```
ciscoasa(config)# crypto ca server
ciscoasa
(config-ca-server)
# otp expiration 24
ciscoasa
(config-ca-server)
#
```

The following example resets the OTP duration to the default of 72 hours:

```
ciscoasa(config)# crypto ca server
ciscoasa
(config-ca-server)
)# no otp expiration
ciscoasa
(config-ca-server)
#
```

Related Commands

Command	Description
crypto ca server	Provides access to the ca server configuration mode command set, which allows you to configure and manage the local CA.
enrollment-retrieval	Specifies the time in hours that an enrolled user can retrieve a PKCS12 enrollment file.
show crypto ca server	Displays the certificate authority configuration.

output console

To send the output of the **action** commands to the console, use the **output console** command in event manager applet configuration mode. To remove the console as an output destination, use the **no** form of this command.

output console
no output console

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Event manager applet configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
9.2(1)	This command was added.

Usage Guidelines

Use this command to send the output of the **action** commands to the console.

Examples

The following example sends the output of the **action** commands to the console:

```
ciscoasa(config-applet)# output console
```

Related Commands

Command	Description
output file append	Writes the **action** command output to a single file, but that file is appended to every time.
output file new	Sends the output of the **action** commands to a new file for each applet that is invoked.
output file overwrite	Writes the **action** command output to a single file, which is truncated every time.
output file rotate	Creates a set of files that are rotated.
output none	Discards any output from the **action** commands.

output file

To redirect the **action** command output to a specified file, use the **output file** command in event manager applet configuration mode. To remove the specified action, use the **no** form of this command.

output file [**append** *filename* | **new** | **overwrite** *filename* | **rotate** *n*]
no output file [**append** *filename* | **new** | **overwrite** *filename* | **rotate** *n*]

Syntax Description

append *filename*	Continuously appends output to the specified filename, which is a local (to the ASA) filename.
new	Creates a new file for output named eem-*applet -timestamp* .log, in which *applet* is the name of the event manager applet and *timestamp* is a dated timestamp in the format of YYYYMMDD-hhmmss.
overwrite *filename*	Writes output to the specified filename, but truncates the output each time an event manager applet is invoked.
rotate *n*	Creates a file for output named eem-*applet -x* .log, in which *applet* is the name of the event manager applet, and *x* is the file number. When a new file is to be written, the oldest file is deleted, and all subsequent files are renumbered before the first file is written. The newest file is indicated by 0, and the oldest file is indicated by the highest number (*n* -1). The *n* argument specifies the rotate value. Valid values range from 2 - 100.

Command Default

No default behavior or values.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Event manager applet configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
9.2(1)	This command was added.

Usage Guidelines

Use the **output file** command to redirect the **action** command output to a specified file.

Examples

The following example appends output to a single file:

```
ciscoasa(config-applet)# output file append examplefile1
```

The following example sends the output of the **action** commands to a new file:

```
ciscoasa(config-applet)# output file new
```

The following example writes output to a single, truncated file:

```
ciscoasa(config-applet)# output file overwrite examplefile1
```

The following example creates a set of files that are rotated:

```
ciscoasa(config-applet)# output file rotate 50
```

Related Commands

Command	Description
output console	Sends the output of the **action** commands to the console.
output none	Discards any output from the **action** commands.

output none

To discard any output from the **action** commands, use the **output none** command in event manager applet configuration mode. To retain output from the **action** commands, use the **no** form of this command.

output none
no output none

Syntax Description

This command has no arguments or keywords.

Command Default

The default is to discard any output from **action** commands.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Event manager applet configuration	• Yes	• Yes	• Yes	• Yes	—

Command History

Release	Modification
9.2(1)	This command was added.

Usage Guidelines

Use this command to discard any output from the **action** commands.

Examples

The following example discards any output from the **action** commands:

```
ciscoasa(config-applet)# output none
```

Related Commands

Command	Description
output console	Sends the output of the **action** commands to the console.
output file append	Writes the **action** command output to a single file, but that file is appended to every time.
output file new	Sends the output of the **action** commands to a new file for each applet that is invoked.
output file overwrite	Writes the **action** command output to a single file, which is truncated every time.
output file rotate	Creates a set of files that are rotated.

outstanding (Deprecated)

Note The last supported release of this command was Version 9.5(1).

To limit the number of unauthenticated e-mail proxy sessions, use the **outstanding** command in the applicable e-mail proxy configuration mode. To remove the attribute from the configuration, use the **no** form of this command.

outstanding { *number* }
no outstanding

Syntax Description

number	The number of unauthenticated sessions permitted. The range is from 1 to 1000.

Command Default

The default is 20.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Pop3s	• Yes	—	• Yes	•	—
Imap4s	• Yes	—	• Yes	—	—
Smtps	• Yes	—	• Yes	—	—

Command History

Release	Modification
7.0(1)	This command was added.
9.5(2)	This command was deprecated.

Usage Guidelines

Use the no version of this command to remove the attribute from the configuration, which permits an unlimited number of unauthenticated sessions. This also limit s DOS attacks on the e-mail ports.

E-mail proxy connections have three states:

- **1.**A new e-mail connection enters the “unauthenticated” state.
- **2.**When the connection presents a username, it enters the “authenticating” state.
- **3.**When the ASA authenticates the connection, it enters the “authenticated” state.

If the number of connections in the unauthenticated state exceeds the configured limit, the ASA terminates the oldest unauthenticated connection, preventing overload. It does not terminate authenticated connections.

Examples

The following example shows how to set a limit of 12 unauthenticated sessions for POP3S e-mail proxy.

```
ciscoasa
(config)#
 pop3s
ciscoasa(config-pop3s)
#
 outstanding 12
```

override-account-disable (Deprecated)

Note The last supported release for this command was Version 9.5(1).

To override an account-disabled indication from a AAA server, use the **override-account-disable** command in tunnel-group general-attributes configuration mode. To disable an override, use the **no** form of this command.

override-account-disable
no override-account-disable

Syntax Description

This command has no arguments or keywords.

Command Default

This command is disabled by default.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Tunnel-group general-attributes configuration	• Yes	—	• Yes	—	—

Command History

Release	Modification
7.1(1)	This command was added.
9.5(2)	This command was deprecated.

Usage Guidelines

This command is valid for servers, such as RADIUS with NT LDAP, and Kerberos, that return an “account-disabled” indication.

You can configure this attribute for IPsec RA and WebVPN tunnel-groups.

Examples

The following example allows overriding the “account-disabled” indicator from the AAA server for the WebVPN tunnel group “testgroup”:

```
ciscoasa(config)# tunnel-group testgroup type webvpn
ciscoasa(config)# tunnel-group testgroup general-attributes
ciscoasa(config-tunnel-general)# override-account-disable
ciscoasa(config-tunnel-general)#
```

The following example allows overriding the “account-disabled” indicator from the AAA server for the IPsec remote access tunnel group “QAgroup”:

```
ciscoasa(config)# tunnel-group QAgroup type ipsec-ra
ciscoasa(config)# tunnel-group QAgroup general-attributes
ciscoasa(config-tunnel-general)# override-account-disable
ciscoasa(config-tunnel-general)#
```

Related Commands

Command	Description
clear configure tunnel-group	Clears the tunnel-group database or the configuration for a particular tunnel group.
tunnel-group general-attributes	Configures the tunnel-group general-attributes values.

override-svc-download

To configure the connection profile to override the group policy or username attributes configuration for downloading an AnyConnect or SSL VPN client, use the **override-svc-download** command from tunnel-group webvpn attributes configuration mode. To remove the command from the configuration, use the **no** form of the command:

override-svc-download enable
no override-svc-download enable

Command Default

The default is disabled. The ASA does not override the group policy or username attributes configuration for downloading the client.

Command Modes

The following table shows the modes in which you can enter the command:

Command Mode	Firewall Mode		Security Context		
	Routed	Transparent	Single	Multiple	
				Context	System
Tunnel-group webvpn configuration	• Yes	—	• Yes	—	—

Command History

Release	Modification
8.0(2)	This command was added.

Usage Guidelines

The security appliance allows clientless, AnyConnect, or SSL VPN client connections for remote users based on whether clientless and/or SSL VPN is enabled in the group policy or username attributes with the **vpn-tunnel-protocol** command. The **svc ask** command further modifies the client user experience by prompting the user to download the client or return to the WebVPN home page.

However, you may want clientless users logging in under specific tunnel groups to not experience delays waiting for the download prompt to expire before being presented with the clientless SSL VPN home page. You can prevent delays for these users at the connection profile level with the **override-svc-download** command. This command causes users logging through a connection profile to be immediately presented with the clientless SSL VPN home page regardless of the **vpn-tunnel-protocol** or **svc ask** command settings.

Examples

In the following example, the user enters tunnel-group webvpn attributes configuration mode for the connection profile >*engineering* and enables the connection profile to override the group policy and username attribute settings for client download prompts:

```
ciscoasa(config)# tunnel-group engineering webvpn-attributes
ciscoasa(config-tunnel-webvpn)# override-svc-download
```

Related Commands

Command	Description
show webvpn svc	Displays information about installed SSL VPN clients.
svc	Enables or requires the SSL VPN client for a specific group or user.
svc image	Specifies a client package file that the ASA expands in cache memory for downloading to remote PCs.