CHAPTER 39

police through pppoe client secondary Commands

police

To apply QoS policing to a class map, use the **police** command in class configuration mode. To remove the rate-limiting requirement, use the **no** form of this command. Policing is a way of ensuring that no traffic exceeds the maximum rate (in bits/second) that you configure, thus ensuring that no one traffic flow can take over the entire resource. When traffic exceeds the maximum rate, the ASA drops the excess traffic. Policing also sets the largest single burst of traffic allowed.

police {**output** | **input**} *conform-rate* [*conform-burst*] [**conform-action** [**drop** | **transmit**] [**exceed-action** [**drop** | **transmit**]]]

no police

Syntax Description

conform-burst	Specifies the maximum number of instantaneous bytes allowed in a sustained burst before throttling to the conforming rate value, between 1000 and 512000000 bytes.
conform-action	Sets the action to take when the rate is less than the *conform_burst* value.
conform-rate	Sets the rate limit for this traffic flow; between 8000 and 2000000000 bits per second.
drop	Drops the packet.
exceed-action	Sets the action to take when the rate is between the *conform-rate* value and the *conform-burst* value.
input	Enables policing of traffic flowing in the input direction.
output	Enables policing of traffic flowing in the output direction.
transmit	Transmits the packet.

Defaults

No default behavior or variables.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.
7.2(1)	Added the **input** option. Policing traffic in the inbound direction is now supported.

Usage Guidelines

To enable policing, use the Modular Policy Framework:

1. **class-map**—Identify the traffic on which you want to perform policing.

2. **policy-map**—Identify the actions associated with each class map.
   a. **class**—Identify the class map on which you want to perform actions.
   b. **police**—Enable policing for the class map.
3. **service-policy**—Assigns the policy map to an interface or globally.

Note The **police** command merely enforces the maximum speed and burst rate, forcing them to the conforming rate value. It does not enforce the **conform-action** or the **exceed-action** specification if these are present.

Note When the conform-burst parameter is omitted, the default value is assumed to be 1/32 of the conform-rate in bytes (that is, with a conform rate of 100,000, the default conform-burst value would be 100,000/32 = 3,125). Note that the conform-rate is in bits/second, whereas the conform-burst is in bytes.

You can configure each of the QoS features alone if desired for the ASA. Often, though, you configure multiple QoS features on the ASA so you can prioritize some traffic, for example, and prevent other traffic from causing bandwidth problems.

See the following supported feature combinations per interface:

- Standard priority queuing (for specific traffic) + Policing (for the rest of the traffic).

  You cannot configure priority queueing and policing for the same set of traffic.
- Traffic shaping (for all traffic on an interface) + Hierarchical priority queueing (for a subset of traffic).

Typically, if you enable traffic shaping, you do not also enable policing for the same traffic, although the ASA does not restrict you from configuring this.

See the following guidelines:

- QoS is applied unidirectionally; only traffic that enters the interface to which you apply the policy map is affected (or exits the interface, depending on the whether you specify **input** or **output**).
- If a service policy is applied or removed from an interface that has existing traffic already established, the QoS policy is not applied or removed from the traffic stream. To apply or remove the QoS policy for such connections, you must clear the connections and re-establish them. See the **clear conn** command.
- To-the-box traffic is not supported.
- Traffic to and from a VPN tunnel bypass interface is not supported.
- When you match a tunnel group class map, only outbound policing is supported.

Examples

The following is an example of a **police** command for the output direction that sets the conform rate to 100,000 bits per second, a burst value of 20,000 bytes, and specifies that traffic that exceeds the burst rate will be dropped:

```
hostname(config)# policy-map localpolicy1
hostname(config-pmap)# class-map firstclass
hostname(config-cmap)# class localclass
hostname(config-pmap-c)# police output 100000 20000 exceed-action drop
hostname(config-cmap-c)# class class-default
hostname(config-pmap-c)#
```

The following example shows how to do rate-limiting on traffic destined to an internal web server:

```
hostname# access-list http_traffic permit tcp any 10.1.1.0 255.255.255.0 eq 80
hostname# class-map http_traffic
hostname(config-cmap)# match access-list http_traffic
hostname(config-cmap)# policy-map outside_policy
hostname(config-pmap)# class http_traffic
hostname(config-pmap-c)# police input 56000
hostname(config-pmap-c)# service-policy outside_policy interface outside
hostname(config)#
```

Related Commands

class	Specifies a class-map to use for traffic classification.
clear configure policy-map	Remove all policy-map configuration, except that if a policy-map is in use in a service-policy command, that policy-map is not removed.
policy-map	Configures a policy; that is, an association of a traffic class and one or more actions.
show running-config policy-map	Display all current policy-map configurations.

policy

To specify the source for retrieving the CRL, use the **policy** command in ca-crl configuration mode.

policy {**static** | **cdp** | **both**}

Syntax Description

both	Specifies that if obtaining a CRL using the CRL distribution point fails, retry using static CDPs up to a limit of five.
cdp	Uses the CDP extension embedded within the certificate being checked. In this case, the ASA retrieves up to five CRL distributions points from the CDP extension of the certificate being verified and augments their information with the configured default values, if necessary. If the ASA attempt to retrieve a CRL using the primary CDP fails, it retries using the next available CDP in the list. This continues until either the ASA retrieves a CRL or exhausts the list.
static	Uses up to five static CRL distribution points. If you specify this option, specify also the LDAP or HTTP URLs with the **protocol** command.

Defaults

The default setting is **cdp**.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
crl configuration	•	—	•	—	—

Command History

Release	Modification
7.0(1)	This command was introduced.

Examples

The following example enters ca-crl configuration mode, and configures CRL retrieval to occur using the CRL distribution point extension in the certificate being checked or if that fails, to use static CDPs:

```
hostname(configure)# crypto ca trustpoint central
hostname(ca-trustpoint)# crl configure
hostname(ca-crl)# policy both
```

Related Commands

Command	Description
crl configure	Enters ca-crl configuration mode.
crypto ca trustpoint	Enters trustpoint configuration mode.
url	Creates and maintains a list of static URLs for retrieving CRLs.

policy-map

When using the Modular Policy Framework, assign actions to traffic that you identified with a Layer 3/4 class map (the **class-map** or **class-map type management** command) by using the **policy-map** command (without the **type** keyword) in global configuration mode. To remove a Layer 3/4 policy map, use the **no** form of this command.

policy-map *name*

no policy-map *name*

Syntax Description

name	Specifies the name for this policy map up to 40 characters in length. All types of policy maps use the same name space, so you cannot reuse a name already used by another type of policy map.

Defaults

By default, the configuration includes a policy that matches all default application inspection traffic and applies certain inspections to the traffic on all interfaces (a global policy). Not all inspections are enabled by default. You can only apply one global policy, so if you want to alter the global policy, you need to either edit the default policy or disable it and apply a new one. (An interface policy overrides the global policy for a particular feature.)

The default policy includes the following application inspections:

- DNS inspection for the maximum message length of 512 bytes
- FTP
- H323 (H225)
- H323 (RAS)
- RSH
- RTSP
- ESMTP
- SQLnet
- Skinny (SCCP)
- SunRPC
- XDMCP
- SIP
- NetBios
- TFTP
- IP Options

The default policy configuration includes the following commands:

```
class-map inspection_default
 match default-inspection-traffic
policy-map type inspect dns preset_dns_map
 parameters
message-length maximum client auto
message-length maximum 512
```

```
dns-guard
protocol-enforcement
nat-rewrite
policy-map global_policy
 class inspection_default
  inspect dns preset_dns_map
  inspect ftp
  inspect h323 h225 _default_h323_map
  inspect h323 ras _default_h323_map
  inspect ip-options _default_ip_options_map
  inspect netbios
  inspect rsh
  inspect rtsp
  inspect skinny
  inspect esmtp _default_esmtp_map
  inspect sqlnet
  inspect sunrpc
  inspect tftp
  inspect sip
  inspect xdmcp
```

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Configuring Modular Policy Framework consists of four tasks:

1. Identify the Layer 3 and 4 traffic to which you want to apply actions using the **class-map** or **class-map type management** command.
2. (Application inspection only) Define special actions for application inspection traffic using the **policy-map type inspect** command.
3. Apply actions to the Layer 3 and 4 traffic using the **policy-map** command.
4. Activate the actions on an interface using the **service-policy** command.

The maximum number of policy maps is 64, but you can only apply one policy map per interface. You can apply the same policy map to multiple interfaces. You can identify multiple Layer 3/4 class maps in a Layer 3/4 policy map (see the **class** command), and you can assign multiple actions from one or more feature types to each class map.

Examples

The following is an example of a **policy-map** command for connection policy. It limits the number of connections allowed to the web server 10.1.1.1:

```
hostname(config)# access-list http-server permit tcp any host 10.1.1.1
hostname(config)# class-map http-server
hostname(config-cmap)# match access-list http-server

hostname(config)# policy-map global-policy
hostname(config-pmap)# description This policy map defines a policy concerning connection
to http server.
hostname(config-pmap)# class http-server
hostname(config-pmap-c)# set connection conn-max 256
```

The following example shows how multi-match works in a policy map:

```
hostname(config)# class-map inspection_default
hostname(config-cmap)# match default-inspection-traffic
hostname(config)# class-map http_traffic
hostname(config-cmap)# match port tcp eq 80

hostname(config)# policy-map outside_policy
hostname(config-pmap)# class inspection_default
hostname(config-pmap-c)# inspect http http_map
hostname(config-pmap-c)# inspect sip
hostname(config-pmap)# class http_traffic
hostname(config-pmap-c)# set connection timeout tcp 0:10:0
```

The following example shows how traffic matches the first available class map, and will not match any subsequent class maps that specify actions in the same feature domain:

```
hostname(config)# class-map telnet_traffic
hostname(config-cmap)# match port tcp eq 23
hostname(config)# class-map ftp_traffic
hostname(config-cmap)# match port tcp eq 21
hostname(config)# class-map tcp_traffic
hostname(config-cmap)# match port tcp range 1 65535
hostname(config)# class-map udp_traffic
hostname(config-cmap)# match port udp range 0 65535
hostname(config)# policy-map global_policy
hostname(config-pmap)# class telnet_traffic
hostname(config-pmap-c)# set connection timeout tcp 0:0:0
hostname(config-pmap-c)# set connection conn-max 100
hostname(config-pmap)# class ftp_traffic
hostname(config-pmap-c)# set connection timeout tcp 0:5:0
hostname(config-pmap-c)# set connection conn-max 50
hostname(config-pmap)# class tcp_traffic
hostname(config-pmap-c)# set connection timeout tcp 2:0:0
hostname(config-pmap-c)# set connection conn-max 2000
```

When a Telnet connection is initiated, it matches **class telnet_traffic**. Similarly, if an FTP connection is initiated, it matches **class ftp_traffic**. For any TCP connection other than Telnet and FTP, it will match **class tcp_traffic**. Even though a Telnet or FTP connection can match **class tcp_traffic**, the ASA does not make this match because they previously matched other classes.

NetFlow events are configured through Modular Policy Framework. If Modular Policy Framework is not configured for NetFlow, no events are logged. Traffic is matched based on the order in which classes are configured. After a match is detected, no other classes are checked. For NetFlow events, the configuration requirements are as follows:

- A flow-export destination (that is, a NetFlow collector) is uniquely identified by its IP address.
- Supported event types are flow-create, flow-teardown, flow-denied, flow-update, and all, which include the four previously listed event types.
- Use the **flow-export event-type** {**all** | **flow-create** | **flow-denied** | **flow-update** | **flow-teardown**} **destination** command to configure the address of NetFlow collectors and filters to determine which NetFlow records should be sent to each collector.

- Flow-export actions are not supported in interface policies.
- Flow-export actions are only supported in the **class-default** command and in classes with the **match any** or **match access-list** command.
- If no NetFlow collector has been defined, no configuration actions occur.
- NetFlow Secure Event Logging filtering is order-independent.

The following example exports all NetFlow events between hosts 10.1.1.1 and 20.1.1.1 to destination 15.1.1.1.

```
hostname(config)# access-list flow_export_acl permit ip host 10.1.1.1 host 20.1.1.1
hostname(config)# class-map flow_export_class
hostname(config-cmap)# match access-list flow_export_acl
hostname(config)# policy-map global_policy
hostname(config-pmap)# class flow_export_class
hostname(config-pmap-c)# flow-export event-type all destination 15.1.1.1
```

Related Commands

Command	Description
class	Identifies a class map name in the policy map.
clear configure policy-map	Removes all policy map configuration. If a policy map is in use in a **service-policy** command, that policy map is not removed.
class-map	Defines a traffic class map.
service-policy	Assigns the policy map to an interface or globally to all interfaces.
show running-config policy-map	Display all current policy map configurations.

policy-map type inspect

When using the Modular Policy Framework, define special actions for inspection application traffic by using the **policy-map type inspect** command in global configuration mode. To remove an inspection policy map, use the **no** form of this command.

policy-map type inspect *application policy_map_name*

no policy-map [**type inspect** *application*] *policy_map_name*

Syntax Description

application	Specifies the type of application traffic you want to act upon. Available types include: • **dcerpc** • **dns** • **esmtp** • **ftp** • **gtp** • **h323** • **http** • **im** • **ip-options** • **ipsec-pass-thru** • **ipv6** • **mgcp** • **netbios** • **radius-accounting** • **rtsp** • **scansafe** • **sip** • **skinny** • **snmp**
policy_map_name	Specifies the name for this policy map up to 40 characters in length. Names that begin with "_internal" or "_default" are reserved and cannot be used. All types of policy maps use the same name space, so you cannot reuse a name already used by another type of policy map.

Defaults

No default behaviors or values.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was introduced.
8.2(1)	Added the **ipv6** keyword to support IPv6 inspection.
9.0(1)	Added the **scansafe** keyword to support Cloud Web Security.

Usage Guidelines

Modular Policy Framework lets you configure special actions for many application inspections. When you enable an inspection engine using the **inspect** command in the Layer 3/4 policy map (the **policy-map** command), you can also optionally enable actions as defined in an inspection policy map created by the **policy-map type inspect** command. For example, enter the **inspect http http_policy_map** command where http_policy_map is the name of the inspection policy map.

An inspection policy map consists of one or more of the following commands entered in policy-map configuration mode. The exact commands available for an inspection policy map depends on the application.

- **match** command—You can define a **match** command directly in the inspection policy map to match application traffic to criteria specific to the application, such as a URL string. Then you enable actions in match configuration mode such as **drop**, **reset**, **log**, and so on. The **match** commands available depend on the application.
- **class** command—This command identifies an inspection class map in the policy map (see the **class-map type inspect** command to create the inspection class map). An inspection class map includes **match** commands that match application traffic with criteria specific to the application, such as a URL string, for which you then enable actions in the policy map. The difference between creating a class map and using a **match** command directly in the inspection policy map is that you can group multiple matches, and you can reuse class maps.
- **parameters** command—Parameters affect the behavior of the inspection engine. The commands available in parameters configuration mode depend on the application.

You can specify multiple **class** or **match** commands in the policy map.

Some **match** commands can specify regular expressions to match text inside a packet. See the **regex** command and the **class-map type regex** command, which groups multiple regular expressions.

The default inspection policy map configuration includes the following commands:

```
policy-map type inspect dns preset_dns_map
 parameters
 message-length maximum client auto
 message-length maximum 512
 dns-guard
 protocol-enforcement
 nat-rewrite
```

If a packet matches multiple different **match** or **class** commands, then the order in which the ASA applies the actions is determined by internal ASA rules, and not by the order they are added to the policy map. The internal rules are determined by the application type and the logical progression of parsing a packet, and are not user-configurable. For example for HTTP traffic, parsing a Request Method field precedes parsing the Header Host Length field; an action for the Request Method field occurs before the action for the Header Host Length field. For example, the following match commands can be entered in any order, but the **match request method get** command is matched first.

```
hostname(config-pmap)# match request header host length gt 100
hostname(config-pmap-c)# reset
hostname(config-pmap-c)# match request method get
hostname(config-pmap-c)# log
```

If an action drops a packet, then no further actions are performed. For example, if the first action is to reset the connection, then it will never match any further **match** commands. If the first action is to log the packet, then a second action, such as resetting the connection, can occur. (You can configure both the **reset** (or **drop-connection**, and so on.) and the **log** action for the same **match** command, in which case the packet is logged before it is reset for a given match.)

If a packet matches multiple **match** or **class** commands that are the same, then they are matched in the order they appear in the policy map. For example, for a packet with the header length of 1001, it will match the first command below, and be logged, and then will match the second command and be reset. If you reverse the order of the two **match** commands, then the packet will be dropped and the connection reset before it can match the second **match** command; it will never be logged.

```
hostname(config-pmap)# match request header length gt 100
hostname(config-pmap-c)# log
hostname(config-pmap-c)# match request header length gt 1000
hostname(config-pmap-c)# reset
```

A class map is determined to be the same type as another class map or **match** command based on the lowest priority **match** command in the class map (the priority is based on the internal rules). If a class map has the same type of lowest priority **match** command as another class map, then the class maps are matched according to the order they are added to the policy map. If the lowest priority command for each class map is different, then the class map with the higher priority **match** command is matched first.

See the following guidelines when modifying an inspection policy-map:

- HTTP inspection policy maps—If you modify an in-use HTTP inspection policy map (**policy-map type inspect http**), you must remove and reapply the **inspect http** *map* action for the changes to take effect. For example, if you modify the “http-map” inspection policy map, you must remove and readd the **inspect http http-map** command from the layer 3/4 policy:

```
hostname(config)# policy-map test
hostname(config-pmap)# class http0
hostname(config-pmap-c)# no inspect http http-map
hostname(config-pmap-c)# inspect http http-map
```

- All inspection policy maps—If you want to exchange an in-use inspection policy map for a different map name, you must remove the **inspect** *protocol map* command, and readd it with the new map. For example:

```
hostname(config)# policy-map test
hostname(config-pmap)# class sip
hostname(config-pmap-c)# no inspect sip sip-map1
hostname(config-pmap-c)# inspect sip sip-map2
```

Examples

The following is an example of an HTTP inspection policy map and the related class maps. This policy map is activated by the Layer 3/4 policy map, which is enabled by the service policy.

```
hostname(config)# regex url_example example\.com
hostname(config)# regex url_example2 example2\.com
hostname(config)# class-map type regex match-any URLs
hostname(config-cmap)# match regex example
hostname(config-cmap)# match regex example2

hostname(config-cmap)# class-map type inspect http match-all http-traffic
hostname(config-cmap)# match req-resp content-type mismatch
hostname(config-cmap)# match request body length gt 1000
hostname(config-cmap)# match not request uri regex class URLs

hostname(config-cmap)# policy-map type inspect http http-map1
hostname(config-pmap)# class http-traffic
hostname(config-pmap-c)# drop-connection log
hostname(config-pmap-c)# match req-resp content-type mismatch
hostname(config-pmap-c)# reset log
hostname(config-pmap-c)# parameters
hostname(config-pmap-p)# protocol-violation action log

hostname(config-pmap-p)# policy-map test
hostname(config-pmap)# class test (a Layer 3/4 class map not shown)
hostname(config-pmap-c)# inspect http http-map1

hostname(config-pmap-c)# service-policy inbound_policy interface outside
```

Related Commands

Command	Description
class	Identifies a class map name in the policy map.
class-map type inspect	Creates an inspection class map to match traffic specific to an application.
parameters	Enters parameter configuration mode for an inspection policy map.
policy-map	Creates a Layer 3/4 policy map.
show running-config policy-map	Display all current policy map configurations.

policy-server-secret

To configure a secret key used to encrypt authentication requests to a SiteMinder SSO server, use the **policy-server-secret** command in webvpn-sso-siteminder configuration mode. To remove a secret key, use the **no** form of this command.

policy-server-secret *secret-key*

no policy-server-secret

Note This command is required for SiteMinder SSO authentication.

Syntax Description

secret-key	The character string used as a secret key to encrypt authentication communications. There is no minimum or maximum number of characters.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Config-webvpn-sso-siteminder configuration	•	—	•	—	—

Command History

Release	Modification
7.1(1)	This command was introduced.

Usage Guidelines

Single sign-on support, available only for WebVPN, lets users access different secure services on different servers without entering a username and password more than once. You first create the SSO server using the **sso-server** command. For SiteMinder SSO servers, the **policy-server-secret** command secures authentication communications between the ASA and the SSO server.

The command argument, *secret-key*, is similar to a password: you create it, save it, and configure it. It is configured on both the ASA using the `policy-server-secret` command and on the SiteMinder Policy Server using the Cisco Java plug-in authentication scheme.

This command applies only to the SiteMinder type of SSO server.

Examples

The following command, entered in config-webvpn-sso-siteminder mode and including a random character string as an argument, creates a secret key for SiteMinder SSO server authentication communications:

```
hostname(config-webvpn)# sso-server my-sso-server type siteminder
hostname(config-webvpn-sso-siteminder)# policy-server-secret @#ET&
hostname(config-webvpn-sso-siteminder)#
```

Related Commands

Command	Description
max-retry-attempts	Configures the number of times the ASA retries a failed SSO authentication attempt.
request-timeout	Specifies the number of seconds before a failed SSO authentication attempt times out.
show webvpn sso-server	Displays the operating statistics for all SSO servers configured on the security device
sso-server	Creates a single sign-on server.
test sso-server	Tests an SSO server with a trial authentication request.
web-agent-url	Specifies the SSO server URL to which the ASA makes SiteMinder SSO authentication requests.

polltime interface

To specify the data interface poll and hold times in an Active/Active failover configuration, use the **polltime interface** command in failover group configuration mode. To restore the default value, use the **no** form of this command.

polltime interface [**msec**] *time* [**holdtime** *time*]

no polltime interface [**msec**] *time* [**holdtime** *time*]

Syntax Description

holdtime *time*	(Optional) Sets the time during which a data interface must receive a hello message from the peer interface, after which the peer interface is declared failed. Valid values are from 5 to 75 seconds.
interface *time*	Specifies data interface polling period. Valid values are from 3 to 15 seconds. If the optional **msec** keyword is used, the valid values are from 500 to 999 milliseconds.
msec	(Optional) Specifies that the given time is in milliseconds.

Defaults

The poll *time* is 5 seconds.

The **holdtime** *time* is 5 times the poll *time*.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Failover group configuration	•	•	—	—	•

Command History

Release	Modification
7.0(1)	This command was introduced.
7.2(1)	The command was changed to include the optional **holdtime** *time* value and the ability to specify the poll time in milliseconds.

Usage Guidelines

Use the **polltime interface** command to change the frequency that hello packets are sent out on interfaces associated with the specified failover group. This command is available for Active/Active failover only. Use the **failover polltime interface** command in Active/Standby failover configurations.

You cannot enter a **holdtime** value that is less than 5 times the poll time. With a faster poll time, the ASA can detect failure and trigger failover faster. However, faster detection can cause unnecessary switchovers when the network is temporarily congested. Interface testing begins when a hello packet is not heard on the interface for over half the hold time.

You can include both **failover polltime unit** and **failover polltime interface** commands in the configuration.

Note When CTIQBE traffic is passed through a ASA in a failover configuration, you should decrease the failover hold time on the ASA to below 30 seconds. The CTIQBE keepalive timeout is 30 seconds and may time out before failover occurs in a failover situation. If CTIQBE times out, Cisco IP SoftPhone connections to Cisco CallManager are dropped, and the IP SoftPhone clients need to reregister with the CallManager.

Examples

The following partial example shows a possible configuration for a failover group. The interface poll time is set to 500 milliseconds and the hold time to 5 seconds for data interfaces in failover group 1.

```
hostname(config)# failover group 1
hostname(config-fover-group)# primary
hostname(config-fover-group)# preempt 100
hostname(config-fover-group)# polltime interface msec 500 holdtime 5
hostname(config-fover-group)# exit
hostname(config)#
```

Related Commands

Command	Description
failover group	Defines a failover group for Active/Active failover.
failover polltime	Specifies the unit failover poll and hold times.
failover polltime interface	Specifies the interface poll and hold times for Active/Standby failover configurations.

pop3s

To enter POP3S configuration mode, use the **pop3s** command in global configuration mode. To remove any commands entered in POP3S command mode, use the **no** version of this command.

POP3 is a client/server protocol in which your Internet server receives and holds e-mail for you. Periodically, you (or your client e-mail receiver) check your mail-box on the server and download any mail. This standard protocol is built into most popular e-mail products. POP3S lets you receive e-mail over an SSL connection.

pop3s

no pop3

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Examples

The following example shows how to enter POP3S configuration mode:

```
hostname(config)# pop3s
hostname(config-pop3s)#
```

Related Commands

Command	Description
clear configure pop3s	Removes the POP3S configuration.
show running-config pop3s	Displays the running configuration for POP3S.

port

To specify the port an e-mail proxy listens to, use the **port** command in the applicable e-mail proxy command mode. To revert to the default value, use the **no** version of this command.

port {*portnum*}

no port

Syntax Description

portnum	The port for the e-mail proxy to use. To avoid conflicts with local TCP services, use port numbers in the range 1024 to 65535.

Defaults

The default ports for e-mail proxies are as follows:

E-mail Proxy	Default Port
IMAP4S	993
POP3S	995
SMTPS	988

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

To avoid conflicts with local TCP services, use port numbers in the range 1024 to 65535.

Examples

The following example shows how to set port 1066 for the IMAP4S e-mail proxy:

```
hostname(config)# imap4s
hostname(config-imap4s)# port 1066
```

port-channel load-balance

For EtherChannels, to specify the load-balancing algorithm, use the **port-channel load-balance** command in interface configuration mode. To set the value to the default, use the **no** form of this command.

port-channel load-balance {**dst-ip** | **dst-ip-port** | **dst-mac** | **dst-port** | **src-dst-ip** | **src-dst-ip-port** | **src-dst-mac** | **src-dst-port** | **src-ip** | **src-ip-port** | **src-mac** | **src-port** | **vlan-dst-ip** | **vlan-dst-ip-port** | **vlan-only** | **vlan-src-dst-ip** | **vlan-src-dst-ip-port** | **vlan-src-ip** | **vlan-src-ip-port**}

no port-channel load-balance

Syntax Description

dst-ip	Balances the packet load on interfaces according to the following characteristics of the packet: • Destination IP address
dst-ip-port	Balances the packet load on interfaces according to the following characteristics of the packet: • Destination IP address • Destination Port
dst-mac	Balances the packet load on interfaces according to the following characteristics of the packet: • Destination MAC address
dst-port	Balances the packet load on interfaces according to the following characteristics of the packet: • Destination port
src-dst-ip	(Default) Balances the packet load on interfaces according to the following characteristics of the packet: • Source IP address • Destination IP address
src-dst-ip-port	Balances the packet load on interfaces according to the following characteristics of the packet: • Source IP address • Destination IP address • Source Port • Destination Port
src-dst-mac	Balances the packet load on interfaces according to the following characteristics of the packet: • Source MAC address • Destination MAC address

src-dst-port	Balances the packet load on interfaces according to the following characteristics of the packet: • Source port • Destination port
src-ip	Balances the packet load on interfaces according to the following characteristics of the packet: • Source IP address
src-ip-port	Balances the packet load on interfaces according to the following characteristics of the packet: • Source IP address • Source port
src-mac	Balances the packet load on interfaces according to the following characteristics of the packet: • Source MAC address
src-port	Balances the packet load on interfaces according to the following characteristics of the packet: • Source port
vlan-dst-ip	Balances the packet load on interfaces according to the following characteristics of the packet: • VLAN • Destination IP address
vlan-dst-ip-port	Balances the packet load on interfaces according to the following characteristics of the packet: • VLAN • Destination IP address • Destination port
vlan-only	Balances the packet load on interfaces according to the following characteristics of the packet: • VLAN
vlan-src-dst-ip	Balances the packet load on interfaces according to the following characteristics of the packet: • VLAN • Source IP address • Destination IP address
vlan-src-dst-ip-port	Balances the packet load on interfaces according to the following characteristics of the packet: • VLAN • Source IP address • Destination IP address • Source port • Destination port

vlan-src-ip	Balances the packet load on interfaces according to the following characteristics of the packet: • VLAN • Source IP address
vlan-src-ip-port	Balances the packet load on interfaces according to the following characteristics of the packet: • VLAN • Source IP address • Source port

Command Default

The default is **src-dst-ip**.

Command Modes

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

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

Command History

Release	Modification
8.4(1)	We introduced this command.

Usage Guidelines

By default, the ASA balances the packet load on interfaces according to the source and destination IP address (**src-dst-ip**) of the packet. If you want to change the properties on which the packet is categorized, use this command. For example, if your traffic is biased heavily towards the same source and destination IP addresses, then the traffic assignment to interfaces in the EtherChannel will be unbalanced. Changing to a different algorithm can result in more evenly distributed traffic.

The ASA distributes packets to the interfaces in the EtherChannel by hashing the load-balancing criteria. The hash result is a 3-bit value (0 to 7).

The eight hash result values are distributed in a round robin fashion between the channel group interfaces, starting with the interface with the lowest ID (slot/port). For example, all packets with a hash result of 0 go to GigabitEthernet 0/0, packets with a hash result of 1 go to GigabitEthernet 0/1, packets with a hash result of 2 go to GigabitEthernet 0/2, and so on.

Because there are eight hash result values regardless of how many active interfaces are in the EtherChannel, packets might not be distributed evenly depending on the number of active interfaces.

Table 39-1 shows the load balancing amounts per interface for each number of active interfaces. The active interfaces in **bold** have even distribution.

Table 39-1 Load Distribution per Interface

# of Active Interfaces	% Distribution Per Interface							
	1	2	3	4	5	6	7	8
1	**100%**	—	—	—	—	—	—	—
2	**50%**	**50%**	—	—	—	—	—	—
3	37.5%	37.5%	25%	—	—	—	—	—
4	**25%**	**25%**	**25%**	**25%**	—	—	—	—
5	25%	25%	25%	12.5%	12.5%	—	—	—
6	25%	25%	12.5%	12.5%	12.5%	12.5%	—	—
7	25%	12.5%	12.5%	12.5%	12.5%	12.5%	12.5%	—
8	**12.5%**	**12.5%**	**12.5%**	**12.5%**	**12.5%**	**12.5%**	**12.5%**	**12.5%**

If an active interface goes down and is not replaced by a standby interface, then traffic is rebalanced between the remaining links. The failure is masked from both Spanning Tree at Layer 2 and the routing table at Layer 3, so the switchover is transparent to other network devices.

Examples

The following example sets the load-balancing algorithm to use the source and destination IP addresses and ports:

```
hostname(config)# interface port-channel 1
hostname(config-if)# port-channel load-balance src-dst-ip-port
```

Related Commands

Command	Description
channel-group	Adds an interface to an EtherChannel.
interface port-channel	Configures an EtherChannel.
lacp max-bundle	Specifies the maximum number of active interfaces allowed in the channel group.
lacp port-priority	Sets the priority for a physical interface in the channel group.
lacp system-priority	Sets the LACP system priority.
port-channel min-bundle	Specifies the minimum number of active interfaces required for the port-channel interface to become active.
show lacp	Displays LACP information such as traffic statistics, system identifier and neighbor details.
show port-channel	Displays EtherChannel information in a detailed and one-line summary form. This command also displays the port and port-channel information.
show port-channel load-balance	Displays port-channel load-balance information along with the hash result and member interface selected for a given set of parameters.

port-channel min-bundle

For EtherChannels, to specify the minimum number of active interfaces required for the port-channel interface to become active, use the **port-channel min-bundle** command in interface configuration mode. To set the value to the default, use the **no** form of this command.

port-channel min-bundle *number*

no port-channel min-bundle

Syntax Description

number	Specifies the minimum number of active interfaces required for the port-channel interface to become active, between 1 and 8.

Command Default

The default is 1.

Command Modes

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

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

Command History

Release	Modification
8.4(1)	We introduced this command.

Usage Guidelines

Enter this command for a port-channel interface. If the active interfaces in the channel group falls below this value, then the port-channel interface goes down, and could trigger a device-level failover.

Examples

The following example sets the minimum number of active interfaces required for the port-channel to become active to two:

```
hostname(config)# interface port-channel 1
hostname(config-if)# port-channel min-bundle 2
```

Related Commands

Command	Description
channel-group	Adds an interface to an EtherChannel.
interface port-channel	Configures an EtherChannel.
lacp max-bundle	Specifies the maximum number of active interfaces allowed in the channel group.

Command	Description
lacp port-priority	Sets the priority for a physical interface in the channel group.
lacp system-priority	Sets the LACP system priority.
port-channel load-balance	Configures the load-balancing algorithm.
show lacp	Displays LACP information such as traffic statistics, system identifier and neighbor details.
show port-channel	Displays EtherChannel information in a detailed and one-line summary form. This command also displays the port and port-channel information.
show port-channel load-balance	Displays port-channel load-balance information along with the hash result and member interface selected for a given set of parameters.

port-channel span-cluster

To sets this EtherChannel as a spanned EtherChannel in an ASA cluster, use the **port-channel span-cluster** command in interface configuration mode. To disable spanning, use the **no** form of this command.

port-channel span-cluster [**vss-load-balance**]

no port-channel span-cluster [**vss-load-balance**]

Syntax Description

vss-load-balance	(Optional) Enables VSS load balancing. If you are connecting the ASA to two switches in a VSS or vPC, then you should enable VSS load balancing. This feature ensures that the physical link connections between the ASAs to the VSS (or vPC) pair are balanced. You must configure the **vss-id** keyword in the **channel-group** command for each member interface before enabling load balancing.

Command Default

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
9.0(1)	We introduced this command.

Usage Guidelines

You must be in spanned EtherChannel mode (**cluster interface-mode spanned**) to use this feature.

This feature lets you group one or more interfaces per unit into an EtherChannel that spans all units in the cluster. The EtherChannel aggregates the traffic across all the available active interfaces in the channel. A spanned EtherChannel can be configured in both routed and transparent firewall modes. In routed mode, the EtherChannel is configured as a routed interface with a single IP address. In transparent mode, the IP address is assigned to the bridge group, not to the interface. The EtherChannel inherently provides load balancing as part of basic operation.

Examples

The following example creates an EtherChannel (port-channel 2) with the tengigabitethernet 0/8 interface as the only member, and then spans the EtherChannel across the cluster. Two subinterfaces are added to port-channel 2.

```
interface tengigabitethernet 0/8
```

```
   channel-group 2 mode active
   no shutdown
interface port-channel 2
   port-channel span-cluster
interface port-channel 2.10
   vlan 10
   nameif inside
   ip address 10.10.10.5 255.255.255.0
   ipv6 address 2001:DB8:1::5/64
   mac-address 000C.F142.4CDE
interface port-channel 2.20
   vlan 20
   nameif outside
   ip address 209.165.201.1 255.255.255.224
   ipv6 address 2001:DB8:2::8/64
   mac-address 000C.F142.5CDE
```

Related Commands

Command	Description
interface	Enters interface configuration mode.
cluster interface-mode	Sets the cluster interface mode, for either Spanned EtherChannels or individual interfaces.

port-forward

To configure the set of applications that users of clientless SSL VPN session can access over forwarded TCP ports, use the **port-forward** command in webvpn configuration mode.

port-forward {*list_name local_port remote_server remote_port description*}

To configure access to multiple applications, use this command with the same *list_name* multiple times, once for each application.

To remove a configured application from a list, use the **no port-forward** *list_name local_port* command (you need not include the *remote_server* and *remote_port* parameters).

no port-forward *listname localport*

To remove an entire configured list, use the **no port-forward** *list_name* command.

no port-forward *list_name*

Syntax Description

description	Provides the application name or short description that displays on the end user Port Forwarding Java applet screen. Maximum 64 characters.
list_name	Groups the set of applications (forwarded TCP ports) users of clientless SSL VPN sessions can access. Maximum 64 characters.
local_port	Specifies the local port that listens for TCP traffic for an application. You can use a local port number only once for a *list_name*. Enter a port number in the range 1-65535. To avoid conflicts with existing services, use a port number greater than 1024.
remote_port	Specifies the port to connect to for this application on the remote server. This is the actual port the application uses. Enter a port number in the range 1-65535 or port name.
remote_server	Provides the DNS name or IP address of the remote server for an application. If you enter the IP address, you may enter it in either IPv4 or IPv6 format. We recommend using a host name so that you do not have to configure the client applications for a specific IP addresses. The dns server-group command **name-server** must resolve the host name to an IP address.

Defaults

There is no default port forwarding list.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.
8.0(2)	The command mode was changed to webvpn.

Usage Guidelines

Port forwarding does not support Microsoft Outlook Exchange (MAPI) proxy. However, you can configure Smart Tunnel support for Microsoft Outlook Exchange 2010.

Examples

The following table shows the values used for example applications.

Application	Local Port	Server DNS Name	Remote Port	Description
IMAP4S e-mail	20143	IMAP4Sserver	143	Get Mail
SMTPS e-mail	20025	SMTPSserver	25	Send Mail
DDTS over SSH	20022	DDTSserver	22	DDTS over SSH
Telnet	20023	Telnetserver	23	Telnet

The following example shows how to create a port forwarding list called *SalesGroupPorts* that provides access to these applications:

```
hostname(config)# webvpn
hostname(config-webvpn)# port-forward SalesGroupPorts 20143 IMAP4Sserver 143 Get Mail
hostname(config-webvpn)# port-forward SalesGroupPorts 20025 SMTPSserver 25 Send Mail
hostname(config-webvpn)# port-forward SalesGroupPorts 20022 DDTSserver 22 DDTS over SSH
hostname(config-webvpn)# port-forward SalesGroupPorts 20023 Telnetserver 23 Telnet
```

Related Commands

Command	Description
port-forward auto-start	Entered in group-policy webvpn or username webvpn mode, this command starts port forwarding automatically and assigns the specified port forwarding list when the user logs onto a clientless SSL VPN session.
port-forward enable	Entered in group-policy webvpn or username webvpn mode, this command starts assigns the specified port forwarding list when the user logs on, but requires the user to start port forwarding manually, using the **Application Access** > **Start Applications** button on the clientless SSL VPN portal page.
port-forward disable	Entered in group-policy webvpn or username webvpn mode, this command turns off port forwarding.

port-forward-name

To configure the display name that identifies TCP port forwarding to end users for a particular user or group policy, use the **port-forward-name** command in webvpn mode, which you enter from group-policy or username mode. To delete the display name, including a null value created by using the **port-forward-name none** command**,** use the no form of the command. The **no** option restores the default name, "Application Access." To prevent a display name, use the **port-forward none** command.

port-forward-name {**value** *name* | **none**}

no port-forward-name

Syntax Description

none	Indicates that there is no display name. Sets a null value, thereby disallowing a display name. Prevents inheriting a value.
value *name*	Describes port forwarding to end users. Maximum of 255 characters.

Defaults

The default name is "Application Access."

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Examples

The following example shows how to set the name, "Remote Access TCP Applications," for the group policy named FirstGroup:

```
hostname(config)# group-policy FirstGroup attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# port-forward-name value Remote Access TCP Applications
```

Related Commands

Command	Description
webvpn	Use in group-policy configuration mode or in username configuration mode. Lets you enter webvpn mode to configure parameters that apply to group policies or usernames.
webvpn	Use in global configuration mode. Lets you configure global settings for WebVPN.

port-object

To add a port object to a service object group of the type TCP, UDP, or TCP-UDP, use the **port-object** command in object-group service configuration mode. To remove port objects, use the **no** form of this command.

port-object {**eq** *port* | **range** *begin_port end_port*}

no port-object {**eq** *port* | **range** *begin_port end_port*}

Syntax Description

range *begin_port end_port*	Specifies a range of ports (inclusive), between 0 and 65535.
eq *port*	Specifies the decimal number (between 0 and 65535) or name of a TCP or UDP port for a service object.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Object-network service configuration	•	•	•	•	—

Command History

Release	Modification
7.0(1)	We introduced this command.

Usage Guidelines

The **port-object** command is used with the **object-group service** *protocol* command to define an object that is either a specific port or a range of ports.

If a name is specified for a TCP or UDP service, it must be one of the supported TCP or/and UDP names, and must be consistent with the protocol type of the object group. For instance, for a protocol types of tcp, udp, and tcp-udp, the names must be a valid TCP service name, a valid UDP service name, or a valid TCP and UDP service name, respectively.

If a number is specified, translation to its corresponding name (if one exists) based on the protocol type will be made when showing the object.

The following service names are supported:

TCP	UDP	TCP and UDP
bgp	biff	discard
chargen	bootpc	domain
cmd	bootps	echo
daytime	dnsix	pim-auto-rp
exec	nameserver	sunrpc
finger	mobile-ip	syslog
ftp	netbios-ns	tacacs
ftp-data	netbios-dgm	talk
gopher	ntp	
ident	rip	
irc	snmp	
h323	snmptrap	
hostname	tftp	
http	time	
klogin	who	
kshell	xdmcp	
login	isakmp	
lpd		
nntp		
pop2		
pop3		
smtp		
sqlnet		
telnet		
uucp		
whois		
www		

Examples

This example shows how to use the **port-object** command in service configuration mode to create a new port (service) object group:

```
hostname(config)# object-group service eng_service tcp
hostname(config-service)# port-object eq smtp
hostname(config-service)# port-object eq telnet
hostname(config)# object-group service eng_service udp
hostname(config-service)# port-object eq snmp
hostname(config)# object-group service eng_service tcp-udp
hostname(config-service)# port-object eq domain
hostname(config-service)# port-object range 2000 2005
hostname(config-service)# quit
```

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.
object-group	Defines object groups to optimize your configuration.
show running-config object-group	Displays the current object groups.

portal-access-rule

This command allows customers to configure a global clientless SSL VPN access policy to permit or deny clientless SSL VPN sessions based on the data present in HTTP header. If denied, an error code is returned to the clients. This denial is performed before user authentication and thus minimizes the use of processing resources.

portal-access-rule none

portal-access-rule *priority* **[{permit | deny [code** *code***]} {any | user-agent match** *string***}**

no portal-access-rule *priority* **[{permit | deny [code** *code***]} {any | user-agent match** *string***}]**

clear configure webvpn portal-access-rule

Syntax Description

none	Removes all portal access rules. Clientless SSL VPN sessions will not restricted based on HTTP header.
priority	Priority of rule. Range: 1-65535.
permit	Permit access based upon HTTP header.
deny	Deny access based upon HTTP header.
code	Permit or deny access based on a returned HTTP status code. Default: 403.
code	The HTTP status code number based on which you want to permit or deny access. Range: 200-599.
any	Match any HTTP header string.
user-agent match	Enable comparison of strings in HTTP headers.
string	Specify the string to match in the HTTP header. Surround the string you are searching for with wildcards (*) for a match that contains your string or do not use wildcards to specify an exact match of your string. **Note** We recommend using wildcards in your search string. Without them, the rule may not match any strings or many fewer than you expect. If the string you are searching for has a space in it, the string must be enclosed in quotations; for example, "*a string*". When using both quotations and wild cards, your search string would look like this: "**a string**".
no portal-access-rule	Use to delete a single portal-access-rule.
clear configure webvpn portal-access-rule	Equivalent to portal-access-rule none command.

Defaults

portal-access-rule none

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
webvpn configuration mode	•	—	•	—	—

Command History

Release	Modification
8.2(5)	This command was introduced simultaneously in ASA 8.2.5 and 8.4(2)
8.4(2)	This command was introduced simultaneously in ASA 8.2.5 and 8.4(2)

Usage Guidelines

This check is performed prior to user authentication.

Examples

The following example creates three portal access rules:

- Portal access rule 1 denies attempted clientless SSL VPN connections when the ASA returns code 403 and Thunderbird is in the HTTP header.
- Portal access rule 10 permits attempted clientless SSL VPN connections when MSIE 8.0 (Microsoft Internet Explorer 8.0) is in the HTTP header.
- Portal access rule 65535 permits all other attempted clientless SSL VPN connections.

```
hostname(config)# webvpn
hostname(config-webvpn)# portal-access-rule 1 deny code 403 user-agent match *Thunderbird*
hostname(config-webvpn)# portal-access-rule 10 permit user-agent match "*MSIE 8.0*"
hostname(config-webvpn)# portal-access-rule 65535 permit any
```

Related Commands

Command	Description
show run webvpn	Displays webvpn configuration including all portal-access-rules.
show vpn-sessiondb detail webvpn	Display information about VPN sessions. The command includes options for displaying information in full or in detail, lets you specify type of sessions to display, and provides options to filter and sort the information.
debug webvpn request *n*	Enables logging of debug messages at a particular level of debugging. Default: 1. Range: 1-255.

post-max-size

To specify the maximum size allowed for an object to post, use the **post-max-size** command in group-policy webvpn configuration mode. To remove this object from the configuration, use the **no** version of this command.

post-max-size <*size*>

no post-max-size

Syntax Description

size	Specifies the maximum size allowed for a posted object. The range is 0 through 2147483647.

Defaults

The default size is 2147483647.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Group-policy webvpn configuration mode	•	—	•	—	—

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

Setting the size to 0 effectively disallows object posting.

Examples

The following example sets the maximum size for a posted object to 1500 bytes:

```
hostname(config)# group-policy test attributes
hostname(config-group-policy)# webvpn
hostname(config-group-webvpn)# post-max-size 1500
```

Related Commands

Command	Description
download-max-size	Specifies the maximum size of an object to download.
upload-max-size	Specifies the maximum size of an object to upload.

Command	Description
webvpn	Use in group-policy configuration mode or in username configuration mode. Lets you enter webvpn mode to configure parameters that apply to group policies or usernames.
webvpn	Use in global configuration mode. Lets you configure global settings for WebVPN.

pppoe client route distance

To configure an administrative distance for routes learned through PPPoE, use the **pppoe client route distance** command in interface configuration mode. To restore the default setting, use the **no** form of this command.

pppoe client route distance *distance*

no pppoe client route distance *distance*

Syntax Description

distance	The administrative distance to apply to routes learned through PPPoE. Valid values are from 1 to 255.

Defaults

Routes learned through PPPoE are given an administrative distance of 1 by default.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

The **pppoe client route distance** command is checked only when a route is learned from PPPoE. If the **pppoe client route distance** command is entered after a route is learned from PPPoE, the administrative distance specified does not affect the existing learned route. Only routes learned after the command was entered have the specified administrative distance.

You must specify the **setroute** option on the **ip address pppoe** command to obtain routes through PPPoE.

If PPPoE is configured on multiple interfaces, you must use the **pppoe client route distance** command on each of the interfaces to indicate the priority of the installed routes. Enablgin PPPoE clients on multiple interfaces is only supported with object tracking.

You cannot configure failover if you obtain IP addresses using PPPoE.

Examples

The following example obtains the default route through PPPoE on GigabitEhternet0/2. The route is tracked by tracking entry object 1. The SLA operation monitors the availability of the 10.1.1.1 gateway off of the outside interface. If the SLA operation fails, then the secondary route obtained on GigabitEthernet0/3 through PPPoE is used.

```
hostname(config)# sla monitor 123
hostname(config-sla-monitor)# type echo protocol ipIcmpEcho 10.1.1.1 interface outside
hostname(config-sla-monitor-echo)# timeout 1000
hostname(config-sla-monitor-echo)# frequency 3
hostname(config)# sla monitor schedule 123 life forever start-time now
hostname(config)# track 1 rtr 123 reachability
hostname(config)# interface GigabitEthernet0/2
hostname(config-if)# pppoe client route track 1
hostname(config-if)# ip address pppoe setroute
hostname(config)# interface GigabitEthernet0/3
hostname(config-if)# pppoe client secondary track 1
hostname(config-if)# pppoe client route distance 254
hostname(config-if)# ip address pppoe setroute
```

Related Commands

Command	Description
ip address pppoe	Configures the specified interface with an IP address obtained through PPPoE.
ppoe client secondary	Configures tracking for secondary PPPoE client interface.
pppoe client route track	Associates routes learned through PPPoE with a tracking entry object.
sla monitor	Defines an SLA monitoring operation.
track rtr	Creates a tracking entry to poll the SLA.

pppoe client route track

To configure the PPPoE client to associate added routes with a specified tracked object number, use the **pppoe client route track** command in interface configuration mode. To remove the PPPoE route tracking, use the **no** form of this command.

pppoe client route track *number*

no pppoe client route track

Syntax Description

number	The tracking entry object ID. Valid values are from 1 to 500.

Defaults

No default behaviors or values.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

The **pppoe client route track** command is checked only when a route is learned from PPPoE. If the **pppoe client route track** command is entered after a route is learned from PPPoE, the existing learned routes are not associated with a tracking object. Only routes learned after the command was entered are associated with the specified tracking object.

You must specify the **setroute** option on the **ip address pppoe** command to obtain routes through PPPoE.

If PPPoE is configured on multiple interfaces, you must use the **pppoe client route distance** command on each of the interfaces to indicate the priority of the installed routes. Enabling PPPoE clients on multiple interfaces is only supported with object tracking.

You cannot configure failover if you obtain IP addresses using PPPoE.

Examples

The following example obtains the default route through PPPoE on GigabitEhternet0/2. The route is tracked by tracking entry object 1. The SLA operation monitors the availability of the 10.1.1.1 gateway off of the outside interface. If the SLA operation fails, then the secondary route obtained on GigabitEthernet0/3 through PPPoE is used.

```
hostname(config)# sla monitor 123
hostname(config-sla-monitor)# type echo protocol ipIcmpEcho 10.1.1.1 interface outside
hostname(config-sla-monitor-echo)# timeout 1000
hostname(config-sla-monitor-echo)# frequency 3
hostname(config)# sla monitor schedule 123 life forever start-time now
hostname(config)# track 1 rtr 123 reachability
hostname(config)# interface GigabitEthernet0/2
hostname(config-if)# pppoe client route track 1
hostname(config-if)# ip address pppoe setroute
hostname(config)# interface GigabitEthernet0/3
hostname(config-if)# pppoe client secondary track 1
hostname(config-if)# pppoe client route distance 254
hostname(config-if)# ip address pppoe setroute
```

Related Commands

Command	Description
ip address pppoe	Configures the specified interface with an IP address obtained through PPPoE.
ppoe client secondary	Configures tracking for secondary PPPoE client interface.
pppoe client route distance	Assigns an administrative distance to routes learned through PPPoE.
sla monitor	Defines an SLA monitoring operation.
track rtr	Creates a tracking entry to poll the SLA.

pppoe client secondary

To configure the PPPoE client to register as a client of a tracked object and to be brought up or down based on the tracking state, use the **pppoe client secondary** command in interface configuration mode. To remove the client registration, use the **no** form of this command.

pppoe client secondary track *number*

no pppoe client secondary track

Syntax Description

number	The tracking entry object ID. Valid values are from 1 to 500.

Defaults

No default behaviors or values.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

The **pppoe client secondary** command is checked only when PPPoE session starts. If the **pppoe client route track** command is entered after a route is learned from PPPoE, the existing learned routes are not associated with a tracking object. Only routes learned after the command was entered are associated with the specified tracking object.

You must specify the **setroute** option on the **ip address pppoe** command to obtain routes through PPPoE.

If PPPoE is configured on multiple interfaces, you must use the **pppoe client route distance** command on each of the interfaces to indicate the priority of the installed routes. Enabling PPPoE clients on multiple interfaces is only supported with object tracking.

You cannot configure failover if you obtain IP addresses using PPPoE.

Examples

The following example obtains the default route through PPPoE on GigabitEhternet0/2. The route is tracked by tracking entry object 1. The SLA operation monitors the availability of the 10.1.1.1 gateway off of the outside interface. If the SLA operation fails, then the secondary route obtained on GigabitEthernet0/3 through PPPoE is used.

```
hostname(config)# sla monitor 123
hostname(config-sla-monitor)# type echo protocol ipIcmpEcho 10.1.1.1 interface outside
hostname(config-sla-monitor-echo)# timeout 1000
hostname(config-sla-monitor-echo)# frequency 3
hostname(config)# sla monitor schedule 123 life forever start-time now
hostname(config)# track 1 rtr 123 reachability
hostname(config)# interface GigabitEthernet0/2
hostname(config-if)# pppoe client route track 1
hostname(config-if)# ip address pppoe setroute
hostname(config)# interface GigabitEthernet0/3
hostname(config-if)# pppoe client secondary track 1
hostname(config-if)# pppoe client route distance 254
hostname(config-if)# ip address pppoe setroute
```

Related Commands

Command	Description
ip address pppoe	Configures the specified interface with an IP address obtained through PPPoE.
ppoe client secondary	Configures tracking for secondary PPPoE client interface.
pppoe client route distance	Assigns an administrative distance to routes learned through PPPoE.
pppoe client route track	Associates routes learned through PPPoE with a tracking entry object.
sla monitor	Defines an SLA monitoring operation.