telnet

To add Telnet access to the FWSM console and set the idle timeout, use the **telnet** command. To remove Telnet access from a previously set IP address, use the **no** form of this command.

[**no**] **telnet** *local_ip mask interface_name*

telnet timeout *number*

Syntax Description

local_ip	IP address of a host or network that can access the FWSM Telnet console.
mask	Netmask for the local IP.
interface_name	Network interface name.
timeout *number*	Specifies the number of minutes that a Telnet session can be idle before being closed by the FWSM; valid values are from 1 to 1440 minutes.

Defaults

number is **5** minutes.

Command Modes

Security Context Mode: single context mode and multiple context mode

Access Location: context command line

Command Mode: configuration mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.

Usage Guidelines

The **telnet** command allows you to specify which hosts can access the FWSM console with Telnet. You can enable Telnet to the FWSM on all interfaces. However, the FWSM enforces that all Telnet traffic to the lowest security interface is protected by IPSec. To enable a Telnet session to the lowest interface, configure IPSec on the lowest security interface to include IP traffic that is generated by the FWSM and enable Telnet on the interface.

Up to 16 hosts or networks are allowed access to the FWSM console with Telnet, and up to 5 hosts are allowed access to the FWSM simultaneously. Use the **no telnet** or **clear telnet** command to remove Telnet access from a previously set IP address. Use the **telnet timeout** command to set the maximum time that a console Telnet session can be idle before being logged off by the FWSM. The **clear telnet** command does not affect the **telnet timeout** command duration. You cannot use the **no telnet** command with the **telnet timeout** command.

To limit access to a single IP address, use 255 in each octet; for example, 255.255.255.255. You must specify *netmask* as 255.255.255.255 regardless of the class of *local_ip*. Do not use the subnetwork mask of the internal network. The *netmask* is only a bit mask for the IP address in *ip_address*.

If IPSec is operating, you can specify an unsecure interface name, which is typically, the outside interface. At a minimum, you might configure the **crypto map** command to specify an interface name with the **telnet** command.

You must specify an interface namee. The FWSM automatically verifies the IP address against the IP addresses that are specified by the **ip address** commands to ensure that the address that you specify is on an internal interface. If an interface name is specified, the FWSM checks only the host against the interface that you specify.

Use the **passwd** command to set a password for Telnet access to the console. The default is **cisco**. Use the **who** command to view which IP addresses are currently accessing the FWSM console. Use the **kill** command to terminate an active Telnet console session.

If you use the **aaa** command with the **console** keyword, Telnet console access must be authenticated with an authentication server.

Note If you have configured the **aaa** command to require authentication for FWSM Telnet console access and the console login request times out, you can gain access to the FWSM from the serial console by entering the **fwsm** username and the password that was set with the **enable password** command.

Examples

This example shows how to permit hosts 192.168.1.3 and 192.168.1.4 to access the FWSM console through Telnet. In addition, all the hosts on the 192.168.2.0 network are given access.

```
fwsm/context_name(config)# telnet 192.168.1.3 255.255.255.255 inside
fwsm/context_name(config)# telnet 192.168.1.4 255.255.255.255 inside
fwsm/context_name(config)# telnet 192.168.2.0 255.255.255.0 inside
fwsm/context_name(config)# show telnet
192.168.1.3 255.255.255.255 inside
192.168.1.4 255.255.255.255 inside
192.168.2.0 255.255.255.0 inside
```

This example shows how to remove individual entries with the **no telnet** command or all **telnet** commands with the **clear telnet** command:

```
fwsm/context_name(config)# no telnet 192.168.1.3 255.255.255.255 inside
fwsm/context_name(config)# show telnet
192.168.1.4 255.255.255.255 inside
192.168.2.0 255.255.255.0 inside
fwsm/context_name(config)# clear telnet
fwsm/context_name(config)# show telnet
```

This example shows how to change the maximum session idle duration:

```
fwsm/context_name(config)# telnet timeout 10
fwsm/context_name(config)# show telnet timeout
telnet timeout 10 minutes
```

This example shows a Telnet console login session (the password does not display when entered):

```
fwsm# passwd: cisco

Welcome to the FWSM
…
Type help or '?' for a list of available commands.
fwsm>
```

Related Commands

aaa accounting
kill
password/passwd
ssh
who

terminal

To set the terminal line parameter settings, use the **terminal** command.

terminal width *columns*

terminal monitor

terminal [**no**] **monitor**

Syntax Description

width *columns*	Sets the width for displaying information during console sessions; permissible values are 0, which means 511 columns, or a value in the range of 40 to 511.
monitor	Specifies that syslog messages are displayed on this terminal.
no	(Optional) Disables syslog message that displays to this terminal.

Defaults

Width is 80 columns. No monitoring is the default.

Command Modes

Security Context Mode: single context mode and multiple context mode

Access Location: system and context command line

Command Mode: privileged mode and configuration mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.

Usage Guidelines

The **terminal monitor** command allows you to enable or disable the display of syslog messages in the current session for either Telnet or serial access to the FWSM console. Use the **logging monitor** command to enable or disable various levels of syslog messages to the console; use the **terminal no monitor** command to disable the messages on a per-session basis. Use the **terminal monitor** command to restart the syslog messages for the current session.

The **terminal width** command allows you to set the width for displaying command output. The terminal width is controlled by the **terminal width** *nn* command, where *nn* is the width in characters. If you enter a line break, you cannot use the backspace key to return to the previous line.

Examples

This example shows how to enable logging and then disable logging only in the current session:

```
fwsm/context_name(config)# terminal monitor
fwsm/context_name(config)# terminal no monitor
```

Related Commands

clear terminal
logging
show terminal

tftp-server

To specify the default TFTP server address and directory, use the **tftp-server** command. To disable access to the server, use the **no** form of this command.

[**no**] **tftp-server** *interface_name ip_address directory*

Syntax Description

interface_name	Interface name designated by the **nameif** command.
ip	IP address of the TFTP server.
directory	Directory of the configuration file.

Defaults

This command has no default settings.

Command Modes

Security Context Mode: single context mode and multiple context mode

Access Location: context command line

Command Mode: configuration mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.

Usage Guidelines

The **tftp-server** command allows you to specify the IP address of the server that you use to propagate the FWSM configuration files to the FWSM. Use the **tftp-server** command with the **configure net** command to read from the configuration or with the **write net** command to store the configuration in the file that you specify. The FWSM supports only one TFTP server.

The *path* name that you specify in the **tftp-server** is appended to the end of the IP address that you specify in the **configure net** and **write net** commands. Because the *path* name is appended to the IP address that you specify with the **tftp-server** command, you will not need to specify the file and pathname with the **configure net** and **write net** commands. If you specify the full *path* and filename in the **tftp-server** command, the IP address in the **configure net** and **write net** commands can be represented with a colon (**:**).

This is the interface by which the TFTP server IP is accessible.

The *interface_name* argument specifies the interface name designated by the **nameif** command. If you specify the outside interface, a warning message informs you that the outside interface is unsecure.

Caution Specifying an unsecure interface may put your data at a security risk.

The format for *path* differs by the type of operating system of the server that you are using to contain the configuration files for the FWSM. The contents of a path are passed directly to the server without interpretation or checking. The configuration file must exist on the TFTP server. Many TFTP servers require the configuration file to be world-writable to write to it and world-readable to read from it.

Note If the TFTP server to which the FWSM is trying to connect is not running TFTP, the FWSM suspends operation and does not time out. Press the **ESC** key on the FWSM console to abort the TFTP session and return to the command-line prompt.

Examples

This example shows how to specify a TFTP server and then read the configuration from /FWSM/config/test_config directory:

```
fwsm/context_name(config)# tftp-server inside 10.1.1.42 /fwsm/config/test_config
fwsm/context_name(config)# configure net :
```

Related Commands

clear tftp-server
show tftp-server

timeout

To set the maximum idle time duration, use the **timeout** command.

timeout [**xlate** | **conn** | **udp** | **icmp** | **rpc** | **h323** | **h225** | **mgcp** | **sip** | **sip_media** | **uauth** *hh*:*mm*:*ss*]

Syntax Description

xlate	(Optional) Specifies the idle time until a translation slot is freed; the minimum value is 1 minute.
conn	(Optional) Specifies the idle time after which a connection closes; the minimum duration is 5 minutes.
udp	(Optional) Specifies the idle time until a UDP slot is freed; the minimum duration is 1 minute.
icmp	(Optional) Specifies the idle time after which general ICMP states are closed.
rpc	(Optional) Specifies the idle time until an RPC slot is freed; the minimum duration is 1 minute.
h323	(Optional) Specifies the idle time after which an H323 control connection is closed.
h255	(Optional) Specifies the idle time after which H.225 signaling closes.
mgcp	(Optional) Sets the duration for the Media Gateway Control Protocol (MGCP) inactivity timer.
sip	(Optional) Modifies the SIP timer.
sip_media	(Optional) Modifies the media timer, which is used for SIP RTP/RTCP with SIP UDP media packets, instead of the UDP inactivity timeout.
uauth	(Optional) Sets the duration before the authentication and authorization cache times out and the user has to reauthenticate the next connection.
hh:*mm*:*ss*	Timeout.

Defaults

The defaults are as follows:

- **xlate** *hh*:*mm*:*ss* is 3 hours (**03:00:00**).
- **conn** *hh*:*mm*:*ss* is 1 hour (**01:00:00**).
- **half-closed** *hh*:*mm*:*ss* is 10 minutes (**00:10:00**).
- **udp** *hh*:*mm*:*ss* is 2 minutes (**00:02:00**).
- **rpc** *hh*:*mm*:*ss* is 10 minutes (**00:10:00**).
- **h255** *hh*:*mm*:*ss* is 1 hour (**01:00:00**).
- **h323** *hh*:*mm*:*ss* is 5 minutes (**00:05:00**).
- **mgcp** *hh:mm:ss* is 5 minutes (**00:05:00**).

- **sip** *hh:mm:* is 30 minutes (**00:30:00**).
- **sip_media** *hh:mm:ss* is 2 minutes (**00:02:00**).
- **uauth** timer is **absolute**.

Command Modes

Security Context Mode: single context mode and multiple context mode

Access Location: context command line

Command Mode: configuration mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.

Usage Guidelines

The **timeout** command allows you to set the idle time for connection, translation UDP, and RPC slots. If the slot has not been used for the idle time specified, the resource is returned to the free pool. TCP connection slots are freed approximately 60 seconds after a normal connection close sequence.

This command is used with the **show** and **clear uauth** commands.

Note Do not use the **timeout uauth 0:0:0** command if passive FTP is used for the connection or if the **virtual** command is used for web authentication.

The connection timer takes precedence over the translation timer; the translation timer works only after all connections have timed out.

When setting the **conn** *hh***:***mm***:***ss*, use **0:0:0** to never time out a connection.

When setting the **half-closed** *hh***:***mm***:***ss*, use **0:0:0** to never time out a half-closed connection.

When setting the **h255** *hh***:***mm***:***ss*, **h225 00:00:00** means to never tear down H.225 signaling. A timeout value of **h225 00:00:01** disables the timer and closes the TCP connection immediately after all calls are cleared.

The **uauth** *hh***:***mm***:***ss* duration must be shorter than the **xlate** keyword. Set to **0** to disable caching. Do not set to zero if passive FTP is used on the connections.

To disable the **absolute** keyword, set the uauth timer to 0 (zero).

Caution Be careful when using the remote procedure call (RPC) and Network File System (NFS) protocols because they are unsecure protocols.

Examples

This example shows how to configure the maximum idle time durations:

```
fwsm/context_name(config)# timeout uauth 0:5:00 absolute uauth 0:4:00 inactivity
fwsm/context_name(config)# show timeout
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 rpc 0:10:00 h323 0:05:00
sip 0:30:00 sip_media 0:02:00
timeout uauth 0:05:00 absolute uauth 0:04:00 inactivity
```

Related Commands

arp

clear timeout

show timeout

show xlate

show uauth

timers

To configure the OSPF process delay timers, use the **timers** command. To return to the default settings, use the **no** form of this command.

timers {**spf** *spf_delay spf_holdtime* | **lsa-group-pacing** *seconds*}

no timers {**spf** | **lsa-group-pacing**}

Syntax Description

spf *spf_delay*	Specifies the delay time between when OSPF receives a topology change and when it starts a shortest path first (SPF) calculation in seconds, from 0 to 65535.
spf_holdtime	The hold time between two consecutive SPF calculations in seconds; valid values are from 1 to 65534.
lsa-group-pacing *seconds*	Specifies the delay time between when OSPF receives a topology change and when it starts a calculation, and the hold time between two consecutive SPF calculations; valid values are from 10 to 1800 seconds.

Defaults

The defaults are as follows:

- *spf_delay* is **5** second.
- *spf_holdtime* is **10** seconds.
- *seconds* is **240** seconds.

Command Modes

Security Context Mode: single context mode

Access Location: context command line

Command Mode: configuration mode

Firewall Mode: Routed

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.

Usage Guidelines

To configure the delay time between when the OSPF protocol receives a topology change and when it starts a calculation, and the hold time between two consecutive SPF calculations, use the **timers spf** *spf_delay spf_holdtime* subcommand. To return to the default timer values, use the **no timers spf** subcommand.

To change the interval at which the OSPF link-state advertisements (LSAs) are collected into a group and refreshed, checksummed, or aged, use the **timers lsa-group-pacing** *seconds* subcommand. To return to the default timer values, use the **no timers lsa-group-pacing** subcommand.

Examples

This example shows how to configure OSPF process delay timers:

```
fwsm/context_name(config)# timers lsa-group-pacing 40
```

Related Commands

router ospf
show ip ospf
show timers

upgrade-mp

To upgrade the maintenance partition, use the **upgrade-mp** command.

upgrade-mp tftp://*location*/*pathname*

upgrade-mp http[**s**]:**//**[*user*:*password*@]*location* [:*port*]/*pathname*

upgrade-mp tftp[:[[**//***location*][/*pathname*]]]

Syntax Description

location	(Optional) Location of the upgrade software image.
pathname	(Optional) Pathname to the upgrade software image.
user	(Optional) Username.
password	(Optional) User's password.
port	HTTP port.

Defaults

This command has no default settings.

Command Modes

Security Context Mode: single context mode

Access Location: system command line

Command Mode: privileged mode and configuration mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.

Examples

This example show how to upgrade a maintenance partition using TFTP:

```
fwsm(config)# upgrade-mp tftp://10.192.1.1/c6svc-mp.2-1-1.bin.gz
```

uptime

To display the FWSM version and the time that the module has been running, use the **uptime** command.

uptime

Syntax Description

This command has no arguments or keywords.

Defaults

This command has no default settings.

Command Modes

Security Context Mode: single context mode

Access Location: system and context command line

Command Mode: privileged mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.
2.3(1)	This command is not supported in this release. The **show uptime** command is supported.

Examples

This example shows how to display FWSM version and runtime information:

```
FWSM# show uptime
FWSM Firewall Version 2.3(1)11
FWSM Device Manager Version 4.0(1)
Compiled on <?xml:namespace prefix = st1 ns = "urn:schemas-microsoft-com:office:smarttags"
/>Fri 04-Feb-05 00:12 by dalecki
FWSM up 6 hours 21 mins
Hardware:   WS-SVC-FWM-1, 1024 MB RAM, CPU Pentium III 1000 MHz
Flash  V1.01   SMART ATA FLASH DISK @ 0xc321, 20MB
0: gb-ethernet0: irq 5
1: gb-ethernet1: irq 7
2: ethernet0: irq 11
Licensed Features:
Failover:           Enabled
VPN-DES:            Enabled
VPN-3DES:           Enabled
Maximum Interfaces: 256
Cut-through Proxy:  Enabled
Guards:             Enabled
URL-filtering:      Enabled
Throughput:         Unlimited
ISAKMP peers:       Unlimited
```

Related Commands

show uptime

url-block

To enable long URL support and HTTP response buffering for URL filtering services, use the **url-block** command. To disable long URL support and HTTP response buffering for URL filtering services, use the **no** form of this command.

[**no**] **url-block** {**block** *block_buffer_limit*} | {**url-mempool** *memory_pool_size*} | {**url-size** *long_url_size*}

Syntax Description

block *block_buffer_limit*	Specifies the maximum number of blocks that are allowed in the HTTP response buffer, valid values are from 1 to 128.
url-mempool *memory_pool_size*	Specifies the size of the URL buffer memory pool in Kilobytes (KB); valid values are from 2 to 10240 KB.
url-size *long_url_size*	Specifies the maximum allowed URL size in KB; valid values are from 2 to 4 KB.

Defaults

block_buffer_limit is **1** KB.

Command Modes

Security Context Mode: single context mode and multiple context mode

Access Location: context command line

Command Mode: configuration mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.

Usage Guidelines

The **url-block url-size** *long_url_size* command is supported on Websense servers only.

The **url-block** command requires that a valid Websense URL filtering configuration is running on the FWSM. After a valid Websense URL filter is in place, you can use this command to pass the URLs that are longer than 1159 bytes, up to a maximum of 4096 bytes, to the Websense server. The **url-block** command stores the URLs that are longer than 1159 bytes in a buffer and then passes the URL to the Websense server (through a TCP packet stream) so that the Websense server can grant or deny access to that URL.

Examples

This example shows how to enable long URL support and HTTP response buffering for URL filtering services:

```
FWSM(config)# url-block block 128
FWSM(config)# url-block url-mempool 1500
FWSM(config)# url-block url-size 4
```

Related Commands

clear url-block
show url-block

url-cache

To cache web server responses that are pending a permit or deny response from an N2H2 server or Websense server, use the **url-cache** command. To disable caching, use the **no** form of this command.

[**no**] **url-cache** {**dst** | **src_dst**} **size** *kbytes*

Syntax Description

dst	Specifies cache entries that are based on the URL destination address.
src_dst	Specifies cache entries that are based on the both the source address initiating the URL request as well as the URL destination address.
size *kbytes*	Specifies a value for the cache size within the range 1 to 128 KB.

Defaults

This command has no default settings.

Command Modes

Security Context Mode: single context mode and multiple context mode

Access Location: context command line

Command Mode: configuration mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.

Usage Guidelines

The **url-cache** command provides a configuration option to buffer the response from a web server if its response is faster than that from the N2H2 or Websense filtering service server. This command prevents the web server's response from being loaded twice.

When you access a site, the filtering server can allow the FWSM to cache the server address for a certain amount of time, as long as every site hosted at the address is in a category that is permitted at all times. When you access the server again, or if another user accesses the server, the FWSM does not need to consult the filtering server.

Use the **url-cache** command to enable URL caching, set the size of the cache, and display the cache statistics.

Caching stores URL access privileges in memory on the FWSM. When a host requests a connection, the FWSM first looks in the URL cache for matching access privileges instead of forwarding the request to the N2H2 or Websense server. You can disable caching with the **no url-cache** command.

Using the URL cache does not update the Websense accounting logs for Websense protocol version 1. If you are using Websense protocol version 1, you should allow Websense to accumulate logs so that you can view the Websense accounting information. After you get a usage profile that meets your security needs, you can enable **url-cache** to increase throughput. Accounting logs are updated for Websense protocol version 4 and for N2H2 URL filtering while using the **url-cache** command.

Note If you change the settings on the N2H2 or Websense server, disable the cache with the **no url-cache** command and then reenable the cache with the **url-cache** command.

Select **dst** mode if all users share the same URL filtering policy on the N2H2 or Websense server.

Select **src_dst** mode if the users do not share the same URL filtering policy on the N2H2 or Websense server.

Examples

This example shows how to cache all outbound HTTP connections that are based on the source and destination addresses:

```
fwsm/context_name(config)# url-cache src_dst 128
```

Related Commands

clear url-cache
show url-cache stat

url-server

To designate a server running either N2H2 server or Websense servers for use with the **filter** command, use the **url-server** command. To remove the server, use the **no** form of this command.

N2H2 Commands

url-server {*interface_name*} **vendor n2h2 host** *local_ip* [**port** *number*] [**timeout** *seconds*] [**protocol** {**TCP** | **UDP**}]

no url-server {*interface_name*} **vendor n2h2 host** *local_ip* [**port** *number*] [**timeout** *seconds*] [**protocol** {**TCP** | **UDP**}]

Websense Commands

url-server {*interface_name*} **vendor websense host** *local_ip* [**timeout** *seconds*] [**protocol** {**TCP** | **UDP**} *version*]

no url-server {*interface_name*} **vendor websense host** *local_ip* [**timeout** *seconds*] [**protocol** {**TCP** | **UDP**} *version*]

Syntax Description

N2H2

interface_name	Network interface where the authentication server resides. If not specified, the default is inside.
vendor n2h2	Indicates that the URL filtering service vendor is N2H2.
host *local_ip*	IP address of the local server that runs the URL filtering application.
port *number*	(Optional) Specifies the N2H2 filtering application server port number.
timeout *seconds*	(Optional) Specifies the maximum idle time permitted before the FWSM switches to the next server that you specified.
protocol TCP	(Optional) Specifies the TCP protocol.
protocol UDP	(Optional) Specifies the UDP protocol.

Websense

interface_name	Network interface where the authentication server resides.
vendor websense	Indicates that the URL filtering service vendor is Websense.
host *local_ip*	IP address of the local server that runs the URL filtering application.
timeout *seconds*	(Optional) Specifies the maximum idle time that is permitted before the FWSM switches to the next server specified.
protocol TCP	(Optional) Specifies the TCP protocol.
protocol UDP	(Optional) Specifies the UDP protocol.
version	(Optional) Protocol version 1 or 4.

Defaults

The default settings for the N2H2 filtering application are as follows:

- If not specified, the *interface_name* is inside.
- **port** *number* is **4005**.
- **timeout** *seconds* is **5** seconds.
- **protocol** is TCP.

The default settings for Websense are as follows:

- If not specified, the *interface_name* is inside.
- **timeout** *seconds* is **5** seconds.
- **protocol** is TCP.
- *version* is TCP version 1.

Command Modes

Security Context Mode: single context mode and multiple context mode

Access Location: context command line

Command Mode: configuration mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.

Usage Guidelines

The **url-server** command allows you to designate the server running the N2H2 server or Websense server URL. The FWSM supports four URL servers per context in multiple mode; 16 URL servers can be assigned in single mode. However, you can use only one application at a time, either the N2H2 server or the Websense server. Changing the configuration on the FWSM does not update the configuration on the application server. Changing the configuration must be done separately and according to the individual vendor's instructions.

Note For information about filtering by the N2H2 server, refer to this URL:
http://www.n2h2.com.
For information on Websense filtering services, refer to this URL:
http://www.websense.com/

You must configure the **url-server** command before using the **filter** command for HTTPS and FTP. If you remove all URL servers from the server list, then all **filter** commands that are related to URL filtering are also removed.

You cannot run both URL filtering services simultaneously.

For Websense, you can configure TCP using version 1 or version 4. You can configure UDP using version 4 only.

Examples

This example shows how to filter all outbound HTTP connections except those from the 10.0.2.54 host when using N2H2:

```
fwsm/context_name(config)# url-server (perimeter) vendor n2h2 host 10.0.1.1
fwsm/context_name(config)# filter url http 0 0 0 0
fwsm/context_name(config)# filter url except 10.0.2.54 255.255.255.255 0 0
```

This example shows how to filter all outbound HTTP connections except those from the 10.0.2.54 host when using Websense:

```
fwsm/context_name(config)# url-server (perimeter) vendor websense host 10.0.1.1
fwsm/context_name(config)# filter url http 0 0 0 0
fwsm/context_name(config)# filter url except 10.0.2.54 255.255.255.255 0 0
```

Related Commands

aaa authorization

clear url-server

filter ftp

show url-server

username

To set the username for the specified privilege level, use the **username** command. To remove the username and privilege level, use the **no** form of this command.

username *username* {[{**nopassword** | **password** *password*} [**encrypted**]] [**privilege** *level*]}

no username *username*

Syntax Description

username	Name of a specific user in the local FWSM authentication database.
nopassword	(Optional) Specifies that password access is not required.
password *password*	(Optional) Specifies that password access is required and specifies a password.
encrypted	(Optional) Specifies encryption.
privilege *level*	(Optional) Specifies the privilege level for the user.

Defaults

This command has no default settings.

Command Modes

Security Context Mode: single context mode and multiple context mode

Access Location: system and context command line

Command Mode: configuration mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.

Usage Guidelines

The local FWSM user authentication database consists of the users entered with the **username** command. The FWSM **login** command uses this database for authentication.

Examples

This example shows how to set the username for the specified privilege level:

```
fwsm/context_name(config)# username larry nopassword privilege 4
```

Related Commands

clear username
login
privilege
show username

virtual

To access the FWSM virtual server, use the **virtual** command.

virtual http *ip_address* [**warn**]

virtual telnet *ip_address*

Syntax Description

http	Allows web browsers to work correctly with the FWSM **aaa** command. See the "Usage Guidelines" section for additional information.
ip_address	IP address. See the "Usage Guidelines" section for additional information.
warn	(Optional) Notifies the **virtual http** command users that the command was redirected.
telnet	Logs you in and logs you out of the FWSM. See the "Usage Guidelines" section for additional information.

Defaults

This command has no default settings.

Command Modes

Security Context Mode: single context mode and multiple context mode

Access Location: context command line

Command Mode: configuration mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.

Usage Guidelines

The **virtual http** command allows web browsers to work correctly with the FWSM **aaa** command. The **aaa** command assumes that the AAA server database is shared with a web server. The FWSM automatically provides the AAA server and web server with the same information. The **virtual http** command works with the **aaa** command to authenticate the user, separate the AAA server information from the web client's URL request, and direct the web client to the web server. Use the **show virtual http** command to list the commands in the configuration. Use the **no virtual http** command to disable its use.

The **virtual http** command allows you to redirect the web browser's initial connection to the *ip_address,* which resides in the FWSM, authenticating the user, and then redirecting the browser to the URL that the user originally requested. The **virtual http** command accesses the virtual server for use with HTTP. This command is useful for FWSM interoperability with Microsoft IIS and for other authentication servers.

When using HTTP authentication to a site running Microsoft IIS that has "Basic text authentication" or "NT Challenge" enabled, users may be denied access from the Microsoft IIS server because the browser appends the string: "Authorization: Basic=Uuhjksdkfhk==" to the HTTP GET commands. This string contains the FWSM authentication credentials.

For outbound use, the *ip_address* must be entered as an address routed to the FWSM.

For inbound use, the *ip_address* must be entered as an unused global address.

Caution Do not set the **timeout uauth** duration to 0 seconds when using the **virtual** command because this action will prevent HTTP connections to the real web server.

The **virtual telnet** command allows the Virtual Telnet server to provide a way to preauthenticate users who require connections through the FWSM using services or protocols that do not support authentication.

You can use the **virtual telnet** command both to log in and log out of the FWSM. When an unauthenticated user connects through Telnet to the virtual IP address, that user is challenged for a username and password, and then authenticated with the TACACS+ or RADIUS server. Once authenticated, the user sees the message "Authentication Successful" and the authentication credentials are cached in the FWSM for the duration of the uauth timeout.

The Virtual Telnet server provides a way to preauthenticate users who require connections through the FWSM using services or protocols that do not support authentication. Users first connect to the Virtual Telnet server IP address, where the user is prompted for a username and password.

The **warn** keyword is applicable only for text-based browsers where the redirect cannot happen automatically.

Examples

This example shows how to make an inbound connection:

```
fwsm/context_name(config)# static (inside, outside) 209.165.201.1 209.165.201.1 netmask
255.255.255.255
fwsm/context_name(config)# access-list acl_out permit tcp any host 209.165.201.1 eq 80
fwsm/context_name(config)# access-group acl_out in interface outside
fwsm/context_name(config)# aaa authentication include any inbound 209.165.201.1
255.255.255.255 0 0 tacacs+
fwsm/context_name(config)# virtual http 209.165.201.1
```

This example displays the **show virtual** command output:

```
fwsm(config)# show virtual http
virtual http 209.165.201.1
```

After adding the **virtual telnet** command to the configuration and writing the configuration to the Flash partition, the users wanting to start Point-to-Point Tunneling (PPTP) sessions through the FWSM use Telnet to access the *ip_address*.

This example shows how to make a connection to the FWSM:

```
fwsm/context_name(config)# virtual telnet 209.165.201.25
fwsm/context_name(config)# aaa authentication include any outside 209.165.201.1
255.255.255.0 0 tacacs+
fwsm/context_name(config)# static (inside, outside) 209.165.201.25 209.165.201.25 netmask
255.255.255.255
fwsm/context_name(config)# access-list acl_out permit tcp any host 209.165.201.25 eq
telnet
fwsm/context_name(config)# access-group acl_out in interface outside
fwsm/context_name(config)# write memory
```

This example shows how to make a connection to an inside host:

```
fwsm(config)# /unix/host%telnet 209.165.201.30
Trying 209.165.201.25...
Connected to 209.165.201.25.
```

```
Escape character is '^]'.
fwsm(config)# username: username
fwsm(config)# TACACS+ Password: password
Authentication Successful
Connection closed by foreign host.
/unix/host%
```

The *username* and *password* are for the user on the TACACS+ server.

Related Commands

clear virtual

vpngroup

To configure the Cisco VPN client version 3.x (Cisco unified VPN client framework), use the **vpngroup** command.

vpngroup *group_name* {**address-pool** *pool_name*} | {**default-domain** *domain_name*} | {**dns-server** *dns_ip_prim* [*dns_ip_sec*]} | {**idle-time** *idle_seconds*} | {**max-time** *max_seconds*} | {**password** *preshared_key*} | {**split-tunnel** *access_list*} | {**wins-server** *wins_ip_prim* [*wins_ip_sec*]

Syntax Description

group_name	VPN policy group name; the name is an ASCII string with a maximum of 63 characters.
address-pool *pool_name*	Specifies the IP address pool name; the name can be up to 63 characters.
default-domain *domain_name*	Default domain name; the name can be up to 127 characters.
dns-server *dns_ip_prim*	Specifies the IP address of the primary DNS server.
dns_ip_sec	(Optional) IP address of the secondary DNS server.
idle-time *idle_seconds*	Specifies the idle timeout in seconds; valid values are from 60 to 86400 seconds.
max-time *max_seconds*	Specifies the maximum connection time in seconds that the VPN group is allowed; valid values are from 60 to 31536000 seconds.
password *preshared_key*	VPN group preshared key; the maximum is 127 characters.
split-tunnel *access_list*	Specifies the name of the access list for the split-tunnel configuration.
wins-server *wins_ip_prim*	Specifies the IP address of the primary Windows Internet Name Service (WINS) server.
wins_ip_sec	(Optional) IP address of the secondary WINS server.

Defaults

The defaults are as follows:

- *max_seconds* is set to unlimited.
- *idle_seconds* is **1800** seconds (30 minutes).

Command Modes

Security Context Mode: single context mode and multiple context mode

Access Location: system and context command line

Command Mode: configuration mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.

Usage Guidelines

Make sure that you configure the Internet Key Exchange (IKE) mode configuration before you configure support for the Cisco VPN Client. Specify that the FWSM initiates the IKE mode configuration.

For additional information about configuring interoperability with the Cisco VPN Client using the **vpngroup** commands, refer to the *Cisco VPN Configuration Guide*. The Cisco VPN Client supports Windows 2000.

The **vpngroup** command set allows you to configure Cisco VPN Client policy attributes to be associated with a VPN group name and are downloaded to the Cisco VPN Client(s) that are part of the given group. The same VPN group name specified here is configured in the Cisco VPN Client to ensure the matching of VPN client.

Configure a VPN group name of "default" to create a VPN group policy that matches any group name. The FWSM selects the VPN group name "default," if there is no other policy match. The VPN *group_name* is an ASCII string to denote a VPN group. You can make up the name. The maximum name has 63 characters.

The **vpngroup address-pool** command allows you to define a pool of local addresses to be assigned to a VPN group.

Note Both the **vpngroup address-pool** command and the **ip local pool** command enable you to specify a pool of local addresses for assigning dynamic IP addresses to VPN clients. For the Cisco VPN Client, the specified pool of addresses is associated with a given group, which consists of Cisco VPN Client users. We recommend that you use the **vpngroup address-pool** command only if you configure more than one pool of addresses to be used by more than one VPN user group. The **vpngroup address-pool** command allows the FWSM to configure different pools of local addresses for different user groups.

Use the **vpngroup** *group_name* **user-idle-timeout** *user_idle_seconds* command to set the IUA idle timeout.

Use the **vpngroup dns-server** command to enable the FWSM to download an IP address of a DNS server to a Cisco VPN Client as part of an IKE negotiation.

The **vpngroup wins-server** command allows the FWSM to download an IP address of a WINS server to a Cisco VPN Client as part of an IKE negotiation.

To enable the FWSM to download a default domain name to a Cisco VPN Client as part of IKE negotiation, use the **vpngroup default-domain** command.

Use the **vpngroup split-tunnel** command to enable split tunneling on the FWSM. Split tunneling allows a remote VPN client simultaneous encrypted access to the corporate network and clear access to the Internet. When you use the **vpngroup split-tunnel** command, specify the access list name to which you are associating split tunneling of traffic. With split tunneling enabled, the FWSM downloads its local network IP address and netmask specified within the associated access list to the VPN client or as part of the policy push to the client. The VPN client sends the traffic that is destined to the specified local FWSM network through an IPSec tunnel and all other traffic in the clear. The FWSM receives the IPSec-protected packet on its outside interface, decrypts it, and then sends it to its specified local network.

The networks defined in the **access-list deny** commands are not pushed to VPN clients.

The **vpngroup idle-time** command allows you to set the inactivity timeout for a Cisco VPN Client. When the inactivity timeout for all IPSec SAs have expired for a given VPN client, the tunnel is terminated.

The **vpngroup max-time** command allows you to set the maximum connection time for a Cisco VPN Client. When the maximum connection time is reached for a given VPN client, the tunnel is terminated. The connection between the Cisco VPN Client and the FWSM has to be reestablished. The default maximum connection time is set to an unlimited amount of time.

Note The inactivity timeout that is specified with the **vpngroup idle-time** command and the maximum connection time that is specified with the **vpngroup max-time** command for a given Cisco VPN Client take precedence over the commands that are used to set global lifetime timeouts. These commands are the **isakmp policy lifetime** and **crypto map set security-association lifetime seconds** commands.

Configure the VPN group's preshared key employing the **vpngroup password** command to be used during IKE authentication. This preshared key is equivalent to the password that you enter within the **Group Password** box of the Cisco VPN Client while configuring your group access information for a connection entry.

The FWSM-configured password displays in asterisks within the file configuration.

Note Both the **vpngroup password** command and the **isakmp key address** command allow you to specify a preshared key for IKE authentication. We recommend that you use the **vpngroup password** command only if you plan to configure more than one VPN user group. The **vpngroup password** command allows the FWSM to configure different VPN user groups.

Examples

This example shows that the VPN client(s) within the VPN group named as "myVpnGroup" is dynamically assigned with one of the IP addresses from the pool of addresses ranging from 10.140.40.0 to 10.140.40.7. The policy attributes for the group "myVpnGroup" are downloaded to the given VPN client during the policy push to the client. Split tunnelling is enabled. All traffic that is destined for the 10.130.38.0 255.255.255.0 FWSM network from the VPN client is protected by IPSec.

```
fwsm/context_name(config)# access-list 90 permit ip 10.130.38.0 255.255.255.0 10.140.40.0
255.255.255.248

fwsm/context_name(config)# ip local pool vpnpool 10.140.40.1-10.140.40.7

fwsm/context_name(config)# crypto ipsec transform-set esp-sha esp-null esp-sha-hmac
fwsm/context_name(config)# crypto dynamic-map dynmap 50 set transform-set esp-sha
fwsm/context_name(config)# crypto map mapName 10 ipsec-isakmp dynamic dynmap
fwsm/context_name(config)# crypto map mapName client configuration address initiate
fwsm/context_name(config)# crypto map mapName interface outside

fwsm/context_name(config)# isakmp enable outside
fwsm/context_name(config)# isakmp identity hostname
fwsm/context_name(config)# isakmp policy 7 authentication pre-share
fwsm/context_name(config)# isakmp policy 7 encryption 3des
fwsm/context_name(config)# isakmp policy 7 hash md5
fwsm/context_name(config)# isakmp policy 7 group 1

fwsm/context_name(config)# vpngroup myVpnGroup address-pool vpnpool
fwsm/context_name(config)# vpngroup myVpnGroup dns-server 10.131.31.11
fwsm/context_name(config)# vpngroup myVpnGroup wins-server 10.131.31.11
fwsm/context_name(config)# vpngroup myVpnGroup default-domain example.com
fwsm/context_name(config)# vpngroup myVpnGroup split-tunnel 90
fwsm/context_name(config)# vpngroup myVpnGroup idle-time 1800
fwsm/context_name(config)# vpngroup myVpnGroup max-time 86400
fwsm/context_name(config)# vpngroup myVpnGroup password ********
```

Related Commands

clear vpngroup
show vpngroup

who

To display active Telnet administration sessions on the FWSM, use the **who** command.

who [*local_ip*]

show who [*local_ip*]

Syntax Description

local_ip	(Optional) Internal IP address to limit the listing to one IP address or to a network IP address.

Defaults

This command has no default settings.

Command Modes

Security Context Mode: single context mode and multiple context mode

Access Location: system and context command line

Command Mode: privileged mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.

Usage Guidelines

The **who** command allows you to display the FWSM TTY_ID and IP address of each Telnet client that is currently logged into the FWSM. This command is the same as the **show who** command.

Examples

This example shows how to display active Telnet administration sessions on the FWSM:

```
fwsm# who

0: From 192.168.1.3
1: From 192.168.2.2
```

Related Commands

kill
show who
telnet

write

To store, view, or erase the current configuration, use the **write** command.

write net [[*tftp_ip*]:[*filename*]]

write {**erase** | **memory** | **terminal** | **standby**}

Syntax Description

tftp_ip	(Optional) IP address of the TFTP server.
filename	(Optional) Filename to qualify the location of the configuration file on the TFTP server named in *server_ip*.
erase	Clears the Flash partition configuration.
memory	Stores the current configuration in the Flash partition and the activation key value and time stamp for when the configuration was last modified.
terminal	Displays the current configuration on the terminal.
standby	Stores the configuration to the failover standby module from RAM to RAM.

Defaults

This command has no default settings.

Command Modes

Security Context Mode: single context mode and multiple context mode

Access Location: system and context command line

Command Mode: privileged mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
1.1(1)	Support for this command was introduced on the FWSM.

Usage Guidelines

Note The **write standby** command can be used only if the active and standby FWSMs are configured differently.

The **standby** keyword forces the configuration synchronization from the active to the standby module.

The **write net** command allows you to store the current configuration into a file on a TFTP server elsewhere in the network. The **write net** command allows you to use the TFTP server IP address that is specified in the **tftp-server** command. If you specify both the IP address and pathname in the **tftp-server** command, you can specify the **write net**:*filename* command as a colon (:) as follows:

```
fwsm(config)# write net :
```

If you set a filename with the **tftp-server** command, do not specify it in the **write** command; instead, use a colon (:) without a filename. Many TFTP servers require the configuration file to be world-writable to write to it.

Use the **configure net** command to get the configuration from the file.

The **write erase** command clears the Flash partition configuration.

The **write memory** command saves the current running configuration to the Flash partition. Use the **configure memory** command to merge the current configuration with the image that you saved in the Flash partition.

The FWSM allows processing to continue during the **write memory** command.

If another FWSM console user tries to change the configuration while you are executing the **write memory** command, the user receives this message:

```
Another session is busy writing configuration to memory
Please wait a moment for it to finish
```

After the **write memory** command completes, the FWSM allows the other command to complete.

Note Use the **write memory** command only if a configuration has been created with IP addresses for both network interfaces.

The **write terminal** command displays the current configuration in the FWSM's RAM memory. You can also display the configuration that is stored in the Flash partition by using the **show configure** command.

Defaults

The default on the FWSM is to store all configurations in compressed format.

Examples

This example shows how to specify the TFTP server and create a file named new_config in which to store the configuration:

```
fwsm(config)# tftp-server 10.1.1.2 /fwsmfirewall/config/new_config
write net :
```

This example shows how to erase the contents of the Flash partition and reload the FWSM:

```
fwsm(config)# write erase
Erase fwsm configuration in flash partition? [confirm] y
fwsm(config)# reload
Proceed with reload? [confirm] y
```

This example shows how to save the current configuration to the Flash partition:

```
fwsm(config)# write memory
Building configuration…
[OK]
```

This example shows how to display the configuration:

```
fwsm(config)# write terminal
Building configuration…
: Saved
```

Related Commands

configure

write standby

To force the configuration synchronization from the active to the standby module, use the **write standby** command.

write standby

Syntax Description

This command has no arguments or keywords.

Defaults

This command has no default settings.

Command Modes

Security Context Mode: single context mode and multiple context mode

Access Location: system and context command line

Command Mode: privileged mode

Firewall Mode: routed firewall mode and transparent firewall mode

Command History

Release	Modification
2.2(1)	Support for this command was introduced on the FWSM.

Usage Guidelines

The **write standby** command allows you to write the configuration that is stored in RAM on the active failover module to the RAM on the standby module. When the primary module boots, it automatically writes the configuration to the secondary module. Enter the **write standby** command if the primary and secondary module configurations have different information.

Note The **write standby** command can be used only if the active and standby FWSMs are configured differently.

You can use this command in these modes:

- Single Mode—Forces complete configuration synchronization to the standby module.
- Multi-mode, user context—Forces the context configuration to synchronize to the standby module.
- Multi-mode, system context—Forces the complete configuration (system and all user context configuration information) to synchronize to the standby module.

You can also display the configuration that is stored in the Flash partition by using the **show configure** command.

Examples

This example shows how to force the configuration synchronization from the active to the standby module:

```
fwsm(config)# write standby
Building configuration…
[OK]
```

Related Commands

clear failover
configure
failover
failover interface ip
failover interface-policy
failover lan interface
failover lan unit
failover link
failover polltime
failover replication http
failover reset
monitor-interface
show failover
write standby