Global Configuration Mode Commands

Use global configuration mode for setting, viewing, and testing configuration of WAAS software features for the entire device. To enter this mode, enter the **configure** command from privileged EXEC mode. The prompt for global configuration mode consists of the hostname of the WAE followed by (config) and the pound sign (#). You must be in global configuration mode to enter global configuration commands.

```
WAE# configure
WAE(config)#
```

Commands entered in global configuration mode update the running configuration file as soon as they are entered. These changes are not saved into the startup configuration file until you enter the **copy running-config startup-config** EXEC mode command. Once the configuration is saved, it is maintained across WAE reboots.

You also can use global configuration mode to enter specific configuration modes. From global configuration mode you can enter the interface configuration mode, standard ACL configuration mode, or the extended ACL configuration mode.

To exit global configuration mode and return to privileged-level EXEC mode, use either the **exit** or **end** global configuration command:

```
WAE(config)# exit
WAE#
```

(config) aaa accounting

To configure AAA accounting on a WAAS device, use the **aaa accounting** global configuration command. To unconfigure AAA, use the **no** form of this command.

aaa accounting cms enable tacacs+

no aaa accounting cms enable tacacs+

aaa accounting commands {**0** | **15**} **default** {**start-stop** | **stop-only** | **wait-start**} **tacacs**

no aaa accounting commands {**0** | **15**} **default** {**start-stop** | **stop-only** | **wait-start**} **tacacs**

aaa accounting exec default {**start-stop** | **stop-only** | **wait-start**} **tacacs**

no aaa accounting exec default {**start-stop** | **stop-only** | **wait-start**} **tacacs**

aaa accounting system default {**start-stop** | **stop-only**} **tacacs**

no aaa accounting system default {**start-stop** | **stop-only**} **tacacs**

Syntax Description

cms enable tacacs+	Enables accounting for all commands executed internally by the Central Manager. This feature is disabled by default.
commands	Configures accounting for all commands at the specified privilege level.
0	Specifies the user privilege level for a normal user.
15	Specifies the user privilege level for an administrative user.
default	Sets AAA accounting to use the default accounting list.
start-stop	Sends a start accounting notice at the beginning of a process and a stop accounting notice at the end of a process. The start accounting record is sent in the background. The requested user process begins regardless of whether the start accounting notice was received by the accounting server.
stop-only	Sends a stop accounting notice at the end of the process requested by the user.
wait-start	Sends both a start and a stop accounting notice to the accounting server. However, the requested user service does not begin until the start accounting notice is acknowledged. The user cannot execute a CLI command or login until the user is on record. A stop accounting notice is also sent but does not need acknowledgement.
tacacs	Enables use of TACACS+ for accounting.
exec	Enables accounting for user EXEC processes (user shells). When enabled, the EXEC shell accounting reports EXEC terminal session (user shell) events and login and logout by an administrator to the EXEC shell.
system	Enables accounting for all system-level events not associated with users, such as reloads.

Defaults

AAA accounting is disabled by default.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Examples

The following example shows how to configure TACACS+ on the WAAS device, specify that a start accounting notice should be sent at the beginning of the process and a stop accounting notice at the end of the process, and request that the user process should begin regardless of whether the start accounting notice was received by the accounting server:

```
WAE(config)# tacacs key abc
WAE(config)# tacacs server 192.168.50.1 primary
WAE(config)# aaa accounting system default start-stop tacacs
WAE# show aaa accounting
Accounting Type   Record event(s)  Protocol
-----------------------------------------------------------------
Exec shell         unknown            unknown
Command level  0   unknown            unknown
Command level 15   unknown            unknown
System             start-stop         TACACS+
```

The following example shows that the WAAS device is set to record all user EXEC sessions. The command also specifies that a stop accounting notice should be sent to the TACACS+ server at the end of the session.

```
WAE(config)# aaa accounting exec default stop-only tacacs
```

The following example shows that the WAAS device is set to record all CLI commands executed by a normal user. The command also specifies that a stop accounting notice should be sent to the TACACS+ server at the end of each CLI command executed by a normal user.

```
WAE(config)# aaa accounting commands 0 default stop-only tacacs
```

The following example shows that the WAAS device is set to record all CLI commands executed by an administrative user. The command also specifies that a start accounting notice should be sent to the TACACS+ server at the beginning of the process and a stop accounting notice at the end of the process. The CLI command executed by the administrative user does not proceed until the start accounting notice has been acknowledged.

```
WAE(config)# aaa accounting commands 15 default wait-start tacacs
```

The following example shows the EXEC shell accounting report that is available on the TACACS+ server:

```
Wed Apr 14 11:19:19 2004 172.16.0.0 super10 pts/0 172.31.0.0 start
start_time=1081919558 task_id=3028 timezone=PST service=shell
Wed Apr 14 11:19:23 2004 172.16.0.0 super10 pts/0 172.31.0.0
stop stop_time=1081919562 task_id=3028 timezone=PST service=shell
Wed Apr 14 11:22:13 2004 172.16.0.0 normal20 pts/0 via5.abc.com start
start_time=1081919732 task_id=3048 timezone=PST service=shell
Wed Apr 14 11:22:16 2004 172.16.0.0 normal20 pts/0 via5.abc.com stop
stop_time=1081919735 task_id=3048 timezone=PST service=shell
Wed Apr 14 11:25:29 2004 172.16.0.0 admin ftp via5.abc.com start start_time=1081919928
```

```
task_id=3069 timezone=PST service=shell
Wed Apr 14 11:25:33 2004 172.16.0.0 admin ftp via5.abc.com stop stop_time=1081919931
task_id=3069 timezone=PST service=shell
```

The following example shows the system accounting report that is available on the TACACS+ server:

```
Wed Apr 14 08:37:14 2004 172.16.0.0 unknown unknown 0.0.0.0 start start_time=1081909831
task_id=2725 timezone=PST service=system event=sys_acct reason=reload
Wed Apr 14 10:19:18 2004 172.16.0.0 admin ttyS0 0.0.0.0 stop stop_time=1081915955
task_id=5358 timezone=PST service=system event=sys_acct reason=shutdown
```

The following example shows the command accounting report that is available on the TACACS+ server:

```
Wed Apr 14 12:35:38 2004 172.16.0.0 admin ttyS0 0.0.0.0 start start_time=1081924137
task_id=3511 timezone=PST service=shell -lvl=0 cmd=logging console enable
Wed Apr 14 12:35:39 2004 172.16.0.0 admin ttyS0 0.0.0.0 stop stop_time=1081924137
task_id=3511   timezone=PST service=shell priv-lvl=0 cmd=logging console enable
```

In addition to command accounting, the WAAS device records any executed CLI command in the system log (*syslog.txt*). The message format is as follows:

```
ce_syslog(LOG_INFO, CESM_PARSER, PARSER_ALL, CESM_350232,
        "CLI_LOG %s: %s \n", __FUNCTION__, pd->command_line);
```

Related Commands **show aaa accounting**

(config) aaa authorization commands

To authorize commands issued through the CLI by a user on a WAAS device, use the **aaa authorization commands** global configuration command. To disable command authorization, use the **no** form of this command.

aaa authorization commands *level* **default tacacs+**

no aaa authorization commands *level* **default tacacs+**

Syntax Description

level **default tacacs+**	Configures command authorization for commands issued by the CLI user. Commands at the specified privilege level (0 or 15) are authorized. Level 0 authorizes EXEC commands, level 15 authorizes both EXEC and global configuration commands.

Defaults

AAA command authorization is disabled by default.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Command authorization enforces authorization through an external AAA server for each command executed by the user. All commands executed by a CLI user are authorized before they are executed.

When command authorization is configured for level 0, only EXEC commands are authorized, regardless of user level (normal or super).

When command authorization is configured for level 15, EXEC and global configuration commands are authorized, regardless of user level (normal or super).

Once it is configured, command authorization configuration is displayed in the running config. When the running config is copied to the startup config, command authorization is configured as the last config so that during the reload, the startup config need not be authorized.

Only commands executed through the CLI interface are subject to command authorization.

Examples

The following example shows how to configure command authorization for level 15 (authorization for both EXEC and global configuration commands) on the WAAS device:

```
WAE(config)# aaa authorization commands 15 default tacacs+
```

Related Commands

show aaa authorization

(config) accelerator cifs

To enable the CIFS application accelerator, use the **accelerator cifs** global configuration command. To disable the CIFS application accelerator, use the **no** form of this command.

accelerator cifs {[**double-byte-unicode**] | **enable** | **eviction-monitor** {**cumulative-time** *mins* | **duration** *mins* | **enable**} | **dynamic-share** *share* | **clear cache** | **cache server-rename** *oldname newname* | **exception** {**coredump** | **debug** | **no-coredump**}}

no accelerator cifs {[**double-byte-unicode**] | **enable** | **eviction-monitor** {**cumulative-time** *mins* | **duration** *mins* | **enable**} | **dynamic-share** *share* | **clear cache** | **cache server-rename** *oldname newname* | **exception** {**coredump** | **debug** | **no-coredump**}}

Syntax Description

double-byte-unicode	(Optional) Enables support for double-byte Unicode languages for Windows 98 clients.
enable	Enables the CIFS traffic accelerator.
eviction-monitor	Configures cache eviction monitoring.
cumulative time *mins*	Sets the cumulative time in minutes over which aggressive cache eviction should be monitored.
duration *mins*	Sets the duration in minutes for aggressive cache eviction monitoring.
enable	Starts cache eviction monitoring.
dynamic-share *share*	Enables support for CIFS dynamic shares and specifies a path in the format: cifs://server/share
clear cache	Clears the CIFS application accelerator cache and restarts the accelerator.
cache server-rename *oldname newname*	Renames a CIFS file server for the cached data.
exception	Configures the action to be taken if an exception occurs.
coredump	Writes a core file (default).
debug	Hangs the system until it is explicitly restarted.
no-coredump	Restarts the accelerator and does not write a core file.

Defaults

The CIFS accelerator is enabled by default and will start automatically if the Enterprise license is installed. The default exception action is coredump.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

Use the **accelerator cifs enable** command to enable the acceleration of CIFS traffic with the transparent CIFS accelerator (not the legacy mode of acceleration).

The CIFS application accelerator requires that the WINS server be configured. Use the **windows-domain wins-server** global configuration command to configure the WINS server.

Use the **accelerator cifs eviction-monitor** command to enable and monitor the aggressive cache eviction for a cumulative time or duration.

To configure prepositioning directives, use the **accelerator cifs preposition** global configuration command.

Use the **accelerator cifs dynamic-share** command to define a dynamic share, which allows multiple users to access the same share but then be automatically mapped to a different directory based on the user's credentials. Defining a dynamic share allows each user to see a different view of the share, and allows the operation of Access Based Enumeration, if configured on Windows Server.

Note We recommend that you use the WAAS Central Manager GUI to configure dynamic shares because the dynamic share CLI configuration can be overwritten by the Central Manager. For more information, see the "Creating Dynamic Shares" section in the *Cisco Wide Area Application Services Configuration Guide*.

Use the **accelerator cifs cache server-rename** command to rename the data in the cache if the name of a file server changed and you do not want to lose the cached data for the server. The renaming applies to prepositioned files and files cached on demand.

Note Do not specify the name of another existing cached file server as the new name. If you do specify an existing name as the new name, the cached contents of this file server are overwritten with the cached contents of the file server you are renaming.

Examples

The following example shows how to enable the CIFS application accelerator:

```
WAE(config)# accelerator cifs enable
```

Related Commands

show accelerator

show statistics accelerator

(config) windows-domain

(config) accelerator cifs preposition

To configure a CIFS application accelerator preposition directive, use the **accelerator cifs preposition** global configuration command. To disable the application accelerator, use the **no** form of this command.

accelerator cifs preposition [**remove**] *directive_id*

no accelerator cifs preposition [**remove**] *directive_id*

Syntax Description

remove	(Optional) Deletes a preposition directive.
directive_id	ID of an existing preposition directive that you want to change or delete, or a new directive that you want to create.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

Use the **accelerator cifs preposition** command to create and edit preposition directives to be used with the transparent CIFS accelerator. A preposition directive defines a set of files that are to be prepositioned on the WAE device.

The **accelerator cifs preposition** command invokes the preposition configuration submode. For details on the commands available in this submode to configure a preposition directive, see the "Preposition Configuration Mode Commands" section.

Note We recommend that you use the WAAS Central Manager GUI to configure preposition directives. For more information, see the "Creating a Preposition Directive" section in the *Cisco Wide Area Application Services Configuration Guide*.

Note If you create a preposition directive from the CLI before the secure store on the WAE is initialized, you must wait at least two datafeed poll cycles (10 minutes by default) before initializing the secure store; otherwise, the preposition directive will not propagate to the Central Manager because the credentials will not be able to be decrypted on the WAE.

Examples

The following example shows how to create a new CIFS preposition directive with ID 3:

```
WAE(config)# accelerator cifs preposition 3
WAE(config-preposition)
```

Related Commands

show accelerator

show statistics accelerator

(config) windows-domain

(config) accelerator epm

To enable the Endpoint Mapper (EPM) application accelerator, use the **accelerator epm** global configuration command. To disable the EPM application accelerator, use the **no** form of this command.

accelerator epm {**enable** | **exception** {**coredump** | **debug** | **no-coredump**}}

no accelerator epm {**enable** | **exception** {**coredump** | **debug** | **no-coredump**}}

Syntax Description

enable	(Optional) Enables the EPM application accelerator.
exception	(Optional) Configures the action to be taken if an exception occurs.
coredump	Writes a core file (default).
debug	Hangs the system until it is explicitly restarted.
no-coredump	Restarts the accelerator and does not write a core file.

Defaults

The EPM accelerator is enabled by default and will start automatically if the Enterprise license is installed. The default exception action is coredump.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

Use the **accelerator epm enable** command to enable the acceleration of EPM traffic. The EPM accelerator must be enabled for the MAPI accelerator to operate.

Examples

The following example shows how to enable the EPM application accelerator:

```
WAE(config)# accelerator epm enable
```

Related Commands

(config) accelerator mapi

show accelerator

show statistics accelerator

(config) accelerator http

To enable the HTTP application accelerator, use the **accelerator http** global configuration command. To disable the HTTP application accelerator, use the **no** form of this command.

accelerator http {**enable** | **dre-hints** {**access-list** *acl* | **enable**}| **exception** {**coredump** | **debug** | **no-coredump**} | **metadatacache** {**access-list** *acl* | **enable** | **conditional-response enable** | **filter-extension** *extension-list* | **redirect-response enable** | **request-ignore-no-cache enable** | **response-ignore-no-cache enable**| **unauthorized-response enable** | **max-age** *seconds* | **min-age** *seconds* | **filter-extension** *extension-list* | **https** {**access-list** *acl* | **enable**}} | **sharepoint-opt prefetch enable** | **suppress-server-encoding** {**access-list** *acl* | **enable**}}

no accelerator http {enable | **dre-hints** {**access-list** *acl* |**enable**}| **exception** {**coredump** | **debug** | **no-coredump**} | **metadatacache** {**access-list** *acl* | **enable** | **conditional-response enable** | **filter-extension** *extension-list* | **redirect-response enable** | **request-ignore-no-cache enable** | **response-ignore-no-cache enable**| **unauthorized-response enable** | **max-age** *seconds* | **min-age** *seconds* | **filter-extension** *extension-list* | **https {access-list** *acl* | **enable**}} | **sharepoint-opt prefetch enable** | **suppress-server-encoding** {**access-list** *acl* | **enable**}}

Syntax Description

enable	(Optional) Enables the HTTP application accelerator.
dre-hints	Configures HTTP and HTTPS DRE hints feature.
access-list *acl*	Configures the HTTP AO feature subnet to associate an access list to an HTTP AO feature. *acl* refers to an ACL that can be created by the *ip access-list* CLI. See (config) ip access-list, page -664.
exception	(Optional) Configures the action to be taken if an exception occurs.
coredump	Writes a core file (default).
debug	Hangs the system until it is explicitly restarted.
no-coredump	Restarts the accelerator and does not write a core file.
metadatacache	(Optional) Configures metadata caching.
enable	(Optional) Enables metadata caching.
conditional-response enable	(Optional) Enables caching of HTTP 304 messages.
redirect-response enable	(Optional) Enables caching of HTTP 301 messages.
request-ignore-no-cache enable	Configures the metadata cache to ignore cache-control on requests.
response-ignore-no-cache enable	Configures the metadata cache to ignore cache-control on responses.
unauthorized-response enable	(Optional) Enables caching of HTTP 401 messages.
max-age *seconds*	(Optional) Specifies the maximum number of seconds to retain HTTP header information in the cache. The default is 86400 seconds (24 hours). Valid time periods range from 5–2592000 seconds (30 days).
min-age *seconds*	(Optional) Specifies the minimum number of seconds to retain HTTP header information in the cache. The default is 60 seconds. Valid time periods range from 5–86400 seconds (24 hours).

filter-extension *extension-list*	(Optional) String containing a comma-separated list of file extensions to which metadata caching is to be applied. Do not include the dot at the beginning of the file extension. You can specify a maximum of 20 file extensions.
https enable	(Optional) Enables metadata caching for HTTPS traffic.
sharepoint-opt prefetch enable	(Optional) Enables data to be prefetched from the SharePoint server and serve it from the cache to the client.
suppress-server-encoding enable	(Optional) Enables suppression of Accept-Encoding compress, gzip, and deflate request-headers between the client and the server for HTTP and HTTPS.

Defaults

The HTTP accelerator is enabled by default and will start automatically if the Enterprise license is installed. The default exception action is coredump.

The metadata caching feature is disabled by default for all response types. The default max-age is 86400 seconds (24 hours), the default min-age is 60 seconds, and the default filter extension list is empty (meaning that metadata caching is applied to all extension types).

The SharePoint optimization feature is disabled by default.

When suppress-server-encoding is enabled, it suppresses the server compression for both HTTP and HTTPS requests. The suppress server encoding feature is disabled by default.

The DRE hints feature applies to both HTTP and HTTPS requests. It is disabled by default.

The subnet feature is enabled after the subnet configuration is added.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

Use the **accelerator http enable** command to enable the acceleration of HTTP traffic.

You can enable or disable each of three metadata caches (conditional-response, redirect-response, and unauthorized-response) separately. By default they are all enabled when you enable HTTP metadata caching. If you disable the HTTP accelerator, metadata caching is also disabled.

When you enable the suppress-server-encoding feature, the WAE removes the Accept-Encoding header from HTTP requests, preventing the web server from compressing HTTP data that it sends to the client. This allows the WAE to apply its own compression to the HTTP data, typically resulting in much better compression than the web server.

Use the SharePoint optimization feature when you need to access Microsoft Office documents stored on a SharePoint server 2010, using a web browser. Enabling this feature will prefetch the data from the server and serve it from the cache, which reduces latency and improves the user experience.

The DRE hint feature improves DRE performance. This feature is not automatically enabled when metadata caching or the suppress server encoding feature is enabled.

The options **request-ignore-no-cache** and **response-ignore-no-cache** are disabled by default. Because the HTTP accelerator is conservative in caching client request metadata and server response metadata, deployments may want to test with these settings enabled to improve the HTTP metadata cache hit ratio to achieve less latency.

If an existing subnet configuration gets modified or removed, the new configuration applies to new connections only, and does not impact the existing HTTP sessions. The change takes effect only after the change is updated in the kernel. Only one ACL is associated with each feature and a new subnet configuration replaces the old one. Use the **no** command to remove the subnet configuration. If the HTTP AO feature is globally disabled, the feature is not applied to any session. If the HTTP AO feature is globally enabled, and if the acl lookup result for this session is permit, the feature applies to the session; otherwise, it does not apply. HTTP AO bypass-list takes precedence over this feature.

Examples

The following example shows how to enable the HTTP application accelerator:

```
WAE(config)# accelerator http enable
```

The following example shows how to enable and configure the metadata cache to operate only on specific file types:

```
WAE(config)# accelerator http metadatacache enable
WAE(config)# accelerator http metadatacache filter-extension html,css,jpg,gif
```

Related Commands

clear cache

show accelerator

show cache http-metadatacache

show statistics accelerator

(config) accelerator ica

To enable the ICA application accelerator, use the **accelerator ica** global configuration command. To disable the ICA application accelerator, use the **no** form of this command.

accelerator ica {**enable** | **exception** {**coredump** | **debug** | **no-coredump**} | **wansecure-mode** {**always** | **none**}}

accelerator ica {**enable** | **exception** {**coredump** | **debug** | **no-coredump**} | **wansecure-mode** {**always** | **none**}}

Syntax Description

enable	Enables the ICA traffic accelerator.
exception	Configures the action to be taken if an exception occurs.
coredump	Writes a core file (default).
debug	Hangs the system until it is explicitly restarted.
no-coredump	Restarts the accelerator and does not write a core file.
wansecure-mode	Configures the state of WAN Secure mode.
always	Enables WAN Secure mode for ICA.
none	Disables WAN Secure mode for ICA (default).

Defaults

The ICA accelerator is enabled by default. The default exception action is coredump. The default WAN Secure mode state is none.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

Use the **accelerator ica enable** command to enable the acceleration of ICA (Independent Computing Architecture) traffic with the transparent ICA accelerator. The ICA application accelerator provides WAN optimization on a WAAS device for ICAtraffic which is used to access a virtual desktop infrastructure (VDI). This is done through a process that is both automatic and transparent to the client and server.

Use the **accelerator ica wansecure-mode always** command to enable WAN Secure mode for ICA. The WAN Secure mode configuration in both of the peer WAEs must match in order for the ICA accelerator to optimize connections.

WAN Secure mode requires that the SSL application accelerator is enabled. Use the **accelerator ssl enable** global configuration command to enable the SSL accelerator.

Examples

The following example shows how to enable the ICA application accelerator:

```
WAE(config)# accelerator ica enable
```

Related Commands

show accelerator

show statistics accelerator

(config) windows-domain

(config) accelerator mapi

To enable the MAPI application accelerator, use the **accelerator mapi** global configuration command. To disable the MAPI application accelerator, or one of its options, use the **no** form of this command.

accelerator mapi {**enable** | **encryption** | **read-opt** | **write-opt** | **reserved-pool-size maximum-percent** *max_percent* | **wansecure-mode** {**always** | **auto** | **none**} | **exception** {**coredump** | **debug** | **no-coredump**}}

no accelerator mapi {**enable** | **encryption** | **read-opt** | **write-opt** | **reserved-pool-size maximum-percent** *max_percent* | **wansecure-mode** {**always** | **auto** | **none**} | **exception** {**coredump** | **debug** | **no-coredump**}}

Syntax Description

enable	Enables the MAPI traffic accelerator.
encryption	Enables the acceleration of encrypted MAPI traffic.
read-opt	Enables the read-ahead optimization of the MAPI traffic for mail reading.
write-opt	Enables the asynchronous write optimization of the MAPI traffic for mail sending.
reserved-pool-size maximum-percent *max_percent*	Configures the maximum reserved connection pool percent, specified as the percent of the device TFO connection limit, to restrict the maximum connections reserved for MAPI optimization during TFO overload. Range is from 5 to 50. Default is 15.
wansecure-mode	Configures the state of WAN Secure mode.
always	Enables WAN Secure mode for encrypted MAPI acceleration.
auto	Enables WAN Secure mode for encrypted MAPI acceleration only if encrypted traffic is received.
none	Disables WAN Secure mode for encrypted MAPI acceleration.
exception	(Optional) Configures the action to be taken if an exception occurs.
coredump	Writes a core file (default).
debug	Hangs the system until it is explicitly restarted.
no-coredump	Restarts the accelerator and does not write a core file.

Defaults

The MAPI accelerator is enabled by default and will start automatically if the Enterprise license is installed. Encrypted MAPI traffic acceleration is not enabled by default. The read optimization (**read-opt**) and write optimization (**write-opt**) features are enabled by default when the MAPI accelerator is enabled. The default maximum reserved connection pool percent is 15. The default WAN secure mode is auto. The default exception action is coredump.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

Use the **accelerator mapi enable** command to enable MAPI acceleration. This feature supports Microsoft Outlook 2000–2007 clients. Secure connections that use message authentication (signing) or encryption are not accelerated and MAPI over HTTP is not accelerated.

You must enable the EPM accelerator before the MAPI accelerator can operate.

Use the **reserved-pool-size** keyword to restrict the maximum number of connections reserved for MAPI optimization during TFO overload. It is specified as a percent of the TFO connection limit of the platform. Valid percent ranges from 5%-50%. The default is 15% which would reserve approximately 0.5 connection for each client-server Association Group (AG) optimized by MAPI accelerator.

The client maintains at least one AG per server it connects to with an average of about 3 connections per AG. For deployments that observe a greater average number of connections per AG, or where TFO overload is a frequent occurrence, a higher value for the reserved pool size maximum percent is recommended.

Reserved connections would remain unused when the device is not under TFO overload. Reserved connections are released when the AG terminates.

Examples

The following example shows how to enable the MAPI application accelerator:

```
WAE(config)# accelerator mapi enable
```

Related Commands

(config) accelerator epm

show accelerator

show statistics accelerator

(config) accelerator nfs

To enable the NFS application accelerator, use the **accelerator nfs** global configuration command. To disable the NFS application accelerator, use the **no** form of this command.

accelerator nfs {**enable** | **exception** {**coredump** | **debug** | **no-coredump**}}

no accelerator nfs {**enable** | **exception** {**coredump** | **debug** | **no-coredump**}}

Syntax Description

enable	(Optional) Enables the EPM application accelerator.
exception	(Optional) Configures the action to be taken if an exception occurs.
coredump	Writes a core file (default).
debug	Hangs the system until it is explicitly restarted.
no-coredump	Restarts the accelerator and does not write a core file.

Defaults

The NFS accelerator is enabled by default and will start automatically if the Enterprise license is installed. The default exception action is coredump.

Command Modes

global configuration

Device Modes

application-accelerator

Examples

The following example shows how to enable the NFS application accelerator:

```
WAE(config)# accelerator nfs enable
```

Related Commands

show accelerator

show statistics accelerator

(config) accelerator smb

To enable the SMB application accelerator, use the **accelerator smb** global configuration command. To disable the SMB application accelerator, use the **no** form of this command.

accelerator smb {**enable** | **alarm digital-signing enable** | **batch-close-opt enable** | **change-notif size** *size* | **dir-opt** {**enable** | **aging** *seconds*} | **dre-hints dre enable** | **dynamic-share** *name* | **exception** {**coredump** | **debug** | **no-coredump**} | **highest-dialect** {**ntlm0-12** | **smb2-002** | **smb2-1**} **exceed-action** {**handoff** | **mute**} | **invalid-fid-opt enable** | **iobuf size** *mb* | **max-pkt-size size** *kb* | **metadata-opt** {**enable** | **cache-size** *mb* [**force**]} | **namedpipe-opt** {**enable** | **cache-size** *kb* | **resp-cache lifetime** *seconds* | **sess-cache lifetime** *seconds*} | **nf-cache** {**enable** | **aging** *seconds* | **bypass-patterns** *regex* | **size** *mb*} | **office-opt enable** | **optimization bypass-pattern** *regex* | **read-ahead** {**enable** | **buffer-size** *mb* [**force**] | **exhaust-distance** *kb* | **extended-window** *kb* | **hit-threshold** *percentage* | **init-window** *kb* | **max-active** *div* | **wait-distance** *kb*} | **write-opt** {**enable** | **quota-aging** *seconds* | **quota-threshold** *mb*}}

no accelerator smb {**enable** | **alarm digital-signing enable** | **batch-close-opt enable** | **change-notif size** *size* | **dir-opt** {**enable** | **aging** *seconds*} | **dre-hints dre enable** | **dynamic-share** *name* | **exception** {**coredump** | **debug** | **no-coredump**} | **highest-dialect** {**ntlm0-12** | **smb2-002** | **smb2-1**} **exceed-action** {**handoff** | **mute**} | **invalid-fid-opt enable** | **iobuf size** *mb* | **max-pkt-size size** *kb* | **metadata-opt** {**enable** | **cache-size** *mb* [**force**]} | **namedpipe-opt** {**enable** | **cache-size** *kb* | **resp-cache lifetime** *seconds* | **sess-cache lifetime** *seconds*} | **nf-cache** {**enable** | **aging** *seconds* | **bypass-patterns** *regex* | **size** *mb*} | **office-opt enable** | **optimization bypass-pattern** *regex* | **read-ahead** {**enable** | **buffer-size** *mb* [**force**] | **exhaust-distance** *kb* | **extended-window** *kb* | **hit-threshold** *percentage* | **init-window** *kb* | **max-active** *div* | **wait-distance** *kb*} | **write-opt** {**enable** | **quota-aging** *seconds* | **quota-threshold** *mb*}}

Syntax Description

enable	Enables the SMB traffic accelerator.
alarm digital-signing enable	Enables the digital-signing alarm.
batch-close-opt enable	Enables asynchronous close optimization for SMB2 protocol.
change-notif size *size*	Sets the change notification table size. Valid values range from 1–2048 entries. The default is10.
dir-opt enable	Enables directory listing optimization.
aging *seconds*	Configures metadata directory list aging time to the specified number of seconds. If the age of a metadata directory list exceeds this time when the metadata is requested, the entry is considered stale and is updated by retrieving it from the file server.
dre-hints dre enable	Enables DRE and LZ hints.
dynamic-share *name*	Adds the specified share to the existing dynamic share configuration. The share name must use the format //server/share and must not exceed 256 characters.
exception	(Optional) Configures the action to be taken if an exception occurs.
coredump	Writes a core file (default).
debug	Hangs the system until it is explicitly restarted.
no-coredump	Restarts the accelerator and does not write a core file.
highest-dialect	Configures the highest dialect to be optimized.

ntlm0-12	Configures NTLM version 0.12 to be the highest dialect.
smb2-002	Configures SMB version 2.002 to be the highest dialect.
smb2-1	Configures SMB version 2.1 to be the highest dialect.
exceed-action	Configures the action if a request uses a dialect higher than the configured highest dialect to be optimized.
handoff	The connection is handed off to the generic application accelerator.
mute	The connection is removed from the negotiate request.
invalid-fid-opt enable	Enables SMB2 invalid file ID optimization. The SMB accelerator issues a local response to files with invalid file ID values.
iobuf size *mb*	Configures the IOBUF buffer size, in MB, from 50 to 1000.
max-pkt-size *kb*	Configures the maximum SMB packet size, in KB, from 64 to 16384.
metadata-opt enable	Enables metadata optimization.
cache-size *mb*	Configures metadata cache size, in MB, from 50 to 360000.
force	Forces the metadata cache size setting.
namedpipe-opt enable	Enables named pipe optimization.
cache-size *kb*	Configures the size of the named pipe cache, in KB, from 128 to 150000.
resp-cache lifetime *seconds*	Configures the response cache lifetime, in seconds, from 0 to 1024.
sess-cache lifetime *seconds*	Configures the session cache lifetime, in seconds, from 0 to 1024.
nf-cache enable	Enables not-found metadata cache optimization.
aging *seconds*	Configures the length of time, in seconds, that not-found metadata cache entries are held in the cache, from 1 to 60 (the default is 30).
bypass-patterns *regex*	Configures a case-insensitive regular expression that matches filenames to be bypassed by the not-found metadata cache.
size *mb*	Configures the maximum size of the not-found metadata cache, in MB, from 1 to 256 (the default is 32).
office-opt enable	Enables Microsoft Office optimization.
optimization bypass-pattern *regex*	Configures a case-insensitive regular expression that matches filenames to be bypassed for all optimizations.
read-ahead enable	Enables read-ahead optimization.
buffer size *mb*	Configures read-ahead buffer size, in MB, from 50 to 10000.
force	Forces the read-ahead cache size setting.
exhaust-distance *kb*	Configures read-ahead window exhaust distance, in KB, from 128 to 1024 (the default is 196).
extended-window *kb*	Configures read-ahead window exhaust distance, in KB, from 256 to 3200 (the default is 640).
hit-threshold *percentage*	Configures read-ahead hit threshold, as a percentage from 10 to 100 (the default is 70).
init-window *kb*	Configures read-ahead initial window size, in KB, from 128 to 1024 (the default is 196).
max-active *div*	Configures read-ahead maximum active memory usage divisor, from 2 to 10 (the default is 4).

wait-distance *kb*	Configures read-ahead wait distance, in KB, from 128 to 3200 (the default is 512).
write-opt enable	Enables async-write optimization.
quota-aging *seconds*	Configures network share quota threshold aging time, in seconds, from 1 to 120 (the default is 60).
quota-threshold *mb*	Configure network share quota threshold, in MB, from 1 to 1024 (the default is 20).

Defaults

The SMB accelerator is disabled by default.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

The enterprise license is required to start the SMB accelerator. Enabling the SMB accelerator disables the CIFS accelerator if it is enabled.

The EXEC mode command **show running-config** displays non-default settings only. Therefore, the command **no accelerator smb enable** does not show up in the running configuration if the SMB accelerator is disabled, while the **accelerator smb enable** command does display if the SMB accelerator is enabled.

Examples

The following example shows how to enable the SMB application accelerator:

```
WAE(config)# accelerator smb enable
```

Related Commands

show accelerator

show statistics accelerator

(config) accelerator ssl

To enable the SSL application accelerator, use the **accelerator ssl** global configuration command. To disable the SSL application accelerator, use the **no** form of this command.

accelerator ssl {**enable** | **exception** {**coredump** | **debug** | **no-coredump**}}

no accelerator ssl {**enable** | **exception** {**coredump** | **debug** | **no-coredump**}}

Syntax Description

enable	(Optional) Enables the SSL application accelerator.
exception	(Optional) Configures the action to be taken if an exception occurs.
coredump	Writes a core file (default).
debug	Hangs the system until it is explicitly restarted.
no-coredump	Restarts accelerator and does not write a core file.

Defaults

The SSL accelerator is enabled by default and will start automatically if the Enterprise license is installed. The default exception action is coredump.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

Use the **accelerator ssl enable** command to enable the acceleration of SSL traffic. To undo this command, for example to disable SSL acceleration after you have enabled it, use the **no** version of this command.

Examples

The following example shows how to enable the SSL application accelerator:

```
WAE(config)# accelerator ssl enable
```

Related Commands

show accelerator

show statistics accelerator

crypto delete

crypto export

crypto generate

crypto import

(config) crypto pki

(config) crypto ssl

(config-ca) ca-certificate
(config-ca) description
(config-ca) revocation-check

(config) accelerator video

To enable the video application accelerator, use the **accelerator video** global configuration command. To disable the video application accelerator, use the **no** form of this command.

accelerator video {**enable** | **unaccelerated-traffic type** {**all** | **overload**} **action drop** | **max-initial-setup-delay** *seconds* | **windows-media** {**client idle-timeout** *seconds* | **log-forwarding enable**}}

no accelerator video {**enable** | **unaccelerated-traffic type** {**all** | **overload**} **action drop** | **max-initial-setup-delay** *seconds* | **windows-media** {**client idle-timeout** *seconds* | **log-forwarding enable**}}

accelerator video exception {**coredump** | **debug** | **no-coredump**}

no accelerator video exception {**coredump** | **debug** | **no-coredump**}

Syntax Description

enable	Enables the video traffic accelerator.
unaccelerated-traffic type	Configures the handling of video traffic that is not being accelerated due to overload or unsupported transport or format, including Windows Media video on demand traffic and all RTSP traffic that is not for Windows Media.
all	Selects all video traffic that is not being accelerated due to overload or unsupported transport or format, including Windows Media video on demand traffic and all RTSP traffic that is not for Windows Media.
overload	Selects video traffic that is not being accelerated due to an overload condition.
action drop	Drops the specified type of video traffic that is not being accelerated. The connection is actually reset. If you do not specify this action, the default is to handle such traffic with the negotiated TCP optimization policy.
max-initial-setup-delay *seconds*	Sets the maximum number of seconds to wait for the first message from the client and the first response from the server, after the connection is accepted by the video accelerator, and before timing out the connection. Valid values range from 10–180 seconds. The default is 60.
windows-media	Configures Windows Media-specific settings.
client idle-timeout *seconds*	Sets the maximum number of seconds to wait after the initial client request, while the client connection is idle, before timing out the connection. Valid values range from 30–300 seconds. The default is 60.
log-forwarding enable	Enables forwarding of Windows Media logs to the upstream Windows Media Server. Log forwarding is enabled by default.
exception	(Optional) Configures the action to be taken if an exception occurs.
coredump	Writes a core file (default).
debug	Hangs the system until it is explicitly restarted.
no-coredump	Restarts the accelerator and does not write a core file.

Defaults

The video accelerator is enabled by default and will start automatically if both the Enterprise and Video licenses are installed. The default exception action is coredump.

Command Modes global configuration

Device Modes application-accelerator

Usage Guidelines Use the **accelerator video enable** command to enable the acceleration of Windows Media live streaming video traffic that uses the RTSP.

You can configure the video accelerator to discard unaccelerated video traffic by using the **unaccelerated-traffic type** {**all** | **overload**} **action drop** option. If you do not specify this option, the unaccelerated video traffic is handled with the negotiated TCP optimization policy.

Examples The following example shows how to enable the video application accelerator:

```
WAE(config)# accelerator video enable
```

Related Commands **show accelerator**

show statistics accelerator

(config) accelerator windows-print

To enable the Windows print accelerator for print traffic using a CIFS application accelerator, use the **accelerator windows-print** global configuration command. To disable the Windows print accelerator, use the **no** form of this command.

Note To enable the Windows print accelerator for print traffic using an SMB application accelerator, use the **(config) smb accelerator print-opt enable** command.

accelerator windows-print enable

no accelerator windows-print enable

Syntax Description

enable	Enables the Windows print accelerator.

Defaults

The Windows print accelerator is enabled by default and will start automatically if the Enterprise license is installed.

Command Modes

global configuration

Device Modes

application-accelerator

Examples

The following example shows how to enable the Windows print accelerator:

```
WAE(config)# accelerator windows-print enable
```

Related Commands

show statistics windows-print requests

(config) alarm overload-detect

To detect alarm overload situations, use the **alarm overload-detect** global configuration command. To unconfigure alarm parameters, use the **no** form of this command.

alarm overload-detect {**clear** *1-999* [**raise** *10-1000*] | **enable** | **raise** *10-1000* [**clear** *1-999*]}

no alarm overload-detect {**clear** *1-999* [**raise** *10-1000*] | **enable** | **raise** *10-1000* [**clear** *1-999*]}

Syntax Description

clear *1-999*	Specifies the number of alarms per second at which the alarm overload state on the WAAS device is cleared. When the alarm drops below this threshold, the alarm is cleared and the SNMP traps and alarm notifications are again sent to your NMS. **Note** The **alarm overload-detect clear** value must be less than the **alarm overload-detect raise** value.
raise *10-1000*	(Optional) Specifies the number of alarms per second at which the WAAS device enters an alarm overload state and SNMP traps and alarm notifications to your network management station (NMS) are suspended.
enable	Enables the detection of alarm overload situations.

Defaults

clear: 1 alarm per second

raise: 10 alarms per second

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

In the alarm overload state, applications continue to raise alarms and these alarms are recorded within the WAAS device. Use the **show alarms** and **show alarms history** EXEC commands to display all the alarms in the alarm overload state.

Examples

The following example shows how to enable detection of alarm overload:

```
WAE(config)# alarm overload-detect enable
```

The following example shows how to set the threshold for triggering the alarm overload at 100 alarms per second:

```
WAE(config)# alarm overload-detect raise 100
```

The following example shows how to set the level for clearing the alarm overload at 10 alarms per second:

```
WAE(config)# alarm overload-detect clear 10
```

Related Commands

show alarms

(config) asset

To set the tag name for the asset tag string, use the **asset** global configuration command. To remove the asset tag name, use the **no** form of this command.

asset tag *name*

no asset tag *name*

Syntax Description

tag *name*	Sets the asset tag name.

Defaults

No default behaviors or values.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Examples

The following example shows how to configure a tag name for the asset tag string on a WAAS device:

```
WAE(config)# asset tag entitymib
```

(config) authentication configuration

To specify administrative login authorization parameters for a WAAS device, use the **authentication configuration** global configuration mode command. To selectively disable options, use the **no** form of this command.

authentication {**configuration** {**local** | **radius** | **tacacs** | **windows-domain**}
enable [**primary** | **secondary** | **tertiary** | **quaternary**]

no authentication {**configuration** {**local** | **radius** | **tacacs** | **windows-domain**}
enable [**primary** | **secondary** | **tertiary** | **quaternary**]

Syntax Description

configuration	Sets the administrative login authorization (configuration) parameters for the WAAS device.
local	Selects the local database method for the WAAS device.
radius	Selects the RADIUS method for the WAAS device.
tacacs	Selects the TACACS+ method for the WAAS device.
windows-domain	Selects the Windows domain controller method for the WAAS device.
enable	Enables the specified methods for the WAAS device.
primary	(Optional) Specifies the first method that the WAAS device should use.
secondary	(Optional) Specifies the second method that the WAAS device should use.
tertiary	(Optional) Specifies the third method that the WAAS device should use if the primary and secondary methods fail.
quaternary	(Optional) Specifies the fourth method that the WAAS device should use if the primary, secondary, and tertiary methods all fail.

Defaults

The local authentication method is enabled by default.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

The **authentication** command configures both the authentication and authorization methods that govern login and configuration access to the WAAS device.

Note We strongly recommend that you use the WAAS Central Manager GUI instead of the WAAS CLI to configure administrative login authentication and authorization for your WAAS devices, if possible. For information about how to use the WAAS Central Manager GUI to centrally configure administrative login authentication and authorization on a single WAE or group of WAEs, which are registered with a WAAS Central Manager, see the *Cisco Wide Area Application Services Configuration Guide*.

The **authentication login** command determines whether the user has any level of permission to access the WAAS device. The **authentication configuration** command authorizes the user with privileged access (configuration access) to the WAAS device.

The **authentication login local** and the **authentication configuration local** commands use a local database for authentication and authorization.

The **authentication login tacacs** and **authentication configuration tacacs** commands use a remote TACACS+ server to determine the level of user access. The WAAS software supports only TACACS+ and not TACACS or Extended TACACS.

To configure TACACS+, use the **authentication** and **tacacs** commands. To enable TACACS+, use the **tacacs enable** command. For more information on TACACS+ authentication, see the **(config) tacacs** command.

The **authentication login radius** and **authentication configuration radius** commands use a remote RADIUS server to determine the level of user access.

By default, the local method is enabled, with TACACS+ and RADIUS both disabled for login and configuration. Whenever TACACS+ and RADIUS are disabled the local method is automatically enabled. TACACS+, RADIUS, and local methods can be enabled at the same time.

The **primary** option specifies the first method to attempt for both login and configuration; the **secondary** option specifies the method to use if the primary method fails. The **tertiary** option specifies the method to use if both primary and secondary methods fail. The **quaternary** option specifies the method to use if the primary, secondary, and tertiary methods fail. If all methods of an **authentication login** or **authentication configuration** command are configured as primary, or all as secondary or tertiary, local is attempted first, then TACACS+, and then RADIUS.

Enforcing Authentication with the Primary Method

The **authentication fail-over server-unreachable** global configuration command allows you to specify that a failover to the secondary authentication method should occur only if the primary authentication server is unreachable. This feature ensures that users gain access to the WAAS device using the local database only when remote authentication servers (TACACS+ or RADIUS) are unreachable. For example, when a TACACS+ server is enabled for authentication with a user authentication failover configured and the user tries to log in to the WAAS device using an account defined in the local database, login fails. Login succeeds only when the TACACS+ server is unreachable.

You can configure multiple TACACS+ or RADIUS servers; authentication is attempted on the primary server first. If the primary server is unreachable, then authentication is attempted on the other servers in the TACACS+ or RADIUS farm, in order. If authentication fails for any reason other than a server is unreachable, authentication is not attempted on the other servers in the farm. This process applies regardless of the setting of the **authentication fail-over server-unreachable** command.

Login Authentication and Authorization Through the Local Database

Local authentication and authorization uses locally configured login and passwords to authenticate administrative login attempts. The login and passwords are local to each WAAS device and are not mapped to individual usernames.

By default, local login authentication is enabled first. You can disable local login authentication only after enabling one or more of the other administrative login authentication methods. However, when local login authentication is disabled, if you disable all other administrative login authentication methods, local login authentication is reenabled automatically.

Specifying RADIUS Authentication and Authorization Settings

To configure RADIUS authentication on a WAAS device, you must first configure a set of RADIUS authentication server settings on the WAAS device by using the **radius-server** global configuration command. (See the **(config) radius-server** command.)

Use the **authentication login radius** global configuration command to enable RADIUS authentication for normal login mode.

Use the **authentication configuration radius** global configuration command to enable RADIUS authorization.

To disable RADIUS authentication and authorization on a WAAS device, use the **no** form of the **authentication** global configuration command (for example, use the **no authentication login radius enable** command to disable RADIUS authentication).

Specifying TACACS+ Authentication and Authorization Settings

To configure TACACS+ authentication on WAAS devices, you must configure a set of TACACS+ authentication settings on the WAAS device by using the **tacacs** global configuration command. (See the **(config) tacacs** command.)

Server Redundancy

Authentication servers can be specified with the **tacacs host** or **radius-server host** global configuration commands. In the case of TACACS+ servers, the **tacacs host** *hostname* command can be used to configure additional servers. These additional servers provide authentication redundancy and improved throughput, especially when WAAS device load-balancing schemes distribute the requests evenly between the servers. If the WAAS device cannot connect to any of the authentication servers, no authentication takes place and users who have not been previously authenticated are denied access. Secondary authentication servers are queried in order only if the primary server is unreachable. If authentication fails for any other reason, alternate servers are not queried.

Specifying the Windows Domain Login Authentication

You can enable the Windows domain as an administrative login authentication and authorization method for a device or device group. Before you enable Windows authentication, you must first configure the Windows domain controller by using the **windows-domain wins-server** global configuration command. (See the **(config) windows-domain** command.)

Note WAAS supports authentication by a Windows domain controller running only on Windows Server 2000 or Windows Server 2003.

Examples

The following example shows how to query the secondary authentication database if the primary authentication server is unreachable. This feature is referred to as the failover server-unreachable feature.

```
WAE(config)# authentication fail-over server-unreachable
```

If you enable the failover server-unreachable feature on the WAAS device, only two login authentication schemes (a primary and secondary scheme) can be configured on the WAAS device. The WAAS device fails over from the primary authentication scheme to the secondary authentication scheme only if the specified authentication server is unreachable.

To enable authentication privileges using the local, TACACS+, RADIUS, or Windows databases, and to specify the order of the administrative login authentication, use the **authentication login** global configuration command. In the following example, RADIUS is specified as the primary method, TACACS+ as the secondary method, Windows as the third method, and the local database as the fourth method. In this example, four login authentication methods are specified because the failover server-unreachable feature is not enabled on the WAAS device.

```
WAE(config)# authentication login radius enable primary
WAE(config)# authentication login tacacs enable secondary
WAE(config)# authentication login windows-domain enable tertiary
WAE(config)# authentication login local enable quaternary
```

Note If you enable the failover server unreachable feature on the WAAS device, make sure that you specify either **TACACS+ or RADIUS** as the primary scheme for authentication, and specify **local** as the secondary scheme for authentication.

To enable authorization privileges using the local, TACACS+, RADIUS, or Windows databases, and to specify the order of the administrative login authorization (configuration), use the **authentication configuration** global configuration command.

Note Authorization privileges apply to console and Telnet connection attempts, secure FTP (SFTP) sessions, and Secure Shell (SSH Version 2) sessions.

We strongly recommend that you set the administrative login authentication and authorization methods in the same order. For example, configure the WAAS device to use RADIUS as the primary login method, TACACS+ as the secondary login method, Windows as the tertiary method, and the local method as the quaternary method for both administrative login authentication and authorization.

The following example shows that RADIUS is specified as the primary method, TACACS+ as the secondary method, Windows as the third method, and the local database as the fourth method. In this example, four login authorization (configuration) methods are specified because the failover server-unreachable feature is not enabled on the WAAS device.

```
WAE(config)# authentication configuration radius enable primary
WAE(config)# authentication configuration tacacs enable secondary
WAE(config)# authentication configuration windows-domain enable tertiary
WAE(config)# authentication configuration local enable quaternary
```

Note If you enable the failover server unreachable feature on the WAAS device, make sure that you specify either **TACACS+** or **RADIUS** as the primary scheme for authorization (configuration), and specify **local** as the secondary scheme for authorization (configuration).

The following example shows the resulting output of the **show authentication** command:

```
WAE# show authentication user

Login Authentication:         Console/Telnet/Ftp/SSH Session
----------------------------- ------------------------------
local                            enabled (primary)
Windows domain              enabled
Radius                           disabled
Tacacs+                          disabled
```

```
Configuration Authentication: Console/Telnet/Ftp/SSH Session
----------------------------- -------------------------------
local                         enabled (primary)
Radius                        disabled
Tacacs+                       disabled
```

Related Commands

(config) radius-server

show authentication

show statistics radius

show statistics tacacs

(config) tacacs

windows-domain

(config) windows-domain

(config)authentication enable

To configure “enable authentication” to use local "admin" user account password instead of using external authentication servers, use the **authentication enable** global configuration mode command. To disable this, use the **no** form of the command.

authentication enable local

no authentication enable local

Syntax Description

local	Selects the local admin user account password to enable authentication information for the WAAS device.

Defaults

When this command is configured, the local admin user account password is used for enable authentication by default.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

When a user who does not have privileged EXEC level types "enable" at the WAE>prompt, the request for enable access is not sent to the external authentication servers, but is processed on the WAE, using only the local admin user account password to verify the given password and provide access.

Note Critical commands (e.g. configuration and management) require that the user be at the privileged EXEC level. To change to the privileged EXEC level, type "enable" at the WAE> prompt.

Examples

The following example shows how to configure enable authentication by using local admin user account password.

```
WAE(config)# authentication enable local.
```

Related Commands

(config) authentication configuration

show authentication

(config) authentication content-request

To authenticate a request for content, use the **authentication content-request** global configuration mode command. To selectively disable options, use the **no** form of this command.

authentication content-request windows-domain-ctrl disconnected-mode enable

no authentication content-request windows-domain-ctrl disconnected-mode enable

Syntax Description

windows-domain-ctrl	Selects a Windows domain controller for domain server authentication.
disconnected-mode enable	Enables authentication in the disconnected mode.

Defaults

The local authentication method is enabled by default.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

The **authentication** command configures both the authentication and authorization methods that govern login and configuration access to the WAAS device.

Note We strongly recommend that you use the WAAS Central Manager GUI instead of the WAAS CLI to configure administrative login authentication and authorization for your WAAS devices, if possible. For information about how to use the WAAS Central Manager GUI to centrally configure administrative login authentication and authorization on a single WAE or group of WAEs, which are registered with a WAAS Central Manager, see the *Cisco Wide Area Application Services Configuration Guide*.

The **authentication login** command determines whether the user has any level of permission to access the WAAS device. The **authentication configuration** command authorizes the user with privileged access (configuration access) to the WAAS device.

The **authentication login local** and the **authentication configuration local** commands use a local database for authentication and authorization.

The **authentication login tacacs** and **authentication configuration tacacs** commands use a remote TACACS+ server to determine the level of user access. The WAAS software supports only TACACS+ and not TACACS or Extended TACACS.

To configure TACACS+, use the **authentication** and **tacacs** commands. To enable TACACS+, use the **tacacs enable** command. For more information on TACACS+ authentication, see the **(config) tacacs** command.

The **authentication login radius** and **authentication configuration radius** commands use a remote RADIUS server to determine the level of user access.

By default, the local method is enabled, with TACACS+ and RADIUS both disabled for login and configuration. Whenever TACACS+ and RADIUS are disabled the local method is automatically enabled. TACACS+, RADIUS, and local methods can be enabled at the same time.

The **primary** option specifies the first method to attempt for both login and configuration; the **secondary** option specifies the method to use if the primary method fails. The **tertiary** option specifies the method to use if both primary and secondary methods fail. The **quaternary** option specifies the method to use if the primary, secondary, and tertiary methods fail. If all methods of an **authentication login** or **authentication configuration** command are configured as primary, or all as secondary or tertiary, local is attempted first, then TACACS+, and then RADIUS.

Enforcing Authentication with the Primary Method

The **authentication fail-over server-unreachable** global configuration command allows you to specify that a failover to the secondary authentication method should occur only if the primary authentication server is unreachable. This feature ensures that users gain access to the WAAS device using the local database only when remote authentication servers (TACACS+ or RADIUS) are unreachable. For example, when a TACACS+ server is enabled for authentication with a user authentication failover configured and the user tries to log in to the WAAS device using an account defined in the local database, login fails. Login succeeds only when the TACACS+ server is unreachable.

You can configure multiple TACACS+ or RADIUS servers; authentication is attempted on the primary server first. If the primary server is unreachable, then authentication is attempted on the other servers in the TACACS+ or RADIUS farm, in order. If authentication fails for any reason other than a server is unreachable, authentication is not attempted on the other servers in the farm. This process applies regardless of the setting of the **authentication fail-over server-unreachable** command.

Login Authentication and Authorization Through the Local Database

Local authentication and authorization uses locally configured login and passwords to authenticate administrative login attempts. The login and passwords are local to each WAAS device and are not mapped to individual usernames.

By default, local login authentication is enabled first. You can disable local login authentication only after enabling one or more of the other administrative login authentication methods. However, when local login authentication is disabled, if you disable all other administrative login authentication methods, local login authentication is reenabled automatically.

Specifying RADIUS Authentication and Authorization Settings

To configure RADIUS authentication on a WAAS device, you must first configure a set of RADIUS authentication server settings on the WAAS device by using the **radius-server** global configuration command. (See the **(config) radius-server** command.)

Use the **authentication login radius** global configuration command to enable RADIUS authentication for normal login mode.

Use the **authentication configuration radius** global configuration command to enable RADIUS authorization.

To disable RADIUS authentication and authorization on a WAAS device, use the **no** form of the **authentication** global configuration command (for example, use the **no authentication login radius enable** command to disable RADIUS authentication).

Specifying TACACS+ Authentication and Authorization Settings

To configure TACACS+ authentication on WAAS devices, you must configure a set of TACACS+ authentication settings on the WAAS device by using the **tacacs** global configuration command. (See the **(config) tacacs** command.)

Server Redundancy

Authentication servers can be specified with the **tacacs host** or **radius-server host** global configuration commands. In the case of TACACS+ servers, the **tacacs host** *hostname* command can be used to configure additional servers. These additional servers provide authentication redundancy and improved throughput, especially when WAAS device load-balancing schemes distribute the requests evenly between the servers. If the WAAS device cannot connect to any of the authentication servers, no authentication takes place and users who have not been previously authenticated are denied access. Secondary authentication servers are queried in order only if the primary server is unreachable. If authentication fails for any other reason, alternate servers are not queried.

Specifying the Windows Domain Login Authentication

You can enable the Windows domain as an administrative login authentication and authorization method for a device or device group. Before you enable Windows authentication, you must first configure the Windows domain controller by using the **windows-domain wins-server** global configuration command. (See the **(config) windows-domain** command.)

Note WAAS supports authentication by a Windows domain controller running only on Windows Server 2000 or Windows Server 2003.

Examples

The following example shows how to query the secondary authentication database if the primary authentication server is unreachable. This feature is referred to as the failover server-unreachable feature.

```
WAE(config)# authentication fail-over server-unreachable
```

If you enable the failover server-unreachable feature on the WAAS device, only two login authentication schemes (a primary and secondary scheme) can be configured on the WAAS device. The WAAS device fails over from the primary authentication scheme to the secondary authentication scheme only if the specified authentication server is unreachable.

To enable authentication privileges using the local, TACACS+, RADIUS, or Windows databases, and to specify the order of the administrative login authentication, use the **authentication login** global configuration command. In the following example, RADIUS is specified as the primary method, TACACS+ as the secondary method, Windows as the third method, and the local database as the fourth method. In this example, four login authentication methods are specified because the failover server-unreachable feature is not enabled on the WAAS device.

```
WAE(config)# authentication login radius enable primary
WAE(config)# authentication login tacacs enable secondary
WAE(config)# authentication login windows-domain enable tertiary
WAE(config)# authentication login local enable quaternary
```

Note If you enable the failover server unreachable feature on the WAAS device, make sure that you specify either **TACACS+ or RADIUS** as the primary scheme for authentication, and specify **local** as the secondary scheme for authentication.

To enable authorization privileges using the local, TACACS+, RADIUS, or Windows databases, and to specify the order of the administrative login authorization (configuration), use the **authentication configuration** global configuration command.

Note Authorization privileges apply to console and Telnet connection attempts, secure FTP (SFTP) sessions, and Secure Shell (SSH Version 2) sessions.

We strongly recommend that you set the administrative login authentication and authorization methods in the same order. For example, configure the WAAS device to use RADIUS as the primary login method, TACACS+ as the secondary login method, Windows as the tertiary method, and the local method as the quaternary method for both administrative login authentication and authorization.

The following example shows that RADIUS is specified as the primary method, TACACS+ as the secondary method, Windows as the third method, and the local database as the fourth method. In this example, four login authorization (configuration) methods are specified because the failover server-unreachable feature is not enabled on the WAAS device.

```
WAE(config)# authentication configuration radius enable primary
WAE(config)# authentication configuration tacacs enable secondary
WAE(config)# authentication configuration windows-domain enable tertiary
WAE(config)# authentication configuration local enable quaternary
```

Note If you enable the failover server unreachable feature on the WAAS device, make sure that you specify either **TACACS+** or **RADIUS** as the primary scheme for authorization (configuration), and specify **local** as the secondary scheme for authorization (configuration).

The following example shows the resulting output of the **show authentication** command:

```
WAE# show authentication user

Login Authentication:         Console/Telnet/Ftp/SSH Session
----------------------------- ------------------------------
local                         enabled (primary)
Windows domain              enabled
Radius                        disabled
Tacacs+                       disabled

Configuration Authentication: Console/Telnet/Ftp/SSH Session
----------------------------- ------------------------------
local                         enabled (primary)
Radius                        disabled
Tacacs+                       disabled
```

Related Commands

(config) radius-server

show authentication

show statistics radius

show statistics tacacs

(config) tacacs

windows-domain

(config) windows-domain

(config) authentication fail-over

To specify authentication failover if the primary authentication server is unreachable, use the **authentication fail-over** global configuration mode command. To disable this feature, use the **no** form of this command.

authentication fail-over server-unreachable

no authentication fail-over server-unreachable

Syntax Description

server-unreachable	Specifies that the WAAS device is to query the secondary authentication database only if the primary authentication server is unreachable.

Defaults

This feature is disabled by default. This means that the WAAS device tries the other authentication methods if the primary method fails for any reason, not just if the server is unreachable.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

The **authentication** command configures both the authentication and authorization methods that govern login and configuration access to the WAAS device.

Note We strongly recommend that you use the WAAS Central Manager GUI instead of the WAAS CLI to configure administrative login authentication and authorization for your WAAS devices, if possible. For information about how to use the WAAS Central Manager GUI to centrally configure administrative login authentication and authorization on a single WAE or group of WAEs, which are registered with a WAAS Central Manager, see the *Cisco Wide Area Application Services Configuration Guide*.

The **authentication fail-over server-unreachable** global configuration command allows you to specify that a failover to the secondary authentication method should occur only if the primary authentication server is unreachable. This feature ensures that users gain access to the WAAS device using the local database only when remote authentication servers (TACACS+ or RADIUS) are unreachable. For example, when a TACACS+ server is enabled for authentication with a user authentication failover configured and the user tries to log in to the WAAS device using an account defined in the local database, login fails. Login succeeds only when the TACACS+ server is unreachable.

You can configure multiple TACACS+ or RADIUS servers; authentication is attempted on the primary server first. If the primary server is unreachable, then authentication is attempted on the other servers in the TACACS+ or RADIUS farm, in order. If authentication fails for any reason other than a server is unreachable, authentication is not attempted on the other servers in the farm. This process applies regardless of the setting of the **authentication fail-over server-unreachable** command.

Examples

The following example shows how to query the secondary authentication database if the primary authentication server is unreachable. This feature is referred to as the failover server-unreachable feature.

```
WAE(config)# authentication fail-over server-unreachable
```

If you enable the failover server-unreachable feature on the WAAS device, only two login authentication schemes (a primary and secondary scheme) can be configured on the WAAS device. The WAAS device fails over from the primary authentication scheme to the secondary authentication scheme only if the specified authentication server is unreachable.

Note If you enable the failover server unreachable feature on the WAAS device, make sure that you specify either **TACACS+ or RADIUS** as the primary scheme for authentication, and specify **local** as the secondary scheme for authentication.

Related Commands

(config) radius-server

show authentication

show statistics radius

show statistics tacacs

(config) tacacs

windows-domain

(config) windows-domain

(config) authentication login

To set the administrative login authentication parameters for a WAAS device, use the **authentication login** global configuration mode command. To selectively disable options, use the **no** form of this command.

authentication login {**local** | **radius** | **tacacs** | **windows-domain**}
enable [**primary** | **secondary** | **tertiary**| **quaternary**]

no authentication login {**local** | **radius** | **tacacs** | **windows-domain**}
enable [**primary** | **secondary** | **tertiary**| **quaternary**]

Syntax Description

local	Selects the local database method for the WAAS device.
radius	Selects the RADIUS method for the WAAS device.
tacacs	Selects the TACACS+ method for the WAAS device.
windows-domain	Selects the Windows domain controller method for the WAAS device.
enable	Enables the specified methods for the WAAS device.
primary	(Optional) Specifies the first method that the WAAS device should use.
secondary	(Optional) Specifies the second method that the WAAS device should use.
tertiary	(Optional) Specifies the third method that the WAAS device should use if the primary and secondary methods fail.
quaternary	(Optional) Specifies the fourth method that the WAAS device should use if the primary, secondary, and tertiary methods all fail.

Defaults

The local authentication method is enabled by default.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

The **authentication** command configures both the authentication and authorization methods that govern login and configuration access to the WAAS device.

Note We strongly recommend that you use the WAAS Central Manager GUI instead of the WAAS CLI to configure administrative login authentication and authorization for your WAAS devices, if possible. For information about how to use the WAAS Central Manager GUI to centrally configure administrative login authentication and authorization on a single WAE or group of WAEs, which are registered with a WAAS Central Manager, see the *Cisco Wide Area Application Services Configuration Guide*.

The **authentication login** command determines whether the user has any level of permission to access the WAAS device. The **authentication configuration** command authorizes the user with privileged access (configuration access) to the WAAS device.

The **authentication login local** and the **authentication configuration local** commands use a local database for authentication and authorization.

The **authentication login tacacs** and **authentication configuration tacacs** commands use a remote TACACS+ server to determine the level of user access. The WAAS software supports only TACACS+ and not TACACS or Extended TACACS.

To configure TACACS+, use the **authentication** and **tacacs** commands. To enable TACACS+, use the **tacacs enable** command. For more information on TACACS+ authentication, see the **(config) tacacs** command.

The **authentication login radius** and **authentication configuration radius** commands use a remote RADIUS server to determine the level of user access.

By default, the local method is enabled, with TACACS+ and RADIUS both disabled for login and configuration. Whenever TACACS+ and RADIUS are disabled the local method is automatically enabled. TACACS+, RADIUS, and local methods can be enabled at the same time.

The **primary** option specifies the first method to attempt for both login and configuration; the **secondary** option specifies the method to use if the primary method fails. The **tertiary** option specifies the method to use if both primary and secondary methods fail. The **quaternary** option specifies the method to use if the primary, secondary, and tertiary methods fail. If all methods of an **authentication login** or **authentication configuration** command are configured as primary, or all as secondary or tertiary, local is attempted first, then TACACS+, and then RADIUS.

Enforcing Authentication with the Primary Method

The **authentication fail-over server-unreachable** global configuration command allows you to specify that a failover to the secondary authentication method should occur only if the primary authentication server is unreachable. This feature ensures that users gain access to the WAAS device using the local database only when remote authentication servers (TACACS+ or RADIUS) are unreachable. For example, when a TACACS+ server is enabled for authentication with a user authentication failover configured and the user tries to log in to the WAAS device using an account defined in the local database, login fails. Login succeeds only when the TACACS+ server is unreachable.

You can configure multiple TACACS+ or RADIUS servers; authentication is attempted on the primary server first. If the primary server is unreachable, then authentication is attempted on the other servers in the TACACS+ or RADIUS farm, in order. If authentication fails for any reason other than a server is unreachable, authentication is not attempted on the other servers in the farm. This process applies regardless of the setting of the **authentication fail-over server-unreachable** command.

Login Authentication and Authorization Through the Local Database

Local authentication and authorization uses locally configured login and passwords to authenticate administrative login attempts. The login and passwords are local to each WAAS device and are not mapped to individual usernames.

By default, local login authentication is enabled first. You can disable local login authentication only after enabling one or more of the other administrative login authentication methods. However, when local login authentication is disabled, if you disable all other administrative login authentication methods, local login authentication is reenabled automatically.

Specifying RADIUS Authentication and Authorization Settings

To configure RADIUS authentication on a WAAS device, you must first configure a set of RADIUS authentication server settings on the WAAS device by using the **radius-server** global configuration command. (See the **(config) radius-server** command.)

Use the **authentication login radius** global configuration command to enable RADIUS authentication for normal login mode.

Use the **authentication configuration radius** global configuration command to enable RADIUS authorization.

To disable RADIUS authentication and authorization on a WAAS device, use the **no** form of the **authentication** global configuration command (for example, use the **no authentication login radius enable** command to disable RADIUS authentication).

Specifying TACACS+ Authentication and Authorization Settings

To configure TACACS+ authentication on WAAS devices, you must configure a set of TACACS+ authentication settings on the WAAS device by using the **tacacs** global configuration command. (See the **(config) tacacs** command.)

Server Redundancy

Authentication servers can be specified with the **tacacs host** or **radius-server host** global configuration commands. In the case of TACACS+ servers, the **tacacs host** *hostname* command can be used to configure additional servers. These additional servers provide authentication redundancy and improved throughput, especially when WAAS device load-balancing schemes distribute the requests evenly between the servers. If the WAAS device cannot connect to any of the authentication servers, no authentication takes place and users who have not been previously authenticated are denied access. Secondary authentication servers are queried in order only if the primary server is unreachable. If authentication fails for any other reason, alternate servers are not queried.

Specifying the Windows Domain Login Authentication

You can enable the Windows domain as an administrative login authentication and authorization method for a device or device group. Before you enable Windows authentication, you must first configure the Windows domain controller by using the **windows-domain wins-server** global configuration command. (See the **(config) windows-domain** command.)

Note WAAS supports authentication by a Windows domain controller running only on Windows Server 2000 or Windows Server 2003.

Examples

The following example shows how to query the secondary authentication database if the primary authentication server is unreachable. This feature is referred to as the failover server-unreachable feature.

```
WAE(config)# authentication fail-over server-unreachable
```

If you enable the failover server-unreachable feature on the WAAS device, only two login authentication schemes (a primary and secondary scheme) can be configured on the WAAS device. The WAAS device fails over from the primary authentication scheme to the secondary authentication scheme only if the specified authentication server is unreachable.

To enable authentication privileges using the local, TACACS+, RADIUS, or Windows databases, and to specify the order of the administrative login authentication, use the **authentication login** global configuration command. In the following example, RADIUS is specified as the primary method, TACACS+ as the secondary method, Windows as the third method, and the local database as the fourth method. In this example, four login authentication methods are specified because the failover server-unreachable feature is not enabled on the WAAS device.

```
WAE(config)# authentication login radius enable primary
WAE(config)# authentication login tacacs enable secondary
WAE(config)# authentication login windows-domain enable tertiary
WAE(config)# authentication login local enable quaternary
```

Note If you enable the failover server unreachable feature on the WAAS device, make sure that you specify either **TACACS+ or RADIUS** as the primary scheme for authentication, and specify **local** as the secondary scheme for authentication.

To enable authorization privileges using the local, TACACS+, RADIUS, or Windows databases, and to specify the order of the administrative login authorization (configuration), use the **authentication configuration** global configuration command.

Note Authorization privileges apply to console and Telnet connection attempts, secure FTP (SFTP) sessions, and Secure Shell (SSH Version 2) sessions.

We strongly recommend that you set the administrative login authentication and authorization methods in the same order. For example, configure the WAAS device to use RADIUS as the primary login method, TACACS+ as the secondary login method, Windows as the tertiary method, and the local method as the quaternary method for both administrative login authentication and authorization.

The following example shows that RADIUS is specified as the primary method, TACACS+ as the secondary method, Windows as the third method, and the local database as the fourth method. In this example, four login authorization (configuration) methods are specified because the failover server-unreachable feature is not enabled on the WAAS device.

```
WAE(config)# authentication configuration radius enable primary
WAE(config)# authentication configuration tacacs enable secondary
WAE(config)# authentication configuration windows-domain enable tertiary
WAE(config)# authentication configuration local enable quaternary
```

Note If you enable the failover server unreachable feature on the WAAS device, make sure that you specify either **TACACS+** or **RADIUS** as the primary scheme for authorization (configuration), and specify **local** as the secondary scheme for authorization (configuration).

The following example shows the resulting output of the **show authentication** command:

```
WAE# show authentication user

Login Authentication:         Console/Telnet/Ftp/SSH Session
----------------------------- ------------------------------
local                            enabled (primary)
Windows domain              enabled
Radius                           disabled
Tacacs+                          disabled
```

```
Configuration Authentication: Console/Telnet/Ftp/SSH Session
------------------------------ ------------------------------
local                          enabled (primary)
Radius                         disabled
Tacacs+                        disabled
```

Related Commands

(config) radius-server

show authentication

show statistics radius

show statistics tacacs

(config) tacacs

windows-domain

(config) windows-domain

(config) authentication strict-password-policy

To activate the strong password policy on a WAAS device, use the **authentication strict-password-policy** global configuration command. To deactivate the strong password policy and use the standard password policy on a WAAS device, use the **no** form of this command.

authentication strict-password-policy [**max-retry-attempts** *number*]

no authentication strict-password-policy [**max-retry-attempts** *number*]

Syntax Description

max-retry-attempts *number*	(Optional) Specifies the maximum number of failed login attempts allowed before the user is locked out. The range is 1–25; the default is 3.

Defaults

The strong password policy is enabled on the WAAS device.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

When you enable the strong password policy, your user passwords must meet the following requirements:

- The password must be 8 to 31 characters long.
- The password can include both uppercase and lowercase letters (A–Z and a–z), numbers (0–9), and special characters including ~`!@#$%^&*()_+-=[]\{};:,</>.
- The password cannot contain all the same characters (for example, 99999).
- The password cannot contain consecutive characters (for example, 12345).
- The password cannot be the same as the username.
- Each new password must be different from the previous 12 passwords. User passwords expire within 90 days.
- The password cannot contain the characters ' " | (apostrophe, double quote, or pipe) or any control characters.
- The password cannot contain dictionary words.

When you disable the strong password policy, user passwords must meet the following requirements:

- The password must have 1 to 31 characters.
- The password can include both uppercase and lowercase letters (A–Z and a–z), and numbers (0–9).
- The password cannot contain the characters ' " | (apostrophe, double quote, or pipe) or any control characters.

Note When you enable the strong password policy, existing standard-policy passwords will still work. However, these passwords are subject to expiration under the strong password policy.

Examples

The following example shows how to enable the strong password policy:

```
WAE(config)# authentication strict-password-policy
```

The following example shows how to enable the strong password policy and set the maximum retry attempts to 5:

```
WAE(config)# authentication strict-password-policy max-retry-attempts 5
```

The following example shows how to disable the strong password policy:

```
WAE(config)# no authentication strict-password-policy
```

Related Commands

clear users

show authentication

(config) authentication configuration

(config) auto-discovery

To configure a WAE to automatically discover origin servers (such as those servers behind firewalls) that cannot receive TCP packets with setup options and add these server IP addresses to a blacklist for a specified number of minutes, use the **auto-discovery** global configuration command. To disable auto-discovery, use the **no** form of this command.

auto-discovery blacklist {**enable** | **hold-time** *minutes*}

no auto-discovery blacklist {**enable** | **hold-time** *minutes*}

Syntax Description

blacklist	Specifies the TFO auto-discovery blacklist server configuration.
enable	Enables the TFO auto-discovery blacklist operation.
hold-time *minutes*	Specifies the maximum time to hold the blacklisted server address in the cache. The range is 1–10080 minutes. The default is 60 minutes.

Defaults

The default auto-discovery blacklist hold time is 60 minutes.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

Use the **auto-discovery blacklist hold-time** command to adjust the blacklist hold time for the TFO auto-discovery feature. With auto-discovery, the WAE keeps track of origin servers (such as those servers behind firewalls) that cannot receive TCP packets with options and learns not to send out TCP packets with options to these blacklisted servers. When a server IP address is added to the blacklist, it remains on the blacklist for the configured number of minutes. After the hold time expires, subsequent connection attempts will again include TCP options so that the WAE can redetermine if the server can receive them. Resending TCP options periodically is useful because network packet loss could cause a server to be blacklisted erroneously.

Examples

The following example shows how to enable TFO auto-discovery blacklist using the **auto-discovery** command:

```
WAE(config)# auto-discovery blacklist enable
```

Related Commands

show statistics auto-discovery

(config) auto-register

To enable the discovery of a WAE and its automatic registration with the WAAS Central Manager through the Dynamic Host Configuration Protocol (DHCP), use the **auto-register** global configuration command. To disable the autoregistration feature on a WAE, use the **no** form of this command.

auto-register enable [**FastEthernet** *slot/port* | **GigabitEthernet** *slot/port* | **TenGigabitEthernet** *slot/port*]

no auto-register enable [**FastEthernet** *slot/port* | **GigabitEthernet** *slot/port* | **TenGigabitEthernet** *slot/port*] [**preserve-ip**]

Syntax Description

enable	Enables the automatic registration of devices using DHCP with the WAAS Central Manager.
FastEthernet *slot/port*	(Optional) Selects a Fast Ethernet interface for automatic registration using DHCP. Selects slot number and port number of the Fast Ethernet interface. Valid slot values depend on the hardware platform.
GigabitEthernet *slot/port*	(Optional) Selects a Gigabit Ethernet interface for automatic registration using DHCP. Selects slot number and port number of the Gigabit Ethernet interface. Valid slot values depend on the hardware platform.
TenGigabitEthernet *slot/port*	(Optional) Selects a TenGigabitEthernet interface for automatic registration using DHCP. Selects slot number and port number of the 10-Gigabit Ethernet interface. Valid slot values depend on the hardware platform.
preserve-ip	(Optional) Converts a dynamic IP address to a static IP address when you remove the automatic registration from an interface so that the interface remains configured with an IP address.

Defaults

Automatic registration using DHCP is enabled on a WAE by default.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

Usage Guidelines

Autoregistration automatically configures network settings and registers WAEs with the WAAS Central Manager. On bootup, devices that run the WAAS software (with the exception of the WAAS Central Manager) automatically discover the WAAS Central Manager and register with it. You do not have to do any manual configuration on the device. Once the WAE is registered, you can approve the device and configure it remotely using the WAAS Central Manager GUI.

You can use the **auto-register enable** command to allow a WAE to discover the hostname of the WAAS Central Manager through DHCP and to automatically register the device with the WAAS Central Manager. Discovery and registration occur at bootup.

Note You must disable autoregistration when both device interfaces are configured as port-channel interfaces.

Note The DHCP that is used for autoregistration is *not* the same as the interface-level DHCP that is configurable through the **ip address dhcp** interface configuration command.

To assign a static IP address using the **interface** command, you must first disable the automatic registration of devices through DHCP by using the **no auto-register enable** command. If you want to keep the dynamic IP address that had been assigned to the interface, use the **preserve-ip** option to convert it to a static IP address.

After the WAE configures its network settings from DHCP, it needs to know the Central Manager hostname so it can register with the Central Manager.

The WAE queries the DNS server to obtain the Central Manager hostname. For autoregistration to work, you must configure the DNS server with the Central Manager hostname by configuring a DNS SRV (Service Location) record. For more information about autoregistration and how to configure the DNS SRV record, see the section on autoregistration in the "Planning Your WAAS Network" chapter of the *Cisco Wide Area Application Services Configuration Guide*.

Examples

The following example shows how to enable autoregistration on GigabitEthernet port 1/0:

```
WAE(config)# auto-register enable GigabitEthernet 1/0
```

The following example shows how to disable autoregistration on all configured interfaces on the WAE without losing any IP addresses assigned by autoregistration DHCP:

```
WAE(config)# no auto-register enable preserve-ip
```

Related Commands

show auto-register

show running-config

show startup-config

(config) banner

To configure the EXEC, login, and message-of-the-day (MOTD) banners, use the **banner** global configuration command. To disable the banner feature, use the **no** form of this command.

banner {**enable** | {{**exec** | **login** | **motd**} [**message** *text*]}}

no banner {**enable** | {{**exec** | **login** | **motd**} [**message** *text*]}}

Syntax Description

enable	Enables banner support on the WAE.
exec	Configures an EXEC banner.
login	Configures a login banner.
motd	Configures an MOTD banner.
message *text*	(Optional) Specifies a message to be displayed when an EXEC process is created. The message text is on a single line (980 characters maximum). The WAE translates the \n portion of the message to a new line when the banner is displayed to the user.

Defaults

Banner support is disabled by default.

Command Modes

global configuration

Usage Guidelines

The **message** keyword is optional. If you enter a carriage return without specifying the **message** keyword, you will be prompted to enter your message text. For message text on one or more lines, press the **Return** key or enter delimiting characters (\n) to specify a message to appear on a new line. You can enter up to a maximum of 980 characters, including new-line characters (\n). Enter a period (.) at the beginning of a new line to save the message and return to the prompt for the global configuration mode.

Note The EXEC banner content is obtained from the command-line input that you enter when prompted for the input.

After you configure the banners, enter the **banner enable** global configuration command to enable banner support on the appliance. Enter the **show banner** EXEC command to display information about the configured banners.

Examples

The following example shows how to use the **banner motd message** global configuration command to configure the MOTD banner. In this example, the MOTD message consists of a single line of text.

```
WAE(config)# banner motd message This is a WAAS 4.0.7 device
```

The following example shows how to use the **banner motd message** global command to configure a MOTD message that is longer than a single line. In this case, the WAE translates the \n portion of the message to a new line when the MOTD message is displayed to the user.

```
WAE(config)# banner motd message "This is the motd message.
\nThis is a WAAS 4.0.7 device\n"
```

The following example shows how to use the **banner login message** global configuration command to configure a login message that is longer than a single line. In this case, WAE A translates the \n portion of the message to a new line in the login message that is displayed to the user.

```
WAE(config)# banner login message "This is login banner.
\nUse your password to login\n"
```

The following example shows how to enable banner support:

```
WAE(config)# banner enable
```

The following example shows how to use the **banner exec** global configuration command to configure an interactive banner. The **banner exec** command is similar to the **banner motd message** commands except that for the **banner exec** command, the banner content is obtained from the command-line input that the user enters after being prompted for the input.

```
WAE(config)# banner exec
Please type your MOTD messages below and end it with '.' at beginning of line:
(plain text only, no longer than 980 bytes including newline)
This is the EXEC banner.\nUse your WAAS username and password to log in to this WAE.\n
.
Message has 99 characters.
WAE(config)#
```

Assume that a WAE has been configured with the MOTD, login, and EXEC banners as shown in the previous examples. When a user uses an SSH session to log in to the WAE, the user will see a login session that includes a MOTD banner and a login banner that asks the user to enter a login password as follows:

```
This is the motd banner.
This is a WAAS 4.0.7 device
This is login banner.
Use your password to login.

Cisco Wide Area Application Services Engine

admin@wae's password:
```

After the user enters a valid login password, the EXEC banner is displayed, and the user is asked to enter the WAAS username and password as follows:

```
Last login: Fri Oct  1 14:54:03 2004 from client
System Initialization Finished.
This is the EXEC banner.
Use your WAAS username and password to log in to this WAE.
```

After the user enters a valid WAAS username and password, the WAE CLI is displayed. The CLI prompt varies depending on the privilege level of the login account. In the following example, because the user entered a username and password that had administrative privileges (privilege level of 15), the EXEC mode CLI prompt is displayed:

```
WAE#
```

Related Commands

show banner

(config) bridge

To configure a bridge group for use by a virtual blade or by inline interfaces on an AppNav Controller Interface Module, use the **bridge** global configuration command. To unconfigure the bridge group, use the **no** form of this command.

bridge *bridge-id* {**protocol** {**ieee** | **interception**}| **description** *description* | **intercept vlan** {**add** | **except** | **remove**} {**all** | **native** | *list*}| **propagate-link-state**}

no bridge *bridge-id* {**protocol** {**ieee** | **interception**}| **description** *description* | **intercept vlan** {**add** | **except** | **remove**} {**all** | **native** | *list*}| **propagate-link-state**}

Syntax Description

bridge-id	Bridge ID from 1–4. On devices with an AppNav Controller Interface Module that has 12 ports, the bridge ID ranges from 1–5. On devices with an AppNav Controller Interface Module that has 4 ports, the bridge ID must be 1.
protocol	Defines the protocol.
ieee	Specifies the IEEE protocol, used for a virtual blade bridge group. This option is not available on devices operating in appnav-controller mode.
interception	Specifies the interception protocol, used for an inline group interception interface on an AppNav Controller Interface Module. This option is available only on devices operating in appnav-controller mode.
description *description*	(Optional) Specifies a description of the bridge group with up to 200 alphanumeric and space characters. This option is available only on devices operating in appnav-controller mode.
intercept vlan	(Optional) Configures VLANS that this bridge group is to intercept. This option is available only on devices operating in appnav-controller mode.
add	(Optional) Adds VLANS to the list of VLANS that this bridge group is to intercept. This option is available only on devices operating in appnav-controller mode.
except	(Optional) Adds VLANS to the list of VLANS that this bridge group is to intercept. All VLANS (1-4095) are added to the list except those specified with this keyword. This option is available only on devices operating in appnav-controller mode.
remove	(Optional) Removes VLANS from the list of VLANS that this bridge group is to intercept. This option is available only on devices operating in appnav-controller mode.
all	(Optional) Specifies that all VLANS are to be added or removed from the list. This option is available only on devices operating in appnav-controller mode.
native	(Optional) Specifies that the native VLAN is to be added or removed from the list. This option is available only on devices operating in appnav-controller mode.

list	(Optional) Specifies a comma separated list of VLANS or VLAN ranges that are to be added or removed from the list. Valid values range from 1-4094 and include **native** and **all**. This option is available only on devices operating in appnav-controller mode.
propagate-link-state	(Optional) Enables or disables link state propagation on inline bridge group interfaces. This option is available only on devices operating in appnav-controller mode. When link state propagation is enabled and one interface in the bridge group goes down, the system automatically shuts down the other interface.

Defaults

For an inline bridge group, all VLANS are intercepted and link state propagation is enabled.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

Usage Guidelines

This command can create a bridge group for bridging to a virtual blade, by using the **protocol ieee** option. After using this command, create a bridge virtual interface in the bridge group by using the **interface bvi** global configuration command. You must add one physical, port-channel, or standby interface to the bridge group, along with the BVI interface.

This command can also create a bridge group for bridging two inline interfaces for interception on an AppNav Controller Interface Module in an AppNav deployment, by using the **protocol interception** option. The two member interfaces of this kind of bridge group do not need to be similar. For example, one could be a physical interface and one could be a port-channel interface. A standby interface is not allowed in a bridge group for inline interception on an AppNav Controller Interface Module.

Examples

The following example shows how to create and configure a bridge interface for a virtual blade:

```
WAE# configure
WAE(config)# bridge 1 protocol ieee
WAE(config)# interface GigabitEthernet 1/0 bridge-group 1
WAE(config)# interface bvi 1 ip address 10.10.10.10 255.0.0.0
WAE(config)# virtual-blade 2
WAE(config-vb)# interface 1 bridge-group 1
```

The following example shows how to remove a bridge virtual interface:

```
WAE(config)# no bridge 1 protocol ieee
```

The following example shows how to create and configure a bridge group for inline interfaces on an AppNav Controller Interface Module:

```
WAE# configure
WAE(config)# interception-method inline
WAE(config)# bridge 5 protocol interception
WAE(config)# bridge 5 intercept vlan add 100-200,300-350,native
WAE(config)# interface GigabitEthernet 1/10 bridge-group 5
```

```
WAE(config)# interface GigabitEthernet 1/11 bridge-group 5
```

Related Commands

(config) interception-method

(config) interface bvi

show bridge

(config) cdp

To configure the Cisco Discovery Protocol (CDP) options globally on all WAAS device interfaces, use the **cdp** global configuration command. To disable CDP, use the **no** form of this command.

cdp {**enable** | **holdtime** *seconds* | **timer** *seconds*}

no cdp {**enable** | **holdtime** *seconds* | **timer** *seconds*}

Syntax Description

enable	Enables CDP globally.
holdtime *seconds*	Sets the length of time in seconds (10–255) that a receiver keeps CDP packets before they are discarded. The default is 180 seconds.
timer *seconds*	Sets the interval between the CDP advertisements in seconds (5–254). The default is 60 seconds.

Defaults

holdtime: 180 seconds

timer: 60 seconds

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Examples

The following example shows that when CDP is first enabled, the hold time is set to 10 seconds for keeping CDP packets, and then the rate at which CDP packets are sent (15 seconds) is set:

```
WAE(config)# cdp enable
WAE(config)# cdp holdtime 10
WAE(config)# cdp timer 15
```

Related Commands

(config-if) cdp

clear arp-cache

show cdp

(config) central-manager

To specify the WAAS Central Manager role and port number, use the **central-manager** global configuration command in central-manager device mode. To specify the IP address or hostname of the WAAS Central Manager with which a WAE is to register, use the **central-manager** global configuration command in application-accelerator device mode. To negate these actions, use the **no** form of this command.

central-manager {**address** {*hostname* | *ip-address*} | **role** {**primary** | **standby**} | **ui port** *port-num*}

no central-manager {**address** {*hostname* | *ip-address*} | **role** {**primary** | **standby**} | **ui port** *port-num*}

Syntax Description

address	Specifies the hostname or IP address of the WAAS Central Manager with which the WAE should register.
hostname	Hostname of the WAAS Central Manager with which the WAE should register.
ip-address	IP address of the WAAS Central Manager with which the WAE should register.
role	Configures the WAAS Central Manager role to either primary or standby.
primary	Configures the WAAS Central Manager to be the primary WAAS Central Manager for the WAEs that are registered with it.
standby	Configures the WAAS Central Manager to be the standby WAAS Central Manager for the WAEs that are registered with it.
ui	Configures the WAAS Central Manager GUI port address.
port *port-num*	Configures the WAAS Central Manager GUI port (1–65535). The default is port 8443.

Note The **address** option works in the application-accelerator device mode only. The **role** and **ui port** options work in the central-manager device mode only.

Defaults

The WAAS Central Manager GUI is preconfigured to use port 8443.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Examples

The following example shows how to specify that the WAAS device named waas-cm is to function as the primary WAAS Central Manager for the WAAS network:

```
waas-cm(config)# central-manager role primary
```

The following example shows how to specify that the WAE should register with the WAAS Central Manager that has an IP address of 10.1.1.1. This command associates the WAE with the primary WAAS Central Manager so that the WAE can be approved as a part of the WAAS network.

```
WAE(config)# central-manager address 10.1.1.1
```

The following example shows how to configure a new GUI port to access the WAAS Central Manager GUI:

```
WAE(config)# central-manager ui port 8550
```

The following example shows how to configure the WAAS Central Manager as the standby WAAS Central Manager:

```
WAE(config)# central-manager role standby
Switching CDM to standby will cause  all configuration settings made on this CDM to be
lost.
Please confirm you want to continue  [no]?yes
Restarting CMS services
```

(config) class-map

To configure an AppNav or optimization class map, use the **class-map** global configuration command. To unconfigure settings, use the **no** form of this command.

class-map type {**appnav** | **waas**} [**match-all** | **match-any**] *classmap-name* [**rename** *new-name*]

no class-map type {**appnav** | **waas**} [**match-all** | **match-any**] *classmap-name*

Syntax Description

appnav	Configures an AppNav class map.
waas	Configures a WAAS optimization class map.
match-all	(Optional) Specifies that all match conditions must be satisfied to consider the class map matched (logical AND). Valid only on AppNav class maps.
match-any	(Optional) Specifies that any match condition must be satisfied to consider the class map matched (logical OR).
classmap-name	Class map name for AppNav (up to 40 alpha-numeric characters and hyphen, beginning with a letter). Class map name for AppNav-XE (up to 40 alpha-numeric characters and special characters including !@#$%^&*()_+-=[]{}|\ ).
rename *new-name*	(Optional) Renames the class map with the specified new name.

Defaults

For AppNav class maps, match-all is the default when multiple match criteria exist.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

Usage Guidelines

Use the **class-map** command to add or modify class maps and match conditions to identify specific types of traffic for use in policies. This command invokes the Class Map configuration mode, which is indicated by a different prompt (config-cmap). For more information on Class Map configuration mode commands, see the "Class Map Configuration Mode Commands" section. To return to global configuration mode, enter the **exit** command.

You can delete a class map by using the **no** form of this command. You cannot delete a class map if any policies are using it.

When creating a new class map, you must add at least one condition. If any of the conditions specified match an already existing condition in the class-map, no action is taken.

Note You cannot have more than 512 different class maps and 1024 total match conditions.

The WAAS software comes with many class maps and policy rules that help your WAAS system classify and optimize some of the most common traffic on your network. Before you create a new class map or policy rule, we recommend that you review the default class map and policy rules and modify them as appropriate. It is usually easier to modify an existing class map or policy rule than to create a new one. For a list of the default applications, class maps, and policy rules, see the *Cisco Wide Area Application Services Configuration Guide*.

Note We strongly recommend that you use the WAAS Central Manager GUI to centrally configure class maps for your WAAS devices. For more information, see the *Cisco Wide Area Application Services Configuration Guide*.

There is one exception this recommendation. Use the CLI to create an AppNav class map with type Application or Custom, and whose source or destination address has one of the following: an IP address ending in “0.0.0” or a non-Class A IP address ending in “0.0”.

Examples

The following example shows how to configure a WAAS optimization class map:

```
wae(config)# class-map type waas myclass1
wae(config-cmap)# description My class number one
wae(config-cmap)# match protocol mapi tcp source ip 10.10.10.35
wae(config-cmap)# exit
```

The following example shows how to configure an AppNav class map:

```
wae(config)# class-map type appnav myclass1
wae(config-cmap)# match peer 50:3d:e5:9c:8f:aa description SanJose_branch
wae(config-cmap)# exit
```

Related Commands

(config) policy-map

(config) clock

To set the summer daylight saving time and time zone for display purposes, use the **clock** global configuration command. To disable this function, use the **no** form of this command.

clock {**timezone** *timezone hoursoffset* [*minutesoffset*]} |
summertime *timezone* {**date** *startday startmonth startyear starthour endday endmonth endyear offset* | **recurring** {**1-4** *startweekday startmonth starthour endweekday endmonth endhour offset* | **first** *startweekday startmonth starthour endweekday endmonth endhour offset* | **last** *startweekday startmonth starthour endweekday endmonth endhour offset*}}

no clock {**timezone** *timezone hoursoffset* [*minutesoffset*]} |
summertime *timezone* {**date** *startday startmonth startyear starthour endday endmonth endyear offset* | **recurring** {**1-4** *startweekday startmonth starthour endweekday endmonth endhour offset* | **first** *startweekday startmonth starthour endweekday endmonth endhour offset* | **last** *startweekday startmonth starthour endweekday endmonth endhour offset*}}

Syntax Description

timezone *timezone hoursoffset*	Configures the name of the standard time zone and hours offset from UTC (–23 to +23). See Table 3-1 in the "Usage Guidelines" section.
minutesoffset	(Optional) Minutes offset (see Table 3-1 in the "Usage Guidelines" section) from UTC (0–59).
summertime *timezone*	Configures the name of the summer or daylight saving time zone.
date	Configures the absolute summer time.
startday	Date (1–31) to start.
startmonth	Month (January through December) to start.
startyear	Year (1993–2032) to start.
starthour	Hour (0–23) to start in hour:minute (hh:mm) format.
endday	Date (1–31) to end.
endmonth	Month (January through December) to end.
endyear	Year (1993–2032) to end.
endhour	Hour (0–23) to end in hour:minute (hh:mm) format.
offset	Minutes offset from UTC (0–1439). The summer time offset specifies the number of minutes that the system clock moves forward at the specified start time and backward at the end time.
recurring	Configures the recurring summer time.
1-4	Configures the starting week number 1–4.
startweekday	Day of the week (Monday–Friday) to start.
endweekday	Weekday (Monday–Friday) to end.
first	Configures the summer time to recur beginning the first week of the month.
last	Configures the summer time to recur beginning the last week of the month.

Defaults

No default behavior or values.

Command Modes global configuration

Device Modes application-accelerator

central-manager

Usage Guidelines To set and display the local and UTC current time of day without an NTP server, use the **clock timezone** command with the **clock set** command. The **clock timezone** parameter specifies the difference between UTC and local time, which is set with the **clock set** EXEC command. The UTC and local time are displayed with the **show clock detail** EXEC command.

Note Unexpected time changes can result in unexpected system behavior. We recommend reloading the system after changing the system clock.

Use the **clock** *timezone offset* command to specify a time zone, where *timezone* is the desired time zone entry listed in the table below and *0 0* is the offset (ahead or behind) UTC is in hours and minutes. (UTC was formerly known as Greenwich mean time [GMT]).

```
WAE(config)# clock timezone timezone 0 0
```

Note The time zone entry is case sensitive and must be specified in the exact notation listed in Table 3-1. When you use a time zone entry from the time zone table, the system is automatically adjusted for daylight saving time.

Table 3-1 Time Zone—Offsets from UTC

Time Zone	Offset from UTC
Africa/Algiers	+1
Africa/Cairo	+2
Africa/Casablanca	0
Africa/Harare	+2
Africa/Johannesburg	+2
Africa/Nairobi	+3
America/Buenos_Aires	–3
America/Caracas	–4
America/Mexico_City	–6
America/Lima	–5
America/Santiago	–4
Atlantic/Azores	–1
Atlantic/Cape_Verde	–1
Asia/Almaty	+6
Asia/Baghdad	+3
Asia/Baku	+4

Table 3-1 Time Zone—Offsets from UTC (continued)

Time Zone	Offset from UTC
Asia/Bangkok	+7
Asia/Colombo	+6
Asia/Dacca	+6
Asia/Hong_Kong	+8
Asia/Irkutsk	+8
Asia/Jerusalem	+2
Asia/Kabul	+4.30
Asia/Karachi	+5
Asia/Katmandu	+5.45
Asia/Krasnoyarsk	+7
Asia/Magadan	+11
Asia/Muscat	+4
Asia/New Delhi	+5.30
Asia/Rangoon	+6.30
Asia/Riyadh	+3
Asia/Seoul	+9
Asia/Singapore	+8
Asia/Taipei	+8
Asia/Tehran	+3.30
Asia/Vladivostok	+10
Asia/Yekaterinburg	+5
Asia/Yakutsk	+9
Australia/Adelaide	+9.30
Australia/Brisbane	+10
Australia/Darwin	+9.30
Australia/Hobart	+10
Australia/Perth	+8
Australia/Sydney	+10
Canada/Atlantic	–4
Canada/Newfoundland	–3.30
Canada/Saskatchewan	–6
Europe/Athens	+2
Europe/Berlin	+1
Europe/Bucharest	+2
Europe/Helsinki	+2
Europe/London	0

Table 3-1 Time Zone—Offsets from UTC (continued)

Time Zone	Offset from UTC
Europe/Moscow	+3
Europe/Paris	+1
Europe/Prague	+1
Europe/Warsaw	+1
Japan	+9
Pacific/Auckland	+12
Pacific/Fiji	+12
Pacific/Guam	+10
Pacific/Kwajalein	–12
Pacific/Samoa	–11
US/Alaska	–9
US/Central	–6
US/Eastern	–5
US/East-Indiana	–5
US/Hawaii	–10
US/Mountain	–7
US/Pacific	–8

Examples

The following example shows how to specify the local time zone as Pacific Standard Time with an offset of 8 hours behind UTC:

```
WAE(config)# clock timezone US/Pacific -8 0
```

The following example shows how to negate the time zone setting on the WAAS device:

```
WAE(config)# no clock timezone
```

The following example shows how to configure daylight saving time:

```
WAE(config)# clock summertime US/Pacific date 10 October 2005 23:59 29 April 2006 23:59 60
```

Related Commands

clock

show clock

(config) cms

To schedule maintenance and enable the Centralized Management System (CMS) on a WAAS device, use the **cms** global configuration command. To negate these actions, use the **no** form of this command.

cms {**database maintenance** {**full** {**enable** | **schedule** *weekday* **at** *time*}} |
regular {**enable** | **schedule** *weekday* **at** *time*}} | **enable**

no cms {**database maintenance** {**full** {**enable** | **schedule** *weekday* **at** *time*}} |
regular {**enable** | **schedule** *weekday* **at** *time*}} | **enable**

cms rpc timeout {**connection** *5-1800* | **incoming-wait** *10-600* | **transfer** *10-7200*}

no cms rpc timeout {**connection** *5-1800* | **incoming-wait** *10-600* | **transfer** *10-7200*}

Syntax Description

database maintenance	Configures the embedded database clean or reindex maintenance routine.
full	Configures the full maintenance routine and cleans the embedded database tables.
enable	Enables the specified routine or process to be performed on the embedded database tables.
schedule *weekday*	Sets the schedule for performing the maintenance routine to a day of the week. every-day Every day Mon every Monday Tue every Tuesday Wed every Wednesday Thu every Thursday Fri every Friday Sat every Saturday Sun every Sunday
at *time*	Sets the maintenance schedule time of day to start the maintenance routine (0–23:0–59) (hh:mm). at Maintenance time of day Mon every Monday Tue every Tuesday Wed every Wednesday Thu every Thursday Fri every Friday Sat every Saturday Sun every Sunday
regular	Configures the regular maintenance routine and reindexes the embedded database tables.
rpc timeout	Configures the timeout values for remote procedure call connections.
connection *5-1800*	Specifies the maximum time to wait when making a connection. The timeout period is in seconds. The default for the WAAS Central Manager is 30 seconds; the default for a WAE is 180 seconds.

incoming-wait *10-600*	Specifies the maximum time to wait for a client response. The timeout period is in seconds. The default is 30 seconds.
transfer *10-7200*	Specifies the maximum time to allow a connection to remain open. The timeout period is in seconds. The default is 300 seconds.

Defaults

database maintenance regular: enabled

database maintenance full: enabled

connection: 30 seconds for WAAS Central Manager; 180 seconds for a WAE

incoming wait: 30 seconds

transfer: 300 seconds

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Use the **cms database maintenance** global configuration command to schedule routine full maintenance cleaning (vacuuming) or a regular maintenance reindexing of the embedded database. The full maintenance routine runs only when the disk is more than 90 percent full and only runs once a week. Cleaning the tables returns reusable space to the database system.

The **cms enable** global configuration command automatically registers the node in the database management tables and enables the CMS process. The **no cms enable** global configuration command only stops the management services on the WAAS device. Use the **cms deregister** EXEC command to de-register (remove) a WAAS device from the WAAS network.

Tip If you are trying to register a device that had previously been registered with a WAAS Central Manager and the **cms enable** global configuration command fails, use the **cms deregister force** command. If you get an error saying that the management service is not enabled when you use the **cms deregister force** command, delete the device from the WAAS Central Manager.

Examples

The following example shows how to schedule a regular (reindexing) maintenance routine to start every Friday at 11:00 p.m on the WAAS device:

```
WAE(config)# cms database maintenance regular schedule Fri at 23:00
```

The following example shows how to enable the CMS process on a WAAS device:

```
WAE(config)# cms enable
Generating new RPC certificate/key pair
Restarting RPC services

Creating database backup file emerg-debug-db-01-25-2006-15-31.dump
Registering Wide Area Central Manager...
Registration complete.
```

```
Please preserve running configuration using 'copy running-config startup-config'.
Otherwise management service will not be started on reload and node will be shown
'offline' in Wide Area Central Manager UI.
management services enabled
```

Related Commands

cms

show cms

(config) crypto pki

To configure public key infrastructure (PKI) encryption parameters on a WAAS device, use the **crypto pki** global configuration command. To negate these actions, use the **no** form of this command.

crypto pki {**ca** *certificate-authority-name*}

crypto pki global-settings [**ocsp url** *url* | **revocation-check** {**ocsp-cert-url** [**none**] | **ocsp-url** [**none**] }]

Syntax Description

ca *certificate-authority-name*	Configures encryption certificate authority information. Using this command enables certificate authority configuration mode. See PKI Certificate Authority Configuration Mode Commands, page -859.
global-settings	Configures PKI encryption global settings. Using this command enables PKI global settings configuration mode. See PKI Certificate Authority Configuration Mode Commands, page -859.
ocsp url *url*	(Optional) Configures an OCSP URL.
revocation-check	(Optional) Configures certificate revocation methods.
ocsp-cert-url	Specifies to use the URL from the certificate.
none	(Optional) Specifies a null method that returns revocation success.
ocsp-url	Specifies to use the URL from the global OCSP setting.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Use the **crypto pki** global configuration command to enter CA configuration mode or PKI global settings configuration mode.

Examples

The following example puts WAAS into CA configuration mode, editing the “my-ca” certification authority. The mode change is indicated by the system prompt:

```
WAE(config)# crypto pki my-ca
WAE(config-ca)#
```

Related Commands

(config) crypto ssl

(config-ca) ca-certificate

(config-ca) description
(config-ca) revocation-check

(config) crypto ssl

To configure secure sockets layer (SSL) encryption parameters on a WAAS device, use the **crypto ssl** global configuration command. To negate these actions, use the **no** form of this command.

crypto ssl {**cipher-list** *cipher-list-name* | **management-service** |
services {**accelerated-service** *service-name* | **global-settings** | **host-service peering**}}

no crypto ssl {**cipher-list** *cipher-list-name* | **management-service** |
services {**accelerated-service** *service-name* | **global-settings** | **host-service peering**}}

Syntax Description

cipher-list *cipher-list-name*	Configures the SSL cipher suite list. Using this command enables SSL cipher list configuration mode. See the SSL Cipher List Configuration Mode Commands chapter.
management-service	Configures SSL management services. Using this command enables SSL management service configuration mode. See the SSL Management Service Configuration Mode Commands chapter.
services	Configures other SSL services (accelerated, global, and host peering).
accelerated-service *service-name*	Configures SSL accelerated services. Using this command enables SSL accelerated service configuration mode. See the SSL Accelerated Service Configuration Mode Commands chapter.
global-settings	Configures SSL service global settings. Using this command enables SSL service global configuration mode. See the SSL Global Service Configuration Mode Commands chapter.
host-service peering	Configures SSL host peering services. Using this command enables SSL host peering service configuration mode. See the SSL Host Peering Service Configuration Mode Commands chapter.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

Use the **crypto ssl** global configuration command to enter SSL cipher list configuration mode, SSL management service configuration mode, SSL accelerated service configuration mode, SSL service global configuration mode, or SSL host peering service configuration mode.

Examples

The following example puts the WAAS device into SSL cipher list configuration mode, editing the mylist cipher suite list. The mode change is indicated by the system prompt:

```
WAE(config)# crypto ssl cipher-list mylist
WAE(config-cipher-list)#
```

The following example puts the WAAS device into SSL management service configuration mode. The mode change is indicated by the system prompt:

```
WAE(config)# crypto ssl management-service
WAE(config-ssl-mgmt)#
```

The following example puts the WAAS device into SSL accelerated service configuration mode, editing the myservice accelerated service. The mode change is indicated by the system prompt:

```
WAE(config)# crypto ssl services accelerated-service myservice
WAE(config-ssl-accelerated)#
```

The following example puts the WAAS device into SSL global service configuration mode. The mode change is indicated by the system prompt:

```
WAE(config)# crypto ssl services global-settings
WAE(config-ssl-global)#
```

The following example puts the WAAS device into SSL host peering service configuration mode. The mode change is indicated by the system prompt:

```
WAE(config)# crypto ssl services host-service peering
WAE(config-ssl-peering)#
```

Related Commands

(config) crypto pki

(config) device mode

To configure the device mode for the WAAS device, use the **device mode** global configuration command. To reset the mode of operation on your WAAS device, use the **no** form of this command.

device mode {**application-accelerator** | **central-manager** | **appnav-controller**}

no device mode {**application-accelerator** | **central-manager** | **appnav-controller**}

Syntax Description

application-accelerator	Configures the WAAS device to function as a WAAS Accelerator. All of the branch and data center WAEs that are doing traffic optimization must be operating in this mode.
central-manager	Configures the WAAS device to function as a WAAS Central Manager.
appnav-controller	Configures the WAAS device to function as an AppNav Controller in an AppNav deployment.

Defaults

The default device operation mode is application-accelerator.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

appnav-controller

Usage Guidelines

If the WAAS device is operating with an Accelerator only image, you will not be able to convert it to central-manager mode until after you update it with a Full image and reboot. You can use the **show version** EXEC command to check the type of software image the WAE is running.

Examples

The following example shows how to specify central manager as the device mode of a WAAS device:

```
WAE(config)# device mode central-manager
```

The following example shows how to specify application accelerator as the device mode of a WAAS device:

```
WAE(config)# device mode application-accelerator
```

To change the device mode from central-manager to application-accelerator or appnav-controller, you must first use the **cms deregister** command in EXEC mode to disable the Centralized Management System on the Central Manager. Then use the **device mode** command in global configuration mode, as shown in the following example:

```
WAE# cms deregister
WAE(config)# device mode application-accelerator
```

```
WAE# copy running-config startup-config
```

Related Commands

show device-mode

(config) directed-mode

To configure the mode by which traffic is sent between two WAEs, use the **directed-mode** global configuration command. To configure the WAAS device not to use directed mode, use the **no** form of this command.

directed-mode enable [**port** *udp-port*]

no directed-mode enable [**port** *udp-port*]

Syntax Description

enable	Enables directed mode.
port *udp-port*	(Optional) Sets the UDP port number to use to send traffic between two WAEs. The default port is 4050.

Defaults

The default communication mode to a peer WAE is transparent mode (not directed mode).

Command Modes

global configuration

Device Modes

application-accelerator

Examples

The following example shows how to configure a WAE for directed mode on the default UDP port of 4050:

```
WAE(config)# directed-mode enable
```

Related Commands

show statistics auto-discovery

show statistics connection closed

(config) disk disk-name

To disable the disk for online removal, use the **disk disk-name** global configuration command. To reenable the disk, use the **no** form of this command.

disk disk-name *diskxx* **shutdown** [**force**]

no disk disk-name *diskxx* **shutdown** [**force**]

Syntax Description

diskxx	Name of the disk (disk00-disk05).
shutdown	Disables the disk for maintenance.
force	(Optional) Forces a disk to be reenabled when used with the **no** form of this command. This option is not available on RAID-5 systems.

Defaults

Disks are enabled.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

This command is not available on the WAE-7341 and WAE-7371 models. Instead, use the **disk disk-name disk***xx* **replace** EXEC mode command.

You can replace a failed disk or perform a scheduled disk maintenance on the WAE-612. Use the **disk disk-name** *diskxx* **shutdown** global configuration command to manually shut down a disk for a scheduled disk maintenance, or on the WAE-7341 and WAE-7371, use the **disk disk-name** *diskxx* **replace** EXEC command to manually shut down a disk for scheduled disk maintenance. (For the schedule disk maintenance procedure, see the *Cisco Wide Area Application Services Configuration Guide*, Chapter 14.)

Note The **show disks failed-disk-id** EXEC command is not available on WAE-7341 and WAE-7371 models.

Examples

The following example shows how to disable disk00 for online removal using the **disk disk-name** command:

```
WAE(config)# disk disk-name disk00 shutdown
```

Related Commands

(config) disk error-handling

(config) disk logical shutdown
disk
show disks

(config) disk encrypt

To enable disk encryption, use the **disk encrypt** global configuration command. To disable disk encryption, use the **no** form of this command.

disk encrypt enable

no disk encrypt enable

Syntax Description

enable	Enables disk encryption.

Defaults

Disk encryption is disabled by default.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

To view the encryption status details, use the **show disks details** EXEC command. While the file system is initializing, you will see the following message: "`System initialization is not finished, please wait...`" You may also view the disk encryption status to check whether a disk is enabled or disabled in the Central Manager GUI, Device Home window.

Note If you are using a No Payload Encryption (NPE) image, the disk encryption feature has been disabled for use in countries where disk encryption is not permitted.

Examples

The following example shows how to enable disk encryption using the **disk encrypt** command:

```
WAE(config)# disk encrypt enable
```

Related Commands

disk

show disks

(config) disk error-handling

To configure how disk errors are handled on a WAAS device, use the **disk error-handling** global configuration command. To disable automatic remapping of disk errors, use the **no** form of this command.

disk error-handling remap

no disk error-handling remap

Syntax Description

remap	Sets the disk to attempt to remap disk errors automatically.

Defaults

The disk is configured to remap disk errors automatically.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Examples

The following example shows how to disable automatic remapping of disk errors:

```
WAE(config)# no disk error-handling remap
```

Related Commands

disk

show disks

(config) disk logical shutdown

To shut down the RAID-5 logical disk drive, use the **disk logical shutdown** global configuration command. To reenable the RAID-5 logical disk drive, use the **no** form of this command.

disk logical shutdown

no disk logical shutdown [**force**]

Syntax Description

force	(Optional) Forces RAID Logical drive to be reenabled when used with the **no** form of this command.

Defaults

The RAID-5 array is configured by default.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

This command is supported on WAE-7341, WAE-7371, and WAE-674 models only.

Use this command to operate the WAE in diskless mode. In diskless mode, the partitions and disks are not mounted and cannot be used.

You must reload the device for this command to take effect.

After a multiple disk failure or RAID controller failure, and after the drives are replaced and the RAID disk is rebuilt, the logical disk may remain in the error state. To reenable the disk, use the **no disk logical shutdown force** command, then reload the WAE.

Examples

The following example shows how shutdown the RAID-5 logical disk drive using the **disk logical shutdown** command:

```
WAE(config)# disk logical shutdown
```

Related Commands

(config) disk disk-name

(config) disk object-cache extend

To enable extended object cache, use the **disk object-cache extend** global configuration command. To disable this feature, use the **no** form of this command.

disk object-cache extend

no disk object-cache extend

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

When extended object cache is enabled, the object cache space is increased only after saving the configuration and performing a reload.

Note If you have a virtual blade enabled using vbspace of greater than 30 GB, you must stop the virtual blade and remove the configuration before enabling extended object cache. If the virtual blade usage is less than 30 GB (including saved memory state) vbspace content will be preserved, otherwise vbspace content will be erased. In either case, after enabling extended object cache, the vbspace filesystem will be reduced to 30 GB if virtual blade is enabled.

The status of extended object cache can be displayed using the **show disk details** EXEC mode command. The output of this command states whether extended object cache is enabled or disabled.

This feature is supported only on WAVE-694, WAE-674-4G, and WAE-674-8G models.

When a device is changed to AppNav mode, a warning message tells the user that changing the Device mode to AppNav Controller, will forcefully disable disk object-cache extend.The new configuration will take effect after a reload. If the user confirms, the system proceeds with reloading the system configuration and the extended object cache is disabled.

Examples

The following example shows how to enable extended object cache:

```
WAE(config)# disk object-cache extend
Cumulative disk space for all VBs will be reduced to 30GB.
Are you sure want to enable [yes/no]?
```

Related Commands

(config) disk logical shutdown

(config) dre

To enable and configure DRE (Data Redundancy Elimination) auto bypass and load monitor settings, use the **dre** global configuration command. To disable DRE settings, use the **no** form of this command.

dre {**auto-bypass** {**cache-percent** [*percent_no*] | **comp-threshold** [*comp_threshold*] | **enable**} | **load-monitor** {**report** | **threshold** [*threshold*]}}

no dre {**auto-bypass** {**cache-percent** | **comp-threshold** | **enable**} | **load-monitor** {**report** | **threshold**}}

Syntax Description

auto-bypass	Configures DRE auto bypass settings.
cache-percent *percent_no*	Sets the cache size percent threshold for bypass trigger (1-99).
comp-threshold *comp_threshold*	Sets the DRE compression ratio threshold for bypass trigger (1-50).
enable	Enables DRE auto bypass.
load-monitor	Configures load monitor settings.
report	Enables load report.
threshold *threshold*	Sets the DRE load threshold (50-99).

Defaults

Enabled by default.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

Use the **dre auto-bypass** global configuration command to generate an alarm and automatically DRE bypass application traffic.

Examples

The following example shows how to enable DRE auto bypass using the **dre** command:

```
WAE(config)# dre auto-bypass enable
```

Related Commands

(config) dre

(config) end

To exit global configuration mode, use the **end** global configuration command.

end

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Use the **end** command to exit global configuration mode after completing any changes to the running configuration. To save new configurations to NVRAM, use the **write** command.

In addition, you can press **Ctrl-Z** to exit global configuration mode.

Examples

The following example shows how to exit global configuration mode on a WAAS device:

```
WAE(config)# end
WAE#
```

Related Commands

(config) exit

(config) exec-timeout

To configure the length of time that an inactive Telnet or SSH session remains open on a WAAS device, use the **exec-timeout** global configuration command. To revert to the default value, use the **no** form of this command.

exec-timeout *timeout*

no exec-timeout *timeout*

Syntax Description

timeout	Timeout in minutes (0–44640). A value of 0 sets the logout timeout to infinite.

Defaults

The default is 15 minutes.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

A Telnet session or Secure Shell (SSH) session with the WAAS device can remain open and inactive for the interval of time specified by the **exec-timeout** command. When the **exec-timeout** interval elapses, the WAAS device automatically closes the Telnet or SSH session.

Examples

The following example shows how to configure a timeout of 100 minutes:

```
WAE(config)# exec-timeout 100
```

The following example shows how to negate the configured timeout of 100 minutes and revert to the default value of 15 minutes:

```
WAE(config)# no exec-timeout
```

Related Commands

(config) telnet enable

(config) exit

To terminate global configuration mode and return to the privileged-level EXEC mode, use the **exit** command.

exit

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

All modes

Device Modes

application-accelerator

central-manager

Usage Guidelines

This command is equivalent to pressing **Ctrl-Z** or entering the **end** command.

Examples

The following example shows how to terminate global configuration mode and return to the privileged-level EXEC mode:

```
WAE(config)# exit
WAE#
```

Related Commands

(config) end

(config) flow monitor

To enable network traffic flow monitoring and to register the WAE with the tcpstat-v1 collector for traffic analysis, use the **flow monitor** global configuration command. To disable the network traffic flow configuration, use the **no** form of this command.

flow monitor tcpstat-v1 {**enable** | **host** *ip_address*}

no flow monitor tcpstat-v1 {**enable** | **host** *ip_address*}

Syntax Description

tcpstat-v1	Sets the tcpstat-v1 collector configuration.
enable	Enables flow monitoring.
host *ip_address*	Specifies the IP address of the collection control agent.

Defaults

The default configuration has no host address configured and the feature is disabled.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

For information about how to configure flow monitoring on the WAE, see the *Cisco Wide Area Application Services Configuration Guide*, Chapter 15.

Examples

The following example shows how to enable flow monitoring using the **flow monitor** command:

```
WAE(config)# flow monitor tcpstat-v1 enable
```

Related Commands

debug flow

(config) help

To obtain online help for the command-line interface, use the **help** global configuration command. To disable help, use the **no** form of this command.

help

no help

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

EXEC and global configuration

Device Modes

application-accelerator

central-manager

appnav-controller

Usage Guidelines

You can obtain help at any point in a command by entering a question mark (**?**). If nothing matches, the help list will be empty, and you must use the backspace key until entering a **?** shows the available options.

Two styles of help are provided:

- Full help is available when you are ready to enter a command argument (for example, **show ?**) and describes each possible argument.
- Partial help is provided when you enter an abbreviated command and you want to know what arguments match the input (for example, **show stat?**).

Examples

The following example shows the output of the **help** global configuration command:

```
WAE# configure
WAE(config)# help
Help may be requested at any point in a command by entering a question mark '?'. If
nothing matches, the help list will be empty and you must backup until entering a '?'
shows the available options.

Two styles of help are provided:
1. Full help is available when you are ready to enter a command argument.
2. Partial help is provided when an abbreviated argument is entered.
```

The following example shows how to use full help to see what WCCP command arguments are available:

```
WAE# configure
WAE(config)# wccp ?
  access-list      Configure an IP access-list for inbound WCCP encapsulate
                   traffic
```

```
  flow-redirect    Redirect moved flows
  router-list      Router List for use in WCCP services
  shutdown         Wccp Shutdown parameters
  tcp-promiscuous  TCP promiscuous mode service
```

The following example shows how to use partial help to determine the syntax of a WCCP argument:

```
WAE(config)# wccp tcp ?
  service-pair Pair of TCP promiscuous services
```

Related Commands

show running-config

(config) hostname

To configure the network hostname on a WAAS device, use the **hostname** global configuration command. To reset the hostname to the default setting, use the **no** form of this command.

hostname *name*

no hostname *name*

Syntax Description

name	New hostname for the WAAS device; the name is case sensitive. The name may be from 1 to 30 alphanumeric characters.

Defaults

The default hostname is the model number of the WAAS device (for example WAE-612 or WAE-7371).

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Use this command to configure the hostname for the WAAS device. The hostname is used for the command prompts and default configuration filenames. This name is also used for routing, so it conforms to the following rules:

- It can use only alphanumeric characters and hyphens (-).
- The maximum length is 30 characters.
- The following characters are considered illegal and cannot be used when naming a device: @, #, $,%, ^, &, *, (), |, \””/, <>.

Examples

The following example shows how to change the hostname of the WAAS device to *sandbox*:

```
WAE-674(config)# hostname sandbox
Sandbox(config)#
```

The following example shows how to remove the hostname:

```
Sandbox(config)# no hostname
WAE-674(config)#
```

Related Commands

dnslookup

(config) ip

(config-if) ip

show hosts

(config) inetd

To enable FTP and RCP services on a WAAS device, use the **inetd enable** global configuration command. To disable these same services, use the **no** form of this command.

inetd enable {**ftp** | **rcp**}

no inetd enable {**ftp** | **rcp**}

Syntax Description

enable	Enables services.
ftp	Enables FTP services.
rcp	Enables RCP services.

Defaults

FTP is enabled; RCP is disabled.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Inetd (an Internet daemon) is a program that listens for connection requests or messages for certain ports and starts server programs to perform the services associated with those ports. Use the **inetd enable** command with the **ftp** and **rcp** keywords to enable and disable services on the WAAS device. To disable the service, enter the **no** form of the **inetd enable** command. Use the **show inetd** EXEC command to see whether current **inetd** sessions are enabled or disabled.

Examples

The following example shows how to enable an FTP service session on the WAAS device:

```
WAE(config)# inetd enable ftp
```

The following example shows how to disable FTP services:

```
WAE(config)# no inetd enable ftp
```

Related Commands

show inetd

(config) inline

To configure the failover timeout of inline interfaces on a Cisco Interface Module, use the **inline** global configuration command. To unconfigure the failover timeout, use the **no** form of this command.

inline failover timeout {**1** | **5** | **25**}

no inline failover timeout {**1** | **5** | **25**}

Syntax Description

failover timeout {**1** | **5** | **25**}	Sets the failover timeout for the inline interfaces. Valid values are 1, 5, or 25 seconds. The default is 1.

Defaults

1 second.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

This command applies only to the following WAAS devices that use a Cisco Interface Module: WAVE-294, WAVE-594, WAVE-694, WAVE-7541, WAVE-7571, and WAVE-8541. This command does not apply to the TenGigabitEthernet module, which cannot be used in inline mode, or to interfaces on the Cisco AppNav Controller Interface Module.

The **inline failover timeout** command sets the number of seconds the interface should wait before going into bypass mode, after a device or power failure.

Examples

The following example shows how to configure the inline failover timeout for 5 seconds:

```
WAE(config)# inline failover timeout 5
```

Related Commands

(config) bridge

(config) interception-method

(config) interface InlineGroup

(config) interface GigabitEthernet

(config) interface TenGigabitEthernet

(config) inline vlan-id-connection-check

To enable VLAN ID checking on intercepted traffic, use the **inline vlan-id-connection-check** global configuration command. To disable VLAN ID checking, use the **no** form of this command.

inline vlan-id-connection-check

no inline vlan-id-connection-check

Syntax Description

This command has no arguments or keywords.

Defaults

VLAN ID checking is enabled.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Examples

The following example shows how to enable VLAN ID checking of the intercepted traffic on the WAAS device:

```
WAE(config)# inline vlan-id-connection-check
```

The following example shows how to disable VLAN ID checking:

```
WAE(config)# no inline vlan-id-connection-check
```

Related Commands

(config) interface InlineGroup

(config) interface GigabitEthernet

(config) interface TenGigabitEthernet

(config-if) encapsulation dot1Q

(config) interception

To configure traffic interception with an access list, use the **interception** global configuration command. To disable the interception access list, use the **no** form of this command.

interception [**appnav-controller**] **access-list** {*acl-num* | *acl_name*}

no interception [**appnav-controller**] **access-list** {*acl-num* | *acl_name*}

Syntax Description

appnav-controller	Configures an access list for an ANC.
acl_num	Numeric identifier that identifies the ACL to apply to traffic interception. For standard ACLs, the valid range is 1–99; for extended ACLs, the valid range is 100–199.
acl_name	Alphanumeric identifier of up to 30 characters, beginning with a letter that identifies the ACL to apply to traffic interception.

Defaults

No default behaviors or values.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

appnav-controller

Usage Guidelines

Use the **interception** command to apply an access list (ACL) to traffic interception. Packets permitted by the ACL are intercepted for WAAS optimization (on an application accelerator device) or for distribution (on an ANC). Packets denied by the ACL are passed through by WAAS. You can define ACLs by using the **ip access-list standard** or **ip access-list extended** configuration commands.

Note On an ANC the `tcp ... established` extended ACL rule type is not supported.

You can configure only one interception ACL, except if the device is an ANC that is also acting as a WN. In this situation, you can configure one **interception appnav-controller access-list** for the ANC and one **interception access-list** for the WN. The ANC ACL could permit a flow that is subsequently denied by the WN ACL.

If you specify an interception ACL that is not defined, it is considered to be a “permit any” ACL and all traffic is intercepted.

An interception ACL works both with WCCP and inline interception modes.

When used with interface ACLs and WCCP ACLs, the interface ACL is applied first, the WCCP ACL is applied second, and then the interception ACL is applied last.

Examples

The following example shows how to define and apply an ACL that intercepts all traffic except WWW traffic from a particular client:

```
dc-wae(config)# ip access-list extended iacl
dc-wae(config-ext-nacl)# deny tcp host 10.74.2.132 any eq www
dc-wae(config-ext-nacl)# permit ip any any
dc-wae(config-ext-nacl)# exit

dc-wae(config)# interception access-list iacl
```

The following example uses the same ACL for an ANC:

```
anc(config)# interception appnav-controller access-list iacl
```

Related Commands

(config) ip access-list

show ip access-list

(config) interception-method

To configure the traffic interception method, use the **interception-method** global configuration command. To disable the interception method, use the **no** form of this command.

interception-method {**inline** | **appnav-controller** | **wccp** | **vn-service**}[**force**]

no interception-method {**inline** | **appnav-controller** | **wccp** | **vn-service**}[**force**]

Syntax Description

inline	Enables inline traffic interception.
appnav-controller	Enables a WAAS node to receive traffic for optimization from an AppNav Controller in an AppNav deployment. (Available only on devices in application-accelerator device mode.)
wccp	Enables WCCP traffic interception.
vn-service	Enables VPATH traffic interception on a vWAAS instance.
force	Forces the configuration without prompting.

Defaults

No default behaviors or values.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

Usage Guidelines

You must use the **interception-method** command to enable a traffic interception method before configuring other traffic interception settings. Other settings that are specific to a particular traffic interception method are not available until after you use this command to enable the method.

When you are changing the traffic interception method, all configuration settings for the current method are removed before the new method is enabled. You are prompted to confirm before the command proceeds.

Examples

The following example shows how to enable WCCP interception:

```
dc-wae(config)# interception-method wccp
Inline interception method will be removed. Proceed?[yes]: yes
```

Related Commands

(config) bridge

(config) inline

(config) interface InlineGroup

(config) wccp tcp-promiscuous service-pair

show interception-method

(config) interface bvi

To configure a bridge virtual interface, use the **interface bvi** global configuration command. To disable a bridge virtual interface, use the **no** form of this command.

interface bvi *bridge-id* [**description** *text* | **ip address** *ip-address netmask* [**secondary**] | **dhcp** [**client-id** *id*][**hostname** *name*] | **load-interval** *seconds*]

no interface bvi *bridge-id* [**description** *text* | **ip address** *ip-address netmask* [**secondary**] | **dhcp** [**client-id** *id*][**hostname** *name*] | **load-interval** *seconds*]

Syntax Description

bridge-id	Bridge virtual interface. Specify a bridge ID from 1–4.
description *text*	(Optional) Specifies a description of the interface.
ip address *ip-address netmask*	Sets the interface IP address and netmask.
secondary	(Optional) Defines the IP address as a secondary IP address.
dhcp	(Optional) Sets the IP address to the address that is negotiated over Dynamic Host Configuration Protocol (DHCP).
client-id *id*	(Optional) Specifies the client identifier.
hostname *name*	(Optional) Specifies the hostname.
load-interval *seconds*	(Optional) Sets the interval at which to poll the interface for statistics and calculate throughput. Ranges from 30 to 600 seconds. The default is 30 seconds.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

This command configures a bridge virtual interface for bridging to a virtual blade. Before you can use this command, you must create a bridge group by using the **bridge** global configuration command.

When DHCP is configured, the bridge virtual interface gets its IP address from the DHCP server when a physical interface supporting DHCP is added to the bridge group.

A bridge virtual interface is not supported on AppNav Controller Interface Module interfaces.

Examples

The following example shows how to create and configure a bridge interface for a virtual blade:

```
WAE# configure
WAE(config)# bridge 1 protocol ieee
WAE(config)# interface GigabitEthernet 1/0 bridge-group 1
WAE(config)# interface bvi 1 ip address 10.10.10.10 255.0.0.0
```

```
WAE(config)# virtual-blade 2
WAE(config-vb)# interface 1 bridge-group 1
```

The following example shows how to remove the configuration of a bridge virtual interface:

```
WAE(config)# no interface bvi 1
```

Related Commands

(config) bridge

(config) interface GigabitEthernet

(config) interface TenGigabitEthernet

(config) interface GigabitEthernet

To configure a Gigabit Ethernet interface, use the **interface** global configuration command. To disable selected options, restore default values, or enable a shutdown interface, use the **no** form of this command.

interface GigabitEthernet *slot/port* [**autosense** | **bandwidth** {**10** | **100** | **1000**} | **cdp enable** | **channel-group** *index* | **description** *text* | **full-duplex** | **half-duplex** | **ip** {**access-group** {*acl-num* | *acl_name*} {**in** | **out**} | **address** {*ip_address netmask* [**secondary**] | **dhcp** [**client-id** *id*][**hostname** *name*]}} | **load-interval** *seconds* | **mtu** *mtusize* | **shutdown** | **standby** *group-index* [**primary**] | **bridge-group** *bridge-id*]

no interface GigabitEthernet *slot/port* [**autosense** | **bandwidth** {**10** | **100** | **1000**} | **cdp enable** | **channel-group** *index* | **description** *text* | **full-duplex** | **half-duplex** | **ip** {**access-group** {*acl-num* | *acl_name*} {**in** | **out**} | **address** {*ip_address netmask* [**secondary**] | **dhcp** [**client-id** *id*][**hostname** *name*]}} | **load-interval** *seconds* | **mtu** *mtusize* | **shutdown** | **standby** *group-index* [**primary**] | **bridge-group** *bridge-id*]

Syntax Description

GigabitEthernet *slot/port*	Selects a Gigabit Ethernet interface to configure (slot and port number). The slot number and port number are separated with a forward slash character (/). Valid slot and port values depend on the hardware platform.
autosense	(Optional) Sets the GigabitEthernet interface to automatically sense the interface speed.
bandwidth	(Optional) Sets the bandwidth of the specified interface.
10	Sets the bandwidth of the interface to 10 megabits per second (Mbps).
100	Sets the bandwidth of the interface to 100 Mbps.
1000	Sets the bandwidth of the interface to 1000 Mbps. This option is not available on all ports and is the same as autosense.
cdp enable	(Optional) Enables Cisco Discovery Protocol (CDP) on the specified interface.
channel-group *index*	(Optional) Assigns the interface to the EtherChannel with the specified index (1-7).
description *text*	Enters a description of the interface.
full-duplex	(Optional) Sets the interface to full-duplex operation.
half-duplex	(Optional) Sets the interface to half-duplex operation. **Note** We strongly recommend that you do not use half duplex on the WAE, routers, switches, or other devices.
ip	(Optional) Enables IP configuration commands for the interface.
access-group	Configures access control for IP packets on this interface using access control list (ACL).
acl_num	Numeric identifier that identifies the ACL to apply to the current interface. For standard ACLs, the valid range is 1–99; for extended ACLs, the valid range is 100–199.
acl_name	Alphanumeric identifier of up to 30 characters, beginning with a letter that identifies the ACL to apply to the current interface.

in	Applies the specified ACL to inbound packets on the current interface.
out	Applies the specified ACL to outbound packets on the current interface.
address *ip-address netmask*	Sets the interface IP address and netmask.
secondary	(Optional) Specifies that the configured address is a secondary IP address. If this keyword is omitted, the configured address is the primary IP address.
dhcp	(Optional) Sets the IP address to the address that is negotiated over Dynamic Host Configuration Protocol (DHCP).
client-id *id*	(Optional) Specifies the client identifier.
hostname *name*	(Optional) Specifies the hostname.
load-interval *seconds*	(Optional) Sets the interval at which to poll the interface for statistics and calculate throughput. Ranges from 30 to 600 seconds. The default is 30 seconds.
mtu *mtusize*	(Optional) Sets the interface Maximum Transmission Unit (MTU) size in bytes (576–1500).
shutdown	(Optional) Shuts down this interface.
standby *group-index*	(Optional) Sets the standby group number to *group-index*.
primary	(Optional) Sets this interface as the active interface in the standby group.
bridge-group *bridge-id*	Places the interface into the specified bridge group.

Defaults

The first attached interface in a standby group is defined as the active interface. There are no other default behaviors or values.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

central-manager

Usage Guidelines

Although the CLI contains the **no interface** option, you cannot apply the **no** command to an interface. The software displays the following error message: Removing of physical interface is not permitted.

To configure an interface bandwidth on a WAAS device, use the **bandwidth** interface configuration command. The bandwidth is specified in megabits per second (Mbps). Using this option automatically enables autosense on the interface.

Note Changing the interface bandwidth, duplex mode, or MTU can cause network disruption for up to 30 seconds. The best practice is to make such changes when traffic interception is disabled or at an off-peak time when traffic disruption is acceptable.

Using the **cdp enable** command in global configuration mode enables CDP globally on all the interfaces. If you want to control CDP behavior per interface, use the **cdp enable** command in interface configuration mode. The interface level control overrides the global control.

To display the interface identifiers (for example, interface GigabitEthernet 1/0), use the **show running-config** or **show startup-config** commands. The **autosense**, **bandwidth**, **full-duplex**, **half-duplex**, **ip**, and **shutdown** commands are listed separately in this command reference.

Note When you use the **ip address** command to change the IP address of an interface that has been shut down, it automatically brings up that interface by default.

Configuring Multiple Secondary IP Addresses on a Single Physical Interface

Use the **interface secondary** global configuration command to configure more than one IP address on the same interface. By configuring multiple IP addresses on a single interface, the WAAS device can be present in more than one subnet. This configuration allows you to optimize the response time because the content goes directly from the WAAS device to the requesting client without being redirected through a router. The WAAS device becomes visible to the client because they are configured on the same subnet.

You can assign up to four secondary addresses to an interface. These addresses become active only after you configure the primary address. No two interfaces can have the same IP address in the same subnetwork. To set these secondary IP addresses, use the **ip address** command.

If a WAAS device has one physical interface that has multiple secondary IP addresses assigned to it, the egress traffic uses the source IP address that is chosen by IP routing. If the secondary IP addresses of a WAAS device in the same subnet as the primary IP address, then the egress traffic uses the primary IP address only. If the secondary IP addresses are in a different subnet than the primary IP address, then the destination IP address determines which IP address on the WAAS device is used for the egress traffic.

Configuring Interfaces for DHCP

When you configure a WAAS device initially, you can configure a static IP address or use interface-level DHCP to dynamically assign IP addresses to the interfaces on the WAAS device.

If you do not enable interface-level DHCP on the WAAS device, you must manually specify a static IP address and network mask for the WAAS device. If the WAAS device moves to another location in another part of the network, you must manually enter a new static IP address and network mask for this WAAS device.

You can enable an interface for DHCP using the **ip address dhcp client-id** *id* **hostname** *name* interface configuration command. The client identifier is an ASCII value. The WAAS device sends its configured client identifier and hostname to the DHCP server when requesting network information. You can configure DHCP servers to identify the client identifier and the hostname that the WAAS device is sending and then send the specific network settings that are assigned to the WAAS device.

Note You must disable autoregistration before you can manually configure an interface for DHCP. Autoregistration is enabled by default on the first interface of the device.

Defining Interface Descriptions

You can specify a one-line description for a specific interface on a WAAS device. Use the **description** *text* interface configuration command to enter the description for the specific interface. The maximum length of the description text is 240 characters. This feature is supported for the Gigabit Ethernet, 10 Gigabit Ethernet, port-channel, standby, and bridge virtual interfaces.

After you define the description for an interface, use the **show** EXEC commands to display the defined interface descriptions. Enter the **show interface** *interface type slot/port* EXEC command to display the defined description for a specific interface on the WAE.

Configuring a Standby Group

You can associate an interface with a standby group by using the **standby** *group-index* interface configuration command. To make an interface the active interface in a standby group, use the **standby** *group-index* **primary** interface configuration command. If you have already associated an interface with a standby group but have not made it the primary interface, you cannot specify the command again to add the primary designation. First, remove the interface from the standby group, then reassign it, specifying the **primary** option at the same time.

A physical interface can be a member of a standby group or a port channel, but not both.

If a device has only two interfaces, you cannot assign an IP address to both a standby group and a port channel. On such a device, only one virtual interface can be configured with an IP address.

Examples

The following example shows how to configure an attribute of an interface with a single CLI command:

```
WAE(config)# interface GigabitEthernet 1/0 full-duplex
```

The following example shows that an interface can be configured in a sequence of CLI commands:

```
WAE(config)# interface GigabitEthernet 1/0
WAE(config-if)# full-duplex
WAE(config-if)# exit
```

The following example shows how to enable a shut down interface:

```
WAE(config)# no interface GigabitEthernet 1/0 shutdown
```

The following example shows how to add an interface to a channel group:

```
WAE# configure
WAE(config)# interface GigabitEthernet 1/0
WAE(config-if)# channel-group 1
WAE(config-if)# exit
```

The following example shows how to remove an interface from a channel group:

```
WAE(config)# interface GigabitEthernet 1/0
WAE(config-if)# no channel-group 1
WAE(config-if)# exit
```

The following example shows how to assign a secondary IP address on a Gigabit Ethernet interface on a WAAS device:

```
WAE# configure
WAE(config)# interface GigabitEthernet 1/0
WAE(config-if)# ip address 10.10.10.10 255.0.0.0 secondary
```

The following example shows how to configure a description for a Gigabit Ethernet interface:

```
WAE(config)# interface GigabitEthernet 1/0
WAE(config-if)# description This is a GigabitEthernet interface.
```

Related Commands

(config) interface InlineGroup

(config) interface PortChannel

(config) interface standby

(config) interface TenGigabitEthernet

(config) interface virtual

show interface

show running-config

show startup-config

(config) interface InlineGroup

To configure an InlineGroup interface, use the **interface** global configuration command. To disable selected options, restore default values, or enable a shutdown interface, use the **no** form of this command.

interface InlineGroup *slot/grpnumber* [**autosense | bandwidth** {**10 | 100 | 1000**} **| cdp enable | encapsulation dot1q** *VLAN* **| failover timeout** {**1 | 3 | 5 | 10**} **| full-duplex | half-duplex | inline** [**vlan** {**all | native |** *vlan_list*}] **| ip** {**access-group** {*acl-num* | *acl_name*} {**in | out**} **| load-interval** *seconds* **| shutdown**]

no interface InlineGroup *slot/grpnumber* [**autosense | bandwidth** {**10 | 100 | 1000**} **| cdp enable | encapsulation dot1q** *VLAN* **| failover timeout** {**1 | 3 | 5 | 10**} **| full-duplex | half-duplex | inline** [**vlan** {**all | native |** *vlan_list*}] **| ip** {**access-group** {*acl-num* | *acl_name*} {**in | out**} **| load-interval** *seconds* **| shutdown**]

Syntax Description

slot/grpnumber	Slot and inline group number for the selected interface. The slot and inline group number are separated with a forward slash character (/). Valid slot and inline group values depend on the hardware platform.
autosense	(Optional) Sets the Gigabit Ethernet interface to automatically sense the interface speed.
bandwidth	(Optional) Sets the bandwidth of the specified interface.
10	Sets the bandwidth of the interface to 10 megabits per second (Mbps).
100	Sets the bandwidth of the interface to 100 Mbps.
1000	Sets the bandwidth of the interface to 1000 Mbps. This option is not available on all ports and is the same as autosense.
cdp enable	(Optional) Enables Cisco Discovery Protocol (CDP) on the specified interface.
encapsulation dot1q *VLAN*	(Optional) Sets the 802.1Q VLAN ID to be assigned to traffic leaving the WAE through this interface. The VLAN ID can range from 1–4094.
failover timeout	(Optional) Sets the maximum time for the inline group of interfaces to transfer traffic to another port in the group after a failover event. (Applies only to interfaces on the Cisco WAE Inline Network Adapter.)
1	Specifies the number of seconds before a failover occurs (default).
3	Specifies the number of seconds before a failover occurs.
5	Specifies the number of seconds before a failover occurs.
10	Specifies the number of seconds before a failover occurs.
full-duplex	(Optional) Sets the interface to full duplex.
half-duplex	(Optional) Sets the interface to half duplex. **Note** We strongly recommend that you do not use half duplex on the WAE, routers, switches, or other devices.
inline	(Optional) Enables inline interception for an InlineGroup of interfaces.
vlan	(Optional) Modifies the VLAN list parameters.
all	Applies the command to all tagged and untagged packets.
native	Specifies untagged packets.

vlan_list	Comma-separated list of VLAN IDs. Restricts the inline feature to the specified set of VLANs.
ip	(Optional) Enables IP configuration commands for the interface.
access-group	Configures access control for IP packets on this interface using access control list (ACL).
acl_num	Numeric identifier that identifies the ACL to apply to the current interface. For standard ACLs, the valid range is 1–99; for extended ACLs, the valid range is 100–199.
acl_name	Alphanumeric identifier of up to 30 characters, beginning with a letter that identifies the ACL to apply to the current interface.
in	Applies the specified ACL to inbound packets on the current interface.
out	Applies the specified ACL to outbound packets on the current interface.
load-interval *seconds*	(Optional) Sets the interval at which to poll the interface for statistics and calculate throughput. Ranges from 30 to 600 seconds. The default is 30 seconds.
shutdown	(Optional) Shuts down this interface.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

An InlineGroup interface is a logical grouping of a pair of Ethernet ports that are physically contained on the optional Cisco WAE Inline Network Adapter or Cisco Interface Module. This command does not apply to the Cisco AppNav Controller Interface Module; use the **(config) bridge** command to configure an inline bridge on that platform.

You can have multiple InlineGroup interfaces, which allows for multiple bypass-enabled paths for traffic to pass through the WAE appliance, making multiple-router deployments possible. The InlineGroup interfaces provide failover capability and can be assigned to any set of VLANs. (For examples of InlineGroup interface configurations, see the **(config-if) inline** command.)

You can configure the InlineGroup interface for link speed (**bandwidth** or **autosense**) and mode of operation (**half-duplex** or **full-duplex**).

The failover timeout set by this command applies only to interfaces on the Cisco WAE Inline Network Adapter. To set the failover timeout for all interfaces together on the Cisco Interface Module, use the **(config) inline** command.

Note If the VLAN ID that you set with the **encapsulation dot1q** option does not match the VLAN ID expected by the router subinterface, you may not be able to connect to the inline interface IP address.

The inline adapter supports only a single VLAN ID for each inline group interface. If you have configured a secondary address from a different subnet on an inline interface, you must have the same secondary address assigned on the router subinterface for the VLAN.

Note We strongly recommend that you do not use half duplex on the WAE, routers, switches, or other devices. Use of half-duplex impedes system ability to improve performance and should not be used. Double-check each Cisco WAE interface as well as the port configuration on the adjacent device (router, switch, firewall, WAE) to verify that full duplex is configured.

Related Commands

(config) interface GigabitEthernet

(config) interface PortChannel

(config) interface standby

(config) interface TenGigabitEthernet

(config) interface virtual

show interface

show running-config

show startup-config

(config) interface PortChannel

To configure a port-channel interface, use the **interface** PortChannel global configuration command. To disable selected options, restore default values, or enable a shutdown interface, use the **no** form of this command.

interface PortChannel *index* [**description** *text* | **ip** {**access-group** {*acl-num* | *acl_name*} {**in** | **out**} | **address** *ip-address netmask*} | **load-interval** *seconds* | **shutdown** | **standby** *index* | **bridge-group** *bridge-id*]

no interface PortChannel *index* [**description** *text* | **ip** {**access-group** {*acl-num* | *acl_name*} {**in** | **out**} | **address** *ip-address netmask*} | **load-interval** *seconds* | **shutdown** | **standby** *index* | **bridge-group** *bridge-id*]

Syntax Description

PortChannel *index*	Configures an EtherChannel with an interface number of 1–7.
description *text*	(Optional) Enters a description of the interface.
ip	(Optional) Enables IP configuration commands for the interface.
access-group	Configures access control for IP packets on this interface using an access control list (ACL).
acl_num	Numeric identifier that identifies the ACL to apply to the current interface. For standard ACLs, the valid range is 1–99; for extended ACLs, the valid range is 100–199.
acl_name	Alphanumeric identifier of up to 30 characters, beginning with a letter that identifies the ACL to apply to the current interface.
in	Applies the specified ACL to inbound packets on the current interface.
out	Applies the specified ACL to outbound packets on the current interface.
address *ip-address netmask*	Sets the interface IP address and netmask.
load-interval *seconds*	(Optional) Sets the interval at which to poll the interface for statistics and calculate throughput. Ranges from 30 to 600 seconds. The default is 30 seconds.
shutdown	(Optional) Shuts down this interface.
standby *index*	(Optional) Includes the port-channel interface in the specified standby group (1-3).
bridge-group *bridge-id*	(Optional) Places the port-channel interface into the specified bridge group.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

central-manager

Usage Guidelines

Port channels (EtherChannels) for the WAAS software support the grouping of multiple same-speed network interfaces into one virtual interface. This configuration allows you to set or remove a virtual interface that consists of up to four physical interfaces (eight on an AppNav Controller Interface Module). Port channels also provide interoperability with Cisco routers, switches, and other networking devices or hosts that support port channels, load balancing, and automatic failure detection and recovery based on the current link status of each interface. You must configure port channels on the switch or router if you configure it on the WAE.

You cannot add an interface that already has a configured IP address, or is configured as primary or secondary, to a port channel.

You cannot remove a port-channel interface that is configured as the primary interface on a WAE.

Note You cannot use the inline Ethernet interfaces that are located on the Cisco WAE Inline Network Adapter to form a port-channel interface. However, you can use the interfaces on a Cisco Interface Module to form a port-channel interface.

Note No two interfaces can have IP addresses in the same subnet.

Examples

The following example shows how to create a port-channel interface. The port channel is port channel 1 and is assigned an IP address of 10.10.10.10 and a netmask of 255.0.0.0:

```
WAE# configure
WAE(config)# interface PortChannel 1
WAE(config-if)# ip address 10.10.10.10 255.0.0.0
WAE(config-if)# exit
```

The following example shows how to remove a port-channel interface:

```
WAE(config)# interface PortChannel 1
WAE(config-if)# no ip address 10.10.10.10 255.0.0.0
WAE(config-if)# exit
WAE(config)# no interface PortChannel 1
```

Related Commands

(config) interface GigabitEthernet

(config) interface InlineGroup

(config) interface standby

(config) interface TenGigabitEthernet

(config) interface virtual

(config) port-channel

show interface

show running-config

show startup-config

(config) interface standby

To configure a standby interface, use the **interface standby** global configuration command. To disable selected options, restore default values, or enable a shutdown interface, use the **no** form of this command.

interface standby *group-index* {**description** *text* | **ip address** *ip_address netmask* | **load-interval** *seconds* | **shutdown** | **bridge-group** *bridge-id*}

no interface standby *group-index* {**description** *text* | **ip address** *ip_address netmask* | **load-interval** *seconds* | **shutdown** | **bridge-group** *bridge-id*}

Syntax Description

group-index	Standby group interface. Specify a group index of 1–3, depending on the platform.
description *text*	Enters a description of the interface.
ip address *ip_address netmask*	Specifies the IP address and netmask of the interface.
load-interval *seconds*	(Optional) Sets the interval at which to poll the interface for statistics and calculate throughput. Ranges from 30 to 600 seconds. The default is 30 seconds.
shutdown	Shuts down this interface.
bridge-group *bridge-id*	Places the standby interface into the specified bridge group.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

central-manager

Usage Guidelines

Only one standby group is supported on the following WAAS devices: WAE-674, WAE-7341, WAE-7371, WAVE-274, WAVE-474, and WAVE-574. WAVE-294/594/694/7541/7571/8541 devices support up to two standby groups. All devices with the AppNav Controller Interface Module support up to three standby interfaces.

A standby group cannot be removed if it is configured as the system primary interface.

A standby group can have up to two member interfaces.

Note No two interfaces can have IP addresses in the same subnet.

Related Commands

(config) interface GigabitEthernet

(config) interface InlineGroup

(config) interface PortChannel

(config) interface TenGigabitEthernet

(config) interface virtual

show interface

show running-config

show startup-config

(config) interface TenGigabitEthernet

To configure a TenGigabitEthernet interface, use the **interface** global configuration command. To disable selected options, restore default values, or enable a shutdown interface, use the **no** form of this command.

interface TenGigabitEthernet *slot/port* [**cdp enable** | **channel-group** *index* | **description** *text* | **ip** {**access-group** {*acl-num* | *acl_name*} {**in** | **out**} | **address** {*ip_address netmask* [**secondary**] | **dhcp** [**client-id** *id*][**hostname** *name*]}} | **load-interval** *seconds* | **mtu** *mtusize* | **shutdown** | **standby** *group-index* [**primary**] | **bridge-group** *bridge-id*]

no interface TenGigabitEthernet *slot/port* [**cdp enable** | **channel-group** *index* | **description** *text* | **ip** {**access-group** {*acl-num* | *acl_name*} {**in** | **out**} | **address** {*ip_address netmask* [**secondary**] | **dhcp** [**client-id** *id*][**hostname** *name*]}} | **load-interval** *seconds* | **mtu** *mtusize* | **shutdown** | **standby** *group-index* [**primary**] | **bridge-group** *bridge-id*]

Syntax Description

slot/port	TenGigabitEthernet interface to configure (slot and port number). The slot number and port number are separated with a forward slash character (/). Valid slot and port values depend on the hardware platform.
cdp enable	(Optional) Enables Cisco Discovery Protocol (CDP) on the specified interface.
channel-group *index*	(Optional) Assigns the interface to the EtherChannel with the specified index (1–7).
description *text*	Enters a description of the interface.
ip	(Optional) Enables IP configuration commands for the interface.
access-group	Configures access control for IP packets on this interface using access control list (ACL).
acl_num	Numeric identifier that identifies the ACL to apply to the current interface. For standard ACLs, the valid range is 1–99; for extended ACLs, the valid range is 100–199.
acl_name	Alphanumeric identifier of up to 30 characters, beginning with a letter that identifies the ACL to apply to the current interface.
in	Applies the specified ACL to inbound packets on the current interface.
out	Applies the specified ACL to outbound packets on the current interface.
address *ip-address netmask*	Sets the interface IP address and netmask.
secondary	(Optional) Specifies that the configured address is a secondary IP address. If this keyword is omitted, the configured address is the primary IP address.
dhcp	(Optional) Sets the IP address to the address that is negotiated over Dynamic Host Configuration Protocol (DHCP).
client-id *id*	(Optional) Specifies the client identifier.
hostname *name*	(Optional) Specifies the hostname.

load-interval *seconds*	(Optional) Sets the interval at which to poll the interface for statistics and calculate throughput. Ranges from 30 to 600 seconds. The default is 30 seconds.
mtu *mtusize*	(Optional) Sets the interface Maximum Transmission Unit (MTU) size in bytes (576–1500).
shutdown	(Optional) Shuts down this interface.
standby *group-index*	(Optional) Sets the standby group number to *group-index*.
primary	(Optional) Sets this interface as the active interface in the standby group.
bridge-group *bridge-id*	Places the interface into the specified bridge group.

Defaults

The first attached interface in a standby group is defined as the active interface. There are no other default behaviors or values.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

central-manager

Usage Guidelines

Although the CLI contains the **no interface** option, you cannot apply the **no** command to an interface. The software displays the following error message: Removing of physical interface is not permitted.

Note Changing the MTU can cause network disruption for up to 30 seconds. The best practice is to make such changes when traffic interception is disabled or at an off-peak time when traffic disruption is acceptable.

Using the **cdp enable** command in global configuration mode enables CDP globally on all the interfaces. If you want to control CDP behavior per interface, use the **cdp enable** command in interface configuration mode. The interface level control overrides the global control.

To display the interface identifiers (for example, interface TenGigabitEthernet 1/0), use the **show running-config** or **show startup-config** commands. The **ip** and **shutdown** commands are listed separately in this command reference.

Note When you use the **ip address** command to change the IP address of an interface that has been shut down, it automatically brings up that interface by default.

Configuring Multiple Secondary IP Addresses on a Single Physical Interface

Use the **interface secondary** global configuration command to configure more than one IP address on the same interface. By configuring multiple IP addresses on a single interface, the WAAS device can be present in more than one subnet. This configuration allows you to optimize the response time because the content goes directly from the WAAS device to the requesting client without being redirected through a router. The WAAS device becomes visible to the client because they are configured on the same subnet.

You can assign up to four secondary addresses to an interface. These addresses become active only after you configure the primary address. No two interfaces can have the same IP address in the same subnetwork. To set these secondary IP addresses, use the **ip address** command.

If a WAAS device has one physical interface that has multiple secondary IP addresses assigned to it, the egress traffic uses the source IP address that is chosen by IP routing. If the secondary IP addresses of a WAAS device in the same subnet as the primary IP address, then the egress traffic uses the primary IP address only. If the secondary IP addresses are in a different subnet than the primary IP address, then the destination IP address determines which IP address on the WAAS device is used for the egress traffic.

Configuring Interfaces for DHCP

When you configure a WAAS device initially, you can configure a static IP address or use interface-level DHCP to dynamically assign IP addresses to the interfaces on the WAAS device.

If you do not enable interface-level DHCP on the WAAS device, you must manually specify a static IP address and network mask for the WAAS device. If the WAAS device moves to another location in another part of the network, you must manually enter a new static IP address and network mask for this WAAS device.

You can enable an interface for DHCP using the **ip address dhcp client-id** *id* **hostname** *name* interface configuration command. The client identifier is an ASCII value. The WAAS device sends its configured client identifier and hostname to the DHCP server when requesting network information. You can configure DHCP servers to identify the client identifier and the hostname that the WAAS device is sending and then send the specific network settings that are assigned to the WAAS device.

Note You must disable autoregistration before you can manually configure an interface for DHCP. Autoregistration is enabled by default on the first interface of the device.

Defining Interface Descriptions

You can specify a one-line description for a specific interface on a WAAS device. Use the **description** *text* interface configuration command to enter the description for the specific interface. The maximum length of the description text is 240 characters. This feature is supported for the Gigabit Ethernet, 10 Gigabit Ethernet, port-channel, standby, and bridge virtual interfaces.

After you define the description for an interface, use the **show** EXEC commands to display the defined interface descriptions. Enter the **show interface** *interface type slot/port* EXEC command to display the defined description for a specific interface on the WAE.

Configuring a Standby Group

You can associate an interface with a standby group by using the **standby** *group-index* interface configuration command. To make an interface the active interface in a standby group, use the **standby** *group-index* **primary** interface configuration command. If you have already associated an interface with a standby group but have not made it the primary interface, you cannot specify the command again to add the primary designation. First, remove the interface from the standby group, and then reassign it, specifying the **primary** option at the same time.

A physical interface can be a member of a standby group or a port channel, but not both.

If a device has only two interfaces, you cannot assign an IP address to both a standby group and a port channel. On such a device, only one virtual interface can be configured with an IP address.

Examples

The following example shows how to configure an attribute of an interface with a single CLI command:

```
WAE(config)# interface TenGigabitEthernet 1/0 ip access-group 1 in
```

The following example shows that an interface can be configured in a sequence of CLI commands:

```
WAE(config)# interface TenGigabitEthernet 1/0
WAE(config-if)# ip access-group 1 in
WAE(config-if)# exit
```

The following example shows how to enable a shut down interface:

```
WAE(config)# no interface TenGigabitEthernet 1/0 shutdown
```

The following example shows how to add an interface to a channel group:

```
WAE# configure
WAE(config)# interface TenGigabitEthernet 1/0
WAE(config-if)# channel-group 1
WAE(config-if)# exit
```

The following example shows how to remove an interface from a channel group:

```
WAE(config)# interface TenGigabitEthernet 1/0
WAE(config-if)# no channel-group 1
WAE(config-if)# exit
```

The following example shows how to assign a secondary IP address on a TenGigabitEthernet interface:

```
WAE# configure
WAE(config)# interface TenGigabitEthernet 1/0
WAE(config-if)# ip address 10.10.10.10 255.0.0.0 secondary
```

The following example shows how to configure a description for a TenGigabitEthernet interface:

```
WAE(config)# interface TenGigabitEthernet 1/0
WAE(config-if)# description This is a TenGigabitEthernet interface.
```

Related Commands

(config) interface GigabitEthernet

(config) interface InlineGroup

(config) interface PortChannel

(config) interface standby

(config) interface virtual

show interface

show running-config

show startup-config

(config) interface virtual

To configure a virtual interface, use the **interface** virtual global configuration command. To disable selected options, restore default values, or enable a shutdown interface, use the **no** form of this command.

interface virtual *slot/port* [**cdp enable** | **description** *text* |
ip {**access-group** {*acl-num* | *acl_name*} {**in** | **out**} | **address** {*ip_address netmask* [**secondary**] | **dhcp** [**client-id** *id*][**hostname** *name*]}} | **mtu** *mtusize* | **shutdown**]

no interface virtual *slot/port* [**cdp enable** | **description** *text* |
ip {**access-group** {*acl-num* | *acl_name*} {**in** | **out**} | **address** {*ip_address netmask* [**secondary**] | **dhcp** [**client-id** *id*][**hostname** *name*]}} | **mtu** *mtusize* | **shutdown**]

Syntax Description

slot/port	vWAAS interface to configure (slot and port number). The slot range is 1–2; the port range is 0. The slot number and port number are separated with a forward slash character (/).
cdp enable	(Optional) Enables Cisco Discovery Protocol (CDP) on the specified interface.
description *text*	Enters a description of the interface.
ip	(Optional) Enables IP configuration commands for the interface.
access-group	Configures access control for IP packets on this interface using access control list (ACL).
acl_num	Numeric identifier that identifies the ACL to apply to the current interface. For standard ACLs, the valid range is 1–99; for extended ACLs, the valid range is 100–199.
acl_name	Alphanumeric identifier of up to 30 characters, beginning with a letter that identifies the ACL to apply to the current interface.
in	Applies the specified ACL to inbound packets on the current interface.
out	Applies the specified ACL to outbound packets on the current interface.
address *ip-address netmask*	Sets the interface IP address and netmask.
secondary	(Optional) Specifies that the configured address is a secondary IP address. If this keyword is omitted, the configured address is the primary IP address.
dhcp	(Optional) Sets the IP address to the address that is negotiated over Dynamic Host Configuration Protocol (DHCP).
client-id *id*	(Optional) Specifies the client identifier.
hostname *name*	(Optional) Specifies the hostname.
mtu *mtusize*	(Optional) Sets the interface Maximum Transmission Unit (MTU) size in bytes (576–1500).
shutdown	(Optional) Shuts down this interface.

Defaults

No default behavior or values.

Command Modes global configuration

Device Modes application-accelerator

central-manager

Usage Guidelines Using the **cdp enable** command in global configuration mode enables CDP globally on all the interfaces. If you want to control CDP behavior per interface, use the **cdp enable** command in interface configuration mode. The interface level control overrides the global control.

To display the interface identifiers (for example, interface virtual 1/0), use the **show running-config** or **show startup-config** commands.

Note When you use the **ip address** command to change the IP address of an interface that has been shut down, it automatically brings up that interface by default.

Configuring Interfaces for DHCP

When you configure a WAAS device initially, you can configure a static IP address or use interface-level DHCP to dynamically assign IP addresses to the interfaces on the WAAS device.

If you do not enable interface-level DHCP on the WAAS device, you must manually specify a static IP address and network mask for the WAAS device. If the WAAS device moves to another location in another part of the network, you must manually enter a new static IP address and network mask for this WAAS device.

You can enable an interface for DHCP using the **ip address dhcp client-id** *id* **hostname** *name* interface configuration command. The client identifier is an ASCII value. The WAAS device sends its configured client identifier and hostname to the DHCP server when requesting network information. You can configure DHCP servers to identify the client identifier and the hostname that the WAAS device is sending and then send the specific network settings that are assigned to the WAAS device.

Note You must disable autoregistration before you can manually configure an interface for DHCP. Autoregistration is enabled by default on the first interface of the device.

Defining Interface Descriptions

You can specify a one-line description for a specific interface on a WAAS device. Use the **description** *text* interface configuration command to enter the description for the specific interface. The maximum length of the description text is 240 characters.

After you define the description for an interface, use the **show** EXEC commands to display the defined interface descriptions. Enter the **show interface virtual** EXEC command to display the defined description for a virtual interface on the WAE.

Examples The following example shows how to assign a secondary IP address on a virtual interface on a vWAAS device:

```
WAE# configure
WAE(config)# interface virtual 1/0
WAE(config-if)# ip address 10.10.10.10 255.0.0.0 secondary
```

The following example shows how to configure a description for a virtual interface:

```
WAE(config)# interface virtual 1/0
WAE(config-if)# description This is a virtual interface.
```

Related Commands

(config) interface GigabitEthernet

(config) interface InlineGroup

(config) interface PortChannel

(config) interface standby

(config) interface TenGigabitEthernet

show interface

show running-config

show startup-config

(config) ip

To change the initial network device configuration settings, use the **ip** global configuration command. To delete or disable these settings, use the **no** form of this command.

ip {**default-gateway** [**management**] *ip-address* | **domain-name** *name1 name2 name3* | **ftp management** | **host** *hostname ip-address* | **name-server** *ip-addresses* | **radius management** | **tacacs management** | **path-mtu-discovery enable** | **route** [**management**] *dest_addrs net_addrs gateway_addrs* | **tftp management**}

no ip {**default-gateway** [**management**] *ip-address* | **domain-name** *name1 name2 name3* | **ftp management** | **host** *hostname ip-address* | **name-server** *ip-addresses* | **radius management** | **tacacs management** | **path-mtu-discovery enable** | **route** [**management**] *dest_addrs net_addrs [gateway_addrs]* | **tftp management**}

Syntax Description

default-gateway *ip-address*	Specifies the IP address of the default gateway (if not routing IP).
management	Specifies that the default gateway or net route is for the management interface.
domain-name *name1 name2 name3*	Specifies domain names (up to three can be specified).
ftp management	Configures the device to use the management interface for FTP traffic.
host *hostname ip-address*	Adds an entry to the /etc/hosts file on the device, mapping the specified hostname to the specified IP address of the host.
name-server *ip-addresses*	Specifies the address of the name server and IP addresses of the name servers (up to a maximum of eight).
radius management	Configures the device to use the management interface for radius traffic.
tacacs management	Configures the device to use the management interface for tacacs traffic.
path-mtu-discovery enable	Enables RFC 1191 Path Maximum Transmission Unit (MTU) discovery.
route *dest_addrs net_addrs gateway_addrs*	Specifies the net route (destination route address, netmask address, and gateway address).
tftp management	Configures the device to use the management interface for TFTP traffic.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

appnav-controller

Usage Guidelines

To define a default gateway, use the **ip default-gateway** command. If you have designated a management interface, you can configure a different default gateway for the management interface by using the **management** keyword. To remove the IP default gateway, use the **no** form of this command. The WAAS device uses the default gateway to route IP packets when there is no specific route found to the destination.

To define a default domain name, use the **ip domain-name** command. To remove the IP default domain name, use the **no** form of this command. You can enter up to three domain names. If a request arrives without a domain name appended in its hostname, the proxy tries to resolve the hostname by appending *name1*, *name2*, and *name3* in that order until one of these names succeeds.

To add an entry to the /etc/hosts file on the device, mapping a hostname to an IP address, use the **ip host** command. A given hostname can be mapped only to a single IP address, while an IP address can have multiple hostnames mapped to it, each one through a separate issuance of this command. To remove the entry from the /etc/hosts file, use the **no** form of this command. You can use the **show hosts** EXEC command to display the contents of the /etc/hosts file.

To specify the address of one or more name servers to use for name and address resolution, use the **ip name-server** *ip-addresses* command. To disable IP name servers, use the **no** form of this command. For proper resolution of the hostname to the IP address or the IP address to the hostname, the WAAS device uses DNS servers. Use the **ip name-server** command to point the WAAS device to a specific DNS server. You can configure up to eight servers.

Path MTU autodiscovery discovers the MTU and automatically sets the correct value. Use the **ip path-mtu-discovery enable** command to start this autodiscovery utility. By default, this feature is disabled because the WAE does not receive ICMP packets. When this feature is disabled, the sending device uses a packet size that is smaller than 576 bytes and the next hop MTU. Existing connections are not affected when this feature is turned on or off.

Use the **ip route** command to add a specific static route for a network or host. Any IP packet designated for the specified destination uses the configured route.

To configure static IP routing, use the **ip route** command. To remove the route, use the **no** form of this command. Do not use the **ip route 0.0.0.0 0.0.0.0** command to configure the default gateway; use the **ip default-gateway** command instead.

Examples

The following example shows how to configure a default gateway for the WAAS device:

```
WAE(config)# ip default-gateway 192.168.7.18
```

The following example shows how to configure a default gateway for the management interface on the WAAS device, if it is different from the standard default gateway:

```
WAE(config)# ip default-gateway management 192.168.10.35
```

The following example shows how to configure a static IP route for the WAAS device:

```
WAE(config)# ip route 172.16.227.128 255.255.255.0 172.16.227.250
```

The following example shows how to configure a default domain name for the WAAS device:

```
WAE(config)# ip domain-name cisco.com
```

The following example shows how to add an entry to the /etc/hosts file on the WAAS device:

```
WAE(config)# ip host corp-B7 10.11.12.140
```

The following example shows how to configure a name server for the WAAS device:

```
WAE(config)# ip name-server 10.11.12.13
```

Related Commands

show hosts

show ip routes

(config) ip access-list

To create and modify access lists on a WAAS device for controlling access to interfaces or applications, and to define subnets, use the **ip access-list** global configuration command. To disable an access list, use the **no** form of this command.

ip access-list {**standard** {*acl-name* | *acl-num*} | **extended** {*acl-name* | *acl-num*} | **logging**}

no ip access-list {**standard** {*acl-name* | *acl-num*} | **extended** {*acl-name* | *acl-num*} | **logging**}

Syntax Description

standard	Enables standard ACL configuration mode. The CLI enters the standard ACL configuration mode in which all subsequent commands apply to the current standard access list. The (config-std-nacl) prompt appears: WAE(config-std-nacl)# See the "Standard ACL Configuration Mode Commands" section for details about working with entries in a standard access list and the commands available from the standard ACL configuration mode (config-std-nacl)#.
extended	Enables extended ACL configuration mode. The CLI enters the extended ACL configuration mode in which all subsequent commands apply to the current extended access list. The (config-ext-nacl) prompt appears: `WAE(config-ext-nacl)#` See the "Extended ACL Configuration Mode Commands" section for details about working with entries in an extended access list and the commands available from the extended ACL configuration mode (config-ext-nacl)#.
acl-name	Access list to which all commands entered from ACL configuration mode apply, using an alphanumeric string of up to 30 characters, beginning with a letter.
acl-num	Access list to which all commands entered from access list configuration mode apply, using a numeric identifier. For standard access lists, the valid range is 1 to 99; for extended access lists, the valid range is 100 to 199.
logging	Enables logging for all IP access lists.

Defaults

An access list drops all packets unless you configure at least one **permit** entry.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Within ACL configuration mode, you can use the editing commands (**list**, **delete**, and **move**) to display the current condition entries, to delete a specific entry, or to change the order in which the entries will be evaluated. To return to global configuration mode, use the **exit** command at the ACL configuration mode prompt.

To create an entry, use a the **deny** or **permit** keyword and specify the type of packets that you want the WAAS device to drop or to accept for further processing. By default, an access list denies everything because the list is terminated by an implicit **deny any** entry. You must include at least one **permit** entry to create a valid access list.

Note IP ACLs that are defined on a router take precedence over the IP ACLs that are defined on the WAE. IP ACLs that are defined on a WAE take precedence over the WAAS application definition policies that are defined on the WAE.

After creating an access list, you can include the access list in an access group using the **access-group** command, which determines how the access list is applied. You can also apply the access list to a specific application using the appropriate command. A reference to an access list that does not exist is the equivalent of a **permit any** condition statement.

To work with access lists, enter either the **ip access-list standard** or **ip access-list extended** global configuration command. Identify the new or existing access list with a name up to 30 characters long beginning with a letter, or with a number. If you use a number to identify a standard access list, it must be between 1 and 99; for an extended access list, use a number from 100 to 199. You must use a standard access list for providing access to the SNMP server or to the TFTP gateway/server. However, you can use either a standard access list or an extended access list for providing access to the WCCP application.

After you identify the access list, the CLI enters the appropriate configuration mode and all subsequent commands apply to the specified access list. The prompt for each configuration mode is shown in the following examples.

```
WAE(config)# ip access-list standard test
WAE(config-std-nacl)# exit
WAE(config)# ip access-list extended test2
WAE(config-ext-nacl)#
```

To define a subnet, use either a standard or an extended ACL. In an HTTP AO subnet configuration, the **access-list** option must have at least one condition statement in it for it to exist. The list is terminated by an implicit **deny any** (standard access list) or **deny ip any any** (extended access list) condition statement. This statement applies to HTTP AO optimizations unless the ACL has an explicit **permit all** statement in it. If an *acl name* or *acl number* does not exist (if no condition statements exist in the access list), it is considered as an implicit **permit any** (standard access list) or **permit ip any any**(extended access list) condition statement. We recommend that you explicitly add **permit any** or **deny any** at the end of the ACL to make all the conditions clear for the subnet feature.

Use the **ip access-list logging** command to log denied packets.

Examples

The following example shows how to create an access list on the WAAS device. You create this access list to allow the WAAS device to accept all web traffic that is redirected to it but limit host administrative access using SSH:

```
WAE(config)# ip access-list extended example
WAE(config-ext-nacl)# permit tcp any any eq www
WAE(config-ext-nacl)# permit tcp host 10.1.1.5 any eq ssh
WAE(config-ext-nacl)# exit
```

The following example shows how to activate the access list for an interface:

```
WAE(config)# interface gigabitethernet 1/0
WAE(config-if)# ip access-group example in
WAE(config-if)# exit
```

The following example shows how this configuration appears when you enter the **show running-configuration** command:

```
...
!
interface GigabitEthernet 1/0
 ip address 10.1.1.50 255.255.0.0
 ip access-group example in
 exit
. . .
ip access-list extended example
 permit tcp any any eq www
 permit tcp host 10.1.1.5 any eq ssh
 exit
. . .
```

The following example shows how to configure an ACL to define a subnet:

```
WAE(config)# ip access-list extended md_acl
WAE(config-ext-nacl)# permit ip 2.57.34.0 0.0.0.255 2.57.34.0 0.0.0.255
WAE(config-ext-nacl)# exit
WAE(config)# ip access-list standard 10
WAE(config-std-nacl)# deny 1.1.1.0 0.0.0.255
WAE(config-std-nacl)# permit any
WAE(config-std-nacl)# exit
```

(config) ip icmp rate-limit unreachable

To limit the rate at which Internet Control Message Protocol (ICMP) destination unreachable messages are generated, use the **ip icmp rate-limit unreachable** command in global configuration mode. To remove the rate limit, use the no form of this command.

ip icmp rate-limit unreachable df *microseconds*

no ip icmp rate-limit unreachable df *microseconds*

Syntax Description

df	Limits the rate ICMP destination unreachable messages are sent when Type 3 code 4, destination unreachable, don't fragment (DF) bit sent and fragmentation required, is specified in the IP header of the ICMP destination unreachable message.
microseconds	Time limit (in microseconds) in which one ICMP destination unreachable message is sent. The range is 250 microseconds to 1000000 microseconds.

Defaults

The default value is one ICMP destination unreachable message per 500 microseconds.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

This feature is enabled by default. The no ip icmp rate-limit unreachable df command turns off the previously configured rate limit.

The software maintains two timers: one for general destination unreachable messages and one for DF destination unreachable messages. Both share the same time limits and defaults. If the df option is not configured, the ip icmp rate-limit unreachable command sets the time values for DF destination unreachable messages. If the df option is configured, its time values remain independent from those of general destination unreachable messages.

Examples

The following example sets the rate of the ICMP destination unreachable message to one message every 10 microseconds:

```
WAE(config)# ip icmp rate-limit unreachable df 10
```

The following example turns off the previously configured rate limit:

```
WAE(config)# no ip icmp rate-limit unreachable df
```

Related Commands

clear arp-cache

(config-if) ip access-group

show ip access-list

(config) ip unreachables df

(config) ip unreachables df

To enable the generation of Internet Control Message Protocol (ICMP) unreachable messages, use the ip unreachables df command in global configuration mode. To disable this function, use the no form of this command.

ip unreachables df

no ip unreachables df

SyntaxDescription

df	Limits the rate ICMP destination unreachable messages are sent when Type 3 code 4, destination unreachable, don't fragment (DF) bit sent and fragmentation required, is specified in the IP header of the ICMP destination unreachable message.

Defaults

The default value is one ICMP destination unreachable message per 500 microseconds.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

If the software receives a nonbroadcast packet destined for itself that uses an unknown protocol, it sends an ICMP protocol unreachable message back to the source. Similarly, if the software receives a packet that it is unable to deliver to the ultimate destination because it knows of no route to the destination address, it sends an ICMP host unreachable message to the source. This feature is enabled by default.

ExamplesExamples

The following example enables the generation of ICMP unreachable messages, as appropriate, on an interface:

```
WAE(config)# interface ethernet 0
WAE(config)# ip unreachables df
```

Related Commands

clear arp-cache

(config-if) ip access-group

show ip access-list

(config) ip icmp rate-limit unreachable

(config) kerberos

To authenticate a user that is defined in the Kerberos database, use the **kerberos** global configuration command. To disable authentication, use the **no** form of this command.

kerberos {**dns** | ~~**local-realm** *kerberos-realm* | **realm** {*dns-domain* | *host*} *kerberos-realm* | **server** *kerberos-realm* {*hostname* | *ip-address*} [*port-number*]}~~

no kerberos {**dns** | ~~**local-realm** *kerberos-realm* | **realm** {*dns-domain* | *host*} *kerberos-realm* | **server** *kerberos-realm* {*hostname* | *ip-address*} [*port-number*]}~~

Syntax Description

dns	Enables or disables DNS lookup for Kerberos.
~~**local-realm** *kerberos-realm*~~	~~Displays the default Kerberos realm (IP address or name in uppercase letters) for WAAS. Configures a switch to authenticate users defined in the Kerberos database. The default value is a null string.~~
~~**realm**~~	~~Maps a hostname or DNS domain name to a Kerberos realm.~~
~~*dns-domain*~~	~~DNS domain name to map to the Kerberos realm.~~ ~~**Note** The name must begin with a leading dot (.).~~
~~*host*~~	~~Host IP address or name to map to Kerberos host realm.~~
~~*kerberos-realm*~~	~~Kerberos realm (IP address or name in uppercase letters). The default value is a null string.~~
~~**server**~~	~~Specifies the Key Distribution Center (KDC) to use in a given Kerberos realm and, optionally, the port number that the KDC is monitoring.~~
~~*hostname*~~	~~Name of the host running the KDC.~~
~~*ip-address*~~	~~IP address of the host running the KDC.~~
~~*port-number*~~	~~(Optional) Number of the port on the KDC server.~~

Defaults

kerberos-realm: NULL string

port-number: 88

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

All Windows 2000 domains are also Kerberos realms. Because the Windows 2000 domain name is also a DNS domain name, the Kerberos realm name for the Windows 2000 domain name is always in uppercase letters. This capitalization follows the recommendation for using DNS names as realm names in the Kerberos Version 5 protocol document (RFC-1510) and affects only interoperability with other Kerberos-based environments.

Note Your Windows domain server must have a Reverse DNS Zone configured for this command to execute successfully.

The KDC server and all hosts with Kerberos authentication configured must interact within a 5-minute window or authentication will fail. All hosts, especially the KDC, should be running NTP. For information about configuring NTP, see the **(config) ntp** command.

The KDC server and Admin server must have the same IP address. The default port number for both servers is port 88.

The **kerberos** command modifies the krb5.conf file.

Examples

The following example shows how to configure the WAAS device to authenticate with a specified KDC in a specified Kerberos realm. The configuration is then verified.

```
WAE(config)# kerberos ?
  local-realm  Set local realm name
  realm        Add domain to realm mapping
  server       Add realm to host mapping
WAE(config)# kerberos local-realm WAE.ABC.COM
WAE(config)# kerberos realm wae.abc.com WAE.ABC.COM
WAE(config)# kerberos server wae.abc.com 10.10.192.50
WAE(config)# exit
WAE# show kerberos
  Kerberos Configuration:
  -----------------------
    Local Realm: WAE.ABC.COM
    DNS suffix: wae.abc.com
    Realm for DNS suffix: WAE.ABC.COM
    Name of host running KDC for realm:
    Master KDC: 10.10.192.50
    Port: 88
```

Related Commands

show kerberos

(config) kernel kdb

To enable access to the kernel debugger (kdb), use the **kernel kdb** global configuration command. To disable access to the kernel debugger, use the **no** form of this command.

kernel kdb

no kernel kdb

Syntax Description

This command has no arguments or keywords.

Defaults

The kernel debugger is disabled by default.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Once enabled, kdb is automatically activated if kernel problems occur, or you can manually activate it from the local console for the WAAS device. Once activated, all normal functioning of the WAAS device is suspended until kdb is manually deactivated. The kdb prompt looks like this:

```
[0]kdb>
```

To deactivate kdb, enter the **go** command at the kdb prompt. If kdb was automatically activated because of kernel problems, the system generates a core dump and restarts. If you activated kdb manually for diagnostic purposes, the system resumes normal functioning in whatever state it was when you activated kdb. In either case, if you enter the **reboot** command, the system restarts and normal operation resumes.

kdb is disabled by default and you must enter the **kernel kdb** command in global configuration mode to enable it. If kdb has been previously enabled, you can enter the **no kernel kdb** global configuration command to disable it. When kdb is enabled, you can activate it manually from the local console by pressing **Ctrl-_** followed by **Ctrl-B**. On a vWAAS device, kdb can be enabled by pressing the **Esc** key and typing **kdb**.

The WAAS device is often unattended at many sites, and it is desirable for the WAAS device to automatically reboot after generating a core dump instead of requiring user intervention. Disabling the kernel debugger allows automatic recovery.

Examples

The following example shows how to enable, and then disable, access to the kernel debugger:

```
WAE(config)# kernel kdb
WAE(config)# no kernel kdb
```

Related Commands (config) kernel kdump enable

(config) kernel kdump enable

To enable the kernel crash dump mechanism, use the **kernel kdump enable** global configuration command. To disable the kernel crash dump mechanism, use the **no** form of this command.

kernel kdump enable

no kernel kdump enable

Syntax Description

This command has no arguments or keywords.

Defaults

The kernel crash dump mechanism is enabled by default.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

A kernal crash dump file is stored in the following disk location:
/local/local1/crash/*timestamp*/vmcore

The analysis of the kernal crash dump file is stored in the following file:
/local/local1/crash/*timestamp*/analysis.txt

Examples

The following example shows how to enable, and then disable, the kernel crash dump mechanism:

```
WAE(config)# kernel kdump enable
WAE(config)# no kernel kdump enable
```

Related Commands

(config) kernel kdb

show kdump

(config) line

To specify terminal line settings, use the **line** global configuration command. To configure the WAAS device to not check for the carrier detect signal, use the **no** form of this command.

line console carrier-detect

no line console carrier-detect

Syntax Description

console	Configures the console terminal line settings.
carrier-detect	Sets the device to check the carrier detect signal before writing to the console.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Examples

The following example shows how to set the WAAS device to check for the carrier detect signal:

```
WAE(config)# line console carrier-detect
```

(config) logging console

To set system logging to console, use the **logging console** global configuration command. To disable logging functions, use the **no** form of this command.

logging console {**enable** | **priority** *loglevel*}

no logging console {**enable** | **priority** *loglevel*}

Syntax Description

enable	Enables system logging.
priority *loglevel*	Sets which priority level messages to send. Use one of the following keywords or you can specify the numeric priority: • **alert**—Immediate action needed. Priority 1. • **critical**—Immediate action needed. Priority 2. • **debug**—Debugging messages. Priority 7. • **emergency**—System is unusable. Priority 0. • **error**—Error conditions. Priority 3. • **information**—Informational messages. Priority 6. • **notice**—Normal but significant conditions. Priority 5. • **warning**—Warning conditions. Priority 4.

Defaults

Logging: on

Priority of message for console: warning (4)

Log file: /local1/syslog.txt

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Use the **logging** command to set specific parameters of the system log file.

You can configure logging to send various levels of messages to the console using the **logging console priority** option.

Examples

The following example shows how to send messages that have a priority code of “error” (Level 3) to the console:

```
WAE(config)# logging console priority error
```

The following example shows how to disable sending of messages that have a priority code of "error" (level 3) to the console:

```
WAE(config)# no logging console error
```

Related Commands

clear arp-cache

show logging

(config) logging disk

To system logging to a disk file, use the **logging disk** global configuration command. To disable logging functions, use the **no** form of this command.

logging disk {**enable** | **filename** *filename* | **priority** *loglevel* | **recycle** *size*}

no logging disk {**enable** | **filename** *filename* | **priority** *loglevel* | **recycle** *size*}

Syntax Description

enable	Enables system logging.
filename *filename*	Sets the name of the syslog file.
priority *loglevel*	Sets which priority level messages to send. Use one of the following keywords or you can specify the numeric priority: • **alert**—Immediate action needed. Priority 1. • **critical**—Immediate action needed. Priority 2. • **debug**—Debugging messages. Priority 7. • **emergency**—System is unusable. Priority 0. • **error**—Error conditions. Priority 3. • **information**—Informational messages. Priority 6. • **notice**—Normal but significant conditions. Priority 5. • **warning**—Warning conditions. Priority 4.
recycle *size*	Overwrites *syslog.txt* when it surpasses the recycle size (1000000–50000000 bytes).

Defaults

Logging: on

Priority of message for disk log file: debug (7)

Log file: /local1/syslog.txt

Log file recycle size: 10,000,000 bytes

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Use the **logging** command to set specific parameters of the system log file.

The **no logging disk recycle size** command sets the file size to the default value. Whenever the current log file size surpasses the recycle size, the log file is rotated. The log file cycles through at most five rotations, and they are saved as [*log file name*].[*1-5*] under the same directory as the original log. The rotated log file is the one configured using the **logging disk filename** command.

Examples

The following example shows how to send messages that have a priority code of "error" (level 3) to a file:

```
WAE(config)# logging disk priority error
```

Related Commands

clear arp-cache

show logging

(config) logging facility

To set the facility parameter for system logging, use the **logging facility** global configuration command. To disable logging functions, use the **no** form of this command.

logging facility *facility*

no logging facility *facility*

Syntax Description

facility	Facility parameter for syslog messages. Use one of the following keywords: • **auth**—Authorization system • **daemon**—System daemons • **kernel**—Kernel • **local0**—Local use • **local1**—Local use • **local2**—Local use • **local3**—Local use • **local4**—Local use • **local5**—Local use • **local6**—Local use • **local7**—Local use • **mail—**Mail system • **news—**USENET news • **syslog—**Syslog itself • **user—**User process • **uucp—**UUCP system

Defaults

Logging: on

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Examples

The following example shows how to set the facility parameter to authorization system for syslog messages:

```
WAE(config)# logging facility auth
```

Related Commands clear arp-cache
show logging

(config) logging host

To configure system logging to a remote host, use the **logging host** global configuration command. To disable logging functions, use the **no** form of this command.

logging host {*hostname* | *ip-address*} [**port** *port_num* | **priority** *loglevel* | **rate-limit** *message_rate*]

no logging host {*hostname* | *ip-address*} [**port** *port_num* | **priority** *loglevel* | **rate-limit** *message_rate*]

Syntax Description

hostname	Hostname of the remote syslog host. Specify up to four remote syslog hosts. **Note** To specify more than one syslog host, use multiple command lines; specify one host per command.
ip-address	IP address of the remote syslog host. Specify up to four remote syslog hosts. **Note** To specify more than one syslog host, use multiple command lines; specify one host per command.
port *port_num*	(Optional) Specifies the port to be used when logging to a host. The default port is 514.
priority *loglevel*	(Optional) Sets which priority level messages to send. Use one of the following keywords or you can specify the numeric priority: • **alert**—Immediate action needed. Priority 1. • **critical**—Immediate action needed. Priority 2. • **debug**—Debugging messages. Priority 7. • **emergency**—System is unusable. Priority 0. • **error**—Error conditions. Priority 3. • **information**—Informational messages. Priority 6. • **notice**—Normal but significant conditions. Priority 5. • **warning**—Warning conditions. Priority 4.
rate-limit *message_rate*	(Optional) Sets the rate limit (in messages per second) for sending messages to a host. Rate limit is 0-10000 (in messages per second). Setting the rate limit to 0 disables rate limiting.

Defaults

Logging: on

Priority of message for a host: warning (4)

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Use the **logging** command to set specific parameters of the system log file.

To configure the WAAS device to send varying levels of event messages to an external syslog host, use the **logging host** option.

You can configure a WAAS device to send varying levels of messages to up to four remote syslog hosts using the **logging host hostname** command.

Examples

The following example shows how to send messages that have a priority code of "error" (level 3) to the remote syslog host that has an IP address of 172.31.2.160:

```
WAE(config)# logging host 172.31.2.160 priority error
```

Related Commands

clear arp-cache

show logging

(config) ntp

To configure the NTP server and to allow the system clock to be synchronized by a time server, use the **ntp** global configuration command. To disable this function, use the **no** form of this command.

ntp [**authenticate** | **authentication-key** *key-num* [**md5** *authentication-key*] |
server {*ip-address* | *hostname*} [*ip-addresses* | *hostnames*] |
server-with-authentication {*ip-address* | *hostname*} **key** *key-num*]

ntp [**authenticate** | **authentication-key** *authentication-key* [**md5** *encryption-type*] |
server {*ip-address* | *hostname*} [*ip-addresses* | *hostnames*] |
server-with-authentication {*ip-address* | *hostname*} **key** *authentication-key*]

no ntp [**authenticate** | **authentication-key** *key-num* [**md5** *authentication-key*] |
server {*ip-address* | *hostname*} [*ip-addresses* | *hostnames*] |
server-with-authentication {*ip-address* | *hostname*} **key** *key-num*]

Syntax Description

authenticate	(Optional) Authenticates the NTP server.
authentication-key *key-num*	(Optional) Sets the ID of the NTP authentication key. Maximum of 4 authentication keys can be configured. The ID must be a positive integer.
md5 *authentication-key*	(Optional) Sets the value for the NTP authentication key (type MD5). The key value must be from 0 to 4294967295.
server	(Optional) Sets the NTP server IP address for the WAAS device.
ip-address	NTP server IP address.
hostname	NTP server hostname.
ip-addresses	(Optional) IP address of the time server that provides the clock synchronization (maximum of 4).
hostnames	(Optional) Hostname of the time server that provides the clock synchronization (maximum of 4).
server-with-authentication	(Optional) Sets the authentication NTP server IP address for the WAAS device.
key *key-num*	(Optional) Sets the NTP authentication key ID for the authentication NTP server.

Defaults

The default NTP version number is 3.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Note Unexpected time changes can result in unexpected system behavior. We recommend reloading the system after enabling an NTP server.

Examples

The following example shows how to specify the NTP server IP address as the time source for a WAAS device. It also removes this configuration.

```
WAE(config)# ntp 172.16.22.44
WAE(config)# no ntp 172.16.22.44
```

clock

(config) clock

show clock

show ntp

(config) peer

To enable peer optimization, use the **peer** global configuration command. To disable peer optimization, use the **no** form of this command.

peer device-id *deviceid* [**description** *description*] **optimization enable**

no peer device-id *deviceid* [**description** *description*] **optimization enable**

Syntax Description

device-id *deviceid*	Configures the device ID of the peer device with which to enable or disable optimization.
description *hostname*	(Optional) Configures a string that is the device description of the peer device. You should use the hostname of the peer WAE for the description.
optimization enable	Enables optimization with the specified peer.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

Use the **no peer** command to disable optimization between peer devices in a serial cluster.

Use the **peer** command to reenable optimization between peer devices if it has been disabled previously.

The *deviceid* is a hexadecimal string (for example, d4:65:01:40:40:8a) that you can obtain with the **show device-id** or **show hardware** EXEC commands.

You can configure optimization for only one peer device with this command.

Examples

The following example shows how to disable optimization with a serial peer device:

```
WAE(config)# no peer device-id d4:65:01:40:40:8a description wae-sj-dc2 optimization
enable
```

Related Commands

show device-id

show hardware

(config) interception

(config) policy-map

To configure an AppNav or optimization policy map, use the **policy-map** global configuration command. To unconfigure settings, use the **no** form of this command.

policy-map type {**appnav** | **waas**} *policymap-name* [**rename** *new-name*]

no policy-map type {**appnav** | **waas**} *policymap-name*

Syntax Description

appnav	Configures an AppNav policy map.
waas	Configures a WAAS optimization policy map.
policymap-name	Policy map name (up to 40 alpha-numeric characters and hyphen, beginning with a letter).
rename *new-name*	(Optional) Renames the policy map with the specified new name.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

Usage Guidelines

Use the **policy-map** command to add or modify policy maps that associate policy actions with class maps. This command invokes the Policy Map configuration mode, which is indicated by a different prompt (config-pmap). For more information on Policy Class Map configuration mode commands, see the "Policy Map Configuration Mode Commands" section. To return to global configuration mode, enter the **exit** command.

You can delete a policy map by using the **no** form of this command.

The WAAS software comes with many class maps and policy rules that help your WAAS system classify and optimize some of the most common traffic on your network. Before you create a new class map or policy rule, we recommend that you review the default class map and policy rules and modify them as appropriate. It is usually easier to modify an existing class map or policy rule than to create a new one. For a list of the default applications, class maps, and policy rules, see the *Cisco Wide Area Application Services Configuration Guide*.

Note We strongly recommend that you use the WAAS Central Manager GUI to centrally configure policy maps for your WAAS devices. For more information, see the *Cisco Wide Area Application Services Configuration Guide*.

Examples

The following example shows how to configure a WAAS optimization policy map:

```
wae(config)# policy-map type waas myPolicy
wae(config-pmap)# description My optimization policy
wae(config-pmap)# class httpx
wae(config-pmap-c)# optimize full accelerate http application Web
```

The following example shows how to configure an AppNav policy map:

```
wae(config)# policy-map type appnav mypolicy
wae(config-pmap)# description My AppNav policy
wae(config-pmap)# class httpx
wae(config-pmap-c)# distribute service-node-group wng3
wae(config-pmap-c)# monitor-load http
```

Related Commands

(config) class-map

(config) service-policy

(config) port-channel

To configure port channel load-balancing on a WAAS device, use the **port-channel** global configuration command. To set load balancing on the port channel to its default method, use the **no** form of this command.

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

no port-channel load-balance {**src-dst-ip** | **src-dst-ip-port**}

Syntax Description

load-balance	Configures the load-balancing method.
src-dst-ip	Specifies the load-balancing method based on a combination of source and destination IP addresses.
src-dst-ip-port	Specifies the load-balancing method based on a combination of source and destination IP addresses/ports.

Defaults

src-dst-ip-port is the default load-balancing method.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

central-manager

Examples

The following example shows how to configure src-dst-ip load balancing on a port channel and then disable it:

```
WAE(config)# port-channel load-balance src-dst-ip
WAE(config)# no port-channel load-balance src-dst-ip
```

Related Commands

(config) interface PortChannel

(config) primary-interface

To configure the primary interface for a WAAS device, use the **primary-interface** global configuration command. To remove the configured primary interface, use the **no** form of this command.

primary-interface {**BVI** *bridge-id* | **GigabitEthernet** slot/*port* | **PortChannel** *index* | **Standby** *group-index* | **TenGigabitEthernet** slot/*port*} [**management**]

no primary-interface {**BVI** *bridge-id* | **GigabitEthernet** *slot*/*port* | **PortChannel** *index* | **Standby** *group-index* | **TenGigabitEthernet** slot/*port*} [**management**]

Syntax Description

BVI *bridge-id*	Selects a bridge virtual interface as the primary interface of the WAAS device. Specify the bridge ID (1–4).
GigabitEthernet *slot/port*	Selects a Gigabit Ethernet interface as the primary interface of the WAAS device. Valid slot and port values depend on the hardware platform.
PortChannel *index*	Selects a port channel interface as the primary interface of the WAAS device. Specify the port channel index number (1–4).
Standby *group-index*	Selects a standby group as the primary interface of the WAAS device. Specify the standby group number (1–3).
TenGigabitEthernet *slot/port*	Selects a TenGigabitEthernet interface as the primary interface of the WAAS device. Valid slot and port values depend on the hardware platform.
management	Designates the specified interface for management traffic.

Defaults

The default primary interface is the Gigabit Ethernet 0/0 or 1/0 interface, depending on the hardware platform. If this interface is not configured, then the first operational interface on which a link beat is detected becomes the default primary interface. Interfaces with lower number IDs are polled first (for example, Gigabit Ethernet 1/0 is checked before 2/0). The Gigabit Ethernet interfaces are polled before the port-channel interfaces.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

central-manager

Usage Guidelines

You can change the primary interface without disabling the WAAS device. To change the primary interface, reenter the command string and specify a different interface.

Note If you use the **restore factory-default preserve basic-config** command, the configuration for the primary interface is not preserved. If you want to reenable the WAAS device after using the **restore factory-default preserve basic-config** command, make sure to reconfigure the primary interface after the factory defaults are restored.

Setting the primary interface to be a Standby group does not imply that Standby functionality is available. You must configure Standby interfaces using the **interface standby** global configuration command.

Examples

The following example shows how to specify the Gigabit Ethernet slot 1, port 0 as the primary interface on a WAAS device:

```
WAE(config)# primary-interface GigabitEthernet 1/0
```

The following example shows how to specify the Gigabit Ethernet slot 2, port 0 as the primary interface on a WAAS device:

```
WAE(config)# primary-interface GigabitEthernet 2/0
```

The following example shows how to specify port channel interface 1 as the primary interface on a WAAS device:

```
WAE(config)# primary-interface portchannel 1
```

Related Commands

(config) interface GigabitEthernet

(config) interface TenGigabitEthernet

(config) radius-server

To configure a set of RADIUS authentication server settings on the WAAS device, use the r**adius-server** global configuration command. To disable RADIUS authentication server settings, use the **no** form of this command.

radius-server {**host** *hostname* | *hostipaddr* [**primary**] | **key** *keyword* | **retransmit** *retries* | **timeout** *seconds*}

no radius-server {**host** *hostname* | *hostipaddr* [**primary**] | **key** *keyword* | **retransmit** *retries* | **timeout** *seconds*}

Syntax Description

host *hostname*	Specifies a RADIUS server. You can have a maximum of 5 servers.
hostipaddr	IP address of the RADIUS server.
primary	(Optional) Sets the server as the primary server.
key *keyword*	Specifies the encryption key shared with the RADIUS servers. You can have a maximum of 15 characters.
retransmit *retries*	Specifies the number of transmission attempts (1–3) to an active server for a transaction. The default is 2.
timeout *seconds*	Specifies the time to wait for a RADIUS server to reply. The range is from 1 to 20 seconds. The default is 5 seconds.

Defaults

retransmit *retries*: 2

timeout *seconds*: 5

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

RADIUS authentication is disabled by default. You can enable RADIUS authentication and other authentication methods at the same time. You can also specify which method to use first. (See the **(config) authentication configuration** command.)

You can configure multiple RADIUS servers; authentication is attempted on the primary server first. If the primary server is unreachable, then authentication is attempted on the other servers in the RADIUS farm, in the order in which they were configured. If authentication fails for any reason other than a server is unreachable, authentication is not attempted on the other servers in the farm. This process applies regardless of the setting of the **authentication fail-over server-unreachable** command.

Examples

The following example shows how to specify a RADIUS server, specify the RADIUS key, and accept retransmit defaults. You can verify the configuration using the **show radius-server** command.

```
WAE(config)# radius-server host 172.16.90.121
WAE(config)# radius-server key myradiuskey
WAE# show radius-server
Radius Configuration:
---------------------
Radius Authentication is on
    Timeout       = 5
    Retransmit    = 3
    Key           = ****
    Servers
    -------
```

Related Commands

show radius-server

(config) service-insertion

To configure WNs, WNGs, ANCs, an ANCG, and a service context that are part of an AppNav Cluster, use the **service-insertion** global configuration command. To unconfigure settings, use the **no** form of this command.

service-insertion {**pass-through offload enable** {**all** | [**config**] [**intermediate**] [**no-peer**]} | **service-context** *contextname* [**rename** *newname*]| **appnav-controller distribute enable** | **appnav-controller-group** *ancgroupname* | **service-node** | **service-node-group** *sngroupname* | **swap src-ip**}

no service-insertion {**pass-through offload enable** {**all** | [**config**] [**intermediate**] [**no-peer**]} | **service-context** *contextname* | **appnav-controller distribute enable** | **appnav-controller-group** *ancgroupname* | **service-node** | **service-node-group** *sngroupname* | **swap src-ip**}

Syntax Description

pass-through offload enable	Enables pass-through traffic to be passed through at the ANC instead of being distributed to the WN and then passed through.
all	Offload all pass-through connections, including connections passed through due to error conditions.
config	Offload connections passed through due to missing policy configuration.
intermediate	Offload connections passed through due to an intermediate WN.
no-peer	Offload connections passed through due to no peer WN.
service-context *contextname*	Specifies the name of the service context to configure and enters service context configuration mode. If the service context does not exist, this command creates it.
rename *newname*	Renames an existing service context with the specified new name. This command is not available in the configuration submode, only at this top level.
appnav-controller distribute enable	Reenables distribution on an ANC if it has been disabled by the **no** form of this command. The default setting is enabled.
appnav-controller-group *ancgroupname*	Specifies the name of an ANCG to configure and enters AppNav Controller group configuration mode. If the ANCG does not exist, this command creates it.
service-node	Enters service node configuration mode to configure WN settings on the device.
service-node-group *sngroupname*	Specifies the name of a WNG to configure and enters service node group configuration mode to configure WNG settings. If the WNG does not exist, this command creates it.
swap src-ip	Enables swapping of client and WAAS device source IP address fields in intra-cluster traffic.

Defaults

Distribution is enabled on an ANC. Pass-through offload is enabled for **config**, **intermediate**, and **no-peer** reasons.

Command Modes global configuration

Device Modes application-accelerator

appnav-controller

Usage Guidelines Use the **service-insertion** command to configure the entities (WNs, WNGs, ANCs, an ANCG, and a service context) that are part of an AppNav Cluster. Some options of this command initiate configuration submodes, which are indicated by a different prompt (for example, config-scg). For more information on the configuration submode commands, see the following sections:

- **service-context**—"Service Context Configuration Mode Commands"
- **appnav-controller-group**—"AppNav Controller Group Configuration Mode Commands"
- **service-node**—"Service Node Configuration Mode Commands"
- **service-node-group**—"Service Node Group Configuration Mode Commands"

Within configuration submodes, you can use the various commands to define the settings of the entity. To return to global configuration mode, enter the **exit** command.

Each WN (and ANC acting as a WN) in the AppNav Cluster must be configured with WN settings by the **service-insertion service-node** command.

Each ANC in the AppNav Cluster must be configured with the following:

- ANCG settings by the **service-insertion appnav-controller-group** command
- WNG settings by the **service-insertion service-node-group** command
- Service context settings by the **service-insertion service-context** command

You can put an ANC into monitoring mode with the **no service-insertion appnav-controller distribute enable** command. This command stops the ANC from distributing any traffic to WNs for optimization. Instead, all traffic is passed through. This mode can be used for traffic monitoring for deployment sizing or troubleshooting purposes. If one ANC in an AppNav Cluster has this setting, all ANCs operate in monitor mode. You can reenable distribution with the **service-insertion appnav-controller distribute enable** command.

You can use the **service-insertion pass-through offload enable** command on a WN to prevent the ANCs from sending pass-through traffic to that WN. You can use this command on an ANC to prevent the ANC from sending pass-through traffic to any WN. The options allow you to specify what kind of pass-through traffic is to be off loaded by the ANC.

You may want to use the **swap src-ip** option if you are using a port channel for the cluster interface or there is a load balancing device between the ANC and WN. This option may improve the load balancing of traffic that the ANC distributes to WNs for optimization because it load balances based on the client IP address rather than the ANC IP address. (For traffic from the server to the client, it swaps the server IP address with the ANC IP address.) The Central Manager enables this feature automatically if any existing ANCs or WNs have port channel cluster interfaces that you configure with the Central Manager.

Examples The following example shows how to configure and enable a WN:

```
WAE(config)# service-insertion service-node
WAE(config-sn)# description London branch node 1
WAE(config-sn)# authentication sha1 key myauthkey
WAE(config-sn)# shutdown max-wait 120
```

```
WAE(config-sn)# enable
```

The following example shows how to configure and enable two WNGs, each having two WNs, on an ANC:

```
ANC(config)# service-insertion service-node-group LondonNodeGroup
ANC(config-sng)# description London branch node group
ANC(config-sng)# service-node 10.10.10.15 description London branch node 1
ANC(config-sng)# service-node 10.10.10.16 description London branch node 2
ANC(config-sng)# exit
ANC(config)# service-insertion service-node-group ChicagoNodeGroup
ANC(config-sng)# description Chicago branch node group
ANC(config-sng)# service-node 10.10.11.25 description Chicago branch node 1
ANC(config-sng)# service-node 10.10.11.26 description Chicago branch node 2
```

The following example shows how to configure and enable an ANCG with two ANC members, on an ANC:

```
ANC(config)# service-insertion appnav-controller-group myControllerGroup
ANC(config-scg)# description AppNav Controller group
ANC(config-scg)# appnav-controller 10.10.8.15 description Primary controller
ANC(config-scg)# appnav-controller 10.10.8.16 description Secondary controller
```

The following example shows how to configure and enable a service context:

```
WAE(config)# service-insertion service-context mycontext
WAE(config-scxt)# description My service context
WAE(config-scxt)# authentication sha1 key myauthkey
WAE(config-scxt)# appnav-controller-group myControllerGroup
WAE(config-scxt)# service-node-group LondonNodeGroup
WAE(config-scxt)# service-node-group ChicagoNodeGroup
WAE(config-scxt)# service-policy myAppNavPolicy
WAE(config-scxt)# enable
```

The following example shows how to enable pass-through offloading of traffic for all reasons:

```
WAE(config)# service-insertion pass-through offload enable all
```

The following example shows how to enable the source IP address swapping feature:

```
WAE(config)# service-insertion swap src-ip
```

The following example shows how to enable the

Related Commands

show service-insertion

(config) service-policy

To configure AppNav and optimization service policy, use the **service-policy** global configuration command. To unconfigure settings, use the **no** form of this command.

service-policy {**optimize** *policy-map-name* | **type**
{**appnav config** {**remove-all** | **restore-predefined**}|
waas {**config** {**remove-all** | **restore-predefined**}| **set ip dscp** *dscp-marking*}}

no service-policy {**optimize** *policy-map-name* | **type**
{**appnav config** {**remove-all** | **restore-predefined**}|
waas {**config** {**remove-all** | **restore-predefined**}| **set ip dscp** *dscp-marking*}}

Syntax Description

optimize *policy-map-name*	Specifies the active optimization policy map.
type	Specifies an operation on AppNav or optimization policies.
appnav config	Specifies an operation on AppNav policies.
remove-all	Removes all class map and policy map configurations.
restore-predefined	Replaces all class map and policy map configurations with factory default configurations.
waas	Specifies an operation on optimization policies.
set ip dscp *dscp-marking*	Specifies the default DSCP marking value, as shown in Table 3-2.

Defaults

The default DSCP marking value is copy.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

Usage Guidelines

The DSCP field in an IP packet enables different levels of service to be assigned to network traffic. Levels of service are assigned by marking each packet on the network with a DSCP code. DSCP is the combination of IP Precedence and Type of Service (ToS) fields. For more information, see RFC 2474.

A DSCP value is assigned in a policy rule and applies to all traffic associated with a class map. If a DSCP value is not assigned or defined, the default DSCP value is applied to traffic. The global default DSCP value is copy, which copies the DSCP value from the incoming packet and uses it for the outgoing packet.

Table 3-2 lists the valid DSCP marking values that you can specify.

Table 3-2 DSCP Marking Values

DSCP Code	Description
0 - 63	Marks packets with a numeric dscp from 0 to 63.
af11	Marks packets with AF11 dscp (001010).
af12	Marks packets with AF11 dscp (001100).
af13	Marks packets with AF13 dscp (001110).
af21	Marks packets with AF21 dscp (010010).
af22	Marks packets with AF22 dscp (010100).
af23	Marks packets with AF23 dscp (010110).
af31	Marks packets with AF31 dscp (011010).
af32	Marks packets with AF32 dscp (011100).
af33	Marks packets with AF33 dscp (011110).
af41	Marks packets with AF41 dscp (100010).
af42	Marks packets with AF42 dscp (100100).
af43	Marks packets with AF43 dscp (100110).
cs1	Marks packets with CS1 (precedence 1) dscp (001000).
cs2	Marks packets with CS2 (precedence 2) dscp (010000).
cs3	Marks packets with CS3 (precedence 3) dscp (011000).
cs4	Marks packets with CS4 (precedence 4) dscp (100000).
cs5	Marks packets with CS5 (precedence 5) dscp (101000).
cs6	Marks packets with CS6 (precedence 6) dscp (110000).
cs7	Marks packets with CS7 (precedence 7) dscp (111000).
copy	Copies the DSCP value from the incoming packet to the outgoing packet. (default)
default	Marks packets with default dscp (000000).
ef	Marks packets with EF dscp (101110).

Examples

The following example shows how to set the default DSCP marking value to copy:

```
WAE(config)# service-policy type waas set ip dscp copy
```

The following example shows how to restore optimization policies:

```
WAE(config)# service-policy type waas config restore-predefined
```

The following example shows how to remove all AppNav policies:

```
WAE(config)# service-policy type appnav config remove-all
```

Related Commands

show service-policy

(config) class-map

(config) policy-map

(config) smb-conf

To manually configure the parameters for a WAAS device Samba configuration file, *smb.conf*, use the **smb-conf** global configuration command. To return a parameter to its default value, use the **no** form of this command.

smb-conf section {**global**} **name** *attr-name* **value** *attr-value*

no smb-conf section {**global**} **name** *attr-name* **value** *attr-value*

Syntax Description

global	Specifies one of the global print parameters.
name *attr-name*	Specifies the name of the parameter in the specified section that you want to manually configure (up to 80 characters).
value *attr-value*	Specifies the value of the parameter (up to 255 characters).

See Table 3-3 for a description of the parameters for the global, print$, and printers, including the names and default values.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Legacy print services are no longer supported in WAAS 4.4.x and later. We recommend using the Windows print accelerator (see the (config) accelerator windows-print command).

The *smb.conf* file contains a variety of samba Configuration parameters. Global parameters apply to the server. Service level parameters, which define default settings for all other sections and shares, allow you to avoid the need to set the same value repeatedly. You can override these globally set share settings and specify other values for each individual section or share.

Table 3-3 Samba Configuration Parameters

Parameter Name	Default Value	Parameter Description
global parameters		
idmap uid	70000-200000	Range of user IDs allocated for mapping UNIX users to NT user SIDs.
idmap gid	70000-200000	Range of group IDs allocated for mapping UNIX groups to NT group SIDs.

Table 3-3 Samba Configuration Parameters (continued)

Parameter Name	Default Value	Parameter Description
winbind enum users	no	Parameter that does not enumerate domain users using MSRPC.
winbind enum groups	no	Parameter that does not enumerate domain groups using MSRPC.
winbind cache time	10	Time that a domain user or group information remains in the cache before expiring.
winbind use default domain	yes	Use the default domain for users and groups.
lpq cache time	0	Cache time for the results of the **lpq** command.
log file	/local/local1/errorlog/samba.log	Location where print-related errors are logged.
max log size	50	Maximum number of errors the log file can contain. After 50 errors, for each new error logged, the oldest error is removed.
socket options	TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192	Controls on the network layer of the operating system that allows the connection with the client to be tuned. This option is typically used to tune your Samba server for optimal performance for your local network.
smb ports	50139	Available ports on the Samba server.
local master	no	Parameter that sets *nmbd* to be a local master browser on a subnet.
domain master	no	Parameter that sets *nmbd* to be a domain master browser for its given workgroup.
preferred master	no	Parameter that sets *nmbd* to be a preferred master browser for its workgroup
dns proxy	no	DNS proxy that is not enabled.
template homedir	/local/local1/	Home directory on File Engine or WAE.
template shell	/admin-shell	Directory of the administrative shell.
comment	Comment:	Optional description of the print server (or share) that is visible when a client queries the server. This parameter can also be set by the **windows-domain comment** command.
netbios name	MYFILEENGINE	Name of the Samba server hosting print services. This parameter can also be set by the **windows-domain netbios-name** command.
realm	CISCO	Active Directory domain name. Always uppercase. This parameter can also be set by the **windows-domain realm** command.
wins server	10.10.10.1	IP address of the Windows domain server used to authenticate user access to print services. This parameter can also be set by the **windows-domain wins-server** command.
password server	10.10.10.10	Optional IP address of the password server used for authentication of users. This parameter can also be set by the **windows-domain password-server** command.

Table 3-3 Samba Configuration Parameters (continued)

Parameter Name	Default Value	Parameter Description
security	domain	Use Windows domain server for authentication. This parameter can also be set by the **windows-domain security** command.
client schannel	no	Secure channel indicator used for Windows domain server authentication.
ldap ssl	none	Defines whether or not Samba should use SSL when connecting to the LDAP server. The default is unconfigured. If set to "off," SSL is never used when querying the directory server. To enable the LDAPv3 StartTLS extended operation (RFC2830), set to "yes".

Examples

The following example shows how to change the maximum size of the Samba error log file from the default of 50 errors to 75 errors:

```
WAE# smb-conf global max log size 75
```

The following example shows how to change the realm from the default of CISCO to MYCOMPANYNAME:

```
WAE# smb-conf global realm MYCOMPANYNAME
```

The following example shows how to enable LDAP server signing:

```
WAE# smb-conf global name "ldap ssl" value "yes"
```

Related Commands

show smb-conf

windows-domain

(config) accelerator windows-print

(config) windows-domain

(config) snmp-server access-list

To configure a standard access control list on a WAAS device to allow access through an SNMP agent, use the **snmp-server access-list** global configuration command. To remove a standard access control list, use the **no** form of this command.

snmp-server access-list {*num* | *name*}

no snmp-server access-list {*num* | *name*}

Syntax Description

num	Standard access list number (1–99).
name	Standard access list name. You can use a maximum of 30 characters.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

If you are using an SNMP server ACL, you must permit the loopback interface.

Examples

The following example shows how to allow the SNMP agent to check against access control list 12 before accepting or dropping packets:

```
WAE(config)# snmp-server access-list 12
```

Note You must first create access list 12 using the **ip access-list standard** global configuration command.

Related Commands

(config) ip access-list

show running-config

(config) snmp-server community

To enable the SNMP agent on a WAAS device and to set up the community access string to permit access to the SNMP agent, use the **snmp-server community** global configuration command. To disable the SNMP agent and remove the previously configured community string, use the **no** form of this command.

snmp-server community *string* [**group** *groupname* | **rw**]

no snmp-server community *string* [**group** *groupname* | **rw**]

Syntax Description

string	Community string that acts like a password and permits access to the SNMP agent. You can use up to a maximum of 64 characters.
group *groupname*	(Optional) Specifies the group name to which the community string belongs. You can use a maximum of 64 characters.
rw	(Optional) Enables read-write access to this community string.

Defaults

The SNMP agent is disabled and a community string is not configured. When configured, an SNMP community string by default permits read-only access to all objects.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Examples

The following example shows how to enable the SNMP agent and assign the community string comaccess to SNMP:

```
WAE(config)# snmp-server community comaccess
```

The following example shows how to disable the SNMP agent and remove the previously defined community string:

```
WAE(config)# no snmp-server community
```

Related Commands

(config) snmp-server community

(config) snmp-server contact

(config) snmp-server enable traps

(config) snmp-server group

(config) snmp-server host

(config) snmp-server location

(config) snmp-server mib

(config) snmp-server notify inform
(config) snmp-server user
(config) snmp-server view
snmp trigger

(config) snmp-server contact

To set the system server contact string on a WAAS device, use the **snmp-server contact** global configuration command. To remove the system contact information, use the **no** form of this command.

snmp-server contact *line*

no snmp-server contact *line*

Syntax Description

contact *line*	Specifies the text for MIB-II object *sysContact*. This is the identification of the contact person for this managed node.

Defaults

No system contact string is set.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

The system contact string is the value stored in the MIB-II system group *sysContact* object.

Examples

The following example shows how to set a system contact string and then remove it:

```
WAE(config)# snmp-server contact Dial System Operator at beeper # 27345
WAE(config)# no snmp-server contact
```

Related Commands

(config) snmp-server community

(config) snmp-server enable traps

(config) snmp-server group

(config) snmp-server host

(config) snmp-server location

(config) snmp-server mib

(config) snmp-server notify inform

(config) snmp-server user

(config) snmp-server view

snmp trigger

(config) snmp-server enable traps

To enable the WAAS device to send SNMP traps, use the **snmp-server enable traps** global configuration command. To disable all SNMP traps or only SNMP authentication traps, use the **no** form of this command.

snmp-server enable traps [**alarm** [**clear-critical** | **clear-major** | **clear-minor** | **raise-critical** | **raise-major** | **raise-minor**]

snmp-server enable traps config | **entity** | **event**

snmp-server enable traps content-engine [**disk-fail** | **disk-read** | **disk-write** | **overload-bypass** | **transaction-log**]

snmp-server enable traps snmp [**authentication** | **cold-start** | **linkdown** | **linkup**]

Syntax Description

alarm	(Optional) Enables WAAS alarm traps.
clear-critical	(Optional) Enables clear-critical alarm traps.
clear-major	(Optional) Enables clear-major alarm traps.
clear-minor	(Optional) Enables clear-minor alarm traps.
raise-critical	(Optional) Enables raise-critical alarm traps.
raise-major	(Optional) Enables raise-major alarm traps.
raise-minor	(Optional) Enables raise-minor alarm traps.
config	Enables CiscoConfigManEvent traps.
entity	Enables SNMP entity traps.
event	Enables Event MIB traps.
content-engine	Enables SNMP WAAS traps.
disk-fail	(Optional) Enables disk failure error traps.
disk-read	(Optional) Enables disk read error traps.
disk-write	(Optional) Enables disk write error traps.
overload-bypass	(Optional) Enables WCCP overload bypass error traps.
transaction-log	(Optional) Enables transaction log write error traps.
snmp	Enables SNMP-specific traps.
authentication	(Optional) Enables authentication trap.
cold-start	(Optional) Enables cold start trap.
linkdown	(Optional) Enables link down trap.
linkup	(Optional) Enables link up trap.

Defaults

This command is disabled by default. No traps are enabled.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

In the WAAS software the following six generic alarm traps are available in the CISCO-CONTENT-ENGINE-MIB:

Name of Alarm Trap	Severity	Action
cceAlarmCriticalRaised	Critical	Raised
cceAlarmCriticalCleared	Critical	Cleared
cceAlarmMajorRaised	Major	Raised
cceAlarmMajorCleared	Major	Cleared
cceAlarmMinorRaised	Minor	Raised
cceAlarmMinorCleared	Minor	Cleared

Note By default, these six general alarm traps are disabled.

These six general alarm traps provide SNMP and Node Health Manager integration. You can enable or disable each of these six alarm traps through the WAAS CLI.

To configure traps, you must enter the **snmp-server enable traps** command. If you do not enter the **snmp-server enable traps** command, no traps are sent.

The **snmp-server enable traps** command is used with the **snmp-server host** command. Use the **snmp-server host** command to specify which host or hosts receive SNMP traps. To send traps, you must configure at least one host using the **snmp-server host** command.

To allow a host to receive a trap, you must enable both the **snmp-server enable traps** command and the **snmp-server host** command for that host.

You must enable SNMP with the **snmp-server community** command.

To disable the sending of the MIB-II SNMP authentication trap, you must enter the command **no snmp-server enable traps snmp authentication**.

Examples

The following example shows how to enable the WAAS device to send all traps to the host 172.31.2.160 using the community string public:

```
WAE(config)# snmp-server enable traps
WAE(config)# snmp-server host 172.31.2.160 public
```

The following example shows how to disable all traps:

```
WAE(config)# no snmp-server enable traps
```

Related Commands

(config) snmp-server community

(config) snmp-server contact

(config) snmp-server group
(config) snmp-server host
(config) snmp-server location
(config) snmp-server mib
(config) snmp-server notify inform
(config) snmp-server user
(config) snmp-server view
snmp trigger

(config) snmp-server group

To define a user security model group for a WAAS device, use the **snmp-server group** global configuration command. To remove the specified group, use the **no** form of this command.

snmp-server group *name* {**v1** [**notify** *name*] [**read** *name*] [**write** *name*] |
v2c [**notify** *name*] [**read** *name*] [**write** *name*] |
v3 {**auth** [**notify** *name*] [**read** *name*] [**write** *name*] |
noauth [**notify** *name*] [**read** *name*] [**write** *name*] |
priv [**notify** *name*] [**read** *name*] [**write** *name*]}}

no snmp-server group *name* {**v1** [**notify** *name*] [**read** *name*] [**write** *name*] |
v2c [**notify** *name*] [**read** *name*] [**write** *name*] |
v3 {**auth** [**notify** *name*] [**read** *name*] [**write** *name*] |
noauth [**notify** *name*] [**read** *name*] [**write** *name*] |
priv [**notify** *name*] [**read** *name*] [**write** *name*]}}

Syntax Description

group *name*	Specifies the SNMP group. You can enter a maximum of 64 characters.
v1	Specifies the group using the Version 1 Security Model.
notify *name*	(Optional) Specifies a notify view name for the group that enables you to specify a notify, inform, or trap. You can enter a maximum of 64 characters.
read *name*	(Optional) Specifies a read view name for the group that enables you to view only the contents of the agent. You can enter a maximum of 64 characters.
write	(Optional) Specifies a write view name for the group that enables you to enter data and configure the contents of the agent. You can enter a maximum of 64 characters.
v2c	Specifies the group using the Version 2c Security Model.
v3	Specifies the group using the User Security Model (SNMPv3).
auth	Specifies the group using the AuthNoPriv Security Level.
noauth	Specifies the group using the noAuthNoPriv Security Level.
priv	Specifies the group using the AuthPriv Security Level.

Defaults

The default is that no user security model group is defined.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

The maximum number of SNMP groups that can be created is 10.

Select one of three SNMP security model groups: Version 1 (**v1**) Security Model, Version 2c (**v2c**) Security Model, or the User Security Model (**v3** or SNMPv3). Optionally, you then specify a notify, read, or write view for the group for the particular security model chosen. The **v3** option allows you to specify the group using one of three security levels: **auth** (AuthNoPriv Security Level), **noauth** (noAuthNoPriv Security Level), or **priv** (AuthPriv Security Level).

Examples

The following example shows how to define a user security model group named acme that uses the SNMP version 1 security model and a view name of mymib for notifications:

```
WAE(config)# snmp-server group acme v1 notify mymib
```

Related Commands

(config) snmp-server community

(config) snmp-server contact

(config) snmp-server enable traps

(config) snmp-server host

(config) snmp-server location

(config) snmp-server mib

(config) snmp-server notify inform

(config) snmp-server user

(config) snmp-server view

snmp trigger

(config) snmp-server host

To specify the recipient of a host SNMP trap operation, use the **snmp-server host** global configuration command. To remove the specified host, use the **no** form of this command.

snmp-server host {*hostname* | *ip-address*} *communitystring*
[**v2c** [**retry** *number*] [**timeout** *seconds*] |
[**v3** {**auth** [**retry** *number*] [**timeout** *seconds*] |
noauth [**retry** *number*] [**timeout** *seconds*] |
priv [**retry** *number*] [**timeout** *seconds*]}]

no snmp-server host {*hostname* | *ip-address*} *communitystring*
[**v2c** [**retry** *number*] [**timeout** *seconds*] |
[**v3** {**auth** [**retry** *number*] [**timeout** *seconds*] |
noauth [**retry** *number*] [**timeout** *seconds*] |
priv [**retry** *number*] [**timeout** *seconds*]}]

Syntax Description

hostname	Hostname of the SNMP trap host that will be sent in the SNMP trap messages from the WAAS device.
ip-address	IP address of the SNMP trap host that will be sent in the SNMP trap messages from the WAAS device.
communitystring	Password-like community string sent in the SNMP trap messages from the WAE. You can enter a maximum of 64 characters.
v2c	(Optional) Specifies the Version 2c Security Model.
retry *number*	(Optional) Sets the count for the number of retries (1–10) for the inform request. (The default is 2 tries.)
timeout *seconds*	(Optional) Sets the timeout for the inform request (1–1000 seconds). The default is 15 seconds.
v3	(Optional) Specifies the User Security Model (SNMPv3).
auth	Sends a notification using the AuthNoPriv Security Level.
noauth	Sends a notification using the noAuthNoPriv Security Level.
priv	Sends a notification using the AuthPriv Security Level.

Defaults

This command is disabled by default. No traps are sent. If enabled, the default version of the SNMP protocol used to send the traps is SNMP Version 1.

retry *number*: 2 retries

timeout: 15 seconds

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

If you do not enter an **snmp-server host** command, no traps are sent. To configure the WAAS device to send SNMP traps, you must enter at least one **snmp-server host** command. To enable multiple hosts, you must enter a separate **snmp-server host** command for each host. The maximum number of **snmp-server host** commands is four.

When multiple **snmp-server host** commands are given for the same host, the community string in the last command is used.

The **snmp-server host** command is used with the **snmp-server enable traps** command to enable SNMP traps.

You must enable SNMP with the **snmp-server community** command.

Examples

The following example shows how to send the SNMP traps defined in RFC 1157 to the host specified by the IP address 172.16.2.160. The community string is comaccess:

```
WAE(config)# snmp-server enable traps
WAE(config)# snmp-server host 172.16.2.160 comaccess
```

The following example shows how to remove the host 172.16.2.160 from the SNMP trap recipient list:

```
WAE(config)# no snmp-server host 172.16.2.160
```

Related Commands

(config) snmp-server community

(config) snmp-server contact

(config) snmp-server enable traps

(config) snmp-server group

(config) snmp-server location

(config) snmp-server mib

(config) snmp-server notify inform

(config) snmp-server user

(config) snmp-server view

snmp trigger

(config) snmp-server location

To set the SNMP system location string on a WAAS device, use the **snmp-server location** global configuration command. To remove the location string, use the **no** form of this command.

snmp-server location *line*

no snmp-server location *line*

Syntax Description

location *line*	Specifies the text for MIB-II object *sysLocation*. This string describes the physical location of this node.

Defaults

No system location string is set.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

The system location string is the value stored in the MIB-II system group system location object. You can see the system location string with the **show snmp** EXEC command.

Examples

The following example shows how configure a system location string:

```
WAE(config)# snmp-server location Building 3/Room 214
```

Related Commands

(config) snmp-server community

(config) snmp-server contact

(config) snmp-server enable traps

(config) snmp-server group

(config) snmp-server host

(config) snmp-server mib

(config) snmp-server notify inform

(config) snmp-server user

(config) snmp-server view

snmp trigger

(config) snmp-server mib

To configure persistence for the SNMP Event MIB, use the **snmp-server mib** global configuration command. To disable the Event MIB, use the **no** form of this command.

snmp-server mib persist event

no snmp-server mib persist event

Syntax Description

persist	Configures MIB persistence.
event	Enables MIB persistence for the Event MIB.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

The Event MIB can set the threshold on any MIB variables supported by the WAAS software and store the threshold permanently on the disk.

The WAAS software implementation of SNMP supports the following MIBs:

- ACTONA-ACTASTORE-MIB
- CISCO-CONFIG-MAN-MIB
- CISCO-CDP-MIB
- CISCO-CONTENT-ENGINE-MIB (partial)
- CISCO-ENTITY-ASSET-MIB
- CISCO-SMI
- CISCO-TC
- ENTITY-MIB
- EVENT-MIB
- HOST-RESOURCES-MIB
- MIB-II
- SNMP-COMMUNITY-MIB
- SNMP-FRAMEWORK-MIB
- SNMP-NOTIFICATION-MIB
- SNMP-TARGET-MIB
- SNMP-USM-MIB

- SNMPv2
- SNMP-VACM-MIB

Note The WAAS software supports six generic alarm traps in the CISCO-CONTENT-ENGINE-MIB for SNMP and Node Health Manager integration.

Examples

The following example shows how to set persistence for the Event MIB:

```
WAE(config)# snmp-server mib persist event
```

Related Commands

(config) snmp-server community
(config) snmp-server contact
(config) snmp-server enable traps
(config) snmp-server group
(config) snmp-server host
(config) snmp-server location
(config) snmp-server notify inform
(config) snmp-server user
(config) snmp-server view
snmp trigger

(config) snmp-server notify inform

To configure the SNMP notify inform request on a WAAS device, use the **snmp-server notify inform** global configuration command. To return the setting to the default value, use the **no** form of this command.

snmp-server notify inform

no snmp-server notify inform

Syntax Description

This command has no arguments or keywords.

Defaults

If you do not enter the **snmp-server notify inform** command, the default is an SNMP trap request.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Examples

The following example shows how to configure an SNMP notify inform request versus the default SNMP trap:

```
WAE(config)# snmp-server notify inform
```

Related Commands

(config) snmp-server community

(config) snmp-server contact

(config) snmp-server enable traps

(config) snmp-server group

(config) snmp-server host

(config) snmp-server location

(config) snmp-server mib

(config) snmp-server user

(config) snmp-server view

snmp trigger

(config) snmp-server trap-source

To set the source interface from which SNMP traps are sent on a WAAS device, use the **snmp-server trap-source** global configuration command. To remove the trap source configuration, use the **no** form of this command.

snmp-server trap-source {**GigabitEthernet** *slot/port* | **PortChannel** *index* | **Standby** *grpnumber* | **TenGigabitEthernet** *slot/port*}

no snmp-server trap-source {**GigabitEthernet** *slot/port* | **PortChannel** *index* | **Standby** *grpnumber* | **TenGigabitEthernet** *slot/port* | **bvi** *bridge-id*}

Syntax Description

GigabitEthernet *slot/port*	Selects a Gigabit Ethernet interface to configure as the trap source. The slot number and port number are separated with a forward slash character (/). Valid slot and port values depend on the hardware platform.
PortChannel *index*	Selects a port channel (1–4) to configure as the trap source.
Standby *grpnumber*	Selects a standby group (1–3) to configure as the trap source.
TenGigabitEthernet *slot/port*	Selects a TenGigabitEthernet interface to configure as the trap source. The slot number and port number are separated with a forward slash character (/). Valid slot and port values depend on the hardware platform.
bvi *bridge-id*	Selects a bridge virtual interface (1–4) to configure as the trap source.

Defaults

No system trap source is set.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Examples

The following example shows how to configure gigabit Ethernet interface 1/0 as the trap source:

```
WAE(config)# snmp-server trap-source gigabitethernet 1/0
```

Related Commands

(config) snmp-server community

(config) snmp-server contact

(config) snmp-server enable traps

(config) snmp-server group

(config) snmp-server host

(config) snmp-server mib

(config) snmp-server notify inform
(config) snmp-server user
(config) snmp-server view
snmp trigger

(config) snmp-server user

To define a user who can access the SNMP server, use the **snmp-server user** global configuration command. To remove access, use the **no** form of this command.

snmp-server user *name group*
[**auth** {**md5** *password* [**priv** *password*] |
sha *password* [**priv** *password*]} |
remote *octetstring* [**auth** {**md5** *password* [**priv** *password*] |
sha *password* [**priv** *password*]}]]

no nmp-server user *name group*
[**auth** {**md5** *password* [**priv** *password*] |
sha *password* [**priv** *password*]} |
remote *octetstring* [**auth** {**md5** *password* [**priv** *password*] |
sha *password* [**priv** *password*]}]]

Syntax Description

name group	Name and group of the SNMP user. Use letters, numbers, dashes, and underscores, but no blanks. The name specifies the user on the SNMP host who wants to communicate with the SNMP agent on the WAAS device. You can enter a maximum of 32 characters for the name. The group specifies the group to which the SNMP user belongs. You can enter a maximum of 64 characters for the group.
auth	(Optional) Configures user authentication parameters.
md5 *password*	Configures HMAC MD5 user authentication password.
priv *password*	(Optional) Configures authentication HMAC-MD5 user private password. You can enter a maximum of 256 characters.
sha *password*	Configures the HMAC-SHA authentication password. You can enter a maximum of 256 characters.
remote *octetstring*	(Optional) Specifies the globally unique identifier (engineID) for a remote SNMP entity (for example, the SNMP network management station) for at least one of the SNMP users (10 to 64 characters, not counting colons). To send an SNMPv3 inform message, you must configure at least one SNMPv3 user with a remote SNMP ID option on the WAAS device. The SNMP ID is entered in octet string form. For example, if the IP address of a remote SNMP entity is 192.147.142.129, then the octet string would be 00:00:63:00:00:00:a1:c0:93:8e:81. (Colons will be removed in the show running-config command output.)

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Examples

The following example shows how to create an SNMPv3 user account on the WAAS device. The SNMPv3 user is named acme and belongs to the group named admin. Because this SNMP user account has been set up with no authentication password, the SNMP agent on the WAAS device does not perform authentication on SNMP requests from this user.

```
WAE(config)# snmp-server user acme admin
```

Related Commands

(config) snmp-server community

(config) snmp-server contact

(config) snmp-server enable traps

(config) snmp-server group

(config) snmp-server host

(config) snmp-server location

(config) snmp-server mib

(config) snmp-server notify inform

(config) snmp-server view

snmp trigger

(config) snmp-server view

To define an SNMPv2 MIB view on a WAAS device, use the **snmp-server view** global configuration command. To remove the MIB view definition, use the **no** form of this command.

snmp-server view *viewname MIBfamily* {**excluded** | **included**}

no snmp-server view *viewname MIBfamily* {**excluded** | **included**}

Syntax Description

viewname MIBfamily	Name of this family of view subtrees and a subtree of the MIB. You can enter a maximum of 64 characters.
excluded	Excludes the MIB family from the view.
included	Includes the MIB family in the view.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Examples

The following example shows how to define an SNMPv2 MIB view:

```
WAE(config)# snmp-server view fileview ciscoFileEngineMIB included
```

Related Commands

(config) snmp-server community

(config) snmp-server contact

(config) snmp-server enable traps

(config) snmp-server group

(config) snmp-server host

(config) snmp-server location

(config) snmp-server mib

(config) snmp-server notify inform

(config) snmp-server user

snmp trigger

(config) sshd

To enable the SSH daemon on a WAAS device, use the **sshd** global configuration command. To disable the SSH daemon on a WAAS device, use the **no** form of this command.

sshd {**allow-non-admin-users** | **enable** | **password-guesses** *number* | **timeout** *seconds* }

no sshd {**allow-non-admin-users** | **enable** | **password-guesses** *number* | **timeout** *seconds* }

Syntax Description

allow-non-admin-users	Allows nonadministrative users to gain SSH access to the chosen device (or device group). By default, this option is disabled. **Note** Nonadministrative users are nonsuperuser administrators. All nonsuperuser administrators have restricted access to a WAAS device because their login accounts have a privilege level of 0. Superuser administrators have full access to a WAAS device because their login accounts have the highest level of privileges, a privilege level of 15.
enable	Enables the SSH daemon on a WAAS device.
password-guesses *number*	Specifies the maximum number of allowable password guesses per connection (1–3). The default is 3.
timeout *seconds*	Configures the number of seconds for which an SSH session will be active during the negotiation (authentication) phase between the client and server before it times out. The SSH login grace time value in seconds is 1–99999. The default is 300. If you have established an SSH connection to the WAAS device but have not entered the username when prompted at the login prompt, the connection will be terminated by the WAAS device if the grace period expires even after a successful login.

Defaults

By default, the SSH daemon is disabled on a WAAS device. If you use the **sshd enable** command to enable the SSH daemon on a WAAS device, the following default settings are used:

password-guesses *number*: 3 guesses

timeout *seconds*: 300 seconds

version: ssh version 2 protocol is enabled

Note The SSH version 1 protocol is no longer supported. Only the SSH version 2 protocol is supported by the WAAS device.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Before you enable the **sshd** command, use the **ssh-key-generate** command to generate a private and a public host key, which the client uses to verify the server identity.

Although the **sshd password-guesses** command specifies the number of allowable password guesses from the SSH server side, the actual number of password guesses for an SSH login session is determined by the combined number of allowable password guesses of the SSH server and the SSH client. Some SSH clients limit the maximum number of allowable password guesses to three (or to one in some cases), even though SSH server side allows more than this number of guesses.

When you enter the **sshd password-guesses** command and specify *n* allowable password guesses, certain SSH clients interpret this *number* as *n*+1. For example, when configuring the number of guesses to two by issuing the command **sshd password-guesses 2** for a particular device, SSH sessions from some SSH clients will allow three password guesses.

Note You can use the Telnet daemon with the WAAS device. SSH does not replace Telnet.

Examples

The following example shows how to enable and configure a Secure Shell daemon on the WAAS device:

```
WAE(config)# sshd enable
WAE(config)# sshd timeout 20
```

Related Commands

(config) ssh-key-generate

(config) ssh-key-generate

To generate the SSH host key for a WAAS device, use the **ssh-key-generate** global configuration command. To remove the SSH key, use the **no** form of this command.

ssh-key-generate [**key-length** *length*]

no ssh-key-generate [**key-length** *length*]

Syntax Description

key-length *length*	(Optional) Configures the length of the SSH key. The number of bits is 512–2048.

Defaults

key-length *length*: 1024 bits

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Before you enter the **sshd enable** command, enter the **ssh-key-generate** command to generate a private and a public host key, which the client programs use to verify a server identity.

When you use an SSH client and log in to a WAAS device, the public key for the SSH daemon that is running on the device is recorded in the client machine known_hosts file in your home directory. If you regenerate the host key by specifying the number of bits in the **key-length** command option, you must delete the old public key entry associated with the WAAS device in the known_hosts file before running the SSH client program to log in to the WAAS device. When you use the SSH client program after deleting the old entry, the known_hosts file is updated with the new SSH public key for the WAAS device.

Examples

The following example shows how to generate an SSH public key and then enables the SSH daemon on the WAAS device:

```
WAE(config)# ssh-key-generate
Ssh host key generated successfully
Saving the host key to box ...
Host key saved successfully
WAE(config)# sshd enable
Starting ssh daemon ...
Ssh daemon started successfully
```

Related Commands

(config) sshd

(config) stats-collector logging

To configure the statistics collector for the SMB accelerator, use the **stats-collector logging** global configuration command. To unconfigure the statistics collector, use the **no** form of this command.

stats-collector logging {**enable** | **rate** {**10** | **30**}}

no stats-collector logging {**enable** | **rate** {**10** | **30**}}

Syntax Description

enable	Enables the statistics collector.
rate {**10**	**30**}	Configures the collection interval to 10 or 30 seconds.

Defaults

The statistics collector is disabled. The collection interval is set to 30 seconds.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

Usage Guidelines

This command configures periodic statistics logging for the SMB application accelerator. After enabling logging, you can disable it with the **no** form of the command. Statistics for the most recent 14 days are saved.

Examples

The following example shows how to enable statistics collection:

```
WAE(config)# stats-collector logging enable
```

The following example shows how to disable statistics collection:

```
WAE(config)# no stats-collector logging enable
```

Related Commands

copy monitoring-log

(config) system jumbomtu

To configure a jumbo MTU on all devices interfaces, use the **system jumbomtu** global configuration command. To remove the jumbo MTU, use the **no** form of this command.

system jumbomtu *size*

no system jumbomtu *size*

Syntax Description

size	Configures the size of the MTU (576–9000 or 9216 bytes, depending on platform).

Defaults

MTU size is 1500 bytes.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

Usage Guidelines

This command is available only on the following platforms: WAE-674/7341/7371, WAVE-294/594/694/7541/7571/8541, and vWAAS. This command changes the MTU setting for all interfaces on the device, including logical interfaces with at least one physical member, and may cause current active connections to time out. After you change the MTU using this command, you cannot change the MTU of individual interfaces.

Examples

The following example shows how to configure a jumbo MTU:

```
WAE(config)# system jumbomtu 9000
Changing system mtu setting will change the MTU values on all the interfaces. This may
cause the current active connections in the device to timeout.
Are you sure you want to do this? (y/n) [n]y
```

Related Commands

show interface

(config) tacacs

To configure TACACS+ server parameters on a WAAS device, use the **tacacs** global configuration command. To disable individual options, use the **no** form of this command.

tacacs {**host** {*hostname* | *ip-address*} [**primary** | **port** *number*] | **key** *keyword* | **password ascii** | **retransmit** *retries* | **timeout** *seconds*}

no tacacs {**host** {*hostname* | *ip-address*} | **key** *keyword* | **password ascii** | **retransmit** *retries* | **timeout** *seconds*}

Syntax Description

host	Specifies a server address.
hostname	Hostname of the TACACS+ server.
ip-address	IP address of the TACACS+ server.
primary	(Optional) Sets the server as the primary server.
port *number*	Sets the port number of the TACACS+ server. If not specified, the default port 49 is used.
key *keyword*	Sets the security word. An empty string is the default.
password ascii	Specifies ASCII as the TACACS+ password type.
retransmit *retries*	Sets the number of times that requests are retransmitted to a server. The number of retry attempts allowed is 1–3. The default is 2 retry attempts.
timeout *seconds*	Sets the number of seconds to wait before a request to a server is timed out. The timeout is in seconds (1–20). The default is 5 seconds.

Defaults

port *number*: 49

keyword: none (empty string)

timeout *seconds*: 5

retries: 2

password: The default password type is PAP.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

To enable user authentication with a TACACS+ server, use the **authentication** global configuration command. (See the **(config) authentication configuration** command.)

Note When AAA Command Authorization is enabled for a device through the Central Manager GUI, TACACS+ CLI configuration changes are not allowed and **tacacs** commands will fail.

You can use the TACACS+ remote database to maintain login and configuration privileges for administrative users. The **tacacs host** command allows you to configure the network parameters required to access the remote database.

Use the **tacacs key** command to specify the TACACS+ key, used to encrypt the packets transmitted to the server. This key must be the same as the one specified on the server daemon. The maximum number of characters in the key must not exceed 32 printable ASCII characters. An empty key string is the default. All leading spaces are ignored; spaces within and at the end of the key string are not ignored. Double quotes are not required even if there are spaces in the key.

Note If you configure a TACACS+ key on the WAAS device (the TACACS+ client), make sure that you configure an identical key on the external TACACS+ server. Do not use the following characters: backwards single quote (`), double quote ("), pipe (|), closing bracket (]), number sign (#), or backslash (\).

The **tacacs timeout** is the number of seconds that the WAAS device waits before declaring a timeout on a request to a particular TACACS+ server. The range is from 1 to 20 seconds, with 5 seconds as the default. The number of times that the WAAS device repeats a retry-timeout cycle before trying the next TACACS+ server is specified by the **tacacs retransmit** command. The default is two retry attempts.

Three unsuccessful login attempts are permitted. TACACS+ logins may appear to take more time than local logins depending on the number of TACACS+ servers and the configured timeout and retry values.

Use the **tacacs password ascii** command to specify the TACACS+ password type as ASCII. The default password type is PAP (Password Authentication Protocol).When the **no tacacs password ascii** command is used to disable the ASCII password type, the password type is once again reset to PAP.

If you do not use the **primary** keyword to specify the primary server, the primary server is the first one configured. If you remove the primary server by using the **no tacacs host** command, the first configured server (other than the removed server) becomes the primary server.

You can configure multiple TACACS+ servers; authentication is attempted on the primary server first. If the primary server is unreachable, then authentication is attempted on the other servers in the TACACS+, in the order in which they were configured. If authentication fails for any reason other than a server is unreachable, authentication is not attempted on the other servers in the farm. This process applies regardless of the setting of the **authentication fail-over server-unreachable** command.

Examples

The following example shows how to configure the key used in encrypting packets:

```
WAE(config)# tacacs key human789
```

The following example shows how to configure the host named spearhead as the primary TACACS+ server:

```
WAE(config)# tacacs host spearhead primary
```

The following example shows how to set the timeout interval for the TACACS+ server:

```
WAE(config)# tacacs timeout 10
```

The following example shows how to set the number of times that authentication requests are retried (retransmitted) after a timeout:

```
WAE(config)# tacacs retransmit 5
```

The following example shows the password type to be PAP by default:

```
WAE# show tacacs
```

```
    Login Authentication for Console/Telnet Session: enabled (secondary)
    Configuration Authentication for Console/Telnet Session: enabled (secondary)

    TACACS+ Configuration:
    ----------------------
    TACACS+ Authentication is off
    Key        = *****
    Timeout    = 5
    Retransmit = 2
    Password type: pap

    Server                          Status
    ----------------------------   ------
    10.107.192.148                primary
    10.107.192.168
    10.77.140.77
```

You can configure the password type to be ASCII using the **tacacs password ascii** command. You can then verify the changes using the **show tacacs** command.

```
WAE(config)# tacacs password ascii
WAE(config)# exit
WAE# show tacacs
    Login Authentication for Console/Telnet Session: enabled (secondary)
    Configuration Authentication for Console/Telnet Session: enabled (secondary)

    TACACS+ Configuration:
    ----------------------
    TACACS+ Authentication is off
    Key        = *****
    Timeout    = 5
    Retransmit = 2
    Password type: ascii

    Server                          Status
    ----------------------------   ------
    10.107.192.148                primary
    10.107.192.168
    10.77.140.77
```

Related Commands

- **(config) authentication configuration**
- **show authentication**
- **show statistics authentication**
- **show statistics tacacs**
- **show tacacs**

(config) tcp

To configure TCP parameters on a WAAS device, use the **tcp** global configuration command. To disable TCP parameters, use the **no** form of this command.

tcp {**cwnd-base** *segments* | **ecn enable** | **increase-xmit-timer-value** *value* | **init-ss-threshold** *value* | **keepalive-probe-cnt** *count* | **keepalive-probe-interval** *seconds* | **keepalive-timeout** *seconds*}

no tcp {**cwnd-base** *segments* | **ecn enable** | **increase-xmit-timer-value** *value* | **init-ss-threshold** *value* | **keepalive-probe-cnt** *count* | **keepalive-probe-interval** *seconds* | **keepalive-timeout** *seconds*}

Syntax Description

cwnd-base *segments*	Sets initial send congestion window in segments (1–10).
ecn enable	Enables TCP explicit congestion notification.
increase-xmit-timer-value *value*	Specifies the factor (1-3) used to modify the length of the retransmit timer by 1 to 3 times the base value determined by the TCP algorithm. **Note** Use this keyword with caution. The keyword can improve throughput when TCP is used over slow reliable connections but should never be changed in an unreliable packet delivery environment.
init-ss-threshold *value*	Sets initial slow-start threshold value (2-10).
keepalive-probe-cnt *count*	Specifies the length of time that the WAAS device keeps an idle connection open. The number of probe counts is 1–10.
keepalive-probe-interval *seconds*	Specifies the number of times that the WAAS device retries a connection. The keepalive probe interval is in seconds (1–300).
keepalive-timeout *seconds*	Specifies the length of time that the WAAS device keeps a connection open before disconnecting. The keepalive timeout is in seconds (1–3600).

Defaults

tcp cwnd-base: 2

tcp increase-xmit-timer-value: 1

tcp init-ss-threshold: 2 segments

tcp keepalive-probe-cnt: 4

tcp keepalive-probe-interval: 75 seconds

tcp keepalive-timeout: 90 seconds

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

The following are the usage guidelines for this command:

Caution Be careful using these parameters. In nearly all environments, the default TCP settings are adequate. Fine tuning of TCP settings is for network administrators who are experienced and have a full understanding of TCP operation details. See the *Cisco Wide Area Application Services Configuration Guide* for more information.

Use the **tcp keepalive-probe-cnt** global configuration command to specify how many times the WAAS device should attempt to connect to the device before closing the connection. The count can be from 1 to 10. The default is 4 attempts.

Use the **tcp keepalive-probe-interval** global configuration command to specify how often the WAAS device is to send out a TCP keepalive. The interval can be from 1 to 120 seconds. The default is 75 seconds.

Use the **tcp keepalive-timeout** global configuration command to wait for a response (the device does not respond) before the WAAS device logs a miss. The timeout can be from 1 to 120 seconds. The default is 90 seconds.

Examples

The following example shows how to enable a TCP explicit congestion notification:

```
WAE(config)# tcp ecn enable
```

Related Commands

clear arp-cache

show statistics tcp

show tcp

(config) telnet enable

To enable Telnet on a WAAS device, use the **telnet enable** global configuration command. To disable this feature, use the **no** form of this command.

telnet enable

no telnet enable

Syntax Description

This command has no arguments or keywords.

Defaults

By default, the Telnet service is enabled on a WAAS device.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Use terminal emulation software to start a Telnet session with a WAAS device.

You must use a console connection instead of a Telnet session to define device network settings on the WAAS device. However, after you have used a console connection to define the device network settings, you can use a Telnet session to perform subsequent configuration tasks.

Note Messages transported between the client and the device are not encrypted.

Examples

The following example shows how to enable the use of Telnet on the WAAS device:

```
WAE(config)# telnet enable
```

Related Commands

telnet

show telnet

(config) tfo exception

To configure exception handling for Traffic Flow Optimization (TFO), use the **tfo exception** global configuration command. To disable TFO exception handling configuration, use the **no** form of this command.

tfo exception {**coredump** | **debug** | **no-coredump**}

no tfo exception {**coredump** | **debug** | **no-coredump**}

Syntax Description

coredump	Writes a core file (default).
debug	Hangs the system until it is explicitly restarted.
no-coredump	Restarts the accelerator and does not write a core file.

Defaults

The default is coredump.

Command Modes

global configuration

Device Modes

application-accelerator

Examples

The following example shows how to write TFO exeption handling to a core file using the **tfo exception** command:

WAE(config)# **`tfo exception coredump`**

Related Commands

(config) tfo optimize

(config) tfo optimize

To configure a WAE for Traffic Flow Optimization (TFO), use the **tfo optimize** global configuration command. To disable TFO optimization, use the **no** form of this command.

tfo optimize {**DRE** {**yes** | **no**} **compression** {**LZ** | **none**} | **full**}

no tfo optimize {**DRE** {**yes** | **no**} **compression** {**LZ** | **none**} | **full**}

Syntax Description

DRE	Configures TFO optimization with or without Data Redundancy Elimination (DRE).
yes	Enables DRE.
no	Disables DRE.
compression	Configures TFO optimization with or without generic compression.
LZ	Configures TFO optimization with Lempel-Ziv (LZ) compression.
none	Configures TFO optimization with no compression.
full	Configures TFO optimization with DRE and LZ compression. Using this keyword is the same as specifying the **tfo optimize DRE yes compression LZ** command.

Defaults

The default TFO optimization on a WAAS device is **tfo optimize full**.

Command Modes

global configuration

Device Modes

application-accelerator

Examples

The following example shows to configures TFO optimization with DRE and full compression using the **tfo optimize** command:

```
WAE(config)# tfo optimize DRE yes compression full
```

Related Commands

show statistics tfo

(config) tfo tcp adaptive-buffer-sizing

To configure a WAE for Traffic Flow Optimization (TFO) with TCP adaptive buffering, use the **tfo tcp adaptive-buffer-sizing** global configuration command. To disable adaptive buffer sizing or to unconfigure the buffer size, use the **no** form of this command.

tfo tcp adaptive-buffer-sizing {enable | receive-buffer-max *size* **| send-buffer-max** *size*}

no tfo tcp adaptive-buffer-sizing {enable | receive-buffer-max *size* **| send-buffer-max** *size*}

Syntax Description

enable	Enables TCP adaptive buffer sizing.
receive-buffer-max *size*	Sets the maximum size of the receive buffer. Valid values range from 1 to 32768 KB.
send-buffer-max *size*	Sets the maximum size of the send buffer. Valid values range from 1 to 32768 KB.

Defaults

Adaptive buffering is enabled by default. The default maximum send and receive buffer sizes depend on the WAE device model.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

If you would rather use preallocated and unchanging send and receive buffers, you can configure them with the following global configuration commands: **tfo tcp optimized-receive-buffer**, **tfo tcp optimized-send-buffer**, **tfo tcp original-receive-buffer**, and **tfo tcp original-send-buffer**. You can turn off adaptive buffer sizing by using the **no tfo tcp adaptive-buffer-sizing** command.

Examples

The following example shows how to configure a WAE for Traffic Flow Optimization (TFO) with TCP adaptive buffering using the **tfo tcp adaptive-buffer-sizing** command:

```
WAE(config)# tfo tcp adaptive-buffer-sizing enable
```

Related Commands

(config) tfo tcp optimized-mss

(config) tfo tcp optimized-receive-buffer

(config) tfo tcp optimized-send-buffer

(config) tfo tcp original-receive-buffer

(config) tfo tcp original-send-buffer

show tfo tcp

(config) tfo tcp keepalive

To configure a WAE for Traffic Flow Optimization (TFO) with TCP keepalives, use the **tfo tcp keepalive** global configuration command. To disable TFO TCP keepalives, use the **no** form of this command.

tfo tcp keepalive

no tfo tcp keepalive

Syntax Description

This command has no arguments or keywords.

Defaults

Keepalives are disabled by default.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

This command enables TCP keepalives on the TFO optimized sockets (the connection between two peer WAEs).

Examples

The following example shows how to configure a WAE for Traffic Flow Optimization with TCP keepalives using the **tfo tcp keepalive** command:

```
WAE(config)# tfo tcp keepalive
```

Related Commands

(config) tfo tcp optimized-mss

(config) tfo tcp optimized-receive-buffer

(config) tfo tcp optimized-send-buffer

(config) tfo tcp original-mss

(config) tfo tcp original-receive-buffer

(config) tfo tcp original-send-buffer

(config) tfo tcp optimized-mss

To configure a WAE for Traffic Flow Optimization (TFO) with an optimized-side TCP maximum segment size, use the **tfo tcp optimized-mss** global configuration command. To disable this function, use the **no** form of this command.

tfo tcp optimized-mss *segment-size*

no tfo tcp optimized-mss *segment-size*

Syntax Description

segment-size	Optimized side TCP max segment size (512–9216).

Defaults

The default value of the segment size is 1432 bytes. If a jumbo MTU is configured, the default segment size is the jumbo MTU value – 68 bytes.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

This command sets the TCP maximum segment size on TFO optimized sockets (the connection between two peer WAEs).

Examples

The following example shows how to configure a WAE for Traffic Flow Optimization with an optimized-side TCP maximum segment size of 512 using the **tfo tcp optimized-mss** command:

```
WAE(config)# tfo tcp optimized-mss 512
```

Related Commands

(config) tfo tcp keepalive

(config) tfo tcp optimized-receive-buffer

(config) tfo tcp optimized-send-buffer

(config) tfo tcp original-mss

(config) tfo tcp original-receive-buffer

(config) tfo tcp original-send-buffer

(config) tfo tcp optimized-receive-buffer

To configure a WAE for Traffic Flow Optimization (TFO) with an optimized-side receive buffer, use the **tfo tcp optimized-receive-buffer** global configuration command. To disable this function, use the **no** form of this command.

tfo tcp optimized-receive-buffer *buffer-size*

no tfo tcp optimized-receive-buffer *buffer-size*

Syntax Description

buffer-size	Receive buffer size in kilobytes. Valid values range from 1 to 32768 KB.

Defaults

32 KB

Command Modes

global configuration

Device Modes

application-accelerator

Examples

The following example shows how to configure a WAE for Traffic Flow Optimization with a 32 KB optimized-side receive buffer using the **tfo tcp optimized-receive-buffer** command:

```
WAE(config)# tfo tcp optimized-receive-buffer 32
```

Related Commands

(config) tfo tcp keepalive

(config) tfo tcp optimized-mss

(config) tfo tcp optimized-send-buffer

(config) tfo tcp original-mss

(config) tfo tcp original-receive-buffer

(config) tfo tcp original-send-buffer

(config) tfo tcp optimized-send-buffer

To configure a WAE for Traffic Flow Optimization (TFO) with an optimized-side send buffer, use the **tfo tcp optimized-send-buffer** global configuration command. To disable this function, use the **no** form of this command.

tfo tcp optimized-send-buffer *buffer-size*

no tfo tcp optimized-send-buffer *buffer-size*

Syntax Description

buffer-size	Send buffer size in kilobytes. Valid values range from 1 to 32768 KB.

Defaults

32 KB

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

The buffer should be equal to or greater than twice the Bandwidth Delay Product (BDP). The BDP is equivalent to the bandwidth (in bits per second) * latency (in seconds). For example, for a 45-Mbps link with a 150-ms (0.15 sec) round-trip delay, the BDP is 45 Mbps * 0.15 sec = 6.75 Mb, or 0.844 MB (844 KB). In this case, you could set the buffer size to 2000 KB.

Examples

The following example shows how to configure a WAE for Traffic Flow Optimization with a 32 KB optimized-side send buffer using the **tfo tcp optimized-send-buffer** command:

```
WAE(config)# tfo tcp optimized-send-buffer 32
```

Related Commands

(config) tfo tcp keepalive

(config) tfo tcp optimized-mss

(config) tfo tcp optimized-receive-buffer

(config) tfo tcp original-mss

(config) tfo tcp original-receive-buffer

(config) tfo tcp original-send-buffer

(config) tfo tcp original-mss

To configure a WAE for Traffic Flow Optimization (TFO) with an unoptimized-side TCP maximum segment size, use the **tfo tcp original-mss** global configuration command. To disable this function, use the **no** form of this command.

tfo tcp original-mss *segment-size*

no tfo tcp original-mss *segment-size*

Syntax Description

segment-size	Original (end-point) side TCP max segment size (512–9216).

Defaults

The default value of the segment size is 1432 bytes. If a jumbo MTU is configured, the default segment size is the jumbo MTU value – 68 bytes.

Command Modes

global configuration

Device Modes

application-accelerator

Examples

The following example shows how to configure a WAE for Traffic Flow Optimization with a 1432 byte unoptimized-side TCP maximum segment size using the **tfo tcp original-mss** command:

```
WAE(config)# tfo tcp original-mss 1432
```

Related Commands

(config) tfo tcp keepalive

(config) tfo tcp optimized-mss

(config) tfo tcp optimized-receive-buffer

(config) tfo tcp optimized-send-buffer

(config) tfo tcp original-receive-buffer

(config) tfo tcp original-send-buffer

(config) tfo tcp original-receive-buffer

To configure a WAE for Traffic Flow Optimization (TFO) with an unoptimized-side receive buffer, use the **tfo tcp original-receive-buffer** global configuration command. To disable this function, use the **no** form of this command.

tfo tcp original-receive-buffer *buffer-size*

no tfo tcp original-receive-buffer *buffer-size*

Syntax Description

buffer-size	Receive buffer size in kilobytes. Valid values range from 1 to 32768 KB.

Defaults

32 KB

Command Modes

global configuration

Device Modes

application-accelerator

Examples

The following example shows how to configure a WAE for Traffic Flow Optimization with a 32 KB unoptimized-side receive buffer using the **tfo tcp original-receive-buffer** command:

```
WAE(config)# tfo tcp original-receive-buffer 32
```

Related Commands

(config) tfo tcp keepalive

(config) tfo tcp optimized-mss

(config) tfo tcp optimized-receive-buffer

(config) tfo tcp optimized-send-buffer

(config) tfo tcp original-mss

(config) tfo tcp original-send-buffer

(config) tfo tcp original-send-buffer

To configure a WAE for Traffic Flow Optimization (TFO) with an unoptimized-side send buffer, use the **tfo tcp original-send-buffer** global configuration command. To disable this function, use the **no** form of this command.

tfo tcp original-send-buffer *buffer-size*

no tfo tcp original-send-buffer *buffer-size*

Syntax Description

buffer-size	Send buffer size in kilobytes. Valid values range from 1 to 32768 KB.

Defaults

32 KB

Command Modes

global configuration

Device Modes

application-accelerator

Examples

The following example shows how to configure a WAE for Traffic Flow Optimization with a 32 KB unoptimized-side receive buffer using the **tfo tcp original-send-buffer** command:

```
WAE(config)# tfo tcp original-send-buffer 32
```

Related Commands

(config) tfo tcp keepalive

(config) tfo tcp optimized-mss

(config) tfo tcp optimized-receive-buffer

(config) tfo tcp optimized-send-buffer

(config) tfo tcp original-mss

(config) tfo tcp original-receive-buffer

(config) threshold-monitor

To configure monitoring thresholds, use the **threshold-monitor** global configuration command. To restore default settings, use the **no** form of this command.

threshold-monitor {**appnav-controller asymmetric-flow-query-failure value** *number* | **accelerator cifs** {**directory resources** *no_of _resources* | **ff-average-local-response-time** *milliseconds* | **ff-average-remote-response-time** *milliseconds* | **open-files** *number*} | **system load** *percent* | **cpu** {*percent* | **enable** } | **softirq enable**}

no threshold-monitor {**appnav-controller asymmetric-flow-query-failure value** *number* | **accelerator cifs** {**directory resources** *no_of _resources* | **ff-average-local-response-time** *milliseconds* | **ff-average-remote-response-time** *milliseconds* | **open-files** *number*} | **system load** *percent* | **cpu** {*percent* | **enable** } | **softirq enable**}

Syntax Description

appnav-controller asymmetric-flow-query -failure value *number*	Sets the asymmetric connections threshold to the specified number of asymmetric connections (1-3000 in thousands) in a one-minute interval.
accelerator cifs	Configures the threshold values for the accelerator cifs.
directory resources *no_of _resources*	Sets the directory resources threshold to the specified number (10-700). The directory resources indicate the number of available directory resources in the system.
ff-average-local-response-time *milliseconds*	Sets the find first average local response time threshold to the specified time (10–9999999) in milliseconds. The find first average local response time parameter indicates the average response time for find first requests that are served locally.
ff-average-remote-response-time *milliseconds*	Sets the first find average remote response time threshold to the specified time (10–9999999) in milliseconds. The find first average remote response time parameter indicates the average response time of find first requests that are served by the file server.
open-files *number*	Sets the open files threshold to the specified number (10-9999999). The open files counter maintains the number of open files and directories.
system load *percent*	Sets the system load threshold to the specified percentage (80–100) of rated connection capacity.
cpu	Configures the threshold value for CPU load monitoring.
percent	Sets the CPU threshold to the specified percentage (80–100) of the CPU usage for the system CPU load monitoring and for monitoring the softirq CPU usage. The default CPU threshold is 95 percent.
enable	Enables CPU load monitoring.
softirq enable	Enables load monitoring of CPU utilization on the CPUs that are processing incoming TCP packets.

Defaults

The asymmetric connections threshold is 600,000.

The system load percentage is 95 percent of rated connection capacity for the device.

The CPU load percentage is 95 percent of the the total CPU usage.

Command Modes global configuration

Device Modes application-accelerator

appnav-controller

Usage Guidelines An asymmetric connection occurs if an ANC receives a SYN-ACK packet for which it does not have a flow state and for which no other ANC in the cluster has a flow state. This indicates a problem where the ANC is not intercepting both directions of the connection.

If the asymmetric connections threshold is exceeded during a one-minute interval, a "Total failed asymmetric flow queries has crossed threshold limit" alarm is raised. This alarm is cleared after the number of asymmetric flow learning failures drops below half of the configured threshold for five consecutive minutes. This threshold applies only to ANCs in an AppNav deployment.

The CIFS accelerator performs below the optimum level when certain resources are running low, if the find first requests are too many, or if there are too many open files.

These parameters can be monitored by setting thresholds. If the configured threshold for any CIFS application accelerator is exceeded on a WAE, the relevant threshold alarm is raised. The alarm is cleared when the relevant count falls to less than the configured threshold. The alarms are CIFS 'Find First' Local Operation Above Threshold, CIFS 'Find First' Remote Operation Above Threshold, CIFS Directory Resources Low, and CIFS 'Open Files' Above Threshold.

The maximum value of the open-file count is platform dependent. For WAE 294, the maximum number of open files can be preconfigured to 500; for WAE 574, the maximum number is 1500; for WAE 674, the maximum number is 12000 and for WAE 7571, the maximum number of open files is 64000.

The system load percentage threshold refers to the percentage of connection capacity used for application accelerators and TFO connections on a WAE. If the configured load threshold for any application accelerator or TFO connections is exceeded on a WAE, the connection threshold exceeded alarm is raised. This alarm is cleared when the connection count falls to 10 percent less than the configured threshold (85 percent by default).

The CPU load threshold refers to the CPU load utilization on a WAE. When the average CPU utilization on the device exceeds the set threshold for 2 minutes, the device stops accepting new connections and passes any new connections through. When the average CPU utilization falls below the threshold for 2 minutes, the device resumes accepting optimized connections. You can disable CPU load monitoring by using the no form of the CPU enable command.

Examples The following example shows how to configure an asymmetric connection threshold of 100,000:

```
WAE(config)# threshold-monitor appnav-controller asymmetric-flow-query-failure value 100
```

The following example shows how to configure an open files threshold of 500:

```
WAE-231-03(config)# threshold-monitor accelerator cifs open-files 500
```

The following example shows how to configure a system load threshold of 90 percent:

```
WAE(config)# threshold-monitor system load 90
```

Related Commands

show statistics accelerator

show statistics connection

show statistics tfo

(config) transaction-logs

To configure and enable transaction logging on a WAE, use the **transaction-logs** global configuration command. To disable a transaction logging option, use the **no** form of this command.

transaction-logs {**accelerator video windows-media** | **flow**} **enable**

transaction-logs flow access-list *acl-name*

transaction-logs {**accelerator video windows-media** | **flow**} **archive interval** *seconds*

transaction-logs {**accelerator video windows-media** | **flow**} **archive interval every-day** {**at** *hour:minute* | **every** *hours*}

transaction-logs {**accelerator video windows-media** | **flow**} **archive interval every-hour** {**at** *minute* | **every** *minutes*}

transaction-logs {**accelerator video windows-media** | **flow**} **archive interval every-week** [**on** *weekdays* **at** *hour:minute*]

transaction-logs {**accelerator video windows-media** | **flow**} **archive max-file-size** *filesize*

transaction-logs {**accelerator video windows-media** | **flow**} **export compress**

transaction-logs {**accelerator video windows-media** | **flow**} **export enable**

transaction-logs {**accelerator video windows-media** | **flow**} **export ftp-server** {*hostname* | *servipaddrs*} [**management**] *login passw directory*

transaction-logs {**accelerator video windows-media** | **flow**} **export interval** *minutes*

transaction-logs {**accelerator video windows-media** | **flow**} **export interval every-day** {**at** *hour:minute* | **every** *hours*}

transaction-logs {**accelerator video windows-media** | **flow**} **export interval every-hour** {**at** *minute* | **every** *minutes*}

transaction-logs {**accelerator video windows-media** | **flow**} **export interval every-week** [**on** *weekdays* **at** *hour:minute*]

transaction-logs {**accelerator video windows-media** | **flow**} **export sftp-server** {*hostname* | *servipaddrs*} [**management**] *login passw directory*

Syntax Description

accelerator video windows-media	Specifies the video accelerator transaction log feature for Windows Media transactions.
flow	Specifies the TFO flow transaction log feature.
enable	Enables the transaction log feature.
access-list *acl-name*	Configures an access list name to restrict logged traffic. Only traffic that is included in the access list is logged.
archive	Configures archive parameters.
interval *seconds*	Determines how frequently the archive file is to be saved. Value is in seconds (120–604800).

every-day	Archives using intervals of 1 day or less.
at *hour:minute*	Specifies the local time at which to archive each day (hh:mm).
every *hours*	Specifies the interval in hours. The interval aligns with midnight. The intervals are as follows: 1 Hourly 12 Every 12 hours 2 Every 2 hours 24 Every 24 hours 3 Every 3 hours 4 Every 4 hours 6 Every 6 hours 8 Every 8 hours
every-hour	Specifies intervals of 1 hour or less.
at *minute*	Sets the time at each hour. The minute alignment for the hourly task is from 0 to 59.
every *minutes*	Specifies the interval in minutes for hourly task that aligns with the top of the hour. The intervals are as follows: 10 Every 10 minutes 15 Every 15 minutes 2 Every 2 minutes 20 Every 20 minutes 30 Every 30 minutes 5 Every 5 minutes
every-week	Specifies intervals of 1 or more times a week.
on *weekdays*	(Optional) Sets the day of the week and the weekdays on which to perform the task. You can specify one or more weekdays: Fri Every Friday Mon Every Monday Sat Every Saturday Sun Every Sunday Thu Every Thursday Tue Every Tuesday Wed Every Wednesday
max-file-size *filesize*	Specifies the maximum size in kilobytes (1000–2000000) of the archive file to be maintained on the local disk.
export	Configures file export parameters. The FTP export feature can support up to four servers. Each server must be configured with a username, password, and directory that are valid for that server.
compress	Enables compression of archived log files into a zip format before exporting them to external FTP servers.
ftp-server	Sets the FTP server to receive exported archived files.
hostname	Hostname of the target server.
servipaddrs	IP address of the target server.
management	Uses the designated management interface for exporting the log files.
login	User login to target server (1–10080).
passw	User password to target server (less than 40 characters).
directory	Target directory path for exported files on the server.

interval *minutes*	Specifies the interval in minutes (1–10080) at which to export a file.
sftp-server	Sets the Secure File Transfer Protocol (SFTP) server to receive exported archived files.

Defaults

The default settings for the logging feature are as follows:

archive: disabled

enable: disabled

export compress: disabled

export: disabled

archive interval: every day, every one hour

archive max-file-size: 2,000,000 KB

export interval: every day, every one hour

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

Related Commands

clear arp-cache

show transaction-logging

transaction-log

(config) username

To establish username authentication on a WAAS device, use the **username** global configuration command. To disable this feature, use the **no** form of this command.

username *name* {**passwd** | **privilege** {**0** | **15**}}

no username *name* {**passwd** | **privilege** {**0** | **15**}}

Syntax Description

name	Username.
passwd	Configures the password interactively.
privilege	Sets the user privilege level.
0	Specifies the user privilege level for the normal user.
15	Specifies the user privilege level for the superuser.

Defaults

The default administrator account is as follows:

- Username: admin
- Password: default
- Privilege: superuser (15)

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Note We strongly recommend that you use the WAAS Central Manager GUI instead of the WAAS CLI to configure passwords and privilege levels for users on your WAAS devices, if possible. For information about how to use the WAAS Central Manager GUI to centrally configure and administer users on a single WAE or group of WAEs, which are registered with a WAAS Central Manager, see the *Cisco Wide Area Application Services Configuration Guide*.

Examples

The following example demonstrates how passwords and privilege levels are reconfigured:

```
WAE(config)# username bwhidney passwd
Warning: User configuration performed via CLI may be overwritten
by the central manager.  Please use the central manager to configure
user accounts.
New WAAS password:
Retype new WAAS password:

WAE(config)# username abeddoe privilege 15
Warning: User configuration performed via CLI may be overwritten
by the central manager.  Please use the central manager to configure
```

user accounts.

Related Commands

show user

(config) virtual-blade

To configure virtual blades on your WAAS device, use the **virtual-blade** global configuration command. To negate these actions, use the **no** form of this command.

virtual-blade {*virtual-blade-number* | **enable**}

no virtual-blade {*virtual-blade-number* | **enable**}

Syntax Description

virtual-blade-number	Number of the virtual blade that you want to edit. This value can be from 1 through 6, depending on the number of virtual blades supported on the device. Using this command enables virtual blade configuration mode. See the "Virtual Blade Configuration Mode Commands" section for more information.
enable	Enables the virtual blade feature on your WAAS device. You must reboot the device after executing this command.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

Use the **virtual-blade** global configuration command to enter virtual blade configuration mode. This command is available only on WAAS devices that support virtual blades. See the "Virtual Blade Configuration Mode Commands" section for more information.

Examples

The following example shows how to enable the virtual blade feature on your device:

```
WAE(config)# virtual-blade enable
WAE(config)# exit
```

The following example shows that after a reload, you can enter the **show** EXEC command to see the new virtual blade resource allocation:

```
# show virtual-blade
Virtual-blade resources:
  VB Memory: 299MiB configured, 1749MiB available.
  VB Disk space: 0GiB configured, 204GiB available.
  /local1/vbs: 128MiB used, 214203MiB available
  CPU(s) assigned: 3 4
Virtual-blade(s) state:
  virtual-blade 2 has incomplete configuration
```

The following example puts your device into virtual blade configuration mode, editing virtual blade 2. The mode change is indicated by the system prompt:

```
WAE(config)# virtual-blade 2
WAE(config-vb)#
```

Related Commands

show virtual-blade

(config-vb) autostart

(config-vb) boot

(config-vb) cpu-list

(config-vb) description

(config-vb) device

(config-vb) disk

(config-vb) interface

(config-vb) memory

(config-vb) vnc

(config) vn-service vpath

To enable VPATH interception on your vWAAS device, use the **vn-service vpath** global configuration command. To disable this feature, use the **no** form of this command.

vn-service vpath

no vn-service vpath

Syntax Description

This command has no arguments or keywords.

Defaults

VPATH interception is disabled by default.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

Use the **vn-service vpath** global configuration command to enable VPATH interception on your vWAAS device. VPATH intercepts traffic from the VM server and redirects it to a vWAAS device for WAN optimization, and then returns the response back to the VEM. The vWAAS egress traffic received by VEM is forwarded without further VPATH interception.

Note Only one type of interception can be enabled at a time on a vWAAS device (VPATH or WCCP).

The following example shows how to enable VPATH interception on a vWAAS device:

```
WAE(config)# vn-service vpath
```

Related Commands

show statistics vn-service vpath

clear statistics vn-service vpath

(config) wccp access-list

To configure an IP access list on a WAE for inbound WCCP GRE encapsulated traffic, use the **wccp access-list** global configuration command. To disable this feature, use the **no** form of this command.

wccp access-list {*acl-number* | *ext-acl-number* | *acl-name*}

no wccp access-list {*acl-number* | *ext-acl-number* | *acl-name*}

Syntax Description

acl-number	Standard IP access list number (1–99).
ext-acl-number	Extended IP access list number (100–199).
acl-name	Name of the access list. You can use a maximum of 30 characters.

Defaults

WCCP access lists are not configured by default.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

The **wccp access-list** *number* global configuration command configures an access control list to allow access to WCCP applications. See the *Cisco Wide Area Application Services Configuration Guide* for a detailed description of how to use standard IP ACLs to control WCCP access on a WAE.

Note WCCP works only with IPv4 networks. WCCP commands are available only after the interception method is set to WCCP by the **interception-method** command.

Examples

The following example shows how to configure the WAE to apply IP access list number 10 to the inbound WCCP traffic:

```
WAE(config)# wccp access-list 10
```

The following example shows sample output from the **show ip access-list** EXEC command from a WAE that has several WCCP access lists configured:

```
WAE(config)# show ip access-list
Space available:
    40 access lists
   489 access list conditions

Standard IP access list 10
   1 deny 10.1.1.1
   2 deny any
     (implicit deny any: 0 matches)
   total invocations: 0
Standard IP access list 98
```

```
    1 permit any
      (implicit deny any: 0 matches)
    total invocations: 0
Extended IP access list 100
    1 permit icmp any any
      (implicit fragment permit: 0 matches)
      (implicit deny ip any any: 0 matches)
    total invocations: 0
Extended IP access list 101
    1 permit ip any any
      (implicit fragment permit: 0 matches)
      (implicit deny ip any any: 0 matches)
    total invocations: 0
Extended IP access list 102
    1 permit icmp 0.0.1.1 255.255.0.0 any
      (implicit fragment permit: 0 matches)
      (implicit deny ip any any: 0 matches)
    total invocations: 0
Extended IP access list 111
    1 permit gre 0.1.1.1 255.0.0.0 any
      (implicit fragment permit: 0 matches)
      (implicit deny ip any any: 0 matches)
    total invocations: 0
Extended IP access list 112
    1 permit ip any any
      (implicit fragment permit: 0 matches)
      (implicit deny ip any any: 0 matches)
    total invocations: 0
Extended IP access list 113
    1 permit gre 0.1.1.1 255.0.0.0 any
      (implicit fragment permit: 0 matches)
      (implicit deny ip any any: 0 matches)
    total invocations: 0
Extended IP access list ext_acl_2
    1 permit gre any any
      (implicit fragment permit: 0 matches)
      (implicit deny ip any any: 0 matches)
    total invocations: 0
Extended IP access list extended_ip_acl
    1 permit tcp any eq 2 any eq exec
      (implicit fragment permit: 0 matches)
      (implicit deny ip any any: 0 matches)
    total invocations: 0

Interface access list references:
  PortChannel     2    inbound   extended_ip_acl
  PortChannel     2    outbound  101

Application access list references:
  snmp-server                    standard  2
    UDP ports: none (List Not Defined)
  WCCP                           either    10
    Any IP Protocol
```

Related Commands

show ip access-list

show wccp

(config) wccp flow-redirect

To enable WCCP flow redirection on a WAE, use the **wccp flow-redirect** global configuration command. To disable flow redirection, use the **no** form of this command.

wccp flow-redirect enable [**timeout** *seconds*]

no wccp flow-redirect enable

Syntax Description

enable	Enables flow redirection (protection).
timeout *seconds*	Sets the maximum amount of time for which to enable flow protection, in seconds (0-86400). If you do not specify this option, flow protection is enabled with no timeout (indefinitely).

Defaults

Disabled

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

Use the **wccp flow-redirect** global configuration command to implement WCCP flow protection. Flow protection is designed to keep the TCP flow intact as well as to not overwhelm WAEs when they are first started up or are reassigned new traffic. This feature also has a slow start mechanism where the WAEs try to take a load appropriate for their capacity.

In a WCCP cache farm, the flow protection timeout value configured in the lead WAE is effective regardless of the values configured in the other WAEs in cache farm. After the timeout value is expired, the flow flush request is sent by the lead WAE to all other WAEs in the cache farm.

Note WCCP works only with IPv4 networks. WCCP commands are available only after the interception method is set to WCCP by the **interception-method** command.

Note Designs that require redirected frames to be returned to the originating router are not compatible with the WCCP flow protection feature.

Examples

The following example shows how to enable WCCP flow protection on a WAE for one hour:

```
WAE(config)# wccp flow-redirect enable timeout 3600
```

Related Commands show wccp

(config) wccp router-list

To configure a router list for WCCP Version 2, use the **wccp router-list** global configuration command. To disable this function, use the **no** form of this command.

wccp router-list *number ip-address*

no wccp router-list *number ip-address*

Syntax Description

number	Router list number (1–7).
ip-address	IP address of the router to add to the list. You can specify up to 32 IP addresses, each separated by the space character.

Defaults

Disabled

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

Usage Guidelines

Each router list can contain up to 32 routers and you can have up to 8 router lists.

Note The WAAS Central Manager uses router list number 8 for a default router list that contains the default gateway.

Note The **ip wccp** global configuration command must be used to enable WCCP on each router that is included on the router list.

WCCP works only with IPv4 networks. WCCP commands are available only after the interception method is set to WCCP by the **interception-method** command.

Examples

The following example shows that router list number 2 is created and contains a single router (the WCCP Version 2-enabled router with IP address 192.168.68.98):

```
WAE(config)# wccp router-list 2 192.168.68.98
```

The following example shows how to delete the router list number 2 created in the previous example:

```
WAE(config)# no wccp router-list 2 192.168.68.98
```

The following example shows how to create a router list (router list 1) with two routers and then configure the WAE to accept redirected TCP traffic from the WCCP Version 2-enabled router on router list 1:

```
WAE(config)# wccp router-list 1 10.10.10.2 10.10.10.3
WAE(config)# wccp tcp-promiscuous service-pair 61 62
WAE(config-wccp-service)# router-list-num 1
WAE(config-wccp-service)# enable
```

Related Commands

(config) wccp tcp-promiscuous service-pair

(config) wccp shutdown

To set the maximum time interval after which the WAE will perform a clean shutdown of the WCCP, use the **wccp shutdown** global configuration command. To disable the clean shutdown, use the **no** form of this command.

wccp shutdown max-wait *seconds*

no wccp shutdown max-wait *seconds*

Syntax Description

max-wait *seconds*	Sets the clean shutdown time interval. The time is in seconds (0–86400). The default is 120 seconds

Defaults

The maximum time interval before a clean shutdown is 120 seconds.

Command Modes

global configuration

Device Modes

application-accelerator

Usage Guidelines

To prevent broken TCP connections, the WAE performs a clean shutdown of the WCCP after you enter the **reload** command or disable WCCP. The WAE does not reboot until either all connections have been serviced or the configured **max-wait** interval has elapsed.

Note WCCP works only with IPv4 networks. WCCP commands are available only after the interception method is set to WCCP by the **interception-method** command.

Examples

The following example shows how to configure the WAE to wait 1000 seconds:

```
WAE(config)# wccp shutdown max-wait 1000
```

The following example shows how to shut down WCCP Version 2 on the WAE by entering the **no enable** WCCP command. After you enter this command, the WAE waits 1000 seconds before it shuts down WCCP Version 2.

```
WAE(config)# wccp tcp-promiscuous service-pair 61 62
WAE(config-wccp-service)# no enable
```

A countdown message appears, indicating how many seconds remain before WCCP will be shut down on the WAE:

```
WCCP clean shutdown initiated
Waiting for shutdown ok (999 seconds) . Press ^C to skip waiting
WCCP clean shutdown wait time expired
```

Related Commands (config) wccp tcp-promiscuous service-pair

(config) wccp tcp-promiscuous service-pair

To configure the Web Cache Coordination Protocol (WCCP) Version 2 TCP promiscuous mode service, use the **wccp tcp-promiscuous service-pair** global configuration command. To negate these actions, use the **no** form of this command.

wccp tcp-promiscuous {**service-pair** *serviceID serviceID+1* | *serviceID*}

no wccp tcp-promiscuous {**service-pair** *serviceID serviceID+1* | *serviceID*}

Syntax Description

service-pair *serviceID serviceID+1*	Specifies a pair of IDs for the WCCP service on devices configured as application accelerators. Valid values are two consecutive numbers from 1-100, inclusive.
serviceID	Specifies one ID for the WCCP service. A valid value is from 1-100, inclusive. On devices operating as AppNav Controllers, you can specify either one or two service IDs.

Defaults

No default behavior or values.

Command Modes

global configuration

Device Modes

application-accelerator

appnav-controller

Usage Guidelines

Use the **wccp tcp-promiscuous service-pair** command to configure and enable the WCCP interception method. This command initiates the WCCP configuration mode as indicated by the (config-wccp-service) prompt. For more information on WCCP configuration mode commands, see the "WCCP Configuration Mode Commands" section.

Within WCCP configuration mode, you can use the various commands (**egress-method**, **failure-detection**, and so on) to define WCCP settings. To return to global configuration mode, enter the **exit** command.

You must use the **enable** WCCP configuration command to enable the WCCP service.

You must configure two WCCP service IDs on WAEs operating in application-acceleration mode. On WAEs operating as AppNav Controllers, you can specify either one or two service IDs.

Note WCCP works only with IPv4 networks. WCCP commands are available only after the interception method is set to WCCP by the **interception-method** global configuration command.

Examples

The following example shows how to configure WCCP service IDs 61 and 62 and put a WAE into WCCP configuration mode:

```
WAE(config)# wccp tcp-promiscuous service-pair 61 62
WAE(config-wccp-service)#
```

The following example shows how to configure WCCP service ID 61 and put an AppNav Controller into WCCP configuration mode:

```
WAE(config)# wccp tcp-promiscuous 61
WAE(config-wccp-service)#
```

Related Commands

(config) wccp router-list

(config) wccp shutdown

show wccp

(config) windows-domain

To configure Windows domain server options on a WAAS device, use the **windows-domain** global configuration command. To disable this feature, use the **no** form of this command.

windows-domain {**administrative group** {**normal-user** | **super-user**} *groupname* | **comment** *string* | **encryption-service** {**enable** | **identity** *name* [**default** | **enable** | **machine-account** | **match** | **password** | **user-account**]} | **ldap-sign-and-seal enable** | **machine-account-password lifespan** *duration* | **netbios-name** *name* | **password-server** {*hostname* | *ipaddress*} | **realm** *kerberos-realm* | **wins-server** {*hostname* | *ipaddress*} | **workgroup** *name* | **security ADS**}

no windows-domain {**administrative group** {**normal-user** | **super-user**} *groupname* | **comment** *string* | **encryption-service** {**enable** | **identity** *name*} | **ldap-sign-and-seal enable** | **machine-account-password lifespan** *duration* | **netbios-name** | **password-server** {*hostname* | *ipaddress*} | **realm** *kerberos-realm* | **wins-server** {*hostname* | *ipaddress*} | **workgroup** *name* | **security ADS**}

Syntax Description

administrative	Sets administrative options.
group	Sets an administrative group name.
normal-user	Sets the administrative group name for the normal user (privilege 0).
super-user	Sets the administrative group name for the superuser (privilege 15).
groupname	Name of the administrative group.
comment *string*	Specifies a comment for the Windows domain server. This is a text string.
encryption-service	Configures encrypted service.
enable	Enables encrypted service.
identity *name*	Specifies the encrypted service identity to manage. The name is the WAAS tag-name identifier.
default	Sets the identity as the default match.
machine-account	Specifies machine account identity.
match	Specifies a match.
password	Specifies the password for the identity.
user-account *name*	Defines and edits the user account identity.
ldap-sign-and-seal	Configures the LDAP sign and seal service.
enable	Enables the LSAP sign and seal service. This service is disabled by default.
machine-account-password	Configures the password settings.
lifespan *duration*	Configures the lifespan duration in seconds. The minimum is 1 hour, the maximum is 60 days, and the default is 30 days.
netbios-name *name*	Specifies the NetBIOS name of the WAE. The NetBIOS name must not consist of only numbers; it must include some letters.
password-server	Specifies the password server used to verify a client password.
hostname	Hostname of the password server.
ipaddress	IP address of the password server.

realm *kerberos-realm*	Specifies the Kerberos realm to use for authentication. The realm is used as the Active Directory Service (ADS) equivalent of the NT4 domain. This argument is valid only when Kerberos ADS mode is used. The value is an IP address or name (in uppercase letters) of the Kerberos realm. The Kerberos realm is typically set to the DNS name of the Kerberos server or Active Directory domain. The default value is a null string. Example: kerberos-realm = MYBOX.MYCOMPANY.COM
wins-server	Specifies the Windows Internet Naming Service (WINS) server.
hostname	Hostname of the WINS server.
ipaddress	IP address of the WINS server.
workgroup *name*	Specifies the name of the workgroup (or domain) in which the WAAS device resides.
security	Sets Kerberos authentication.
ADS	Specifies the Active Directory Service.

Defaults

Windows domain options are disabled by default.

Command Modes

global configuration

Device Modes

application-accelerator

central-manager

Usage Guidelines

Use this global configuration command to set the Windows domain server parameters for a WAAS device.

When you enable Kerberos authentication, the default **realm** is DOMAIN.COM and the **security** is ADS. If you disable Kerberos authentication, the **security** is domain.

Note WAAS supports authentication by a Windows domain controller running only on Windows Server 2000 or Windows Server 2003.

Examples

The following example shows how to configure the Windows domain server at 10.10.24.1 for a WAAS device with a NetBIOS name of myWaasDevice in the ABC domain. It also identifies the password server:

```
WAE(config)# windows-domain wins-server 10.10.24.1
WAE(config)# windows-domain password-server 10.10.100.4
WAE(config)# windows-domain netbios-name myWaasDevice
WAE(config)# windows-domain workgroup ABC
```

The following example shows how to configure the windows domain server when Kerberos authentication is enabled using the **kerberos** command:

```
WAE(config)# windows-domain realm ABC.COM
```

```
WAE(config)# windows-domain security ADS

 =============== checking new config using testparm ===================

Load smb config files from /state/actona/conf/smb.conf
Processing section "[print$]"
Processing section "[printers]"
Loaded services file OK.

WAE(config)# exit
WAE# show windows-domain
  Login Authentication for Console/Telnet Session: enabled

  Windows domain Configuration:
  -----------------------------
    Workgroup:
    Comment: Comment:
    Net BIOS: MYWAASDEVICE
    Realm: ABC
    WINS Server: 10.10.10.1
    Password Server: 10.10.10.10
    Security: ADS
```

Related Commands

(config) kerberos

show windows-domain

windows-domain