CHAPTER 10

database path through debug xml Commands

database path

To specify a path or location for the local CA server database, use the **database** command in CA server configuration mode. To reset the path to flash memory, the default setting, use the **no** form of this command.

[no] database path *mount-name directory-path*

Syntax Description

directory-path	Specifies the path to a directory on the mount point where the CA files are stored.
mount-name	Specifies the mount name.

Defaults

By default, the CA server database is stored in flash memory.

Command Modes

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

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

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

The local CA files stored in the database include the certificate database, user database files, temporary PKCS12 files, and the current CRL file. The *mount-name* is the same as the *name* argument for the **mount** command used to specify a file system for the security appliance.

Note These CA files are internal stored files and should not be modified.

Examples

The following example defines the mount point for the CA database as cifs_share. It also defines the database files directory on the mount point as ca_dir/files_dir.

```
hostname(config)# crypto ca server
hostname(config-ca-server)# database path cifs_share ca_dir/files_dir/
hostname(config-ca-server)#
```

Related Commands

Command	Description
crypto ca server	Provides access to CA Server Configuration mode CLI command set, which allows the user to configure and manage a local CA.
crypto ca server user-db write	Writes the user information configured in the local CA database to disk.
debug crypto ca server	Shows debug messages when the user configures the local CA server.
mount	Makes Common Internet File System (CIFS) and/or File Transfer Protocol (FTPFS) file systems accessible to the security appliance
show crypto ca server	Displays the characteristics of the CA configuration on the security appliance.
show crypto ca server cert-db	Displays the certificates issued by the CA server.

ddns (DDNS-update-method)

To specify a DDNS update method type, use the **ddns** command in DDNS-update-method mode. To remove an update method type from the running configuration, use the **no** form of this command.

ddns [**both**]

no ddns [**both**]

Syntax Description

both	(Optional) Specifies updating to both the DNS A and PTR resource records (RRs).

Defaults

Update only A RRs.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
DDNS-update-method	•	—	•	•	—

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

Dynamic DNS (DDNS) updates the name to address and address to name mappings maintained by DNS. Of the two methods for performing DDNS updates—the IETF standard defined by RFC 2136 and a generic HTTP method—the security appliance supports the IETF method in this release.

Name and address mappings are contained in two types of resource records (RR):

- The A resource record contains domain name to IP address mappings.
- The PTR resource record contains IP address to domain name mappings.

DDNS updates can be used to maintain consistent information between the A and PTR RR types.

When issued in DDNS-update-method configuration mode, the **ddns** command defines whether the update is just to A RR, or to both A RR and PTR RR.

Examples

The following example configures updating to both the A and PTR RRs for the DDNS update method named ddns-2:

```
hostname(config)# ddns update method ddns-2
hostname(DDNS-update-method)# ddns both
```

Related Commands

Command	Description
ddns update (interface config mode)	Associates a dynamic DNS (DDNS) update method with a security appliance interface or a DDNS update hostname.
ddns update method (global config mode)	Creates a method for dynamically updating DNS resource records.
dhcp-client update dns	Configures the update parameters that the DHCP client passes to the DHCP server.
dhcpd update dns	Enables a DHCP server to perform DDNS updates.
interval maximum	Configures the maximum interval between update attempts by a DDNS update method.

ddns update (interface configuration)

To associate a dynamic DNS (DDNS) update method with a security appliance interface or an update hostname, use the **ddns update** command in interface configuration mode. To remove the association between the DDNS update method and the interface or the hostname from the running configuration, use the **no** form of this command.

ddns update [*method-name* | **hostname** *hostname*]

no ddns update [*method-name* | **hostname** *hostname*]

Syntax Description

hostname	Specifies that the next term in the command string is a hostname.
hostname	Specifies a hostname to be used for updates.
method-name	Specifies a method name for association with the interface being configured.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

After defining a DDNS update method, you must associate it with a security appliance interface to trigger DDNS updates.

A hostname could be a Fully Qualified Domain Name (FQDN) or just a hostname. If just a hostname, the security appliance appends a domain name to the hostname to create a FQDN.

Examples

The following example associates the interface GigabitEthernet0/2 with the DDNS update method named ddns-2 and the hostname hostname1.example.com:

```
hostname(config)# interface GigabitEthernet0/2
hostname(config-if)# ddns update ddns-2
hostname(config-if)# ddns update hostname hostname1.example.com
```

Related Commands

Command	Description
ddns (DDNS-update-method mode)	Specifies a DDNS update method type for a created DDNS method.
ddns update method (global config mode)	Creates a method for dynamically updating DNS resource records.
dhcp-client update dns	Configures the update parameters that the DHCP client passes to the DHCP server.
dhcpd update dns	Enables a DHCP server to perform DDNS updates.
interval maximum	Configures the maximum interval between update attempts by a DDNS update method.

ddns update method (global configuration mode)

To create a method for dynamically updating a DNS resource records (RRs), use the **ddns update method** command in global configuration mode. To remove a dynamic DNS (DDNS) update method from the running configuration, use the **no** form of this command.

ddns update method *name*

no ddns update method *name*

Syntax Description

name	Specifies the name of a method for dynamically updating DNS records.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

DDNS updates the name to address and address to name mappings maintained by DNS. The update method configured by the **ddns update method** command determines what and how often dynamic DNS updates are performed. Of the two methods for performing DDNS updates—the IETF standard defined by RFC 2136 and a generic HTTP method—the security appliance supports the IETF method in this release.

Name and address mappings are contained in two types of resource records (RR):

- The A resource record contains domain name to IP address mappings.
- The PTR resource record contains IP address to domain name mappings.

DDNS updates can be used to maintain consistent information between the A and PTR RR types.

Note Before **ddns update method** will work, you must configure a reachable default DNS server using the **dns** command with domain lookup enabled on the interface.

Examples

The following example configures the DDNS update method named ddns-2:

```
hostname(config)# ddns update method ddns-2
```

Related Commands

Command	Description
ddns (DDNS-update-method mode)	Specifies a DDNS update method type for a created DDNS method.
ddns update (interface config mode)	Associates a dynamic DNS (DDNS) update method with a security appliance interface or a DDNS update hostname.
dhcp-client update dns	Configures the update parameters that the DHCP client passes to the DHCP server.
dhcpd update dns	Enables a DHCP server to perform dynamic DNS updates.
interval maximum	Configures the maximum interval between update attempts by a DDNS update method.

debug aaa

To show debug messages for AAA, use the **debug aaa** command in privileged EXEC mode. To stop showing AAA messages, use the **no** form of this command.

debug aaa [**accounting** | **authentication** | **authorization** | **common** | **internal** | **vpn** [*level*]]

no debug aaa

Syntax Description

accounting	(Optional) Show debug messages for accounting only.
authentication	(Optional) Show debug messages for authentication only.
authorization	(Optional) Show debug messages for authorization only.
common	(Optional) Show debug messages for different states within the AAA feature.
internal	(Optional) Show debug messages for AAA functions supported by the local database only.
level	(Optional) Specifies the debug level. Valid with the **vpn** keyword only.
vpn	(Optional) Show debug messages for VPN-related AAA functions only.

Defaults

The default *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was modified to include new keywords.

Usage Guidelines

The **debug aaa** command displays detailed information about AAA activity. The **no debug all** or **undebug all** commands turn off all enabled debugs.

Examples

The following example enables debugging for AAA functions supported by the local database:

```
hostname(config)# debug aaa internal
debug aaa internal enabled at level 1
hostname(config)# uap allocated. remote address: 10.42.15.172, Session_id: 2147483841
uap freed for user . remote address: 10.42.15.172, session id: 2147483841
```

Related Commands

Command	Description
show running-config aaa	Displays running configuration related to AAA.

debug appfw

To display detailed information about application inspection, use the **debug appfw** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

debug appfw [**chunk** | **event** | **eventverb** | **regex**]

no debug appfw [**chunk** | **event** | **eventverb** | **regex**]

Syntax Description

chunk	(Optional) Displays runtime information about processing of chunked transfer encoded packets.
event	(Optional) Displays debug information about packet inspection events.
eventverb	(Optional) Displays the action taken by the security appliance in response to an event.
regex	(Optional) Displays information about matching patterns with predefined signatures.

Defaults

All options are enabled by default.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	•	—

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

The **debug appfw** command displays detailed information about HTTP application inspection. The **no debug all** or **undebug all** commands turn off all enabled debug commands.

Examples

The following example enables the display of detailed information about application inspection:

```
hostname# debug appfw
```

Related Commands

Commands	Description
http-map	Defines an HTTP map for configuring enhanced HTTP inspection.
inspect http	Applies a specific HTTP map to use for application inspection.

debug arp

To show debug messages for ARP, use the **debug arp** command in privileged EXEC mode. To stop showing debug messages for ARP, use the **no** form of this command.

debug arp

no debug arp

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for ARP:

```
hostname# debug arp
```

Related Commands

Command	Description
arp	Adds a static ARP entry.
show arp statistics	Shows ARP statistics.
show debug	Shows all enabled debuggers.

debug arp-inspection

To show debug messages for ARP inspection, use the **debug arp-inspection** command in privileged EXEC mode. To stop showing debug messages for ARP inspection, use the **no** form of this command.

debug arp-inspection

no debug arp-inspection

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for ARP inspection:

```
hostname# debug arp-inspection
```

Related Commands

Command	Description
arp	Adds a static ARP entry.
arp-inspection	For transparent firewall mode, inspects ARP packets to prevent ARP spoofing.
show debug	Shows all enabled debuggers.

debug asdm history

To view debug information for ASDM, use the **debug asdm history** command in privileged EXEC mode.

debug asdm history *level*

Syntax Description

level	(Optional) Specifies the debug level.

Defaults

The default *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was changed from the **debug pdm history** command to the **debug asdm history** command.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables level 1 debugging of ASDM:

```
hostname# debug asdm history
debug asdm history enabled at level 1

hostname#
```

Related Commands

Command	Description
show asdm history	Displays the contents of the ASDM history buffer.

debug auto-update

To display auto-update client and server debugging information, use the **debug auto-update** command in privileged EXEC mode. To disable the display of auto-update client and server debugging information, use the **no** form of this command.

debug auto-update client | server [*level*]

no debug auto-update client | server [*level*]

Syntax Description

client	The auto-update client.
level	(Optional) Sets the debugging message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set this parameter to a higher number.
server	The Auto Update server.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	—	•

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables auto-update client and server debugging messages. The **show debug auto-update** command indicates that auto-update client and server debugging messages are enabled.

```
hostname# debug auto-update client
hostname# debug auto-update server
hostname# show debug auto-update
debug auto-update client enabled at level 1
debug auto-update server enabled at level 1
```

Related Commands

Command	Description
show debug auto	Displays the current auto-update debugging configuration.

debug boot-mem

To display boot memory debugging information, use the **debug boot-mem** command in privileged EXEC mode. To disable the display of debugging information, use the **no** form of this command.

debug boot-mem [*level*]

no debug boot-mem [*level*]

Syntax Description

level	(Optional) Sets the debugging message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set this parameter to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables boot memory debugging messages. The **show debug boot-mem** command indicates that boot memory debugging messages are enabled.

```
hostname# debug boot-mem
debug boot-mem enabled at level 1
hostname# show debug boot-mem
debug boot-mem enabled at level 1
```

Related Commands

Command	Description
show debug boot	Displays the current boot memory debugging configuration.

debug context

To show debug messages when you add or delete a security context, use the **debug context** command in privileged EXEC mode. To stop showing debug messages for contexts, use the **no** form of this command.

debug context [*level*]

no debug context [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default level is 1.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for context management:

```
hostname# debug context
```

Related Commands

Command	Description
context	Creates a security context in the system configuration and enters context configuration mode.
show context	Shows context information.
show debug	Shows all enabled debuggers.

debug boot-mem

To display boot memory debugging information, use the **debug boot-mem** command in privileged EXEC mode. To disable the display of debugging information, use the **no** form of this command.

debug boot-mem [*level*]

no debug boot-mem [*level*]

Syntax Description

level	(Optional) Sets the debugging message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set this parameter to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables boot memory debugging messages. The **show debug boot-mem** command indicates that boot memory debugging messages are enabled.

```
hostname# debug boot-mem
debug boot-mem enabled at level 1
hostname# show debug boot-mem
debug boot-mem enabled at level 1
```

Related Commands

Command	Description
show debug boot	Displays the current boot memory debugging configuration.

debug cplane

To show debug messages about the control plane that connects internally to an SSM, use the **debug cplane** command in privileged EXEC mode. To stop showing debug messages for the control plane, use the **no** form of this command.

debug cplane [*level*]

no debug cplane [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default level is 1.

Command Modes

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

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

Command History

Release	Modification
7.0	This command was introduced.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for the control plane:

```
hostname# debug cplane
```

Related Commands

Command	Description
hw-module module recover	Recovers an intelligent SSM by loading a recovery image from a TFTP server.
hw-module module reset	Shuts down an SSM and performs a hardware reset.
hw-module module reload	Reloads the intelligent SSM software.

Command	Description
hw-module module shutdown	Shuts down the SSM software in preparation for being powered off without losing configuration data.
show module	Shows SSM information.

debug crypto ca

To show debug messages for PKI activity (used with CAs), use the **debug crypto ca** command in privileged EXEC mode. To stop showing debug messages for PKI, use the **no** form of this command.

debug crypto ca [**messages** | **transactions**] [*level*]

no debug crypto ca [**messages** | **transactions**] [*level*]

Syntax Description

messages	(Optional) Shows only debug messages for PKI input and output messages.
transactions	(Optional) Shows only debug messages for PKI transactions.
level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number. Level 1 (the default) shows messages only when errors occur. Level 2 shows warnings. Level 3 shows informational messages. Levels 4 and up show additional information for troubleshooting.

Defaults

By default, this command shows all debug messages. The default level is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	•	—

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for PKI:

```
hostname# debug crypto ca
```

Related Commands

Command	Description
debug crypto engine	Shows debug messages for the crypto engine.
debug crypto ipsec	Shows debug messages for IPSec.
debug crypto isakmp	Shows debug messages for ISAKMP.

debug crypto ca server

To set the local CA server debug message level and begin listing associated debug messages, use the **debug crypto ca server** command in ca server configuration mode. To stop listing all debug messages, use the **no** form of the command.

debug crypto ca server [*level*]

no debug crypto ca server [*level*]

Syntax Description

level	Sets the debug message level to display, the range of values is between 1 and 255.

Defaults

The default debug level is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
CA server configuration	•	—	•	—	—
Global configuration	•	—	•	—	—
Privileged EXEC	•	—	•	—	—

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks. Levels 5 and higher are reserved for raw data dumps and should be avoided during normal debugging because of excessive debug output.

Examples

The following example sets the debug level to 3:

```
hostname(config-ca-server)# debug crypto ca server 3
hostname(config-ca-server)#
```

The following example turns off all debugging:

```
hostname(config-ca-server)# no debug crypto ca server
hostname(config-ca-server)#
```

Related Commands

Command	Description
cdp-url	Specifies the certificate revocation list (CRL) distribution point (CDP) to be include in the certificates issued by the CA.
crypto ca server	Provides access to CA Server Configuration mode CLI command set, which allows you to configure and manage the local CA.
database path	Specifies a path or location for the local CA server database.
show crypto ca server	Displays the characteristics of the certificate authority configuration on the security appliance in ASCII text format.
show crypto ca server certificate	Displays the local CA configuration in base64 format.
show crypto ca server crl	Displays the current CRL of the local CA.

debug crypto engine

To show debug messages for the crypto engine, use the **debug crypto engine** command in privileged EXEC mode. To stop showing debug messages for the crypto engine, use the **no** form of this command.

debug crypto engine [*level*]

no debug crypto engine [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default level is 1.

Command Modes

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

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

Command History

Release	Modification
7.0	This command was introduced.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for the crypto engine:

```
hostname# debug crypto engine
```

Related Commands

Command	Description
debug crypto ca	Shows debug messages for the CA.
debug crypto ipsec	Shows debug messages for IPSec.
debug crypto isakmp	Shows debug messages for ISAKMP.

debug crypto ipsec

To show debug messages for IPSec, use the **debug crypto ipsec** command in privileged EXEC mode. To stop showing debug messages for IPSec, use the **no** form of this command.

debug crypto ipsec [*level*]

no debug crypto ipsec [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default level is 1.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for IPSec:

```
hostname# debug crypto ipsec
```

Related Commands

Command	Description
debug crypto ca	Shows debug messages for the CA.
debug crypto engine	Shows debug messages for the crypto engine.
debug crypto isakmp	Shows debug messages for ISAKMP.

debug crypto isakmp

To show debug messages for ISAKMP, use the **debug crypto isakmp** command in privileged EXEC mode. To stop showing debug messages for ISAKMP, use the **no** form of this command.

debug crypto isakmp [**timers**] [*level*]

no debug crypto isakmp [**timers**] [*level*]

Syntax Description

timers	(Optional) Shows debug messages for ISAKMP timer expiration.
level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number. Level 1 (the default) shows messages only when errors occur. Levels 2 through 7 show additional information. Level 254 shows decrypted ISAKMP packets in a human readable format. Level 255 shows hexadecimal dumps of decrypted ISAKMP packets.

Defaults

The default level is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	—	—

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for ISAKMP:

```
hostname# debug crypto isakmp
```

Related Commands

Command	Description
debug crypto ca	Shows debug messages for the CA.
debug crypto engine	Shows debug messages for the crypto engine.
debug crypto ipsec	Shows debug messages for IPSec.

debug ctiqbe

To show debug messages for CTIQBE application inspection, use the **debug ctiqbe** command in privileged EXEC mode. To stop showing debug messages for CTIQBE application inspection, use the **no** form of this command.

debug ctiqbe [*level*]

no debug ctiqbe [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

To see the current debug command settings, enter the **show debug** command. To stop the debug output, enter the **no debug** command. To stop all debug messages from being displayed, enter the **no debug all** command.

Note Enabling the **debug ctiqbe** command may slow down traffic on busy networks.

Examples

The following example enables debug messages at the default level (1) for CTIQBE application inspection:

```
hostname# debug ctiqbe
```

Related Commands

Command	Description
inspect ctiqbe	Enables CTIQBE application inspection.
show ctiqbe	Displays information about CTIQBE sessions established through the security appliance.
show conn	Displays the connection state for different connection types.
timeout	Sets the maximum idle time duration for different protocols and session types.

debug ctl-provider

To show debug messages for Certificate Trust List providers, use the **debug ctl-provider** command in privileged EXEC mode. To stop showing debug messages, use the **no** form of this command.

debug ctl-provider [**errors** | **events** | **parser**]

no debug ctl-provider [**errors** | **events** | **parser**]

Syntax Description

errors	Specifies CTL provider error debugging.
events	Specifies CTL provider event debugging.
parser	Specifies CTL provider parser debugging.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	•	—

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for CTL provider:

```
hostname# debug ctl-provider
```

Related Commands

Command	Description
ctl	Parses the CTL file from the CTL client and install trustpoints.
ctl-provider	Configures a CTL provider instance in CTL provider mode.
export	Specifies the certificate to be exported to the client
service	Specifies the port to which the CTL provider listens.

debug dap

To enable logging of Dynamic Access Policy events, use the **debug dap** command in privileged EXEC mode. To disable the logging of DAP debug messages, use the **no** form of this command.

debug dap {errors | trace}

no debug dap [errors | trace]

Syntax Description

errors	Specifies DAP processing errors.
trace	Specifies a DAP function trace.

Defaults

No default value or behaviors.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	—	—

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

The high priority assigned to debugging output can render the system unusable. For this reason, use debug commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use debug commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased debug command processing overhead will affect system use.

Examples

The following example shows how to enable DAP trace debugging:

```
hostname # debug dap trace
hostname #
```

Related Commands

Command	Description
dynamic-access-policy-record	Creates a DAP record.

debug ddns

To show debug messages for DDNS, use the **debug ddns** command in privileged EXEC mode. To disable debug messages, use the **no** form of this command.

debug ddns

no debug ddns

Syntax Description

This command has no arguments or keywords.

Defaults

The default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

The **debug ddns** command displays detailed information about DDNS. The **undebug ddns** turns off DDNS debugging information as does the **no debug ddns** command.

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example shows an example of enabling DDNS debug messages:

```
hostname# debug ddns
debug ddns enabled at level 1
```

Related Commands

Command	Description
ddns (DDNS-update-method mode)	Specifies a DDNS update method type for a created DDNS method.
ddns update (interface config mode)	Associates a DDNS update method with a security appliance interface or a DDNS update hostname.
ddns update method (global config mode)	Creates a method for dynamically updating DNS resource records.
show running-config ddns	Displays the type and interval of all configured DDNS methods in the running configuration.

debug dhcpc

To enable debugging of the DHCP client, use the **debug dhcpc** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

debug dhcpc {**detail** | **packet** | **error**} [*level*]

no debug dhcpc {**detail** | **packet** | **error**} [*level*]

Syntax Description

detail	Displays detail event information that is associated with the DHCP client.
error	Displays error messages that are associated with the DHCP client.
level	(Optional) Specifies the debug level. Valid valuse range from 1 to 255.
packet	Displays packet information that is associated with the DHCP client.

Defaults

The default debug level is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	—	•	•	—

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Displays DHCP client debug information.

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example shows how to enable debugging for the DHCP client:

```
hostname# debug dhcpc detail 5
debug dhcpc detail enabled at level 5
```

Related Commands

Command	Description
show ip address dhcp	Displays detailed information about the DHCP lease for an interface.
show running-config interface	Displays the running configuration of the specified interface.

debug dhcpd

To enable debugging of the DHCP server, use the **debug dhcpd** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

debug dhcpd {**event** | **packet**} [*level*]

no debug dhcpd {**event** | **packet**} [*level*]

Syntax Description

event	Displays event information that is associated with the DHCP server.
level	(Optional) Specifies the debug level. Valid valuse range from 1 to 255.
packet	Displays packet information that is associated with the DHCP server.

Defaults

The default debug level is 1.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

The **debug dhcpd event** command displays event information about the DHCP server. The **debug dhcpd packet** command displays packet information about the DHCP server.

Use the **no** form of the d**ebug dhcpd** commands to disable debugging.

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example shows an example of enabling DHCP event debugging:

```
hostname# debug dhcpd event
debug dhcpd event enabled at level 1
```

Related Commands

Command	Description
show dhcpd	Displays DHCP binding, statistic, or state information.
show running-config dhcpd	Displays the current DHCP server configuration.

debug dhcpd ddns

To enable debugging of the DHCP DDNS, use the **debug dhcpd ddns** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

debug dhcpd ddns [*level*]

no debug dhcpd ddns [*level*]

Syntax Description

level	(Optional) Specifies the debug level. Valid values range from 1 to 255.

Defaults

The default debug level is 1.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

The **debug dhcpd ddns** command displays detailed information about DHCP and DDNS. The **undebug dhcpd ddns** command turns off DHCP and DDNS debugging information as does the **no debug dhcpd ddns** command.

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example shows DHCP DDNS debugging being enabled:

```
hostname# debug dhcpd ddns
debug dhcpd ddns enabled at level 1
```

Related Commands

Command	Description
dhcpd update dns	Enables a DHCP server to perform dynamic DNS updates.
show running-config dhcpd	Displays the current DHCP server configuration.
show running-config ddns	Display the DDNS update methods of the running configuration.

debug dhcprelay

To enable debugging of the DHCP relay server, use the **debug dhcpreleay** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

debug dhcprelay {**event** | **packet** | **error**} [*level*]

no debug dhcprelay {**event** | **packet** | **error**} [*level*]

Syntax Description

error	Displays error messages that are associated with the DHCP relay agent.
event	Displays event information that is associated with the DHCP relay agent.
level	(Optional) Specifies the debug level. Valid valuse range from 1 to 255.
packet	Displays packet information that is associated with the DHCP relay agent.

Defaults

The default debug level is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	—	•	•	—

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example shows how to enable debugging for DHCP relay agent error messages:

```
hostname# debug dhcprelay error
debug dhcprelay error enabled at level 1
```

Related Commands

Command	Description
clear configure dhcprelay	Removes all DHCP relay agent settings.
clear dhcprelay statistics	Clears the DHCP relay agent statistic counters.
show dhcprelay statistics	Displays DHCP relay agent statistic information.
show running-config dhcprelay	Displays the current DHCP relay agent configuration.

debug disk

To display file system debugging information, use the **debug disk** command in privileged EXEC mode. To disable the display of debugging information, use the **no** form of this command.

debug disk {**file** | **file-verbose** | **filesystem**} [*level*]

no debug disk {**file** | **file-verbose** | **filesystem**}

Syntax Description

file	Enables file-level disk debug messages.
file-verbose	Enables verbose file-level disk debug messages
filesystem	Enables file system debug messages.
level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set this parameter to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	—	•

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables file-level disk debugging messages. The **show debug** command reveals that file-level disk debugging messages are enabled. The **dir** command causes several debugging messages.

```
hostname# debug disk file
debug disk file enabled at level 1
hostname# show debug
debug vpn-sessiondb  enabled at level 1
```

```
hostname# dir
IFS: Opening: file flash:/, flags 1, mode 0
IFS: Opened: file flash:/ as fd 3
IFS: Getdent: fd 3
IFS: Getdent: fd 3
IFS: Getdent: fd 3
IFS: Getdent: fd 3

Directory of flash:/
IFS: Close: fd 3
IFS: Opening: file flash:/, flags 1, mode 0

4      -rw-  5124096     14:42:27 Apr 04 2005  cdisk.binIFS: Opened: file flash:/ as fd 3

9      -rw-  5919340     14:53:39 Apr 04 2005  ASDMIFS: Getdent: fd 3

11     drw-  0           15:18:56 Apr 21 2005  syslog
IFS: Getdent: fd 3
IFS: Getdent: fd 3
IFS: Getdent: fd 3
IFS: Close: fd 3

16128000 bytes total (5047296 bytes free)
```

Related Commands

Command	Description
show debug	Displays current debugging configuration.

debug dns

To show debug messages for DNS, use the **debug dns** command in privileged EXEC mode. To stop showing debug messages for DNS, use the **no** form of this command.

debug dns [**resolver** | **all**] [*level*]

no debug dns [**resolver** | **all**] [*level*]

Syntax Description

all	(Default) Shows all messages, including messages about the DNS cache.
level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.
resolver	(Optional) Shows only DNS resolver messages.

Defaults

The default level is 1. If you do not specify any keywords, the security appliance shows all mesages.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for DNS:

```
hostname# debug dns
```

Related Commands

Command	Description
class-map	Defines the traffic class to which to apply security actions.
inspect dns	Enables DNS application inspection.
policy-map	Associates a class map with specific security actions.
service-policy	Applies a policy map to one or more interfaces.

debug eap

To enable logging of EAP events to debug NAC messaging, use the **debug eap** command in privileged EXEC mode. To disable the logging of EAP debug messages, use the **no** form of this command.

debug eap {**all | errors | events | packets | sm**}

no debug eap [**all | errors | events | packets | sm**]

Syntax Description

all	Enables logging of debug messages about all EAP information.
errors	Enables logging of EAP packet errors.
events	Enables logging of EAP session events.
packets	Enables logging of debug messages about EAP packet information.
sm	Enables logging of debug messages about EAP state machine information.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

When you use this command, the security appliance records EAP session state changes and EAP status query events, and generates a complete record of EAP and packet contents in hexadecimal format.

The high priority assigned to debugging output can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables the logging of all EAP session events:

```
hostname# debug eap events
hostname#
```

The following example enables the logging of all EAP debug messages:

```
hostname# debug eap all
hostname#
```

The following example disables the logging of all EAP debug messages:

```
hostname# no debug eap
hostname#
```

Related Commands

Command	Description
debug eou	Enables logging of EAPoUDP events to debug NAC messaging.
debug nac	Enables logging of NAC events.
eou initialize	Clears the resources assigned to one or more NAC sessions and initiates a new, unconditional posture validation for each of the sessions.
eou revalidate	Forces immediate posture revalidation of one or more NAC sessions.
show debug	Displays current debug configuration.

debug eigrp fsm

To display debug information the DUAL finite state machine, use the **debug eigrp fsm** command in privileged EXEC mode. To disable the debug information display, use the **no** form of this command.

debug eigrp fsm

no debug eigrp fsm

Syntax Description

This command has no arguments or keywords.

Defaults

No default behaviors or values.

Command Modes

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

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

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

This command lets you observe EIGRP feasible successor activity and to determine whether route updates are being installed and deleted by the routing process.

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output from the **debug eigrp fsm** command:

```
hostname# debug eigrp fsm

DUAL: dual_rcvupdate(): 172.25.166.0 255.255.255.0 via 0.0.0.0 metric 750080/0
DUAL: Find FS for dest 172.25.166.0 255.255.255.0. FD is 4294967295, RD is 4294967295
found
DUAL: RT installed 172.25.166.0 255.255.255.0 via 0.0.0.0
DUAL: dual_rcvupdate(): 192.168.4.0 255.255.255.0 via 0.0.0.0 metric 4294967295/4294967295
DUAL: Find FS for dest 192.168.4.0 255.255.255.0. FD is 2249216, RD is 2249216
DUAL:   0.0.0.0 metric 4294967295/4294967295not found Dmin is 4294967295
DUAL: Dest 192.168.4.0 255.255.255.0 not entering active state.
DUAL: Removing dest 192.168.4.0 255.255.255.0, nexthop 0.0.0.0
```

```
DUAL: No routes. Flushing dest 192.168.4.0 255.255.255.0
```

In the fist line, DUAL stands for diffusing update algorithm. It is the basic mechanism within EIGRP that makes the routing decisions. The next three fields are the Internet address and mask of the destination network and the address through which the update was received. The metric field shows the metric stored in the routing table and the metric advertised by the neighbor sending the information. If shown, the term "Metric... inaccessible" usually means that the neighbor router no longer has a route to the destination, or the destination is in a hold-down state.

In the following output, EIGRP is attempting to find a feasible successor for the destination. Feasible successors are part of the DUAL loop avoidance methods. The FD field contains more loop avoidance state information. The RD field is the reported distance, which is the metric used in update, query, or reply packets.

The indented line with the "not found" message means a feasible successor was not found for 192.168.4.0 and EIGRP must start a diffusing computation. This means it begins to actively probe (sends query packets about destination 192.168.4.0) the network looking for alternate paths to 192.164.4.0.

```
DUAL: Find FS for dest 192.168.4.0 255.255.255.0. FD is 2249216, RD is 2249216
DUAL:   0.0.0.0 metric 4294967295/4294967295not found Dmin is 4294967295
```

The following output indicates the route DUAL successfully installed into the routing table:

```
DUAL: RT installed 172.25.166.0 255.255.255.0 via 0.0.0.0
```

The following output shows that no routes to the destination were discovered and that the route information is being removed from the topology table:

```
DUAL: Dest 192.168.4.0 255.255.255.0 not entering active state.
DUAL: Removing dest 192.168.4.0 255.255.255.0, nexthop 0.0.0.0
DUAL: No routes. Flushing dest 192.168.4.0 255.255.255.0
```

Related Commands

Command	Description
show eigrp topology	Displays the EIGRP topology table.

debug eigrp neighbors

To display debug information for neighbors discovered by EIGRP, use the **debug eigrp neighbors** command in privileged EXEC mode. To disable the debug information display, use the **no** form of this command.

debug eigrp neighbors [**siatimer** | **static**]

no debug eigrp neighbors [**siatimer** | **static**]

Syntax Description

siatimer	(Optional) Displays EIGRP stuck in active messages.
static	(Optional) Displays EIGRP static neighbor messages.

Defaults

No default behaviors or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	—	•	—	—

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output from the **debug eigrp neighbors static** command. The example shows a static neighbor being added, and then removed, and the corresponding debug messages.

```
hostname# debug eigrp neighbors static

EIGRP Static Neighbors debugging is on

hostname# configure terminal
hostname(config) router eigrp 100
hostname(config-router)# neighbor 10.86.194.3 interface outside
hostname(config-router)#

EIGRP: Multicast Hello is disabled on Ethernet0/0!
```

```
EIGRP: Add new static nbr 10.86.194.3 to AS 100 Ethernet0/0

hostname(config-router)# no neighbor 10.86.194.3 interface outside
hostname(config-router)#

EIGRP: Static nbr 10.86.194.3 not in AS 100 Ethernet0/0 dynamic list
EIGRP: Delete static nbr 10.86.194.3 from AS 100 Ethernet0/0
EIGRP: Multicast Hello is enabled on Ethernet0/0!

hostname(config-router)# no debug eigrp neighbors static

EIGRP Static Neighbors debugging is off
```

Related Commands

Command	Description
neighbor	Defines an EIGRP neighbor.
show eigrp neighbors	Displays the EIGRP neighbor table.

debug eigrp packets

To display debug information for EIGRP packets, use the **debug eigrp packets** command in privileged EXEC mode. To disable the debug information display, use the **no** form of this command.

debug eigrp packets [**SIAquery | SIAreply | ack | hello | probe | query | reply | request | retry | stub | terse | update | verbose**]

no debug eigrp packets [**SIAquery | SIAreply | ack | hello | probe | query | reply | request | retry | stub | terse | update | verbose**]

Syntax Description

ack	(Optional) Limits the debug output to EIGRP ack packets.
hello	(Optional) Limits the debug output to EIGRP hello packets.
probe	(Optional) Limits the debug output to EIGRP probe packets.
query	(Optional) Limits the debug output to EIGRP query packets.
reply	(Optional) Limits the debug output to EIGRP reply packets.
request	(Optional) Limits the debug output to EIGRP request packets.
retry	(Optional) Limits the debug output to EIGRP retry packets.
SIAquery	(Optional) Limits the debug output to EIGRP stuck in active query packets.
SIAreply	(Optional) Limits the debug output to EIGRP stuck in active reply packets.
stub	(Optional) Limits the debug output to EIGRP stub routing packets.
terse	(Optional) Displays all EIGRP packets except hello packets.
update	(Optional) Limits the debug output to EIGRP update packets.
verbose	(Optional) Outputs all packet debug messages.

Defaults

No default behaviors or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	—	•	—	—

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

You can specify more than one packet type in a single command, for example:

```
debug eigrp packets query reply
```

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output from the **debug eigrp packets** command:

```
hostname# debug eigrp packets

EIGRP: Sending HELLO on Ethernet0/1
       AS 109, Flags 0x0, Seq 0, Ack 0
EIGRP: Sending HELLO on Ethernet0/1
       AS 109, Flags 0x0, Seq 0, Ack 0
EIGRP: Sending HELLO on Ethernet0/1
       AS 109, Flags 0x0, Seq 0, Ack 0
EIGRP: Received UPDATE on Ethernet0/1 from 192.195.78.24,
       AS 109, Flags 0x1, Seq 1, Ack 0
EIGRP: Sending HELLO/ACK on Ethernet0/1 to 192.195.78.24,
       AS 109, Flags 0x0, Seq 0, Ack 1
EIGRP: Sending HELLO/ACK on Ethernet0/1 to 192.195.78.24,
       AS 109, Flags 0x0, Seq 0, Ack 1
EIGRP: Received UPDATE on Ethernet0/1 from 192.195.78.24,
       AS 109, Flags 0x0, Seq 2, Ack 0
```

The output shows transmission and receipt of EIGRP packets. The sequence and acknowledgment numbers used by the EIGRP reliable transport algorithm are shown in the output. Where applicable, the network-layer address of the neighboring router is also included.

Related Commands

Command	Description
show eigrp traffic	Displays the number of EIGRP packets sent and received.

debug eigrp transmit

To display transmittal messages sent by EIGRP, use the **debug eigrp transmit** command in privileged EXEC mode. To disable the debug information display, use the **no** form of this command.

debug eigrp transmit [**ack**] [**build**] [**detail**] [**link**] [**packetize**] [**peerdown**] [**sia**] [**startup**] [**strange**]

no debug eigrp transmit [**ack**] [**build**] [**detail**] [**link**] [**packetize**] [**peerdown**] [**sia**] [**startup**] [**strange**]

Syntax Description

ack	(Optional) Information for acknowledgment (ACK) messages sent by the system.
build	(Optional) Build information messages (messages that indicate that a topology table was either successfully built or could not be built).
detail	(Optional) Additional detail for debug output.
link	(Optional) Information regarding topology table linked-list management.
packetize	(Optional) Information regarding packetize events.
peerdown	(Optional) Information regarding the impact on packet generation when a peer is down.
sia	(Optional) Stuck-in-active messages.
startup	(Optional) Information regarding peer startup and initialization packets that have been transmitted.
strange	(Optional) Unusual events relating to packet processing.

Defaults

If at least one transmittal event is not specified, all transmittal events are shown in the debug output.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	—	•	—	—

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

You can specify more than one transmittal event in a single command, For example:

```
hostname# debug eigrp ack build link
```

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output from the **debug eigrp transmit** command. The example shows a **network** command being entered and the transmittal event debug message that is generated.

```
hostname# debug eigrp transmit

EIGRP Transmission Events debugging is on

    (ACK, PACKETIZE, STARTUP, PEERDOWN, LINK, BUILD, STRANGE, SIA, DETAIL)

hostname# configure terminal
hostname(config)# router eigrp 100
hostname(config-router)# network 10.86.194.0 255.255.255.0

DNDB UPDATE 10.86.194.0 255.255.255.0, serno 0 to 1, refcount 0

hostname(config-router)# no debug eigrp transmit

EIGRP Transmission Events debugging is off
```

Related Commands

Command	Description
show eigrp traffic	Displays the number of EIGRP packets sent and received.

debug eigrp user-interface

To display debug information for EIGRP user events, use the **debug eigrp user-interface** command in privileged EXEC mode. To disable the debug information display, use the **no** form of this command.

debug eigrp user-interface

no debug eigrp user-interface

Syntax Description

This command has no arguments or keywords.

Defaults

No default behaviors or values.

Command Modes

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

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

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output from the **debug eigrp user-interface** command. The output is caused by an administrator removing a **passive-interface** command from an EIGRP configuration.

```
hostname# debug eigrp user-interface

EIGRP UI Events debugging is on

hostname# configure terminal
hostname(config) router eigrp 100
hostname(config-router)# no passive-interface inside

CSB2AF: FOUND (AS=100, Name=, VRF=0, AFI=ipv4)

hostname(config-router)# no debug eigrp user-interface

EIGRP UI Events debugging is off
```

Related Commands

Command	Description
router eigrp	Enables an EIGRP routing process and enters router configuration mode.
show running-config eigrp	Displays the EIGRP commands in the running configuration.

debug email

To display e-mail debugging information, use the **debug email** command in privileged EXEC mode. To disable the display of e-mail debugging information, use the **no** form of this command.

debug email [*level*]

no debug email [*level*]

Syntax Description

level	(Optional) Sets the debugging message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables e-mail debugging messages. The **show debug email** command indicates that e-mail debugging messages are enabled.

```
hostname# debug email
debug email enabled at level 1
hostname# show debug email
debug email enabled at level 1
```

Related Commands

Command	Description
show debug	Displays the current debug configuration.

debug entity

To display MIB debug information, use the **debug entity** command in privileged EXEC mode. To disable the display of debug information, use the **no** form of this command.

debug entity [*level*]

no debug entity

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables MIB debug messages. The **show debug** command reveals that MIB debug messages are enabled.

```
hostname# debug entity
debug entity  enabled at level 1
hostname# show debug
debug entity  enabled at level 1
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug eou

To enable logging of EAPoUDP events to debug NAC messaging, use the **debug eou** command in privileged EXEC mode. To disable the logging of EAPoUDP debug messages, use the **no** form of this command.

debug eou {**all** | **eap** | **errors** | **events** | **packets** | **sm**}

no debug eou [**all** | **eap** | **errors** | **events** | **packets** | **sm**]

Syntax Description

all	Enables logging of debug messages about all EAPoUDP information.
eap	Enables logging of debug messages about EAPoUDP packets.
errors	Enables logging of EAPoUDP packet errors.
events	Enables logging of EAPoUDP session events.
packets	Enables logging of debug messages about EAPoUDP packet information.
sm	Enables logging of debug messages about EAPoUDP state machine information.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

When you use this command, the security appliance records EAPoUDP session state changes and timer events, and generates a complete record of EAPoUDP header and packet contents in hexadecimal format.

The high priority assigned to debugging output can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables the logging of all EAPoUDP session events:

```
hostname# debug eou events
```

```
hostname#
```

The following example enables the logging of all EAPoUDP debug messages:

```
hostname# debug eou all
hostname#
```

The following example disables the logging of all EAPoUDP debug messages:

```
hostname# no debug eou
hostname#
```

Related Commands

Command	Description
debug eap	Enables logging of EAP events to debug NAC messaging.
debug nac	Enables logging of NAC events.
eou initialize	Clears the resources assigned to one or more NAC sessions and initiates a new, unconditional posture validation for each of the sessions.
eou revalidate	Forces immediate posture revalidation of one or more NAC sessions.
show debug	Displays current debug configuration.

debug esmtp

To show debug messages for SMTP/ESMTP application inspection, use the **debug esmtp** command in privileged EXEC mode. To stop showing debug messages for SMTP/ESMTP application inspection, use the **no** form of this command.

debug esmtp [*level*]

no debug esmtp [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

To see the current debug command settings, enter the **show debug** command. To stop the debug output, enter the **no debug** command. To stop all debug messages from being displayed, enter the **no debug all** command.

Note Enabling the **debug esmtp** command may slow down traffic on busy networks.

Examples

The following example enables debug messages at the default level (1) for SMTP/ESMTP application inspection:

```
hostname# debug esmtp
```

Related Commands

Command	Description
class-map	Defines the traffic class to which to apply security actions.
inspect esmtp	Enables ESMTP application inspection.
policy-map	Associates a class map with specific security actions.
service-policy	Applies a policy map to one or more interfaces.
show conn	Displays the connection state for different connection types, including SMTP.

debug fixup

To display detailed information about application inspection, use the **debug fixup** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

debug fixup

no debug fixup

Defaults

All options are enabled by default.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

The **debug fixup** command displays detailed information about application inspection. The **no debug all** or **undebug all** commands turn off all enabled debug commands.

Examples

The following example enables the display of detailed information about application inspection:

```
hostname# debug fixup
```

Related Commands

Commands	Description
class-map	Defines the traffic class to which to apply security actions.
inspect *protocol*	Enables application inspection for specific protocols.
policy-map	Associates a class map with specific security actions.

debug fover

To display failover debug information, use the **debug fover** command in privileged EXEC mode. To disable the display of debug information, use the **no** form of this command.

debug fover {**cable** | **cmd-exec** | **fail** | **fmsg** | **ifc** | **open** | **rx** | **rxdmp** | **rxip** | **switch** | **sync** | **tx** | **txdmp** | **txip** | **verify**}

no debug fover {**cable** | **fail** | **fmsg** | **ifc** | **open** | **rx** | **rxdmp** | **rxip** | **switch** | **sync** | **tx** | **txdmp** | **txip** | **verify**}

Syntax Description

cable	Failover LAN status or serial cable status.
cmd-exec	**failover exec** command execution trace.
fail	Failover internal exception.
fmsg	Failover message.
ifc	Network interface status trace.
open	Failover device open.
rx	Failover message receive.
rxdmp	Failover receive message dump (serial console only).
rxip	IP network failover packet receive.
switch	Failover switching status.
sync	Failover configuration/command replication.
tx	Failover message transmit.
txdmp	Failover transmit message dump (serial console only).
txip	IP network failover packet transmit.
verify	Failover message verify.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was modified. It includes additional debug keywords.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output from the **debug fover cmd-exec** command. After debugging is enabled, a **failover exec** command is entered. The results of the **failover exec** command is shown after the debug output.

```
hostname(config)# debug fover cmd-exec

fover event trace on

hostname(config)# failover exec mate show running-config failover

ci/console: Sending cmd: show runn failovero to peer for execution, seq = 4
ci/console: frep_execv_cmd: replicating exec cmd: show runn failover...
fover_parse: Fover rexec response: seq=4, size=228, data="fail..."
ci/console: Fover rexec waiting at clock tick 2670960
fover_parse: Fover rexec ack: seq = 4, ret_val = 0
ci/console: Fover rexec conteinuer at clock tick: 2671040
ci/console: Fover exec succeeded, seq = 5

failover
failover lan interface failover GigabitEthernet0/3
failover polltime unit 1 holdtime 3
failover key *****
failover link failover GigabitEthernet0/3
failover interface ip failover 10.0.5.1 255.255.255.0 standby 10.0.5.2
ciscoasa(config)#
```

Related Commands

Command	Description
show failover	Displays information about the failover configuration and operational statistics.

debug fsm

To display FSM debug information, use the **debug fsm** command in privileged EXEC mode. To disable the display of debug information, use the **no** form of this command.

debug fsm [*level*]

no debug fsm

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables FSM debug messages. The **show debug** command reveals that FSM debug messages are enabled.

```
hostname# debug fsm
debug fsm  enabled at level 1
hostname# show debug
debug fsm  enabled at level 1
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug ftp client

To show debugging messages for FTP clients, use the **debug ftp client** command in privileged EXEC mode. To stop showing debugging messages for FTP clients, use the **no** form of this command.

debug ftp client [*level*]

no debug ftp client [*level*]

Syntax Description

level	(Optional) Sets the debugging message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set this parameter to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

To view the current debugging command settings, enter the **show debug** command. To stop the debugging output, enter the **no debug** command. To stop all debugging messages from being displayed, enter the **no debug all** command.

Note Enabling the **debug ftp client** command may slow down traffic on busy networks.

Examples

The following example enables debugging messages at the default level (1) for FTP clients:

```
hostname# debug ftp client
```

Related Commands

Command	Description
copy	Uploads or downloads image files or configuration files to or from an FTP server.
ftp mode passive	Configures the mode for FTP sessions.
show running-config ftp mode	Displays the FTP client configuration.

debug generic

To display miscellaneous debug information, use the **debug generic** command in privileged EXEC mode. To disable the display of miscellaneous debug information, use the **no** form of this command.

debug generic [*level*]

no debug generic

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables miscellaneous debug messages. The **show debug** command reveals that miscellaneous debug messages are enabled.

```
hostname# debug generic
debug generic  enabled at level 1
hostname# show debug
debug generic  enabled at level 1
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug gtp

To display detailed information about GTP inspection, use the **debug gtp** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

debug gtp {**error** | **event** | **ha** | **parser**}

no debug gtp {**error** | **event** | **ha** | **parser**}

Syntax Description

error	Displays debug information on errors encountered while processing the GTP message.
event	Displays debug information on GTP events.
ha option	Debugs information on GTP HA events.
parser	Displays debug information for parsing the GTP messages.

Defaults

All options are enabled by default.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

The **debug gtp** command displays detailed information about GTP inspection. The **no debug all** or **undebug all** commands turn off all enabled debug commands.

Note GTP inspection requires a special license.

Examples

The following example enables the display of detailed information about GTP inspection:

```
hostname# debug gtp
```

Related Commands

Commands	Description
clear service-policy inspect gtp	Clears global GTP statistics.
gtp-map	Defines a GTP map and enables GTP map configuration mode.
inspect gtp	Applies a GTP map to use for application inspection.
show service-policy inspect gtp	Displays the GTP configuration.
show running-config gtp-map	Shows the GTP maps that have been configured.

debug h323

To show debug messages for H.323, use the **debug h323** command in privileged EXEC mode. To stop showing debug messages for H.323, use the **no** form of this command.

debug h323 {h225 | h245 | ras} [asn | event]

no debug h323 {h225 | h245 | ras} [asn | event]

Syntax Description

h225	Specifies H.225 signaling.
h245	Specifies H.245 signaling.
ras	Specifies the registration, admission, and status protocol.
asn	(Optional) Displays the output of the decoded protocol data units (PDU)s.
event	(Optional) Displays the signaling events or turns on both traces.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	•	—

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

To see the current debug command settings, enter the **show debug** command. To stop the debug output, enter the **no debug** command. To stop all debug messages from being displayed, enter the **no debug all** command.

Note Enabling the **debug h323** command may slow down traffic on busy networks.

Examples

The following example enables debug messages at the default level (1) for H.225 signaling:

```
hostname# debug h323 h225
```

Related Commands

Command	Description
inspect h323	Enables H.323 application inspection.
show h225	Displays information for H.225 sessions established across the security appliance.
show h245	Displays information for H.245 sessions established across the security appliance by endpoints using slow start.
show h323-ras	Displays information for H.323 RAS sessions established across the security appliance.
timeout h225 | h323	Configures idle time after which an H.225 signalling connection or an H.323 control connection will be closed.

debug http

To display detailed information about HTTP traffic, use the **debug http** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

debug http [*level*]

no debug http [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The defafult for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
7.0	This command was introduced.

Usage Guidelines

The **debug http** command displays detailed information about HTTP traffic. The **no debug all** or **undebug all** commands turn off all enabled debug commands.

Examples

The following example enables the display of detailed information about HTTP traffic:

```
hostname# debug http
```

Related Commands

Commands	Description
http	Specifies hosts that can access the HTTP server internal to the security appliance.
http-proxy	Configures an HTTP proxy server.
http redirect	Redirects HTTP traffic to HTTPS.
http server enable	Enables the security appliance HTTP server.

debug http-map

To show debug messages for HTTP application inspection maps, use the **debug http-map** command in privileged EXEC mode. To stop showing debug messages for HTTP application inspection, use the **no** form of this command.

debug http-map

no debug http-map

Defaults

The default value for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

To see the current debug command settings, enter the **show debug** command. To stop the debug output, enter the **no debug** command. To stop all debug messages from being displayed, enter the **no debug all** command.

Note Enabling the **debug http-map** command may slow down traffic on busy networks.

Examples

The following example enables debug messages at the default level (1) for HTTP application inspection:

```
hostname# debug http-map
```

Related Commands

Command	Description
class-map	Defines the traffic class to which to apply security actions.
debug appfw	Displays detailed information about HTTP application inspection.
http-map	Defines an HTTP map for configuring enhanced HTTP inspection.
inspect http	Applies a specific HTTP map to use for application inspection.
policy-map	Associates a class map with specific security actions.

debug icmp

To display detailed information about ICMP inspection, use the **debug icmp** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

debug icmp trace [*level*]

no debug icmp trace [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.
trace	Displays debug information about ICMP trace activity.

Defaults

All options are enabled.

Command Modes

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

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

Command History

Release	Modification
7.0	This command was introduced.

Usage Guidelines

The **debug icmp** command displays detailed information about ICMP inspection. The **no debug all** or **undebug all** commands turn off all enabled debugs.

Examples

The following example enables the display of detailed information about ICMP inspection:

```
hostname# debug icmp
```

Related Commands

Commands	Description
clear configure icmp	Clears the ICMP configuration.
icmp	Configures access rules for ICMP traffic that terminates at a security appliance interface.
show conn	Displays the state of connections through the security appliance for different protocols and session types.

Commands	Description
show icmp	Displays ICMP configuration.
timeout icmp	Configures idle timeout for ICMP.

debug igmp

To display IGMP debug information, use the **debug igmp** command in privileged EXEC mode. To stop the display of debug information, use the **no** form of this command.

debug igmp [**group** *group_id* | **interface** *if_name*]

no debug igmp [**group** *group_id* | **interface** *if_name*]

Syntax Description

group *group_id*	Displays IGMP debug information for the specified group.
interface *if_name*	Display IGMP debug information for the specified interface.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output from the **debug igmp** command:

```
hostname#debug igmp

IGMP debugging is on
IGMP: Received v2 Query on outside from 192.168.3.2
IGMP: Send v2 general Query on dmz
IGMP: Received v2 Query on dmz from 192.168.4.1
IGMP: Send v2 general Query on outside
IGMP: Received v2 Query on outside from 192.168.3.1
IGMP: Send v2 general Query on inside
IGMP: Received v2 Query on inside from 192.168.1.1
IGMP: Received v2 Report on inside from 192.168.1.6 for 224.1.1.1
IGMP: Updating EXCLUDE group timer for 224.1.1.1
```

Related Commands

Command	Description
show igmp groups	Displays the multicast groups with receivers that are directly connected to the security appliance and that were learned through IGMP.
show igmp interface	Displays multicast information for an interface.

debug ils

To show debug messages for ILS, use the **debug ils** command in privileged EXEC mode. To stop showing debug messages for ILS, use the **no** form of this command.

debug ils [*level*]

no debug ils [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

To see the current debug command settings, enter the **show debug** command. To stop the debug output, enter the **no debug** command. To stop all debug messages from being displayed, enter the **no debug all** command.

Note Enabling the **debug ils** command may slow down traffic on busy networks.

Examples

The following example enables debug messages at the default level (1) for ILS application inspection:

```
hostname# debug ils
```

Related Commands

Command	Description
class-map	Defines the traffic class to which to apply security actions.
inspect ils	Enables ILS application inspection.

Command	Description
policy-map	Associates a class map with specific security actions.
service-policy	Applies a policy map to one or more interfaces.

Related Commands

debug imagemgr

To display Image Manager debugging information, use the **debug imagemgr** command in privileged EXEC mode. To disable the display of Image Manager debugging information, use the **no** form of this command.

debug imagemgr [*level*]

no debug imagemgr

Syntax Description

level	(Optional) Sets the debugging message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set this parameter to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables Image Manager debugging messages. The **show debug** command indicates that Image Manager debugging messages are enabled.

```
hostname# debug imagemgr
debug imagemgr  enabled at level 1
hostname# show debug
debug imagemgr  enabled at level 1
hostname#
```

Related Commands

Command	Description
show debug	Displays the current debugging configuration.

debug inspect tls-proxy

To show debug messages for TLS proxy inspection, use the **debug inspect tls-proxy** command in privileged EXEC mode. To stop showing debug messages, use the **no** form of this command.

debug inspect tls-proxy [**all** | **errors** | **events** | **packets**]

no debug inspect tls-proxy [**all** | **errors** | **events** | **packets**]

Syntax Description

all	Specifies all TLS proxy debugging.
errors	Specifies TLS proxy error debugging.
events	Specifies TLS proxy event debugging.
packets	Specifies TLS proxy packet debugging.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for TLS proxy:

```
hostname# debug inspect tls-proxy
```

Related Commands

Command	Description
client	Defines a cipher suite and sets the local dynamic certificate issuer or keypair.
ctl-provider	Defines a CTL provider instance and enters provider configuration mode.
show tls-proxy	Shows the TLS proxies.
tls-proxy	Defines a TLS proxy instance and sets the maximum sessions.

debug ip eigrp

To display debug information EIGRP protocol packets, use the **debug ip eigrp** command in privileged EXEC mode. To disable the debug information display, use the **no** form of this command.

debug ip eigrp [*as-number*] [*ip-addr mask* | **neighbor** *nbr-addr* | **notifications** | **summary**]

no debug ip eigrp [*as-number*] [*ip-addr mask* | **neighbor** *nbr-addr* | **notifications** | **summary**]

Syntax Description

as-number	(Optional) Specifies the autonomous system number of the EIGRP process for which you are viewing the event log. Because the security appliance only supports one EIGRP routing process, you do not need to specify the autonomous system number.
ip-addr mask	(Optional) Limits debug output to messages that fall within the range defined by the IP address and network mask.
neighbor *nbr-addr*	(Optional) Limits debug output to the specified neighbor.
notifications	(Optional) Limits debug output to EIGRP protocol events and notifications.
summary	(Optional) Limits debug output to summary route processing.
user-interface	(Optional) Limits debug output to user events.

Defaults

If no keywords or arguments are specified, only debug messages from the IPv4 ASDM are displayed.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	—	•	—	—

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

This command helps you analyze the packets that are sent and received on an interface.

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output from the **debug ip eigrp** command:

```
hostname# debug ip eigrp

IP-EIGRP Route Events debugging is on

EIGRP-IPv4(Default-IP-Routing-Table:1): Processing incoming UPDATE packet
EIGRP-IPv4(Default-IP-Routing-Table:1): Ext 192.168.3.0 255.255.255.0 M 386560 - 256000
130560 SM 360960 - 256000 104960
EIGRP-IPv4(Default-IP-Routing-Table:1): Ext 192.168.0.0 255.255.255.0 M 386560 - 256000
130560 SM 360960 - 256000 104960
EIGRP-IPv4(Default-IP-Routing-Table:1): Ext 192.168.3.0 255.255.255.0 M 386560 - 256000
130560 SM 360960 - 256000 104960
EIGRP-IPv4(Default-IP-Routing-Table:1): 172.69.43.0 255.255.255.0, - do advertise out
Ethernet0/1
EIGRP-IPv4(Default-IP-Routing-Table:1): Ext 172.69.43.0 255.255.255.0 metric 371200 -
256000 115200
EIGRP-IPv4(Default-IP-Routing-Table:1): 192.135.246.0 255.255.255.0, - do advertise out
Ethernet0/1
EIGRP-IPv4(Default-IP-Routing-Table:1): Ext 192.135.246.0 255.255.255.0 metric 46310656 -
45714176 596480
EIGRP-IPv4(Default-IP-Routing-Table:1): 172.69.40.0 255.255.255.0, - do advertise out
Ethernet0/1
EIGRP-IPv4(Default-IP-Routing-Table:1): Ext 172.69.40.0 255.255.255.0 metric 2272256 -
1657856 614400
EIGRP-IPv4(Default-IP-Routing-Table:1): 192.135.245.0 255.255.255.0, - do advertise out
Ethernet0/1
EIGRP-IPv4(Default-IP-Routing-Table:1): Ext 192.135.245.0 255.255.255.0 metric 40622080 -
40000000 622080
EIGRP-IPv4(Default-IP-Routing-Table:1): 192.135.244.0 255.255.255.0, - do advertise out
Ethernet0/1
```

Table 10-1describes the significant fields shown in the display.

Table 10-1 debug ip eigrp Field Descriptions

Field	Description
IP-EIGRP:	Indicates IP EIGRP messages.
Ext	Indicates that the following address is an external route rather than an internal route, which would be labeled as Int.
M	Displays the computed metric, which includes the value in the SM field and the cost between this router and the neighbor. The first number is the composite metric. The next two numbers are the inverse bandwidth and the delay, respectively.
SM	Displays the metric as reported by the neighbor.

Related Commands

Command	Description
debug eigrp packets	Displays debug information for EIGRP packets.

debug ipsec-over-tcp

To display IPSec-over-TCP debug information, use the **debug ipsec-over-tcp** command in privileged EXEC mode. To disable the display of debug information, use the **no** form of this command.

debug ipsec-over-tcp [*level*]

no debug ipsec-over-tcp

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables IPSec-over-TCP debug messages. The **show debug** command reveals that IPSec-over-TCP debug messages are enabled.

```
hostname# debug ipsec-over-tcp
debug ipsec-over-tcp  enabled at level 1
hostname# show debug
debug ipsec-over-tcp  enabled at level 1
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug ipv6

To display ipv6 debug messages, use the **debug ipv6** command in privileged EXEC mode. To stop the display of debug messages, use the **no** form of this command.

debug ipv6 {icmp | interface | mld | nd | packet | routing}

no debug ipv6 {icmp | interface | nd | packet | routing}

Syntax Description

icmp	Displays debug messages for IPv6 ICMP transactions, excluding ICMPv6 neighbor discovery transactions.
interface	Displays debug information for IPv6 interfaces.
mld	Displays debug messages for Multicast Listener Discovery (MLD).
nd	Displays debug messages for ICMPv6 neighbor discovery transactions.
packet	Displays debug messages for IPv6 packets.
routing	Displays debug messages for IPv6 routing table updates and route cache updates.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	—	•	•	—

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output for the **debug ipv6 icmp** command:

```
hostname# debug ipv6 icmp
13:28:40:ICMPv6:Received ICMPv6 packet from 2000:0:0:3::2, type 136
13:28:45:ICMPv6:Received ICMPv6 packet from FE80::203:A0FF:FED6:1400, type 135
13:28:50:ICMPv6:Received ICMPv6 packet from FE80::203:A0FF:FED6:1400, type 136
```

```
13:28:55:ICMPv6:Received ICMPv6 packet from FE80::203:A0FF:FED6:1400, type 135
```

Related Commands

Command	Description
ipv6 icmp	Defines access rules for ICMP messages that terminate on a security appliance interface.
ipv6 address	Configures an interface with an IPv6 address or addresses.
ipv6 nd dad attempts	Defines the number of neighbor discovery attempts performed during duplicate address detection.
ipv6 route	Defines a static entry in the IPv6 routing table.

debug iua-proxy

To display IUA proxy debug information, use the **debug iua-proxy** command in privileged EXEC mode. To disable the display of debug information, use the **no** form of this command.

debug iua-proxy [*level*]

no debug iua-proxy

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables IUA-proxy debug messages. The **show debug** command reveals that IUA-proxy debug messages are enabled.

```
hostname# debug iua-proxy
debug iua-proxy  enabled at level 1
hostname# show debug
debug iua-proxy  enabled at level 1
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug kerberos

To display Kerberos authentication debug information, use the **debug kerberos** command in privileged EXEC mode. To disable the display of debug information, use the **no** form of this command.

debug kerberos [*level*]

no debug kerberos

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables Kerberos debug messages. The **show debug** command reveals that Kerberos debug messages are enabled.

```
hostname# debug kerberos
debug kerberos  enabled at level 1
hostname# show debug
debug kerberos  enabled at level 1
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug l2tp

To display L2TP debug information, use the **debug l2tp** command in privileged EXEC mode. To disable the display of debug information, use the **no** form of this command.

debug l2tp {**data** | **error** | **event** | **packet**} *level*

no debug l2tp {**data** | **error** | **event** | **packet**} *level*

Syntax Description

data	displays data packet trace information.
error	Displays error events.
event	Displays L2TP connection events.
packet	Displays packet trace information.
level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables L2TP debug messages for connection events. The **show debug** command reveals that L2TP debug messages are enabled.

```
hostname# debug l2tp event 1
hostname# show debug
debug l2tp event enabled at level 1
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug ldap

To display LDAP debug information, use the **debug ldap** command in privileged EXEC mode. To disable the display of debug information, use the **no** form of this command.

debug ldap [*level*]

no debug ldap

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables LDAP debug messages. The **show debug** command reveals that LDAP debug messages are enabled.

```
hostname# debug ldap
debug ldap  enabled at level 1
hostname# show debug
debug ldap  enabled at level 1
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug mac-address-table

To show debug messages for the MAC address table, use the **debug mac-address-table** command in privileged EXEC mode. To stop showing debug messages for the MAC address table, use the **no** form of this command.

debug mac-address-table [*level*]

no debug mac-address-table [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default level is 1.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for the MAC address table:

```
hostname# debug mac-address-table
```

Related Commands

Command	Description
mac-address-table aging-time	Sets the timeout for dynamic MAC address entries.
mac-address-table static	Adds static MAC address entries to the MAC address table.
mac-learn	Disables MAC address learning.

Command	Description
show debug	Shows all enabled debuggers.
show mac-address-table	Shows MAC address table entries.

debug menu

To display detailed debug information for specific features, use the **debug menu** command in privileged EXEC mode.

debug menu

Caution The **debug menu** command should be used only under the supervision of Cisco TAC.

Syntax Description

This command should be used only under the supervision of Cisco TAC.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

This command should be used only under the supervision of Cisco TAC.

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug mfib

To display MFIB debug information, use the **debug mfib** command in privileged EXEC mode. To stop displaying debug information, use the **no** form of this command.

debug mfib {**db** | **init** | **mrib** | **pak** | **ps** | **signal**} [*group*]

no debug mfib {**db** | **init** | **mrib** | **pak** | **ps** | **signal**} [*group*]

Syntax Description

db	(Optional) Displays debug information for route database operations.
group	(Optional) IP address of the multicast group.
init	(Optional) Displays system initialization activity.
mrib	(Optional) Displays debug information for communication with MFIB.
pak	(Optional) Displays debug information for packet forwarding operations.
ps	(Optional) Displays debug information for process switching operations.
signal	(Optional) Displays debug information for MFIB signaling to routing protocols.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example displays MFIB dabase operation debug information:

```
hostname# debug mfib db
MFIB IPv4 db debugging enabled
```

Related Commands

Command	Description
show mfib	Displays MFIB forwarding entries and interfaces.

debug mgcp

To display detailed information about MGCP application inspection, use the **debug mgcp** command in privileged EXEC mode. To disable debugging, Use the **no** form of this command.

debug mgcp {**messages** | **parser** | **sessions**}

no debug mgcp {**messages** | **parser** | **sessions**}

messages	Displays debug information about MGCP messages.
parser	Displays debug information for parsing MGCP messages.
sessions	Displays debug information about MGCP sessions.

Defaults

All options are enabled.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	•	—

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

The **debug mgcp** command displays detailed information about mgcp inspection. The **no debug all** or **undebug all** commands turn off all enabled debugs.

Examples

The following example enables the display of detailed information about MGCP application inspection:

```
hostname# debug mgcp
```

Related Commands

Commands	Description
class-map	Defines the traffic class to which to apply security actions.
inspect mgcp	Enables MGCP application inspection.
mgcp-map	Defines an MGCP map and enables MGCP map configuration mode.
show mgcp	Displays information about MGCP sessions established through the security appliance.
show conn	Displays the connection state for different connection types.

debug module-boot

To show debugging messages about the SSM booting process, use the **debug module-boot** command in privileged EXEC mode. To stop showing debugging messages for the SSM booting process, use the **no** form of this command.

debug module-boot [*level*]

no debug module-boot [*level*]

Syntax Description

level	(Optional) Sets the debugging message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set this parameter to a higher number.

Defaults

The default *level* is 1.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Enabling the **debug** commands may slow down traffic on busy networks.To view the current debugging command settings, enter the **show debug** command. To stop the debugging output, enter the **no debug** command. To stop all debugging messages from being displayed, enter the **no debug all** command.

Examples

The following example enables debugging messages for the SSM booting process:

```
hostname# debug module-boot
```

Related Commands

Command	Description
hw-module module recover	Recovers an intelligent SSM by loading a recovery image from a TFTP server.
hw-module module reset	Shuts down an SSM and performs a hardware reset.

Command	Description
hw-module module reload	Reloads the intelligent SSM software.
hw-module module shutdown	Shuts down the SSM software in preparation for being powered off without losing configuration data.
show module	Shows SSM information.

debug mrib

To display MRIB debug information, use the **debug mrib** command in privileged EXEC mode. To stop the display of debug information, use the **no** form of this command.

debug mrib {**client** | **io** | **route** [*group*] | **table**}

no debug mrib {**client** | **io** | **route** [*group*] | **table**}

Syntax Description

client	Enables debugging for MRIB client management activity.
io	Enables debugging of MRIB I/O events.
route	Enables debugging of MRIB routing entry activity.
group	Enables debugging of MRIB routing entry activity for the specified group.
table	Enables debugging of MRIB table management activity.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example shows how to enable debugging of MRIB I/O events:

```
hostname# debug mrib io
IPv4 MRIB io debugging is on
```

Related Commands

Command	Description
show mrib client	Displays information about the MRIB client connections.
show mrib route	Displays MRIB table entries.

debug nac

To enable logging of NAC Framework events, use the **debug nac** command in privileged EXEC mode. To disable the logging of NAC debug messages, use the **no** form of this command.

debug nac {**all** | **auth** | **errors** | **events**}

no debug nac {**all** | **auth** | **errors** | **events**}

Syntax Description

all	Enables logging of debug messages about all NAC information.
auth	Enables logging of debug messages about NAC authentication requests and responses.
errors	Enables logging of NAC session errors.
events	Enables logging of NAC session events.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

When you use this command, the security appliance logs the following types of NAC events: initializations, exception list matches, ACS transactions, clientless authentications, default ACL applications, and revalidations.

The high priority assigned to debugging output can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables the logging of all NAC session events:

```
hostname# debug nac events
hostname#
```

The following example enables the logging of all NAC debug messages:

```
hostname# debug nac all
hostname#
```

The following example disables the logging of all NAC debug messages:

```
hostname# no debug nac
hostname#
```

Related Commands

Command	Description
debug eap	Enables logging of Extensible Authentication Protocol events to debug NAC Framework messaging.
debug eou	Enables logging of EAP over UDP events to debug NAC Framework messaging.
show vpn-session_summary.db	Displays the number of IPSec, WebVPN, and NAC sessions.
show vpn-session.db	Displays information about VPN sessions, including NAC results.

debug ntdomain

To display NT domain authentication debug information, use the **debug ntdomain** command in privileged EXEC mode. To disable the display of NT domain debug information, use the **no** form of this command.

debug ntdomain [*level*]

no debug ntdomain

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables NT domain debug messages. The **show debug** command reveals that NT domain debug messages are enabled.

```
hostname# debug ntdomain
debug ntdomain  enabled at level 1
hostname# show debug
debug ntdomain  enabled at level 1
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug ntp

To show debug messages for NTP, use the **debug ntp** command in privileged EXEC mode. To stop showing debug messages for NTP, use the **no** form of this command.

debug ntp {**adjust** | **authentication** | **events** | **loopfilter** | **packets** | **params** | **select** | **sync** | **validity**}

no debug ntp {**adjust** | **authentication** | **events** | **loopfilter** | **packets** | **params** | **select** | **sync** | **validity**}

Syntax Description

adjust	Shows messages about NTP clock adjustments.
authentication	Shows messages about NTP authentication.
events	Shows messages about NTP events.
loopfilter	Shows messages about NTP loop filter.
packets	Shows messages about NTP packets.
params	Shows messages about NTP clock parameters.
select	Shows messages about NTP clock selection.
sync	Shows messages about NTP clock synchronization.
validity	Shows messages about NTP peer clock validity.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for NTP:

```
hostname# debug ntp events
```

Related Commands

Command	Description
ntp authenticate	Enables NTP authentication.
ntp server	Identifies an NTP server.
show debug	Shows all enabled debuggers.
show ntp associations	Shows the NTP servers with which the security appliance is associated.
show ntp status	Shows the status of the NTP association.

debug ospf

To display debug information about the OSPF routing processes, use the **debug ospf** command in privileged EXEC mode. To stop displaying debug information, use the **no** form of this command.

debug ospf [**adj** | **database-timer** | **events** | **flood** | **lsa-generation** | **packet** | **retransmission** | **spf** [**external** | **inter** | **intra**] | **tree**]

no debug ospf [**adj** | **database-timer** | **events** | **flood** | **lsa-generation** | **packet** | **retransmission** | **spf** [**external** | **inter** | **intra**] | **tree**]

Syntax Description

adj	(Optional) Enables the debugging of OSPF adjacency events.
database-timer	(Optional) Enables the debugging of OSPF timer events.
events	(Optional) Enables the debugging of OSPF events.
external	(Optional) Limits SPF debugging to external events.
flood	(Optional) Enables the debugging of OSPF flooding.
inter	(Optional) Limits SPF debugging to inter-area events.
intra	(Optional) Limits SPF debugging to intra-area events.
lsa-generation	(Optional) Enables the debugging of OSPF summary LSA generation.
packet	(Optional) Enables the debugging of received OSPF packets.
retransmission	(Optional) Enables the debugging of OSPF retransmission events.
spf	(Optional) Enables the debugging of OSPF shortest path first calculations. You can limit the SPF debug information by using the **external**, **inter**, and **intra** keywords.
tree	(Optional) Enables the debugging of OSPF database events.

Defaults

Displays all OSPF debug information if no keyword is provided.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	—	•	—	—

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output from the **debug ospf events** command:

```
hostname# debug ospf events
ospf event debugging is on

OSPF:hello with invalid timers on interface Ethernet0
hello interval received 10 configured 10
net mask received 255.255.255.0 configured 255.255.255.0
dead interval received 40 configured 30
```

Related Commands

Command	Description
show ospf	Displays general information about the OSPF routing process.

debug parser cache

To display CLI parser debug information, use the **debug parser cache** command in privileged EXEC mode. To disable the display of CLI parser debug information, use the **no** form of this command.

debug parser cache [*level*]

no debug parser cache

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables CLI parser debug messages. The **show debug** command reveals the current debug configuration. The CLI parser debug messages appear before and after the output of the **show debug** command.

```
hostname# debug parser cache
debug parser cache enabled at level 1
hostname# show debug
parser cache: try to match 'show debug' in exec mode
debug parser cache enabled at level 1
parser cache: hit at index 8
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug pim

To display PI M debug information, use the **debug pim** command in privileged EXEC mode. To stop displaying debug information, use the **no** form of this command.

debug pim [**df-election** [**interface** *if_name* | **rp** *rp*] | **group** *group* | **interface** *if_name* | **neighbor**]

no debug pim [**df-election** [**interface** *if_name* | **rp** *rp*] | **group** *group* | **interface** *if_name* | **neighbor**]

Syntax Description

df-election	(Optional) Displays debug messages for PIM bidirectional DF-election message processing.
group *group*	(Optional) Displays debug information for the specified group. The value for *group* can be one of the following: • Name of the multicast group, as defined in the DNS hosts table or with the domain **ipv4 host** command. • IP address of the multicast group. This is a multicast IP address in four-part dotted-decimal notation.
interface *if_name*	(Optional) When used with the **df-election** keyword, it limits the DF election debug display to information for the specified interface. When used without the **df-election** keyword, displays PIM error messages for the specified interface. **Note** The **debug pim interface** command does not display PIM protocol activity messages; it only displays error messages. To see debug information for PIM protocol activity, use the **debug pim** command without the **interface** keyword. You can use the **group** keyword to limit the display to the specified multicast group.
neighbor	(Optional) Displays only the sent/received PIM hello messages.
rp *rp*	(Optional) Can be either one of the following: • Name of the RP, as defined in the Domain Name System (DNS) hosts table or with the domain **ipv4 host** command. • IP address of the RP. This is a multicast IP address in four-part dotted-decimal notation.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Logs PIM packets received and transmitted and also PIM-related events.

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output from the **debug pim** command:

```
hostname# debug pim
PIM: Received Join/Prune on Ethernet1 from 172.24.37.33
PIM: Received Join/Prune on Ethernet1 from 172.24.37.33
PIM: Received Join/Prune on Tunnel0 from 10.3.84.1
PIM: Received Join/Prune on Ethernet1 from 172.24.37.33
PIM: Received Join/Prune on Ethernet1 from 172.24.37.33
PIM: Received RP-Reachable on Ethernet1 from 172.16.20.31
PIM: Update RP expiration timer for 224.2.0.1
PIM: Forward RP-reachability packet for 224.2.0.1 on Tunnel0
PIM: Received Join/Prune on Ethernet1 from 172.24.37.33
PIM: Prune-list (10.221.196.51/32, 224.2.0.1)
PIM: Set join delay timer to 2 seconds for (10.221.0.0/16, 224.2.0.1) on Ethernet1
PIM: Received Join/Prune on Ethernet1 from 172.24.37.6
PIM: Received Join/Prune on Ethernet1 from 172.24.37.33
PIM: Received Join/Prune on Tunnel0 from 10.3.84.1
PIM: Join-list: (*, 224.2.0.1) RP 172.16.20.31
PIM: Add Tunnel0 to (*, 224.2.0.1), Forward state
PIM: Join-list: (10.0.0.0/8, 224.2.0.1)
PIM: Add Tunnel0 to (10.0.0.0/8, 224.2.0.1), Forward state
PIM: Join-list: (10.4.0.0/16, 224.2.0.1)
PIM: Prune-list (172.24.84.16/28, 224.2.0.1) RP-bit set RP 172.24.84.16
PIM: Send Prune on Ethernet1 to 172.24.37.6 for (172.24.84.16/28, 224.2.0.1), RP
PIM: For RP, Prune-list: 10.9.0.0/16
PIM: For RP, Prune-list: 10.16.0.0/16
PIM: For RP, Prune-list: 10.49.0.0/16
PIM: For RP, Prune-list: 10.84.0.0/16
PIM: For RP, Prune-list: 10.146.0.0/16
PIM: For 10.3.84.1, Join-list: 172.24.84.16/28
PIM: Send periodic Join/Prune to RP via 172.24.37.6 (Ethernet1)
```

Related Commands

Command	Description
show pim group-map	Displays group-to-protocol mapping table.
show pim interface	Displays interface-specific information for PIM.
show pim neighbor	Displays entries in the PIM neighbor table.

debug pix acl

To show pix acl debug messages, use the **debug pix acl** command in privileged EXEC mode. To stop showing debug messages, use the **no** form of this command.

debug pix acl

no debug pix acl

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use debug commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use debug commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased debug command processing overhead will affect system use.

Examples

The following example enables debug messages that :

```
hostname# debug pix acl
```

Related Commands

Command	Description
debug pix process	Shows debug messages for xlate and secondary connections processing.
show debug	Shows all enabled debuggers.

debug pix cls

To show pix cls debug messages, use the **debug pix cls** command in privileged EXEC mode. To stop showing debug messages, use the **no** form of this command.

debug pix cls

no debug pix cls

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use debug commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use debug commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased debug command processing overhead will affect system use.

Examples

The following example enables debug messages that :

```
hostname# debug pix cls
```

Related Commands

Command	Description
debug pix process	Shows debug messages for xlate and secondary connections processing.
show debug	Shows all enabled debuggers.

debug pix pkt2pc

To show debug messages that trace packets sent to the uauth code and that trace the event where the uauth proxy session is cut through to the data path, use the **debug pix pkt2pc** command in privileged EXEC mode. To stop showing debug messages, use the **no** form of this command.

debug pix pkt2pc

no debug pix pkt2pc

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages that trace packets sent to the uauth code and that trace the event where the uauth proxy session is cut through to the data path:

```
hostname# debug pix pkt2pc
```

Related Commands

Command	Description
debug pix process	Shows debug messages for xlate and secondary connections processing.
show debug	Shows all enabled debuggers.

debug pix process

To show debug messages for xlate and secondary connections processing, use the **debug pix process** command in privileged EXEC mode. To stop showing debug messages, use the **no** form of this command.

debug pix process

no debug pix process

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for xlate and secondary connections processing:

```
hostname# debug pix process
```

Related Commands

Command	Description
debug pix pkt2pc	Shows debug messages that trace packets sent to the uauth code and that trace the event where the uauth proxy session is cut through to the data path.
show debug	Shows all enabled debuggers.

debug pix uauth

To showpix uauth debug messages, use the **debug pix uauth** command in privileged EXEC mode. To stop showing debug messages, use the **no** form of this command.

debug pix uauth

no debug pix uauth

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use debug commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use debug commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased debug command processing overhead will affect system use.

Examples

The following example enables debug messages that :

```
hostname# debug pix uauth
```

Related Commands

Command	Description
debug pix process	Shows debug messages for xlate and secondary connections processing.
show debug	Shows all enabled debuggers.

debug pptp

To show debug messages for PPTP, use the **debug pptp** command in privileged EXEC mode. To stop showing debug messages for PPTP, use the **no** form of this command.

debug pptp [*level*]

no debug pptp [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

To see the current debug command settings, enter the **show debug** command. To stop the debug output, enter the **no debug** command. To stop all debug messages from being displayed, enter the **no debug all** command.

Note Enabling the **debug pptp** command may slow down traffic on busy networks.

Examples

The following example enables debug messages at the default level (1) for PPTP application inspection:

```
hostname# debug pptp
```

Related Commands

Command	Description
class-map	Defines the traffic class to which to apply security actions.
inspect pptp	Enables PPTP application inspection.

Command	Description
policy-map	Associates a class map with specific security actions.
service-policy	Applies a policy map to one or more interfaces.

debug radius

To show debug messages for AAA, use the **debug radius** command in privileged EXEC mode. To stop showing RADIUS messages, use the **no** form of this command.

debug radius [**all** | **decode** | **session** | **user** *username*]]

no debug radius

Syntax Description

all	(Optional) Show RADIUS debugging messages for all users and sessions, including decoded RADIUS messages.
decode	(Optional) Show decoded content of RADIUS messages. Content of all RADIUS packets display, including hexadecimal values and the decoded, eye-readable versions of these values.
session	(Optional) Show session-related RADIUS messages. Packet types for sent and received RADIUS messages display but not the packet content.
user	(Optional) Show RADIUS debugging messages for a specific user.
username	Specifies the user whose messages you want to see. Valid with the **user** keyword only.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

The **debug radius** command displays detailed information about RADIUS messaging between the security appliance and a RADIUS AAA server. The **no debug all** or **undebug all** commands turn off all enabled debugs.

Examples

The following example shows decoded RADIUS messages, which happen to be accounting packets:

```
hostname(config)# debug radius decode
hostname(config)# RADIUS packet decode (accounting request)

-------------------------------------
```

```
Raw packet data (length = 216).....
i
Parsed packet data.....
Radius: Code = 4 (0x04)
Radius: Identifier = 105 (0x69)
Radius: Length = 216 (0x00D8)
Radius: Vector: 842E0E99F44C00C05A0A19AB88A81312
Radius: Type = 40 (0x28) Acct-Status-Type
Radius: Length = 6 (0x06)
Radius: Value (Hex) = 0x2
Radius: Type = 5 (0x05) NAS-Port
Radius: Length = 6 (0x06)
Radius: Value (Hex) = 0x1
Radius: Type = 4 (0x04) NAS-IP-Address
Radius: Length = 6 (0x06)
Radius: Value (IP Address) = 10.1.1.1 (0x0A010101)
Radius: Type = 14 (0x0E) Login-IP-Host
Radius: Length = 6 (0x06)
Radius: Value (IP Address) = 10.2.0.50 (0xD0FE1291)
Radius: Type = 16 (0x10) Login-TCP-Port
Radius: Length = 6 (0x06)
Radius: Value (Hex) = 0x50
Radius: Type = 44 (0x2C) Acct-Session-Id
Radius: Length = 12 (0x0C)
Radius: Value (String) =
30 78 31 33 30 31 32 39 66 65                       |  0x130129fe
Radius: Type = 1 (0x01) User-Name
Radius: Length = 9 (0x09)
Radius: Value (String) =
62 72 6f 77 73 65 72                                |  browser
Radius: Type = 46 (0x2E) Acct-Session-Time
Radius: Length = 6 (0x06)
Radius: Value (Hex) = 0x0
Radius: Type = 42 (0x2A) Acct-Input-Octets
Radius: Length = 6 (0x06)
Radius: Value (Hex) = 0x256D
Radius: Type = 43 (0x2B) Acct-Output-Octets
Radius: Length = 6 (0x06)
Radius: Value (Hex) = 0x3E1
Radius: Type = 26 (0x1A) Vendor-Specific
Radius: Length = 30 (0x1E)
Radius: Vendor ID = 9 (0x00000009)
Radius: Type = 1 (0x01) Cisco-AV-pair
Radius: Length = 24 (0x18)
Radius: Value (String) =
69 70 3a 73 6f 75 72 63 65 2d 69 70 3d 31 30 2e     |  ip:source-ip=10.
31 2e 31 2e 31 30                                   |  1.1.10
Radius: Type = 26 (0x1A) Vendor-Specific
Radius: Length = 27 (0x1B)
Radius: Vendor ID = 9 (0x00000009)
Radius: Type = 1 (0x01) Cisco-AV-pair
Radius: Length = 21 (0x15)
Radius: Value (String) =
69 70 3a 73 6f 75 72 63 65 2d 70 6f 72 74 3d 33     |  ip:source-port=3
34 31 33                                            |  413
Radius: Type = 26 (0x1A) Vendor-Specific
Radius: Length = 40 (0x28)
Radius: Vendor ID = 9 (0x00000009)
Radius: Type = 1 (0x01) Cisco-AV-pair
Radius: Length = 34 (0x22)
Radius: Value (String) =
69 70 3a 64 65 73 74 69 6e 61 74 69 6f 6e 2d 69     |  ip:destination-i
70 3d 32 30 38 2e 32 35 34 2e 31 38 2e 31 34 35     |  p=10.2.0.50
Radius: Type = 26 (0x1A) Vendor-Specific
```

```
Radius: Length = 30 (0x1E)
Radius: Vendor ID = 9 (0x00000009)
Radius: Type = 1 (0x01) Cisco-AV-pair
Radius: Length = 24 (0x18)
Radius: Value (String) =
69 70 3a 64 65 73 74 69 6e 61 74 69 6f 6e 2d 70    |  ip:destination-p
6f 72 74 3d 38 30                                  |  ort=80
```

Related Commands

Command	Description
show running-config	Displays the configuration that is running on the security appliance.

debug redundant-interface

To show debug messages about redundant interfaces, use the **debug redundant-interface** command in privileged EXEC mode. To stop showing debug messages for redundant interfaces, use the **no** form of this command.

debug redundant-interface [*level*]

no debug redundant-interfac [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default level is 1.

Command Modes

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

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

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for redundant interfaces:

```
hostname# debug redundant-interface
```

Related Commands

Command	Description
interface redundant	Creates a redundant interface.
member-interface	Assigns a physical interface to a redundant interface.
redundant-interface	Changes the active interface in a redundant interface pair.
show debug	Shows all enabled debuggers.

debug rip

To display debug information for RIP, use the **debug rip** command in privileged EXEC mode. To disable the debug information display, use the **no** form of this command.

debug rip [**database** | **events**]

no debug rip [**database** | **events**]

Syntax Description

database	Displays RIP database events.
events	Displays RIP processing events.

Defaults

All RIP events are shown in the debug output.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	—	•	—	—

Command History

Release	Modification
Preexisting	This command was preexisting.
7.2(1)	The **database** and **events** keywords were added.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output from the **debug rip** command:

```
hostname# debug rip

RIP: broadcasting general request on GigabitEthernet0/1
RIP: broadcasting general request on GigabitEthernet0/2
RIP: Received update from 10.89.80.28 on GigabitEthernet0/1
    10.89.95.0 in 1 hops
    10.89.81.0 in 1 hops
    10.89.66.0 in 2 hops
    172.31.0.0 in 16 hops (inaccessible)
    0.0.0.0 in 7 hops
RIP: Sending update to 255.255.255.255 via GigabitEthernet0/1 (10.89.64.31)
```

```
    subnet 10.89.94.0, metric 1
    172.31.0.0 in 16 hops (inaccessible)
RIP: Sending update to 255.255.255.255 via GigabitEthernet0/2 (10.89.94.31)
    subnet 10.89.64.0, metric 1
    subnet 10.89.66.0, metric 3
    172.31.0.0 in 16 hops (inaccessible)
    default 0.0.0.0, metric 8
RIP: bad version 128 from 192.168.80.43
```

Related Commands

Command	Description
router rip	Configures a RIP process.
show running-config rip	Displays the RIP commands in the running configuration.

debug rtp

To display debug information and error messages for RTP packets associated with H.323 and SIP inspection, use the **debug rtp** command in privileged EXEC mode. To disable the display of debug information, use the **no** form of this command.

debug rtp [*level*]

no debug rtp [*level*]

Syntax Description

level	(Optional) Specifies an optional level of debug.

Defaults

The default *level* is 1.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example shows how to enable debugging for RTP packets using the **debug rtp** command:

```
hostname# debug rtp 255
debug rtp  enabled at level 255
```

Related Commands

Command	Description
policy-map	Creates a Layer 3/4 policy map.

Command	Description
rtp-conformance	Checks RTP packets flowing on the pinholes for protocol conformance in H.323 and SIP.
show running-config policy-map	Displays all current policy map configurations.

debug rtsp

To show debug messages for RTSP application inspection, use the **debug rtsp** command in privileged EXEC mode. To stop showing debug messages for RTSP application inspection, use the **no** form of this command.

debug rtsp [*level*]

no debug rtsp [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

To see the current debug command settings, enter the **show debug** command. To stop the debug output, enter the **no debug** command. To stop all debug messages from being displayed, enter the **no debug all** command.

Note Enabling the **debug rtsp** command may slow down traffic on busy networks.

Examples

The following example enables debug messages at the default level (1) for RTSP application inspection:

```
hostname# debug rtsp
```

Related Commands

Command	Description
class-map	Defines the traffic class to which to apply security actions.
inspect rtsp	Enables RTSP application inspection.
policy-map	Associates a class map with specific security actions.
service-policy	Applies a policy map to one or more interfaces.

debug sdi

To display SDI authentication debug information, use the **debug sdi** command in privileged EXEC mode. To disable the display of SDI debug information, use the **no** form of this command.

debug sdi [*level*]

no debug sdi

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables SDI debug messages. The **show debug** command reveals that SDI debug messages are enabled.

```
hostname# debug sdi
debug sdi  enabled at level 1
hostname# show debug
debug sdi  enabled at level 1
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug sequence

To add a sequence number to the beginning of all debug messages, use the **debug sequence** command in privileged EXEC mode. To disable the use of debug sequence numbers, use the **no** form of this command.

debug sequence [*level*]

no debug sequence

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The defaults are as follows:

- Debug message sequence numbers are disabled.
- The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables sequence numbers in debug messages. The **debug parser cache** command enables CLI parser debug messages. The **show debug** command reveals the current debug configuration. The CLI parser debug messages shown include sequence numbers before each message.

```
hostname# debug sequence
debug sequence  enabled at level 1
hostname# debug parser cache
debug parser cache enabled at level 1
```

```
hostname# show debug
0: parser cache: try to match 'show debug' in exec mode
debug parser cache enabled at level 1
debug sequence  enabled at level 1
1: parser cache: hit at index 8
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug session-command

To show debugging messages for a session to an SSM, use the **debug session-command** command in privileged EXEC mode. To stop showing debugging messages for a session to an SSM, use the **no** form of this command.

debug session-command [*level*]

no debug session-command [*level*]

Syntax Description

level	(Optional) Sets the debugging message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set this parameter to a higher number.

Defaults

The default level is 1.

Command Modes

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debugging messages for a session to an SSM:

```
hostname# debug session-command
```

Related Commands

Command	Description
session	Sessions to an SSM.

debug sip

To show debug messages for SIP application inspection, use the **debug sip** command in privileged EXEC mode. To stop showing debug messages for SIP application inspection, use the **no** form of this command.

debug sip [**ha**]

no debug sip [**ha**]

Syntax Description

ha	(Optional) Display SIP Stateful Failover messages. When this keyword is used with the **debug sip** command on the active unit, debug messages are displayed when SIP state information is sent to the standby unit. When this keyword is used with the **debug sip** command on the standby unit, debug messages are displayed with state updates are received from the active unit.

Defaults

No default behavior or values.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.
8.0(2)	The **ha** keyword was added.

Usage Guidelines

To see the current debug command settings, enter the **show debug** command. To stop the debug output, enter the **no debug** command. To stop all debug messages from being displayed, enter the **no debug all** command.

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output from the **debug sip** command run on the active unit or failover group in a failover pair:

```
hostname# debug sip ha
SIP HA:    Sending    update SESSION message from faddr 10.132.80.120/5060 laddr
10.130.80.4/50295 Call-id: 001201e8-8a36000d-196df7f1-17cfef14@10.130.80.4 From:
sip:1004@10.132.80.120:001201e88a3600124a7fad61-640406c0 To: sip:1009@10.132.80.120:
State:1

SIP HA:    msg sent to peer successful  Version: 1 Action: update Object: session

SIP HA:    Sending    update TX message from faddr 10.132.80.120/5060laddr
10.130.80.4/50295CSeq 101 INVITEState Transaction Calling
```

The following is sample output from the **debug sip** command run on the standby unit or failover group in a failover pair:

```
hostname# debug sip ha
SIP HA:    Message    received from peer, Version: 1 Action: add Object: session

SIP HA:    Created    SIP session for faddr 10.132.80.120/5060 laddr 10.130.80.4/50295
Call-id: 001201e8-8a36000d-196df7f1-17cfef14@10.130.80.4 From:
sip:1004@10.132.80.120:001201e88a3600124a7fad61-640406c0 To: sip:1009@10.132.80.120: 1
total

SIP HA:    Message    received from peer, Version: 1 Action: add Object: tx

SIP HA:    Found an existing session faddr 10.132.80.120/5060 laddr 10.130.80.4/50295
Call-id: 001201e8-8a36000d-196df7f1-17cfef14@10.130.80.4 From:
sip:1004@10.132.80.120:001201e88a3600124a7fad61-640406c0 To: sip:1009@10.132.80.120:

SIP HA:    Created    SIP Transaction    for faddr 10.132.80.120/5060 to   laddr
10.130.80.4/50295CSeq 101 INVITEState Transaction Calling
```

Related Commands

Command	Description
class-map	Defines the traffic class to which to apply security actions.
inspect sip	Enables SIP application inspection.
show conn	Displays the connection state for different connection types.
show sip	Displays information about SIP sessions established through the security appliance.
timeout	Sets the maximum idle time duration for different protocols and session types.

debug skinny

To show debug messages for SCCP (Skinny) application inspection, use the **debug skinny** command in privileged EXEC mode. To stop showing debug messages for SCCP application inspection, use the **no** form of this command.

debug skinny [*level*]

no debug skinny [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

To see the current debug command settings, enter the **show debug** command. To stop the debug output, enter the **no debug** command. To stop all debug messages from being displayed, enter the **no debug all** command.

Note Enabling the **debug skinny** command may slow down traffic on busy networks.

Examples

The following example enables debug messages at the default level (1) for SCCP application inspection:

```
hostname# debug skinny
```

Related Commands

Command	Description
class-map	Defines the traffic class to which to apply security actions.
inspect skinny	Enables SCCP application inspection.
show skinny	Displays information about SCCP sessions established through the security appliance.
show conn	Displays the connection state for different connection types.
timeout	Sets the maximum idle time duration for different protocols and session types.

debug sla monitor

To display debug messages for the SLA monitor operation, use the **debug sla monitor** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

debug sla monitor [**error** | **trace**] [*sla-id*]

no debug sla monitor [*sla-id*]

Syntax Description

error	(Optional) Output IP SLA Monitor Error Messages.
sla-id	(Optional) The ID of the SLA to debug.
trace	(Optional) Output IP SLA Monitor Trace Messages.

Defaults

Both error and trace messages are shown by default.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

Only 32 SLA operations can be debugged at one time.

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables SLA operation error debugging:

```
hostname(config)# debug sla monitor error
```

The following example shows how to display SLA operation trace messages for the specified SLA operation:

```
hostname(config)# debug sla monitor trace 123
```

Related Commands

Command	Description
clear configure route	Removes statically configured **route** commands.
clear route	Removes routes learned through dynamic routing protocols such as RIP.
show route	Displays route information.
show running-config route	Displays configured routes.

debug sqlnet

To show debug messages for SQL*Net application inspection, use the **debug sqlnet** command in privileged EXEC mode. To stop showing debug messages for SQL*Net application inspection, use the **no** form of this command.

debug sqlnet [*level*]

no debug sqlnet [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

To see the current debug command settings, enter the **show debug** command. To stop the debug output, enter the **no debug** command. To stop all debug messages from being displayed, enter the **no debug all** command.

Note Enabling the **debug sqlnet** command may slow down traffic on busy networks.

Examples

The following example enables debug messages at the default level (1) for SQL*Net application inspection:

```
hostname# debug sqlnet
```

Related Commands

Command	Description
class-map	Defines the traffic class to which to apply security actions.
inspect sqlnet	Enables SQL*Net application inspection.
policy-map	Associates a class map with specific security actions.
service-policy	Applies a policy map to one or more interfaces.
show conn	Displays the connection state for different connection types, including SQL*Net.

debug ssh

To display debug information and error messages associated with SSH, use the **debug ssh** command in privileged EXEC mode. To disable the display of debug information, use the **no** form of this command.

debug ssh [*level*]

no debug ssh [*level*]

Syntax Description

level	(Optional) Specifies an optional level of debug.

Defaults

The default *level* is 1.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output from the **debug ssh 255** command:

```
hostname# debug ssh 255
debug ssh  enabled at level 255
SSH2 0: send: len 64 (includes padlen 17)
SSH2 0: done calc MAC out #239
SSH2 0: send: len 32 (includes padlen 7)
SSH2 0: done calc MAC out #240
SSH2 0: send: len 64 (includes padlen 15)
SSH2 0: done calc MAC out #241
SSH2 0: send: len 32 (includes padlen 16)
SSH2 0: done calc MAC out #242
SSH2 0: send: len 64 (includes padlen 7)
SSH2 0: done calc MAC out #243
SSH2 0: send: len 64 (includes padlen 18)
SSH2 0: done calc MAC out #244
```

```
SSH2 0: send: len 64 (includes padlen 8)
SSH2 0: done calc MAC out #245
SSH2 0: send: len 64 (includes padlen 18)
SSH2 0: done calc MAC out #246
SSH2 0: send: len 64 (includes padlen 7)
SSH2 0: done calc MAC out #247
SSH2 0: send: len 64 (includes padlen 18)
SSH2 0: done calc MAC out #248
SSH2 0: send: len 64 (includes padlen 7)
SSH2 0: done calc MAC out #249
SSH2 0: send: len 64 (includes padlen 18)
SSH2 0: done calc MAC out #250
SSH2 0: send: len 64 (includes padlen 8)
SSH2 0: done calc MAC out #251
SSH2 0: send: len 64 (includes padlen 18)
SSH2 0: done calc MAC out #252
SSH2 0: send: len 64 (includes padlen 7)
SSH2 0: done calc MAC out #253
SSH2 0: send: len 64 (includes padlen 18)
SSH2 0: done calc MAC out #254
SSH2 0: send: len 64 (includes padlen 8)
SSH2 0: done calc MAC out #255
SSH2 0: send: len 64 (includes padlen 18)
SSH2 0: done calc MAC out #256
SSH2 0: send: len 64 (includes padlen 7)
SSH2 0: done calc MAC out #257
SSH2 0: send: len 64 (includes padlen 18)
SSH2 0: done calc MAC out #258
```

Related Commands

Command	Description
clear configure ssh	Clears all SSH commands from the running configuration.
show running-config ssh	Displays the current SSH commands in the running configuration.
show ssh sessions	Displays information about active SSH sessions to the security appliance.
ssh	Allows SSH connectivity to the security appliance from the specified client or network.

debug sunrpc

To show debug messages for RPC application inspection, use the **debug sunrpc** command in privileged EXEC mode. To stop showing debug messages for RPC application inspection, use the **no** form of this command.

debug sunrpc [*level*]

no debug sunrpc [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

To see the current debug command settings, enter the **show debug** command. To stop the debug output, enter the **no debug** command. To stop all debug messages from being displayed, enter the **no debug all** command.

Note Enabling the **debug sunrpc** command may slow down traffic on busy networks.

Examples

The following example enables debug messages at the default level (1) for RPC application inspection:

```
hostname# debug sunrpc
```

Related Commands

Command	Description
class-map	Defines the traffic class to which to apply security actions.
inspect sunrpc	Enables Sun RPC application inspection.
policy-map	Associates a class map with specific security actions.
show conn	Displays the connection state for different connection types, including RPC.
timeout	Sets the maximum idle time duration for different protocols and session types.

debug switch ilpm

To show debug messages for models with a built-in switch, such as the ASA 5505 adaptive security appliance, show debug messages for PoE, use the **debug switch ilpm** command in privileged EXEC mode. To stop showing debug messages for PoE, use the **no** form of this command.

debug switch ilpm [**events** | **errors**] [*level*]

no debug switch ilpm [**events** | **errors**] [*level*]

Syntax Description

errors	(Optional) Shows troubleshooting information when there is an error.
events	(Optional) Shows PoE events.
level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

By default, both events and errors are shown if you do not specify a keyword. The default level is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	—	—

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for PoE ports:

```
hostname# debug switch ilpm
```

Related Commands

Command	Description
interface vlan	Adds a VLAN interface.
debug switch manager	Shows debug messages for VLAN assignment and **switchport** command-caused events and errors.
show debug	Shows all enabled debuggers.

debug switch manager

To show debug messages for switch port models with a built-in switch, such as the ASA 5505 adaptive security appliance, show debug messages for VLAN assignment, and **switchport** command-caused events and errors, use the **debug switch manager** command in privileged EXEC mode. To stop showing debug messages for switch ports, use the **no** form of this command.

debug switch manager [**events** | **errors**] [*level*]

no debug switch manager [**events** | **errors**] [*level*]

Syntax Description

errors	(Optional) Shows troubleshooting information when there is an error.
events	(Optional) Shows the switch manager events.
level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

By default, both events and errors are shown if you do not specify a keyword. The default level is 1.

Command Modes

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

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

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

Using **debug** commands might slow down traffic on busy networks.

Examples

The following example enables debug messages for switch ports:

```
hostname# debug switch manager
```

Related Commands

Command	Description
interface vlan	Adds a VLAN interface.
debug switch ilpm	Shows debug messages for PoE.
show debug	Shows all enabled debuggers.

debug tacacs

To display TACACS+ debug information, use the **debug tacacs** command in privileged EXEC mode. To disable the display of TACACS+ debug information, use the **no** form of this command.

debug tacacs [**session** | **user** *username*]

no debug tacacs [**session** | **user** *username*]

Syntax Description

session	Displays session-related TACACS+ debug messages.
user	Displays user-specific TACACS+ debug messages. You can display TACACS+ debug messages for only one user at a time.
username	Specifies the user whose TACACS+ debug messages you want to view.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables TACACS+ debug messages. The **show debug** command reveals that TACACS+ debug messages are enabled.

```
hostname# debug tacacs user admin342
hostname# show debug
debug tacacs user admin342
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug tcp-map

To show debug messages for TCP application inspection maps, use the **debug tcp-map** command in privileged EXEC mode. To stop showing debug messages for TCP application inspection, use the **no** form of this command.

debug tcp-map

no debug tcp-map

Syntax Description

This command has no arguments or keywords.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables debug messages for TCP application inspection maps. The **show debug** command reveals that debug messages for TCP application inspection maps are enabled.

```
hostname# debug tcp-map
debug tcp-map enabled at level 1.
hostname# show debug
debug tcp-map enabled at level 1.
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug timestamps

To add timestamp information to the beginning of all debug messages, use the **debug timestamps** command in privileged EXEC mode. To disable the use of debug timestamps, use the **no** form of this command.

debug timestamps [*level*]

no debug timestamps

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The defaults are as follows:

- Debug timestamp information is disabled.
- The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables timestamps in debug messages. The **debug parser cache** command enables CLI parser debug messages. The **show debug** command reveals the current debug configuration. The CLI parser debug messages shown include timestamps before each message.

```
hostname# debug timestamps
debug timestamps  enabled at level 1
hostname# debug parser cache
debug parser cache enabled at level 1
```

```
hostname# show debug
1982769.770000000: parser cache: try to match 'show debug' in exec mode
1982769.770000000: parser cache: hit at index 8
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug vpn-sessiondb

To display VPN-session database debug information, use the **debug vpn-sessiondb** command in privileged EXEC mode. To disable the display of VPN-session database debug information, use the **no** form of this command.

debug vpn-sessiondb [*level*]

no debug vpn-sessiondb

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	Routed	Transparent	Single	Context	System
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables VPN-session database debug messages. The **show debug** command reveals that VPN-session database debug messages are enabled.

```
hostname# debug vpn-sessiondb
debug vpn-sessiondb  enabled at level 1
hostname# show debug
debug vpn-sessiondb  enabled at level 1
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug wccp

To enable logging of WCCP events, use the **debug wccp** command in privileged EXEC mode. To disable the logging of WCCP debug messages, use the **no** form of this command.

debug wccp {**events | packets | subblocks**}

no debug wccp {**events | packets | subblocks**}

Syntax Description

events	Enables logging of WCCP session events.
packets	Enables logging of debug messages about WCCP packet information.
subblocks	Enables logging of debug messages about WCCP subblocks.

Defaults

No default behavior or values.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	—	•

Command History

Release	Modification
7.2(1)	This command was introduced.

Usage Guidelines

The high priority assigned to debugging output can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables the logging of all WCCP session events:

```
hostname# debug wccp events
hostname#
```

The following example enables the logging of WCCP packet debug messages:

```
hostname# debug wccp packets
hostname#
```

The following example disables the logging of WCCP debug messages:

```
hostname# no debug wccp
```

```
hostname#
```

Related Commands

Command	Description
wccp	Enables support of WCCP.
show debug	Displays current debug configuration.

debug webvpn

To log WebVPN debug messages, use the **debug webvpn** command in privileged EXEC mode. To disable the logging of WebVPN debug messages, use the **no** form of this command.

debug webvpn [**chunk | cifs | citrix | failover | html | javascript | request | response | svc | transformation | url | util | xml**] [*level*]

no debug webvpn [**chunk | cifs | citrix | failover | html | javascript | request | response | svc | transformation | url | util | xml**] [*level*]

Syntax Description

chunk	Displays debug messages about memory blocks used to support WebVPN connections.
cifs	Displays debug messages about connections between CIFS)servers and WebVPN users.
citrix	Displays debug messages about connections between Citrix Metaframe Servers and Citrix ICA clients over WebVPN.
failover	Displays debug messages about equipment failovers affecting WebVPN connections.
html	Displays debug messages about HTML pages sent over WebVPN connections.
javascript	Displays debug messages about JavaScript sent over WebVPN connections.
request	Displays debug messages about requests issued over WebVPN connections.
response	Displays debug messages about responses issued over WebVPN connections.
svc	Displays debug messages about connections to SSL VPN clients over WebVPN.
transformation	Displays debug messages about WebVPN content transformation.
url	Displays debug messages about website requests issued over WebVPN connections.
util	Displays debug messages about CPU utilization dedicated to support connections to WebVPN remote users.
xml	Displays debug messages about JavaScript sent over WebVPN connections.
level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

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

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

Command History

Release	Modification
7.0(1)	This command was introduced.

Usage Guidelines

The high priority assigned to debugging output can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following example enables WebVPN debug messages, specifically for CIFS. The **show debug** command reveals that CIFS debug messages are enabled.

```
hostname# debug webvpn cifs
INFO: debug webvpn cifs enabled at level 1.
hostname# show debug
debug webvpn cifs enabled at level 1
hostname#
```

Related Commands

Command	Description
show debug	Displays current debug configuration.

debug xdmcp

To show debug messages for XDMCP application inspection, use the **debug xdmcp** command in privileged EXEC mode. To stop showing debug messages for XDMCP application inspection, use the **no** form of this command.

debug xdmcp [*level*]

no debug xdmcp [*level*]

Syntax Description

level	(Optional) Sets the debug message level to display, between 1 and 255. The default is 1. To display additional messages at higher levels, set the level to a higher number.

Defaults

The default value for *level* is 1.

Command Modes

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

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

Command History

Release	Modification
Preexisting	This command was preexisting.

Usage Guidelines

To see the current debug command settings, enter the **show debug** command. To stop the debug output, enter the **no debug** command. To stop all debug messages from being displayed, enter the **no debug all** command.

Note Enabling the **debug xdmcp** command may slow down traffic on busy networks.

Examples

The following example enables debug messages at the default level (1) for XDMCP application inspection:

```
hostname# debug xdmcp
```

Related Commands

Command	Description
class-map	Defines the traffic class to which to apply security actions.
inspect xdmcp	Enables XDMCP application inspection.
policy-map	Associates a class map with specific security actions.
service-policy	Applies a policy map to one or more interfaces.

debug xml

To display debugging information for the XML parser, use the **debug xml** command in privileged EXEC mode. To disable the display of debugging information, use the **no** form of this command.

debug xml [**element** | **event**]

no debug xml [**element** | **event**]

Syntax Description

element	(Optional) Displays debugging events that are related to processing individual XML elements.
event	(Optional) Displays XML parsing or error events.

Defaults

If no keywords are specified, all XML parser debugging messages are shown.

Command Modes

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

	Firewall Mode		Security Context		
				Multiple	
Command Mode	**Routed**	**Transparent**	**Single**	**Context**	**System**
Privileged EXEC	•	•	•	•	•

Command History

Release	Modification
8.0(2)	This command was introduced.

Usage Guidelines

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. Moreover, it is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Examples

The following is sample output from the **debug xml element** command:

```
hostname# debug xml element
debug xml element enabled at level 1

XML Executes cmd: hostname hostname
XML Executes cmd: domain-name example.com
XML Executes cmd: names
XML Executes cmd: dns-guard
XML Executes cmd: !
XML Executes cmd: interface Ethernet0
XML Executes cmd:  nameif outside
XML Executes cmd:  security-level 0
```

```
XML Executes cmd:  ip address 192.168.5.151 255.255.255.0 standby 192.168.5.152
XML Executes cmd: interface Ethernet1
XML Executes cmd:  nameif inside
XML Executes cmd:  security-level 100
XML Executes cmd:  ip address 192.168.0.151 255.255.255.0 standby 192.168.0.152
XML Executes cmd: !
XML Executes cmd: boot system flash:/f
XML Executes cmd: ftp mode passive
XML Executes cmd: clock timezone jst 9
XML Executes cmd: dns server-group DefaultDNS
XML Executes cmd:  domain-name cisco.com
_tcp_listen: could not query index for interface 65535 port 23
XML Executes cmd: pager lines 24
XML Executes cmd: logging console debugging
XML Executes cmd: logging buffered debugging
XML Executes cmd: mtu outside 1500
XML Executes cmd: mtu inside 1500
XML Executes cmd: failover
XML Executes cmd: no asdm history enable
XML Executes cmd: arp timeout 14000
XML Executes cmd: route outside 0.0.0.0 0.0.0.0 192.168.5.1 1
XML Executes cmd: timeout xlate 3:00:00
XML Executes cmd: timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 icmp 0:00:02
XML Executes cmd: timeout sunrpc 0:10:00 h323 0:05:00 h225 1:00:00 mgcp 0:05:00 mgcp-pat
0:05:00
XML Executes cmd: timeout sip 0:30:00 sip_media 0:02:00 sip-invite 0:03:00 sip-disconnect
0:02:00
XML Executes cmd: timeout uauth 0:05:00 absolute
XML Executes cmd: username user1 password mbO2jYs13AXlIAGa encrypted
XML Executes cmd: username sugi password EB3OP7Hu2hSu6x/7 encrypted
XML Executes cmd: http server enable
XML Executes cmd: http 0.0.0.0 0.0.0.0 outside
XML Executes cmd: no snmp-server location
XML Executes cmd: no snmp-server contact
XML Executes cmd: snmp-server enable traps snmp authentication linkup linkdown coldstart
XML Executes cmd: telnet timeout 5
XML Executes cmd: ssh timeout 5
XML Executes cmd: console timeout 0
XML Executes cmd: !
XML Executes cmd: class-map inspection_default
XML Executes cmd:  match default-inspection-traffic
XML Executes cmd: !
XML Executes cmd: !
XML Executes cmd: policy-map type inspect dns migrated_dns_map_1
XML Executes cmd:  parameters
XML Executes cmd:   message-length maximum 512
XML Executes cmd: policy-map global_policy
XML Executes cmd:  class inspection_default
XML Executes cmd:   inspect ftp
XML Executes cmd:   inspect h323 h225
XML Executes cmd:   inspect h323 ras
XML Executes cmd:   inspect netbios
XML Executes cmd:   inspect rsh
XML Executes cmd:   inspect rtsp
XML Executes cmd:   inspect skinny
XML Executes cmd:   inspect esmtp
XML Executes cmd:   inspect sqlnet
XML Executes cmd:   inspect sunrpc
XML Executes cmd:   inspect tftp
XML Executes cmd:   inspect sip
XML Executes cmd:   inspect xdmcp
XML Executes cmd: !
XML Executes cmd: service-policy global_policy global
XML error info: cmd-id 87 type info
```

```
XML Executes cmd: prompt hostname context
XML Executes cmd: crashinfo save disable
```

The following is sample output from the **debug xml event** command:

```
hostname# debug xml event
debug xml event enabled at level 1

XML parsing: data = <con... len = 3176
Exit XML parser, ret code = 0
```

Related Commands

Command	Description
show debug	Displays the debugging status for the various **debug** commands.