Network Powered Lighting

clear coap database

To clear the CoAP database, use the **clear coap database** command in user EXEC or privileged EXEC mode.

clear coap database

Command Default

This command has no arguments or keywords.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Example

This example shows how to clear the coap database:

```
Device(config)# clear coap database
```

clear macro auto configuration

To remove the macro applied configuration from the interfaces, use the **clear macro auto configuration** command.

Note Before executing the **clear macro auto configuration** command, you must disable Auto SmartPorts on the switch.

clear macro auto configuration {**all** | **interface** [*interface-id*]}

Syntax Description

all	Removes macro applied configuration from all the interfaces.
interface [*interface-id*]	Removes macro applied configuration from an interface.

Command Default

This command has no default setting.

Command Modes

User EXEC (>)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Use the command to remove configuration applied by macros from all the interfaces or a particular interface on the switch.

You can verify your settings by entering the **show macro auto interface** command in privileged EXEC mode.

Example

This example shows how to remove the configuration from all the switch interfaces:

```
Device(config)# clear macro auto configuration all
```

coap endpoint (coap-proxy configuration)

To configure the COAP Proxy to support multiple IPv4/IPv6 static-endpoints, use the **coap endpoint** command in coap-proxy configuration mode. To return to the default settings, use the **no** form of the command.

coap endpoint {**ipv4** | **ipv6**}[*ip-address*]
no coap endpoint {**ipv4** | **ipv6**}[*ip-address*]

Syntax Description

ipv4 *ip-address*	Specifies IPv4 static endpoint.
ipv6 *ip-address*	Specifies IPv6 static endpoint.

Command Modes

coap-proxy configuration (config-coap-proxy)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Example

This example show how to configure IPv4 static endpoint

```
Device(config)# endpoint ipv4 1.1.1.1
Device(config-coap-proxy)# transport tcp
```

debug coap

To enable debugging of the coap configurations, use the **debug coap** command in privileged EXEC mode.

debug coap {**all** | **database** | **errors** | **events** | **packet** | **trace** | **warnings**}

Syntax Description

all	Displays all coap debug messages.
database	Displays coap database debug messages.
errors	Displays coap error debug messages.
events	Displays coap event debug messages.
packet	Displays coap packet debug messages.
trace	Displays coap trace debug messages.
warnings	Displats coap warning debug messages

Command Default

This command has no arguments or keywords.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Example

The example shows how to enable debugging for coap database:

```
Device# debug coap database
```

device classifier

To enable the device classifier, use the **device classifier** command in global configuration mode. Use the **no** form of this command to disable the device classifier.

device classifier

no device classifier

Command Default

This command is disabled by default.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Use the **no device classifier** command, in global configuration mode, to disable the device classifier. You cannot disable the device classifier while it is being used by features such as Auto SmartPorts (ASP).

Example

This example shows how to enable the ASP device classifier on a switch:

```
Device(config)# device classifier
Device(config)# end
```

list (coap-proxy configuration)

To restrict the IP address range where the lights and their resources can be learnt, use the **list** command in coap-proxy configuration mode. To return to the default settings, use the **no** form of the command.

A maximum of five ip-lists can be configured, irrespective of ipv4 or ipv6, using the **list** command.

list {**ipv4** | **ipv6**}[*list-name*]
no list {**ipv4** | **ipv6**}[*list-name*]

Syntax Description

ipv4 *list-name*	Specifies IPv4 list name.
ipv6 *list-name*	Specifies IPv6 list name.

Command Modes

coap-proxy configuration (config-coap-proxy)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

To access coap-proxy configuration mode, enter the **coap proxy** command in global configuration mode.

Example

This example shows how to restrict the IPv4 address range using a list name.

```
Device(config)# coap proxy
Device(config-coap-proxy)# list ipv4 trial_list
```

macro

To apply a macro to an interface or to apply and debug a macro on an interface, use the **macro** command in interface configuration mode.

macro {**apply** | **trace**}*macro-name* [**parameter** {*value*}][**parameter** {*value*}][**parameter** {*value*}]

Syntax Description

apply	Applies a macro to an interface.
trace	Applies a macro to an interface and then debugs it.
macro-name	Specifies the name of the macro.
parameter *value*	(Optional) Specifies unique parameter values that are specific to the interface. You can enter up to three keyword-value pairs. Parameter keyword matching is case sensitive. All matching occurrences of the keyword are replaced with the corresponding value.

Command Default

This command has no default setting.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

You can use the **macro apply** *macro-name* command to apply and show the macros running on an interface.

You can use the **macro trace** *macro-name* command to apply and then debug the macro to find any syntax or configuration errors.

If a command fails because of a syntax error or a configuration error when you apply a macro, the macro continues to apply the remaining commands to the interface.

When creating a macro that requires the assignment of unique values, use the **parameter** *value* keywords to designate values specific to the interface.

Keyword matching is case sensitive. All matching occurrences of the keyword are replaced with the corresponding value. Any full match of a keyword, even if it is part of a larger string, is considered a match and is replaced by the corresponding value.

Some macros might contain keywords that require a parameter value. You can use the **macro apply** *macro-name* ? command to display a list of any required values in the macro. If you apply a macro without entering the keyword values, the commands are invalid and are not applied.

There are Cisco-default SmartPorts macros embedded in the switch software. You can display these macros and the commands that they contain by using the **show parser macro** command in user EXEC mode.

Follow these guidelines when you apply a Cisco-default SmartPorts macro on an interface:

- Display all macros on the switch by using the **show parser macro** command in user EXEC mode. Display the contents of a specific macro by using the **show parser macro** *macro-name* command in user EXEC mode.
- Keywords that begin with $ mean that a unique parameter value is required. Append the Cisco-default macro with the required values by using the **parameter** *value* keywords.

The Cisco-default macros use the $ character to identify required keywords. You can use the $ character to define keywords when you create a macro.

When you apply a macro to an interface, the macro name is automatically added to the interface. You can display the applied commands and macro names by using the **show running-config interface** *interface-id* command in user EXEC mode.

A macro applied to an interface range behaves the same way as a macro applied to a single interface. When you use an interface range, the macro is applied sequentially to each interface within the range. If a macro command fails on one interface, it is still applied to the remaining interfaces.

You can delete a macro-applied configuration on an interface by entering the **default interface** *interface-id* command in interface configuration mode.

Example

After you use the **macro name** command, in interface configuration mode, you can apply it to an interface. This example shows how to apply a user-created macro called duplex to an interface:

```
Device(config-if)# macro apply duplex
```

To debug a macro, use the **macro trace** command, in interface configuration mode, to find any syntax or configuration errors in the macro as it is applied to an interface.

```
Device(config-if)# macro trace duplex
Applying command...'duplex auto'
%Error Unknown error.
Applying command...'speed nonegotiate'
```

This example shows how to display the Cisco-default cisco-desktop macro and how to apply the macro and set the access VLAN ID to 25 on an interface:

```
Device# show parser macro cisco-desktop
-------------------------------------------------------------
Macro name : cisco-desktop
Macro type : default
# Basic interface - Enable data VLAN only
# Recommended value for access vlan (AVID) should not be 1
switchport access vlan $AVID
switchport mode access
# Enable port security limiting port to a single
# MAC address -- that of desktop
switchport port-security
switchport port-security maximum 1
# Ensure port-security age is greater than one minute
# and use inactivity timer
switchport port-security violation restrict
switchport port-security aging time 2
switchport port-security aging type inactivity
# Configure port as an edge network port
spanning-tree portfast
```

```
spanning-tree bpduguard enable
------------------------------------------------------------------
Device#
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# interface gigabitethernet1/0/4
Device(config-if)# macro apply cisco-desktop $AVID 25
```

macro auto

To configure and apply a global macro using the CLI, use the **macro auto** command in privileged EXEC mode.

Use the **no** form of this command to return to the default setting.

macro auto {**apply** | **config**} *macro-name*

Syntax Description

apply	Applies the macro.
config	Enters the macro parameters.
macro-name	Specifies the macro name.

Command Default

No macros are applied to the switch.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

To remove the macro from the switch, enter the **no** forms of the macro commands.

If you enter the **macro auto config** *macro-name* command, you are prompted to enter values for all the macro parameters.

Use the exact text string when entering the macro-name. The entries are case sensitive.

The user-defined values appear only in the **show macro auto** or **show running-config** command output.

Example

This example shows how to display global macros:

```
Device# macro auto apply ?
CISCO_SWITCH_AAA_ACCOUNTING         Configure aaa accounting parameters
CISCO_SWITCH_AAA_AUTHENTICATION     Configure aaa authentication parameters
CISCO_SWITCH_AAA_AUTHORIZATION      Configure aaa authorization parameters
CISCO_SWITCH_AUTO_IP_CONFIG         Configure the ip parameters
CISCO_SWITCH_AUTO_PCI_CONFIG        Configure PCI compliant parameters
CISCO_SWITCH_DOMAIN_NAME_CONFIG     Configure domain name
CISCO_SWITCH_ETHERCHANNEL_CONFIG    Configure the etherchannel parameters
CISCO_SWITCH_HOSTNAME_CONFIG        Configure hostname
CISCO_SWITCH_HTTP_SERVER_CONFIG     Configure http server
CISCO_SWITCH_LOGGING_SERVER_CONFIG  Configure logging server
CISCO_SWITCH_MGMT_VLAN_CONFIG       Configure management vlan parameters
CISCO_SWITCH_NAME_SERVER_CONFIG     Configure name server parameters
CISCO_SWITCH_NTP_SERVER_CONFIG      Configure NTP server
CISCO_SWITCH_RADIUS_SERVER_CONFIG   Configure radius server
CISCO_SWITCH_SETUP_SNMP_TRAPS       Configure SNMP trap parameters
CISCO_SWITCH_SETUP_USR_CONFIG       Configure the user parameters
CISCO_SWITCH_SNMP_SOURCE_CONFIG     Configure snmp source interface
```

```
CISCO_SWITCH_TACACS_SERVER_CONFIG   Configure tacacs server
CISCO_SWITCH_USER_PASS_CONFIG       Configure username and password

Device# macro auto config ?
CISCO_SWITCH_AAA_ACCOUNTING         Configure aaa accounting parameters
CISCO_SWITCH_AAA_AUTHENTICATION     Configure aaa authentication parameters
CISCO_SWITCH_AAA_AUTHORIZATION      Configure aaa authorization parameters
CISCO_SWITCH_AUTO_IP_CONFIG         Configure the ip parameters
CISCO_SWITCH_AUTO_PCI_CONFIG        Configure PCI compliant parameters
CISCO_SWITCH_DOMAIN_NAME_CONFIG     Configure domain name
CISCO_SWITCH_ETHERCHANNEL_CONFIG    Configure the etherchannel parameters
CISCO_SWITCH_HOSTNAME_CONFIG        Configure hostname
CISCO_SWITCH_HTTP_SERVER_CONFIG     Configure http server
CISCO_SWITCH_LOGGING_SERVER_CONFIG  Configure logging server
CISCO_SWITCH_MGMT_VLAN_CONFIG       Configure management vlan parameters
CISCO_SWITCH_NAME_SERVER_CONFIG     Configure name server parameters
CISCO_SWITCH_NTP_SERVER_CONFIG      Configure NTP server
CISCO_SWITCH_RADIUS_SERVER_CONFIG   Configure radius server
CISCO_SWITCH_SETUP_SNMP_TRAPS       Configure SNMP trap parameters
CISCO_SWITCH_SETUP_USR_CONFIG       Configure the user parameters
CISCO_SWITCH_SNMP_SOURCE_CONFIG     Configure snmp source interface
CISCO_SWITCH_TACACS_SERVER_CONFIG   Configure tacacs server
CISCO_SWITCH_USER_PASS_CONFIG       Configure username and password
```

This example shows how to display the parameters for a specific macro:

```
Device# macro auto config CISCO_SWITCH_AUTO_IP_CONFIG ?
CISCO_SWITCH_DOMAIN_NAME_CONFIG     domain name parameters
CISCO_SWITCH_LOGGING_SERVER_CONFIG  logging host parameters
CISCO_SWITCH_NAME_SERVER_CONFIG     name server parameters
CISCO_SWITCH_NTP_SERVER_CONFIG      ntp server parameters
LINE                                Provide parameters of form [Parameters
                                    name=value]
<cr>
```

```
Device# macro auto config CISCO_SWITCH_AUTO_PCI_CONFIG ?
CISCO_SWITCH_AAA_ACCOUNTING         aaa accounting parameters
CISCO_SWITCH_AAA_AUTHENTICATION     aaa authentication parameters
CISCO_SWITCH_AAA_AUTHORIZATION      aaa authorization parameters
CISCO_SWITCH_HTTP_SERVER_CONFIG     http server parameters
CISCO_SWITCH_RADIUS_SERVER_CONFIG   radius server parameters
CISCO_SWITCH_TACACS_SERVER_CONFIG   tacacs server parameters
LINE                                Provide parameters of form [Parameters
                                    name=value]
<cr>
```

```
Device# macro auto config CISCO_SWITCH_SETUP_SNMP_TRAPS ?
CISCO_SWITCH_SNMP_SOURCE_CONFIG     snmp source parameters
LINE                                Provide parameters of form [Parameters
                                    name=value]
<cr>
```

```
Device# macro auto config CISCO_SWITCH_SETUP_USR_CONFIG ?CISCO_AUTO_TIMEZONE_CONFIG timezone
 parameters
CISCO_SWITCH_HOSTNAME_CONFIG        hostname parameter
LINE                                Provide parameters of form [Parameters
                                    name=value]
<cr>
```

This example shows how to set macro parameters and apply the macro using the CLI:

```
Device# macro auto config CISCO_SWITCH_ETHERCHANNEL_CONFIG
Enter the port channel id[1-48] for 3K & 2350,[1-6] for 2K: 2
Enter the port channel type, Layer:[2-3(L3 not supported on 2K)]: 2
Enter etherchannel mode for the interface[auto/desirable/on/active/passive]: active
Enter the channel protocol[lacp/none]: lacp
Enter the number of interfaces to join the etherchannel[8-PAGP/MODE:ON,16-LACP]: 7
Enter interface name[GigabitEthernet3/0/3]: gigabitethernet1/0/1
Enter interface name[GigabitEthernet3/0/3]: gigabitethernet1/0/2
Enter interface name[GigabitEthernet3/0/3]: gigabitethernet1/0/3
Enter interface name[GigabitEthernet3/0/3]: gigabitethernet1/0/4
Enter interface name[GigabitEthernet3/0/3]: gigabitethernet1/0/5
Enter interface name[GigabitEthernet3/0/3]: gigabitethernet1/0/6
Enter interface name[GigabitEthernet3/0/3]: gigabitethernet1/0/7
Do you want to apply the parameters? [yes/no]: yes
Enter configuration commands, one per line. End with CNTL/Z.
Enter configuration commands, one per line. End with CNTL/Z.
Enter configuration commands, one per line. End with CNTL/Z.
Enter configuration commands, one per line. End with CNTL/Z.
Enter configuration commands, one per line. End with CNTL/Z.
Enter configuration commands, one per line. End with CNTL/Z.
Enter configuration commands, one per line. End with CNTL/Z.
Device# macro auto apply CISCO_SWITCH_ETHERCHANNEL_CONFIG
Enter configuration commands, one per line. End with CNTL/Z.
Device#
```

macro auto apply (Cisco IOS shell scripting capability)

To configure and apply a global macro using the Cisco IOS shell scripting capability, use the **macro auto apply** command in privileged EXEC mode. Use the **no** form of this command to return to the default setting.

macro auto apply *macro-name*

Syntax Description

apply	Applies the macro.
macro-name	Specifies the macro name.

Command Default

No macros are applied to the switch.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

To remove the macro from the switch, enter the **no** forms of the macro commands.

Use the exact text string when entering the *macro-name*. The entries are case sensitive.

The user-defined values appear only in the **show macro auto** or **show running-config** command output.

You can also use the Cisco IOS shell scripting capability to set the parameters. For examples, see the

"Configuring and Applying Global Macros" section in the "Configuring Auto Smartports and Static Smartports Macros" chapter.

Example

This example shows how to display global macros:

```
Device# macro auto apply ?

CISCO_SWITCH_AAA_ACCOUNTING         Configure aaa accounting parameters
CISCO_SWITCH_AAA_AUTHENTICATION     Configure aaa authentication parameters
CISCO_SWITCH_AAA_AUTHORIZATION      Configure aaa authorization parameters
CISCO_SWITCH_AUTO_IP_CONFIG         Configure the ip parameters
CISCO_SWITCH_AUTO_PCI_CONFIG        Configure PCI compliant parameters
CISCO_SWITCH_DOMAIN_NAME_CONFIG     Configure domain name
CISCO_SWITCH_ETHERCHANNEL_CONFIG    Configure the etherchannel parameters
CISCO_SWITCH_HOSTNAME_CONFIG        Configure hostname
CISCO_SWITCH_HTTP_SERVER_CONFIG     Configure http server
CISCO_SWITCH_LOGGING_SERVER_CONFIG  Configure logging server
CISCO_SWITCH_MGMT_VLAN_CONFIG       Configure management vlan parameters
CISCO_SWITCH_NAME_SERVER_CONFIG     Configure name server parameters
CISCO_SWITCH_NTP_SERVER_CONFIG      Configure NTP server
CISCO_SWITCH_RADIUS_SERVER_CONFIG   Configure radius server
CISCO_SWITCH_SETUP_SNMP_TRAPS       Configure SNMP trap parameters
CISCO_SWITCH_SETUP_USR_CONFIG       Configure the user parameters
CISCO_SWITCH_SNMP_SOURCE_CONFIG     Configure snmp source interface
```

```
CISCO_SWITCH_TACACS_SERVER_CONFIG   Configure tacacs server
CISCO_SWITCH_USER_PASS_CONFIG       Configure username and password
```

macro auto config (Cisco IOS shell scripting capability)

To configure and apply a global macro, use the **macro auto config** command in privileged EXEC mode. Use the **no** form of this command to return to the default setting.

macro auto config *macro-name* [*parameter=value* [*parameter=value*]**...**]

Syntax Description

config	Enters the macro parameters.
macro-name	Specifies the macro name.
parameter=value [parameter=value] ...	*parameter=value*—Replaces values for global macro parameter values. Enter values in the form of name value pair separated by a space: <name1>=<value1> [<name2>=<value2>...]

Command Default

No macros are applied to the switch.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

To remove the macro from the switch, enter the **no** forms of the macro commands.

If you enter the **macro auto config** *macro-name* command, you are prompted to enter values for all the macro parameters.

Use the exact text string when entering the *macro-name* and *parameters*. The entries are case sensitive.

The user-defined values appear only in the **show macro auto** or **show running-config** command output.

You can also use the Cisco IOS shell scripting capability to set the parameters. For examples, see the "Configuring and Applying Global Macros" section in the "Configuring Auto Smartports and Static Smartports Macros" chapter.

macro auto control

To specify when the switch applies an Auto Smartports macro based on the detection method, device type, or trigger (referred to as event trigger control), use the **macro auto control** command in interface configuration mode. Use the **no** form of this command to disable trigger-to-macro mapping. The switch then does not apply macros based on event triggers.

macro auto control {**detection** [**cdp**] [**lldp**] [**mac-address**] | **device** [**ip-camera**] [**media-player**] [**phone**] [**lightweight-ap**] [**access-point**] [**router**] [**switch**] | **trigger** [**last-resort**]}
no macro auto control {**detection** [**cdp**] [**lldp**] [**mac-address**] | **device** [**ip-camera**] [**media-player**] [**phone**] [**lightweight-ap**] [**access-point**] [**router**] [**switch**] | **trigger** [**last-resort**]}

Syntax Description

detection [cdp] [lldp] [mac-address]	detection—Sets one or more of these as an event trigger: • (Optional) **cdp**—CDP messages • (Optional) **lldp**—LLDP messages • (Optional) **mac-address**—User-defined MAC address groups
device [access-point] [ip-camera] [lightweight-ap] [media-player] [phone] [router] [switch]	device—Sets one or more of these devices as an event trigger: • (Optional) **access-point**—Autonomous access point • (Optional) **ip-camera**—Cisco IP video surveillance camera • (Optional) **lightweight-ap**—Lightweight access point • (Optional) **media-player**—Digital media player • (Optional) **phone**—Cisco IP phone • (Optional) **router**—Cisco router • (Optional) **switch**—Cisco switch
trigger [last-resort]	trigger—Sets a specific event trigger. • (Optional) **last-resort**—Last-resort trigger.

Command Default

The switch uses the device type as the event trigger. If the switch cannot determine the device type, it uses MAC address groups, MAB messages, 802.1x authentication messages, and LLDP messages in random order.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

If you do not set event triggers, the switch uses the device type as the event trigger. If the switch cannot determine the device type, it uses MAC address groups, MAB messages, 802.1x authentication messages, and LLDP messages in random order.

To verify that a macro is applied to an interface, use the **show macro auto interface** command in user EXEC mode.

Example

This example shows how to set LLDP messages and MAC address groups as event triggers:

```
Device(config)# interface gigabitethernet 5/0/2
Device(config-if)# macro auto control detection lldp mac-address
Device(config-if)# exit
Device(config)# end
```

This example shows how to set access points, video surveillance cameras, and digital media players as event triggers:

Note The switch applies a built-in macro only when it detects an access point, video surveillance camera, or digital media player.

```
Device(config)# interface gigabitethernet 5/0/1
Device(config-if)# macro auto control device access-point ip-camera media-player
Device(config-if)# exit
Device(config)# end
```

macro auto execute

To replace built-in macro default values and to configure mapping from an event trigger to a built-in or user-defined macro, use the **macro auto execute** command in global configuration mode.

macro auto execute *event trigger* {**builtin** *built-in macro* | **remote** *url*} {*parameter=value*} {*function contents*}
no macro auto execute *event trigger* {**builtin** *built-in macro* | **remote** *url*} {*parameter=value*} {*function contents*}

Syntax Description

event trigger	Defines mapping from an event trigger to a built-in macro. Specifies an event trigger: • CISCO_CUSTOM_EVENT • CISCO_DMP_EVENT • CISCO_IPVSC_EVENT • CISCO_LAST_RESORT_EVENT • CISCO_PHONE_EVENT • CISCO_ROUTER_EVENT • CISCO_SWITCH_EVENT • CISCO_WIRELESS_AP_EVENT • CISCO_WIRELESS_LIGHTWEIGHT_AP_EVENT • WORD—Apply a user-defined event trigger such as a MAC address group

builtin *built-in macro name*	(Optional) Specifies a builtin built-in macro name: • CISCO_AP_AUTO_SMARTPORT Specify the parameter value: NATIVE_VLAN=1 • CISCO_DMP_AUTO_SMARTPORT Specify the parameter value: ACCESS_VLAN=1. • CISCO_IPVSC_AUTO_SMARTPORT Specify the parameter value: ACCESS_VLAN=1. • CISCO_LWAP_AUTO_SMARTPORT Specify the parameter value: ACCESS_VLAN=1. • CISCO_PHONE_AUTO_SMARTPORT Specify the parameter values: ACCESS_VLAN=1 and VOICE_VLAN=2. • CISCO_ROUTER_AUTO_SMARTPORT Specify the parameter value: NATIVE_VLAN=1. • CISCO_SWITCH_AUTO_SMARTPORT Specify the parameter value: NATIVE_VLAN=1.
parameter=value	(Optional) *parameter=value*—Replaces default values for parameter values shown for the *bultin-macro name*, for example, ACCESS_VLAN=1. Enter new values in the form of name value pair separated by a space: [<name1>=<value1> <name2>=<value2>...].
{function contents}	(Optional) *{function contents}*— Specifies a user-defined macro to associate with the trigger. Enter the macro contents within braces. Begin the Cisco IOS shell commands with the left brace and end the command grouping with the right brace.

remote *url*	(Optional) Specifies a remote server location: • The syntax for the local flash file system on the standalone switch or the stack's active switch: **flash**: The syntax for the local flash file system on a stack member: **flash** *member number*: The syntax for the FTP: **ftp**:[[//*username*[:*password*]@*location*]/*directory*]/*filename* The syntax for an HTTP server: **http**://[[*username*:*password*]@]{*hostname* | *host-ip*}[/*directory*]/*filename* The syntax for a secure HTTP server: **https**://[[*username*:*password*]@]{*hostname* | *host-ip*}[/*directory*]/*filename* The syntax for the NVRAM: **nvram**://[[*username*:*password*]@][/*directory*]/*filename* The syntax for the Remote Copy Protocol (RCP): **rcp**:[[//*username@location*]/*directory*]/*filename* The syntax for the Secure Copy Protocol (SCP): **scp**:[[//*username@location*]/*directory*]/*filename* The syntax for the TFTP: **tftp**:[[//*location*]/*directory*]/*filename*

Command Default

None

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Use the **macro auto execute** command to replace the built-in macro default values with values that are specific to your switch.

The switch automatically maps from event triggers to built-in macros. The built-in macros are system-defined macros in the software image. You can also create user-defined macros by using the Cisco IOS shell scripting capability.

You can create new event triggers by using the **shell trigger** commands in global configuration mode. Use the **show shell triggers** command in privileged EXEC to display the contents of the user-defined triggers and macros.

You can use the **macro auto mac-address-group** command in global configuration mode to create event triggers for devices that do not support Cisco Discovery Protocol (CDP) or Link Layer Discovery Protocol (LLDP).

You can use the remote macro feature to store macros in a central location for designated network switches to use. You can then maintain and update the macro files for use by multiple switches. Use **remote** *url* to configure the remote server location and macro path information. There are no specific file extension requirements for saved macro files.

Auto Smartports macros and antimacros (the antimacro is the portion of the applied macro that removes it at link down) have these guidelines and limitations:

- You can delete or change the built-in macros. However, you can override a built-in macro by creating a user-defined macro with the same name. To restore the original built-in macro, delete the user-defined macro.
- If you enable both the **macro auto device** and the **macro auto execute** commands, the parameters specified in the command last executed are applied to the switch. Only one command is active on the switch.
- To avoid system conflicts when macros are applied, remove all port configurations except for 802.1x authentication.
- Do not configure port security when enabling Auto SmartPorts on the switch.
- If the macro conflicts with the original configuration, either the macro does not apply some of the original configuration commands, or the antimacro does not remove them. (The antimacro is the portion of the applied macro that removes the macro at a link-down event.)
- For example, if 802.1x authentication is enabled, you cannot remove the switchport-mode access configuration. Remove the 802.1x authentication before removing the switchport mode configuration.
- A port cannot be a member of an EtherChannel when you apply Auto SmartPorts macros.
- The built-in-macro default data VLAN is VLAN 1. The default voice VLAN is VLAN 2. If your switch uses different access, native, or voice VLANs, use the **macro auto device** or the **macro auto execute** commands to configure the values.
- For 802.1x authentication or MAC authentication bypass (MAB), to detect non-Cisco devices, configure the RADIUS server to support the Cisco attribute-value pair **auto-smart-port=***event trigger*
- The switch supports Auto SmartPort macros only on directly connected devices. Multiple device connections, such as hubs, are not supported.
- If authentication is enabled on a port, the switch ignores a MAC address trigger if authentication fails.
- The order of CLI commands within the macro and the corresponding antimacro can be different.

.

Example

This example shows how to use two built-in macros for connecting Cisco switches and Cisco IP phones to the switch. This example modifies the default voice VLAN, access VLAN, and native VLAN for the trunk interface:

```
Device(config)# !!! the next command modifies the access and voice vlans
Device(config)# !!! for the built in Cisco IP phone auto smartport macro
Device(config)# macro auto execute CISCO_PHONE_EVENT builtin CISCO_PHONE_AUTO_SMARTPORT
ACCESS_VLAN=10 VOICE_VLAN=20
Device(config)# !!! the next command modifies the Native vlan used for inter switch trunks
```

```
Device(config)# macro auto execute CISCO_SWITCH_EVENT builtin CISCO_SWITCH_AUTO_SMARTPORT
NATIVE_VLAN=10
Device(config)# !!! the next command enables auto smart ports globally
Device(config)# macro auto global processing
Device(config)# exit
Device# !!! here is the running configuration of the interface connected
Device# !!! to another Cisco Switch after the Macro is applied
Device# show running-config interface gigabitethernet1/0/1
Building configuration...

Current configuration : 284 bytes
!
interface GigabitEthernet1/0/1
 switchport trunk encapsulation dot1q
 switchport trunk native vlan 10
 switchport mode trunk
 srr-queue bandwidth share 10 10 60 20
 queue-set 2
 priority-queue out
 mls qos trust cos
 auto qos voip trust
 macro description CISCO_SWITCH_EVENT
end
```

This example shows how to map a user-defined event trigger called media player to a user-defined macro

1. Connect the media player to an 802.1x- or MAB-enabled switch port.
2. On the RADIUS server, set the attribute-value pair to auto-smart-port=DMP_EVENT
3. On the switch, create the event trigger DMP_EVENT, and enter the user-defined macro commands.
4. The switch recognizes the attribute-value pair=DMP_EVENT response from the RADIUS server and applies the macro associated with this event trigger.

```
Device(config)# shell trigger DMP_EVENT mediaplayer
Device(config)# macro auto execute DMP_EVENT {
if [[ $LINKUP == YES ]]; then
conf t
 interface $INTERFACE
   macro description $TRIGGER
   switchport access vlan 1
   switchport mode access
   switchport port-security
   switchport port-security maximum 1
   switchport port-security violation restrict
   switchport port-security aging time 2
   switchport port-security aging type inactivity
   spanning-tree portfast
   spanning-tree bpduguard enable
   exit
fi
if [[ $LINKUP == NO ]]; then
conf t
interface $INTERFACE
     no macro description $TRIGGER
     no switchport access vlan 1
     if [[ $AUTH_ENABLED == NO ]]; then
        no switchport mode access
     fi
```

```
     no switchport port-security
     no switchport port-security maximum 1
     no switchport port-security violation restrict
     no switchport port-security aging time 2
     no switchport port-security aging type inactivity
     no spanning-tree portfast
     no spanning-tree bpduguard enable
     exit
fi
```

Table 1: Supported Cisco IOS Shell Keywords

Command	Description
{	Begin the command grouping.
}	End the command grouping.
[[	Use as a conditional construct.
]]	Use as a conditional construct.
else	Use as a conditional construct.
==	Use as a conditional construct.
fi	Use as a conditional construct.
if	Use as a conditional construct.
then	Use as a conditional construct.
-z	Use as a conditional construct.
$	Variables that begin with the $ character are replaced with a parameter value.
#	Use the # character to enter comment text.

Table 2: Unsupported Cisco IOS Shell Reserved Keywords

Command	Description
|	Pipeline.
case	Conditional construct.
esac	Conditional construct.
for	Looping construct.
function	Shell function.
in	Conditional construct.
select	Conditional construct.

Command	Description
time	Pipeline.
until	Looping construct.
while	Looping construct.

macro auto global control

To specify when the switch applies an Auto Smartports macro based on the device type or trigger (referred to as event trigger control), use the **macro auto global control** command in global configuration mode. Use the **no** form of this command to disable trigger-to-macro mapping.

macro auto global control {**detection** [**cdp**] [**lldp**][**mac-address**] | **device** [**access-point**] [**ip-camera**] [**lightweight-ap**] [**media-player**] [**phone**] [**router**] [**switch**] | **trigger** [**last-resort**]}
no macro auto global control {**detection** [**cdp**] [**lldp**] [**mac-address**] | **device** [**access-point**] [**ip-camera**] [**lightweight-ap**] [**media-player**] [**phone**] [**router**] [**switch**] | **trigger** [**last-resort**]}

Syntax Description

detection [cdp] [lldp] [mac-address]	detection—Sets one or more of these as an event trigger: • (Optional) **cdp**—CDP messages • (Optional) **lldp**—LLDP messages • (Optional) **mac-address**—User-defined MAC address groups
device [access-point] [ip-camera] [lightweight-ap] [media-player] [phone] [router] [switch]	device—Sets one or more of these devices as an event trigger: • (Optional) **access-point**—Autonomous access point • (Optional) **ip-camera**—Cisco IP video surveillance camera • (Optional) **lightweight-ap**—Lightweight access point • (Optional) **media-player**—Digital media player • (Optional) **phone**—Cisco IP phone • (Optional) **router**—Cisco router • (Optional) **switch**—Cisco switch
trigger [last-resort]	trigger—Sets a specific event trigger. • (Optional) **last-resort**—Last-resort trigger.

Command Default

The switch uses the device type as the event trigger. If the switch cannot determine the device type, it uses MAC address groups, MAB messages, 802.1x authentication messages, and LLDP messages in random order.

Command Modes Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines If you do not set event triggers, the switch uses the device type as the event trigger. If the switch cannot determine the device type, it uses MAC address groups, MAB messages, 802.1x authentication messages, and LLDP messages in random order.

To verify that a macro is applied to a switch, use the **show macro auto global** command in user EXEC mode.

Example

This example shows how to set CDP messages, LLDP messages and MAC address groups as event triggers:

```
Device(config)# macro auto global control detection cdp lldp mac-address
Device(config)# end
```

This example shows how to set autonomous access points, lightweight access points, and IP phones:

```
Device(config)# macro auto global control device access-point lightweight-ap phone
Device(config)# end
```

macro auto global processing

To enable Auto SmartPorts macros on the switch, use the **macro auto global processing** command in global configuration mode. Use the **no** form of this command to disable the macros.

macro auto global processing

no macro auto global processing

Command Default

Auto Smartports is disabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Use the **macro auto global processing** command to globally enable macros on the switch. To disable macros on a specific port, use the **no macro auto processing** command in interface mode.

When using 802.1x or MAB authentication, you need to configure the RADIUS server to support the Cisco attribute-value pair **auto-smart-port**=*event trigger*. If authentication fails, the macro is not applied. If the 802.1x or MAB authentication fails on the interface, the switch does not use the fallback CDP event trigger.

When CDP-identified devices advertise multiple capabilities, the switch chooses a capability first by switch and then by router.

To verify that a macro is applied to an interface, use the **show macro auto interface**command in privileged EXEC mode.

Example

This example shows how to enable Auto SmartPorts on the switch and to disable the feature on a specific interface:

```
Device(config)# macro auto global processing
Device(config)# interface gigabitethernet 0/1
Device(config-if)# no macro auto processing
Device(config-if)# exit
Device(config)#
```

macro auto mac-address-group

To create an event trigger for devices that do not support Cisco Discovery Protocol (CDP) or Link Layer Discover Protocol (LLDP), use the **macro auto mac-address-group** command in global configuration mode. Use the **no** form of this command to delete the group.

macro auto mac-address-group *name* {**mac-address list** *list* | **oui** {**list** *list* | **range** *start-value* **size** *number*}}
no macro auto mac-address-group *name* {**mac-address list** *list* | **oui** {**list** *list* | **range** *start-value* **size** *number*}}

Syntax Description

name	Specifies the group name.
ui	(Optional) Specifies an operationally unique identifier (OUI) **list** or **range**. • **list**—Enter an OUI list in hexadecimal format separated by spaces. • **range**—Enter the starting OUI hexadecimal value (*start-value*). • **size**—Enter the length of the range (number) from 1 to 5 to create a list of sequential addresses.
mac-address list *list*	(Optional) Configures a list of MAC addresses separated by a space.

Command Default

No groups are defined.

Command Modes

Group configuration (config-addr-grp-mac)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Use the **macro auto mac-address-group** command to create an event trigger for devices that do not support CDP or LLDP. Use the MAC address group as a trigger to map to a built-in or user-defined macro by using the **macro auto execute** command. At link-up the switch detects the device type and applies the specified macro.

The switch supports up to ten MAC address groups. Each group can have up to 32 OUI and 32 MAC configured addresses.

Example

This example shows how to create a MAC-address-group event trigger called *address_trigger* and how to verify your entries:

```
Device(config)# macro auto mac-address-group mac address_trigger
Device(config-addr-grp-mac)# mac-address list 2222.3333.3334 22.33.44 a.b.c
Device(config-addr-grp-mac)# oui list 455555 233244
```

```
Device(config-addr-grp-mac)# oui range 333333 size 2
Device(config-addr-grp-mac)# exit
Device(config)# end
Device# show running configuration
!
!macro auto mac-address-group address_trigger
 oui list 333334
 oui list 333333
 oui list 233244
 oui list 455555
 mac-address list 000A.000B.000C
 mac-address list 0022.0033.0044
 mac-address list 2222.3333.3334
!

<output truncated>
```

macro auto processing

To enable Auto SmartPorts macros on an interface, use the **macro auto processing** command in interface configuration mode. Use the no form of this command to disable the macros.

macro auto processing

no macro auto processing

Command Default

Auto SmartPorts is disabled.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Use the **macro auto processing** command, in interface configuration mode, to enable macros on a specific interface. To disable macros on a specific interface, use the no macro auto processing command, in interface configuration mode.

A port cannot be a member of an EtherChannel when you apply Auto SmartPorts macros. If you use EtherChannels, disable Auto SmartPorts on the EtherChannel interface by using the **no macro auto processing** command. The EtherChannel interface applies the configuration to the member interfaces.

To verify that a macro is applied to an interface, use the **show macro auto interface** command in privileged EXEC mode.

Example

This example shows how to enable Auto SmartPorts on the switch and to disable the feature on a specific interface:

```
Device(config)# interface gigabitethernet 0/1
Device(config-if)# no macro auto processing
Device(config-if)# exit
Device(config)# macro auto global processing
```

macro auto sticky

To configure macros to remain active after a link-down event, referred to as macro persistence, use the **macro auto sticky** command in global configuration mode. Use the **no** form of this command to disable the macro persistence.

macro auto sticky
no macro auto sticky

Command Default

Macro persistence is disabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Use the **macro auto sticky** command so that macros remain active after a link-down event.

Example

This example shows how to enable macro persistence on an interface:

```
Device(config)# interface gigabitethernet 5/0/2
Device(config-if)# macro auto port sticky
Device(config-if)# exit
Device(config)# end
```

macro auto trigger

To enter the configure-macro-trigger mode and define a trigger for a device that has no built-in trigger and associate the trigger with a device or profile, use the **macro auto trigger** command in global configuration mode. To remove the user-defined trigger, use the **no** form of this command.

macro auto trigger *trigger_name* {**device** | **exit** | **no** | **profile**}
no macro auto trigger *trigger_name* {**device** | **exit** | **no** | **profile**}

Syntax Description

trigger_name	Specifies a trigger to be associated with the device type or profile name.
device	Specifies a device name to map to the named trigger.
exit	Exits device group configuration mode.
no	Removes any configured device.
profile	Specifies a profile name to map to the named trigger.

Command Default

No user-defined triggers are configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

If a device is classified by the Device Classifier, but does not have a built-in trigger defined, use the **macro auto trigger** command, in global configuration mode, to define a trigger based on a device name or a profile name. After you enter the command, the switch is in the configure-macro-trigger mode and the **device**, **exit**, **no**, and **profile** keywords are visible. In this mode, you can provide a device name or a profile name to map to the trigger. It is not necessary to map the trigger to both a device name and a profile name. If you map the trigger to both names, the trigger-to-profile name mapping has preference for macro application.

You must use this command to configure a trigger when you configure a user-defined macro. The trigger name is required for the custom macro configuration.

After the device is profiled, you must add the complete string to the device-group database.

Example

This example shows how to configure a user-defined trigger for a profile called DMP_EVENT mediaplayer for use with a media player that has no built-in trigger:

```
Device(config)# macro auto trigger DMP
Device(config-macro-trigger)# profile mediaplayer-DMP
Device(config-macro-trigger)# exit
```

macro description

To enter a description about which macros are applied to an interface, use the **macro description** command in interface configuration mode. Use the **no** form of this command to remove the description. This command is mandatory for Auto SmartPorts to work.

macro description *text*
no macro description *text*

Syntax Description

description *text*	Enters a description about the macros that are applied to the specified interface.

Command Default

This command has no default setting.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Use the **description** keyword to associate comment text or the macro name with an interface. When multiple macros are applied on a single interface, the description text is from the last applied macro.

You can verify your settings by entering the **show parser macro description** command in privileged EXEC mode.

Example

This example shows how to add a description to an interface:

```
Device(config-if)# macro description duplex settings
```

macro global

To apply a macro to a switch or to apply and debug a macro on a switch, use the **macro global** command in global configuration mode.

macro global {**apply** | **trace**} *macro-name* [**parameter** {*value*}][**parameter** {*value*}][**parameter** {*value*}] parameter

Syntax Description

apply	Applies a macro to the switch.
trace	Applies a macro to a switch and debugs the macro.
macro-name	Specifies the name of the macro.
parameter *value*	(Optional) Specifies unique parameter values that are specific to the switch. You can enter up to three keyword-value pairs. Parameter keyword matching is case sensitive. All matching occurrences of the keyword are replaced with the corresponding value.

Command Default

This command has no default setting.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Note You can delete a global macro-applied configuration on a switch only by entering the no version of each command in the macro.

Use the **macro global apply** *macro-name* command to apply the macro to an interface.

Use the **macro global trace** *macro-name* command to apply and then debug the macro to find any syntax or configuration errors.

If a command fails when you apply a macro because of a syntax error or a configuration error, the macro continues to apply the remaining commands to the switch.

When creating a macro that requires the assignment of unique values, use the **parameter** *value* keywords to designate values specific to the switch.

Keyword matching is case sensitive. All matching occurrences of the keyword are replaced with the corresponding value. Any full match of a keyword, even if it is part of a larger string, is considered a match and is replaced by the corresponding value.

Some macros might contain keywords that require a parameter value. You can use the **macro global apply** *macro-name* ? command to display a list of any required values in the macro. If you apply a macro without entering the keyword values, the commands are invalid and are not applied.

There are Cisco-default Smartports macros embedded in the switch software. You can display these macros and the commands they contain by using the **show parser macro** command in user EXEC mode.

Follow these guidelines when you apply a Cisco-default Smartports macro on a switch:

- Display all macros on the switch by using the **show parser macro** command. Display the contents of a specific macro by using the **show parser macro name** *macro-name* command.
- Keywords that begin with $ mean that a unique parameter value is required. Append the Cisco-default macro with the required values by using the **parameter** *value* keywords.

  The Cisco-default macros use the $ character to help identify required keywords. There is no restriction on using the $ character to define keywords when you create a macro.

When you apply a macro to a switch, the macro name is automatically added to the switch. You can display the applied commands and macro names by using the **show running-config** command.

Example

After you have created a new macro by using the **macro auto execute** command, you can apply it to a switch. This example shows how to view the **snmp** macro, how to apply the macro, set the hostname to test-server, and set the IP precedence value to 7:

```
Device# show parser macro name snmp
Macro name : snmp
Macro type : customizable

#enable port security, linkup, and linkdown traps
snmp-server enable traps port-security
snmp-server enable traps linkup
snmp-server enable traps linkdown
#set snmp-server host
snmp-server host ADDRESS
#set SNMP trap notifications precedence
snmp-server ip precedence VALUE

----------------------------------------------------
Switch(config)# macro global apply snmp ADDRESS test-server VALUE 7
```

To debug a macro, use the **macro global trace** command to find any syntax or configuration errors in the macro when you apply it to a switch. In this example, the **ADDRESS** parameter value was not entered, the **snmp-server host** command failed, and the remainder of the macro is applied to the switch:

```
Device(config)# macro global trace snmp VALUE 7
Applying command...‘snmp-server enable traps port-security’
Applying command...‘snmp-server enable traps linkup’
Applying command...‘snmp-server enable traps linkdown’
Applying command...‘snmp-server host’
%Error Unknown error.
Applying command...‘snmp-server ip precedence 7’
```

macro global description

To enter a description about the macros that are applied to a switch, use the **macro global description** command in global configuration mode. Use the **no** form of this command to remove the description.

macro global description *text*

no macro global description *text*

Syntax Description

description *text*	Enters a description about the macros that are applied to the switch.

Command Default

This command has no default setting.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Use the **description** keyword to associate comment text or the macro name with a switch. When multiple macros are applied on a switch, the description text is from the last applied macro.

You can verify your settings by entering the **show parser macro description** command in privileged EXEC mode.

Example

This example shows how to add a description to a switch:

```
Device(config)# macro global description udld aggressive mode enabled
```

max-endpoints (coap-proxy configuration)

To specify the maximum number of endpoints that can be learnt on the device, use the **max-endpoints** command in coap-proxy configuration mode. To return to the default settings, use the **no** form of the command.

max-endpoints *number*
no max-endpoints

Syntax Description

number	Range is from 1 to 500

Command Default

The default number of endpoints is 10.

Command Modes

coap-proxy configuration (config-coap-proxy)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

To access coap-proxy configuration mode, enter the **coap proxy** command in global configuration mode.

Example

This example shows how to specify maximum endpoints as 12 that can be learnt on the device.

```
Device(config)# coap proxy
Device(config-coap-proxy)# max-endpoints 12
```

port-dtls (coap-proxy configuration)

To configure a Datagram Transport Layer Security (DTLS) port, use the **port-dtls** command in coap-proxy configuration mode. To return to the default settings, use the **no** form of the command.

port-dtls *number*
no port-dtls

Syntax Description

number	Range is from 1 to 65000.

Command Default

The default port is 5683.

Command Modes

coap-proxy configuration (config-coap-proxy)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

To access coap-proxy configuration mode, enter the **coap proxy** command in global configuration mode.

Example

This example shows how to configure a dtls port .

```
Device(config)# coap proxy
Device(config-coap-proxy)# port-dtls 5899
```

port-unsecure (coap-proxy configuration)

To configure a port, use the **port-unsecure** command in coap-proxy configuration mode. To return to the default settings, use the **no** form of the command.

port-unsecure *number*
no port-dtls

Syntax Description

number	Range is from 1 to 65000.

Command Default

The default port is 5683.

Command Modes

coap-proxy configuration (config-coap-proxy)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

To access coap-proxy configuration mode, enter the **coap proxy** command in global configuration mode.

Example

This example shows how to configure a port .

```
Device(config)# coap proxy
Device(config-coap-proxy)# port-unsecure 5899
```

resource directory (coap-proxy configuration)

To unicast upstream resource directory server to which the switch can act as a COAP client, use the **resource directory** command in coap-proxy configuration mode. To return to the default settings, use the **no** form of the command.

A maximum of five ip-lists can be configured, for each ipv4 or ipv6, using the resource directory command.

resource directory {**ipv4** | **ipv6**}[*ip-address*]
no resource directory

Syntax Description

ipv4 *ip-address*	Specifies IPv4 address.
ipv6 *ip-address*	Specifies IPv6 address.

Command Modes

coap-proxy configuration (config-coap-proxy)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

To access coap-proxy configuration mode, enter the **coap proxy** command in global configuration mode.

Example

This example shows how to unicast upstream resource directory server to which the switch can act as a COAP client.

```
Device(config)# coap proxy
Device(config-coap-proxy)# resource-directory ipv4 192.168.1.1
```

security (coap-proxy configuration)

To configure CoAP security features, use the **security** command in coap-proxy configuration mode. To return to the default settings, use the **no** form of the command.

security {**none** [{**ipv4** { *ip-address ip-mask/prefix*} | **ipv6**{ *ip-address ip-mask/prefix*} | **list**{*ipv4-list-name ipv6-list-name*}}] | **dtls** {[**id-trustpoint** {*identity-trustpoint label*}][**verification-trustpoint** { *verification-trustpoint*}] | [{**ipv4** { *ip-address ip-mask/prefix*} | **ipv6**{ *ip-address ip-mask/prefix*} | **list**{*ipv4-list-name ipv6-list-name*}}]}}
no security

Syntax Description

none	Indicates no security on that port. **Note** A maximum of five ipv4 and five ipv6 addresses can be associated.
dtls	The DTLS security takes RSA trustpoint and Verification trustpoint which are optional. Without 1.1.0.0 255.255.0.0 Verification trustpoint it does the normal Public Key Exchange. **Note** A maximum of five ipv4 and five ipv6 addresses can be associated.

Command Modes

coap-proxy configuration (config-coap-proxy)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

To access coap-proxy configuration mode, enter the **coap proxy** command in global configuration mode.

Example

This example shows how to configure no security on the port.

```
Device(config)# coap proxy
Device(config-coap-proxy)# security none ipv4 1.1.0.0 255.255.0.0
```

shell trigger

To create an event trigger, use the **shell trigger** command in global configuration mode. Use the **no** form of this command to delete the trigger.

shell trigger *identifier description*

no shell trigger *identifier description*

Syntax Description

identifier	Specifies the event trigger identifier. The identifier should have no spaces or hyphens between words.
description	Specifies the event trigger description text.

Command Default

System-defined event triggers:

- CISCO_DMP_EVENT
- CISCO_IPVSC_AUTO_EVENT
- CISCO_PHONE_EVENT
- CISCO_SWITCH_EVENT
- CISCO_ROUTER_EVENT
- CISCO_WIRELESS_AP_EVENT
- CISCO_WIRELESS_LIGHTWEIGHT_AP_EVENT

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Use this command to create user-defined event triggers for use with the **macro auto device** and the **macro auto execute** commands.

To support dynamic device discovery when using IEEE 802.1x authentication, you need to configure the RADIUS authentication server to support the Cisco attribute-value pair: **auto-smart-port**=*event trigger*.

Example

This example shows how to create a user-defined event trigger called RADIUS_MAB_EVENT:

```
Device(config)# shell trigger RADIUS_MAB_EVENT MAC_AuthBypass Event
Device(config)# end
```

show coap dtls endpoints

To display the CoAP dtls endpoints, use the **show coap dtls endpoints** command in user EXEC or privileged EXEC mode.

show coap dtls endpoints

Command Default

This command has no arguments or keywords.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Example

This example shows how to display the CoAP dtls endpoint:

```
Device# show coap dtls endpoints
#     Index StateString StateValue  Port IP
------------------------------------------------------------------
```

show coap endpoints

To display the CoAP endpoints, use the **show coap endpoints** command in user EXEC or privileged EXEC mode.

show coap endpoints

Command Default

This command has no arguments or keywords.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Example

This example shows how to display the CoAP endpoint

```
Device# show coap endpoints
List of all endpoints :

Code : D - Discovered , N - New
#    Status  Age(s)    LastWKC(s)    IP
---------------------------------------------------------------

Endpoints - Total : 0 Discovered : 0 New : 0
```

show coap globals

To display the CoAP globals, use the **show coap globals** command in user EXEC or privileged EXEC mode.

show coap globals

Command Default

This command has no arguments or keywords.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Example

The following is sample output from the **show coap globals** command:

This example shows how to display the CoAP configuration:

```
Device# show coap dtls globals
Coap System Timer Values :
 Discovery  : 120 sec
 Cache Exp  : 5 sec
 Keep Alive : 120 sec
 Client DB  : 5 sec
 Query Queue: 500 ms
 Ack delay  : 500 ms
 Timeout    : 5 sec
 Ageout     : 300 sec

Max Endpoints      : 10

Max DTLS Endpoints : 20
Resource Disc Mode : POST
```

show coap resources

To display the CoAP resources, use the **show coap resources** command in user EXEC or privileged EXEC mode.

show coap resources

Command Default

This command has no arguments or keywords.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Example

This example shows how to display the CoAP resources:

```
Device# show coap resources
Link format data =

</>
</cisco/flood>
</cisco/context>
</cisco/showtech>
</cisco/discover>
</cisco/sleep>
</cisco/lldp>
```

show coap stats

To display the CoAP stats, use the **show coap stats** command in user EXEC or privileged EXEC mode.

show coap stats

Command Default

This command has no arguments or keywords.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Example

This example shows how to display the CoAP stats:

```
Device# show coap stats
Coap Stats :
Endpoints    : 0
Requests     : 20
Ext Queries  : 0
New Endpoints: 0
```

show coap version

To display the CoAP version, use the **show coap version** command in user EXEC or privileged EXEC mode.

show coap version

Command Default

This command has no arguments or keywords.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Example

This example shows how to display the CoAP version:

```
Device# show coap version
CoAP version 1.0.5
RFC 7252
```

show device classifier attached

To display the devices connected to a switch and their associated properties, use the **show device classifier attached** command in user EXEC mode.

show device classifier attached [{**detail** | **interface** *interface_id* | **mac-address** *mac_address*}]

Syntax Description

detail	Displays detailed device classifier information.
interface *interface_id*	Displays information about devices attached to the specified interface.
mac *mac_address*	Displays device information for the specified endpoint.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Use this command to display the devices connected to a switch. Use the **show device classifier attached** command in privileged EXEC mode to display the configurable parameters for a device.

Example

This example shows how to use the **show device classifier attached** command with no optional keywords to view the devices connected to the switch:

```
Device# show device classifier attached
MAC_Address       Port_Id     Profile Name
==============    =======     ============================
000a.b8c6.1e07    Gi1/0/2    Cisco-Device
001f.9e90.1250    Gi1/0/4    Cisco-AP-Aironet-1130
=====================================================
```

This example shows how to use the **show device classifier attached** command in privileged EXEC mode with the optional **mac-address** keyword to view summary information about the connected device with the specified MAC address:

```
Device# show device classifier attached mac-address 001f.9e90.1250
MAC_Address       Port_Id     Profile Name
==============    =======     ============================
001f.9e90.1250    Gi1/0/4    Cisco-AP-Aironet-1130
============================================================
```

This example shows how to use the **show device classifier attached** command in privileged EXEC mode with the optional **mac-address** and **detail** keywords to view detailed information about the connected device with the specified MAC address:

```
Device# show device classifier attached mac-address 001f.9e90.1250 detail
MAC_Address       Port_Id     Certainty Parent    ProfileType      Profile Name
     Device_Name
==============    =======     ========= ======    ===========  ============================
    ===========
001f.9e90.1250    Gi1/0/4      40       2         Built-in       Cisco-AP-Aironet-1130
    cisco AIR-LAP1131AG-E-K9
=============================================================================================
```

This example shows how to use the **show device classifier attached** command in privileged EXEC mode with the optional **interface** keyword to view summary information about the device connected to the specified interface:

```
Device# show device classifier attached interface gi 1/0/2
MAC_Address       Port_Id          Profile Name
==============    =======     ============================
000a.b8c6.1e07    Gi1/0/2    Cisco-Device
=================================================
```

This example shows how to use the **show device classifier attached** command in privileged EXEC mode with the optional **interface** and **detail** keywords to view detailed information about the device connected to the specified interface:

```
Device# show device classifier attached interface gi 1/0/2 detail
MAC_Address       Port_Id     Certainty Parent    ProfileType  Profile Name
     Device_Name
==============    =======     ========= ======    ===========  ============================
    ===========
000a.b8c6.1e07    Gi1/0/2      10       0        Default       Cisco-Device        cisco
WS-C2960-48TT-L
=============================================================================================
```

show device classifier clients

To display the clients using the device classifier facility on the switch, use the **show device classifier clients** command in user EXEC mode.

show device classifier clients

Command Default

This command has no arguments or keywords.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Device classifier (DC) is enabled by default when you enable a client application (for example, Auto SmartPorts) that uses its functionality. Use the **show device classifier clients** command to display the clients that are using the DC feature on the switch.

As long as any clients are using the DC, you cannot disable it by using the **no device classifier** command. If you attempt to disable the DC while a client is using it, an error message appears.

Example

This example shows how to use the **show device classifier clients** command to view the clients using the DC on the switch:

```
Device# show device classifier clients
Client Name
===================
Auto Smart Ports

This example shows the error message that appears when you attempt to disable DC while a
client is using it:
Switch(config)# no device classifier
These subsystems should be disabled before disabling Device classifier
Auto Smart Ports

% Error - device classifier is not disabled
```

show device classifier profile type

To display all the device types recognized by the device classifier, use the **show device classifier profile type** command in user EXEC mode.

show device classifier profile type [{**table** [{*built-in default*}] | **string** *filter_string*}]

Syntax Description

table	Displays device classification in a table.
built-in	Displays device classification information from the built-in device table.
default	Displays device classification information from the default device table.
filter *string*	Displays information for devices that match the filter.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

This command displays all the device types recognized by the device classification engine. The number of available device types is the number of profiles stored on the switch. Because the number of profiles can be very large, you can use the **filter** keyword to limit the command output.

Example

This example shows how to use the **show device classifier profile type** command in privileged EXEC mode with no optional keywords to view the devices recognized by the device classifier:

```
Device# show device classifier profile type table
  Valid         Type       Profile Name                          min Conf   ID
===========  =========  ==================                       ========  ====
  Valid        Default      Apple-Device                             10        0
  Valid        Default      Aruba-Device                             10        1
  Valid        Default      Avaya-Device                             10        2
  Valid        Default      Avaya-IP-Phone                           20        3
  Valid        Default      BlackBerry                               20        4
  Valid        Default      Cisco-Device                             10        5
  Valid        Default      Cisco-IP-Phone                           20        6
  Valid        Default      Cisco-IP-Phone-7902                      70        7
  Valid        Default      Cisco-IP-Phone-7905                      70        8
  Valid        Default      Cisco-IP-Phone-7906                      70        9
  Valid        Default      Cisco-IP-Phone-7910                      70        10
  Valid        Default      Cisco-IP-Phone-7911                      70        11
  Valid        Default      Cisco-IP-Phone-7912                      70        12
  Valid        Default      Cisco-IP-Phone-7940                      70        13
  Valid        Default      Cisco-IP-Phone-7941                      70        14
  Valid        Default      Cisco-IP-Phone-7942                      70        15
```

Valid	Default	Cisco-IP-Phone-7945	70	16
Valid	Default	Cisco-IP-Phone-7945G	70	17
Valid	Default	Cisco-IP-Phone-7960	70	18
Valid	Default	Cisco-IP-Phone-7961	70	19
Valid	Default	Cisco-IP-Phone-7962	70	20
Valid	Default	Cisco-IP-Phone-7965	70	21
Valid	Default	Cisco-IP-Phone-7970	70	22
Valid	Default	Cisco-IP-Phone-7971	70	23
Valid	Default	Cisco-IP-Phone-7975	70	24
Valid	Default	Cisco-IP-Phone-7985	70	25
Valid	Default	Cisco-IP-Phone-9971	70	26
Valid	Default	Cisco-WLC-2100-Series	40	27
Valid	Default	DLink-Device	10	28
Valid	Default	Enterasys-Device	10	29
Valid	Default	HP-Device	10	30
Valid	Default	HP-JetDirect-Printer	30	31
Valid	Default	Lexmark-Device	10	32
Valid	Default	Lexmark-Printer-E260dn	30	33
Valid	Default	Microsoft-Device	10	34
Valid	Default	Netgear-Device	10	35
Valid	Default	NintendoWII	10	36
Valid	Default	Nortel-Device	10	37
Valid	Default	Nortel-IP-Phone-2000-Series	20	38
Valid	Default	SonyPS3	10	39
Valid	Default	XBOX360	20	40
Valid	Default	Xerox-Device	10	41
Valid	Default	Xerox-Printer-Phaser3250	30	42
Valid	Default	Aruba-AP	20	43
Valid	Default	Cisco-Access-Point	10	44
Valid	Default	Cisco-IP-Conference-Station-7935	70	45
Valid	Default	Cisco-IP-Conference-Station-7936	70	46
Valid	Default	Cisco-IP-Conference-Station-7937	70	47
Valid	Default	DLink-DAP-1522	20	48
Valid	Default	Cisco-AP-Aironet-1130	30	49
Valid	Default	Cisco-AP-Aironet-1240	30	50
Valid	Default	Cisco-AP-Aironet-1250	30	51
Valid	Default	Cisco-AIR-LAP	25	52
Valid	Default	Cisco-AIR-LAP-1130	30	53
Valid	Default	Cisco-AIR-LAP-1240	50	54
Valid	Default	Cisco-AIR-LAP-1250	50	55
Valid	Default	Cisco-AIR-AP	25	56
Valid	Default	Cisco-AIR-AP-1130	30	57
Valid	Default	Cisco-AIR-AP-1240	50	58
Valid	Default	Cisco-AIR-AP-1250	50	59
Invalid	Default	Sun-Workstation	10	60
Valid	Default	Linksys-Device	20	61
Valid	Default	LinksysWAP54G-Device	30	62
Valid	Default	HTC-Device	10	63
Valid	Default	MotorolaMobile-Device	10	64
Valid	Default	VMWare-Device	10	65
Valid	Default	ISE-Appliance	10	66
Valid	Built-in	Cisco-Device	10	0
Valid	Built-in	Cisco-Router	10	1
Valid	Built-in	Router	10	2
Valid	Built-in	Cisco-IP-Camera	10	3
Valid	Built-in	Cisco-IP-Camera-2xxx	30	4
Valid	Built-in	Cisco-IP-Camera-2421	50	5
Valid	Built-in	Cisco-IP-Camera-2500	50	6
Valid	Built-in	Cisco-IP-Camera-2520	50	7
Valid	Built-in	Cisco-IP-Camera-2530	50	8
Valid	Built-in	Cisco-IP-Camera-4xxx	50	9
Valid	Built-in	Cisco-Transparent-Bridge	8	10
Valid	Built-in	Transparent-Bridge	8	11
Valid	Built-in	Cisco-Source-Bridge	10	12

```
Valid        Built-in     Cisco-Switch                    10        13
Valid        Built-in     Cisco-IP-Phone                  20        14
Valid        Built-in     IP-Phone                        20        15
Valid        Built-in     Cisco-DMP                       10        16
Valid        Built-in     Cisco-DMP-4305G                 70        17
Valid        Built-in     Cisco-DMP-4310G                 70        18
Valid        Built-in     Cisco-DMP-4400G                 70        19
Valid        Built-in     Cisco-WLC-2100-Series           40        20
Valid        Built-in     Cisco-Access-Point              10        21
Valid        Built-in     Cisco-AIR-LAP                   30        22
Valid        Built-in     Cisco-AIR-AP                    30        23
Valid        Built-in     Linksys-Device                  20        24
```

show macro auto

To display Auto Smartports macro information, use the **show macro auto** command in user EXEC mode.

show macro auto {**address-group** *address-group-name* | **device** [**access-point**] [**ip-camera**] [**lightweight-ap**] [**media-player**] [**phone**] [**router**] [**switch**] | **global** [*event_trigger*] | **interface** [*interface_id*]}

Syntax Description

address-group [*address-group-name*]	Displays address-group information. (Optional) *address-group-name*—Displays information for the specified address group.
device [*access-point*] [*ip-camera*] [*lightweight-ap*] [*media-player*] [*phone*] [*router*] [*switch*]	Displays device information about one or more devices. • (Optional) **access-point**—Autonomous access point • (Optional) **ip-camera**—Cisco IP video surveillance camera • (Optional) **lightweight-ap**—Lightweight access point • (Optional) **media-player**—Digital media player • (Optional) phone—Cisco IP phone • (Optional) **router**—Cisco router • (Optional) **switch**—Cisco switch
global [*event_trigger*]	Displays Auto Smartports information about the switch. (Optional) *event_trigger*—Displays information about the specified event trigger.
interface [*interface_id*]	Displays interface status. (Optional) *interface_id*—isplays information about the specified interface.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Use this command to display the Auto SmartPorts information for the switch. Use the **show macro auto device** command to display the configurable parameters for a device.

Example

This example shows how to use the **show macro auto device** to view the configuration on the switch:

```
Device# show macro auto device
Device:lightweight-ap
Default Macro:CISCO_LWAP_AUTO_SMARTPORT
Current Macro:CISCO_LWAP_AUTO_SMARTPORT
Configurable Parameters:ACCESS_VLAN
Defaults Parameters:ACCESS_VLAN=1
Current Parameters:ACCESS_VLAN=1

Device:access-point
Default Macro:CISCO_AP_AUTO_SMARTPORT
Current Macro:CISCO_AP_AUTO_SMARTPORT
Configurable Parameters:NATIVE_VLAN
Defaults Parameters:NATIVE_VLAN=1
Current Parameters:NATIVE_VLAN=1

Device:phone
Default Macro:CISCO_PHONE_AUTO_SMARTPORT
Current Macro:CISCO_PHONE_AUTO_SMARTPORT
Configurable Parameters:ACCESS_VLAN VOICE_VLAN
Defaults Parameters:ACCESS_VLAN=1 VOICE_VLAN=2
Current Parameters:ACCESS_VLAN=1 VOICE_VLAN=2

Device:router
Default Macro:CISCO_ROUTER_AUTO_SMARTPORT
Current Macro:CISCO_ROUTER_AUTO_SMARTPORT
Configurable Parameters:NATIVE_VLAN
Defaults Parameters:NATIVE_VLAN=1
Current Parameters:NATIVE_VLAN=1

Device:switch
Default Macro:CISCO_SWITCH_AUTO_SMARTPORT
Current Macro:CISCO_SWITCH_AUTO_SMARTPORT
Configurable Parameters:NATIVE_VLAN
Defaults Parameters:NATIVE_VLAN=1
Current Parameters:NATIVE_VLAN=1

Device:ip-camera
Default Macro:CISCO_IP_CAMERA_AUTO_SMARTPORT
Current Macro:CISCO_IP_CAMERA_AUTO_SMARTPORT
Configurable Parameters:ACCESS_VLAN
Defaults Parameters:ACCESS_VLAN=1
Current Parameters:ACCESS_VLAN=1

Device:media-player
Default Macro:CISCO_DMP_AUTO_SMARTPORT
Current Macro:CISCO_DMP_AUTO_SMARTPORT
Configurable Parameters:ACCESS_VLAN
Defaults Parameters:ACCESS_VLAN=1
Current Parameters:ACCESS_VLAN=1
```

This example shows how to use the **show macro auto address-group name** command to view the TEST3 address group configuration on the switch:

```
Device# show macro auto address-group TEST3MAC Address Group Configuration:
```

```
Group Name OUI  MAC ADDRESS
-------------------------------------------------------------
TEST3 2233.33     0022.0022.0022
2233.34
```

show parser macro

To display the parameters for all configured macros or for one macro on the switch, use the **show parser macro** command in user EXEC mode.

show parser macro {**brief** | **description** [**interface** *interface-id*] | **name** *macro-name*}

Syntax Description

brief	(Optional) Displays the name of each macro.
description [**interface** *interface-id*]	(Optional) Displays all macro descriptions or the description of a specific interface.
name *macro-name*	(Optional) Displays information about a single macro identified by the macro name.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Example

This is a partial output example from the **show parser macro** command. The output for the Cisco-default macros varies depending on the switch platform and the software image running on the switch:

```
Device# show parser macro
Total number of macros = 6
------------------------------------------------------------------
Macro name : cisco-global
Macro type : default global
# Enable dynamic port error recovery for link state
# failures
errdisable recovery cause link-flap
errdisable recovery interval 60

<output truncated>

------------------------------------------------------------------
Macro name : cisco-desktop
Macro type : default interface
# macro keywords $AVID
# Basic interface - Enable data VLAN only
# Recommended value for access vlan (AVID) should not be 1
switchport access vlan $AVID
switchport mode access

<output truncated>

------------------------------------------------------------------
Macro name : cisco-phone
```

```
Macro type : default interface
# Cisco IP phone + desktop template
# macro keywords $AVID $VVID
# VoIP enabled interface - Enable data VLAN
# and voice VLAN (VVID)
# Recommended value for access vlan (AVID) should not be 1
switchport access vlan $AVID
switchport mode access

<output truncated>

------------------------------------------------------------------
Macro name : cisco-switch
Macro type : default interface
# macro keywords $NVID
# Access Uplink to Distribution
# Do not apply to EtherChannel/Port Group
# Define unique Native VLAN on trunk ports
# Recommended value for native vlan (NVID) should not be 1
switchport trunk native vlan $NVID

<output truncated>

------------------------------------------------------------------
Macro name : cisco-router
Macro type : default interface
# macro keywords $NVID
# Access Uplink to Distribution
# Define unique Native VLAN on trunk ports
# Recommended value for native vlan (NVID) should not be 1
switchport trunk native vlan $NVID

<output truncated>

------------------------------------------------------------------
Macro name : snmp
Macro type : customizable

#enable port security, linkup, and linkdown traps
snmp-server enable traps port-security
snmp-server enable traps linkup
snmp-server enable traps linkdown
#set snmp-server host
snmp-server host ADDRESS
#set SNMP trap notifications precedence
snmp-server ip precedence VALUE

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

This example shows the output from the **show parser macro name** command:

```
Device# show parser macro name standard-switch10
Macro name : standard-switch10
Macro type : customizable
macro description standard-switch10
# Trust QoS settings on VOIP packets
auto qos voip trust
# Allow port channels to be automatically formed
channel-protocol pagp
```

This example shows the output from the **show parser macro brief** command:

```
Device# show parser macro brief
    default global   : cisco-global
    default interface: cisco-desktop
    default interface: cisco-phone
    default interface: cisco-switch
    default interface: cisco-router
    customizable     : snmp
```

This exampe shows the output from the **show parser macro description** command:

```
Device# show parser macro description
Global Macro(s): cisco-global
Interface    Macro Description(s)
---------------------------------------------------------------
Gi1/0/1        standard-switch10
Gi1/0/2       this is test macro
---------------------------------------------------------------
```

This example shows the output from the **show parser macro description interface** command:

```
Device# show parser macro description interface gigabitethernet1/0/2
Interface    Macro Description
---------------------------------------------------------------
Gi1/0/2      this is test macro
---------------------------------------------------------------
```

show shell

To display shell information, use the **show shell** command in user EXEC mode.

show shell [{**enviornment** | **functions** [{**brief** *shell_function*}] | **triggers**}]

Syntax Description

environment	(Optional) Displays shell environment inf
functions [**brief** |*shell_function*]	(Optional) Displays macro information. • **brief**—Names of the shell functions. • *shell_function*—Name of a shell func
triggers	(Optional) Displays event trigger informat

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

Use this command to display the shell information for the switch.

Example

This example shows how to use the **show shell** *triggers* command to view the event triggers in the switch software:

```
Device# term shell
Device# show shell triggers
User defined triggers
---------------------
Built-in triggers
-----------------
Trigger Id: CISCO_CUSTOM_EVENT
Trigger description: Custom macroevent to apply user defined configuration
Trigger environment: User can define the macro
Trigger mapping function: CISCO_CUSTOM_AUTOSMARTPORT

Trigger Id: CISCO_DMP_EVENT
Trigger description: Digital media-player device event to apply port configuration
Trigger environment: Parameters that can be set in the shell - $ACCESS_VLAN=(1)
 The value in the parenthesis is a default value
Trigger mapping function: CISCO_DMP_AUTO_SMARTPORT

Trigger Id: CISCO_IPVSC_EVENT
Trigger description: IP-camera device event to apply port configuration
Trigger environment: Parameters that can be set in the shell - $ACCESS_VLAN=(1)
 The value in parenthesis is a default value
Trigger mapping function: CISCO_IP_CAMERA_AUTO_SMARTPORT
```

```
Trigger Id: CISCO_LAST_RESORT_EVENT
Trigger description: Last resortevent to apply port configuration
Trigger environment: Parameters that can be set in the shell - $ACCESS_VLAN=(1)
 The value in the parenthesis is a default value
Trigger mapping function: CISCO_LAST_RESORT_SMARTPORT

Trigger Id: CISCO_PHONE_EVENT
Trigger description: IP-phone device event to apply port configuration
Trigger environment: Parameters that can be set in the shell - $ACCESS_VLAN=(1)
and $VOICE_VLAN=(2), The value in the parenthesis is a default value
Trigger mapping function: CISCO_PHONE_AUTO_SMARTPORT

Trigger Id: CISCO_ROUTER_EVENT
Trigger description: Router device event to apply port configuration
Trigger environment: Parameters that can be set in the shell - $NATIVE_VLAN=(1)
 The value in the parenthesis is a default value
Trigger mapping function: CISCO_ROUTER_AUTO_SMARTPORT

Trigger Id: CISCO_SWITCH_ETHERCHANNEL_CONFIG
Trigger description: etherchannel parameter
Trigger environment: $INTERFACE_LIST=(),$PORT-CHANNEL_ID=(),
                              $EC_MODE=(),$EC_PROTOCOLTYPE=(),
                              PORT-CHANNEL_TYPE=()
Trigger mapping function: CISCO_ETHERCHANNEL_AUTOSMARTPORT

Trigger Id: CISCO_SWITCH_EVENT
Trigger description: Switch device event to apply port configuration
Trigger environment: Parameters that can be set in the shell - $NATIVE_VLAN=(1)
 The value in the parenthesis is a default value
Trigger mapping function: CISCO_SWITCH_AUTO_SMARTPORT

Trigger Id: CISCO_WIRELESS_AP_EVENT
Trigger description: Autonomous ap device event to apply port configuration
Trigger environment: Parameters that can be set in the shell - $NATIVE_VLAN=(1)
 The value in the parenthesis is a default value
Trigger mapping function: CISCO_AP_AUTO_SMARTPORT

Trigger Id: CISCO_WIRELESS_LIGHTWEIGHT_AP_EVENT
Trigger description: Lightweight-ap device event to apply port configuration
Trigger environment: Parameters that can be set in the shell - $ACCESS_VLAN=(1)
  The value in the parenthesis is a default value
Trigger mapping function: CISCO_LWAP_AUTO_SMARTPORT

Trigger Id: word
Trigger description: word
Trigger environment:
Trigger mapping function:
```

This example shows how to use the **show shell functions** command to view the built-in macros in the switch software:

```
Device# show shell functions
#User defined functions:

#Built-in functions:
function CISCO_AP_AUTO_SMARTPORT () {
    if [[ $LINKUP == YES ]]; then
        conf t
            interface  $INTERFACE
                macro description $TRIGGER
                switchport trunk encapsulation dot1q
                switchport trunk native vlan $NATIVE_VLAN
                switchport trunk allowed vlan ALL
```

```
                switchport mode trunk
                switchport nonegotiate
                auto qos voip trust
                mls qos trust cos
                if [[ $LIMIT == 0 ]]; then
                  default srr-queue bandwidth limit
                else
                  srr-queue bandwidth limit $LIMIT
                fi
                if [[ $SW_POE == YES ]]; then
                   if [[ $AP125X == AP125X ]]; then
                      macro description AP125X
                      macro auto port sticky
                      power inline port maximum 20000
                   fi
                fi
             exit
        end
    fi
    if [[ $LINKUP == NO ]]; then
        conf t
            interface  $INTERFACE
                no macro description
                no switchport nonegotiate
                no switchport trunk native vlan $NATIVE_VLAN
                no switchport trunk allowed vlan ALL
                no auto qos voip trust
                no mls qos trust cos
                default srr-queue bandwidth limit
                if [[ $AUTH_ENABLED == NO ]]; then
                  no switchport mode
                  no switchport trunk encapsulation
                fi
                if [[ $STICKY == YES ]]; then
                   if [[ $SW_POE == YES ]]; then
                      if [[ $AP125X == AP125X ]]; then
                         no macro auto port sticky
                         no power inline port maximum
                      fi
                   fi
                fi
            exit
        end
    fi
}
<output truncated>
```

start (coap-proxy configuration)

To start CoAP on the switch, use the **start** command in coap-proxy configuration mode.

start

Command Modes

coap-proxy configuration (config-coap-proxy)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

To access coap-proxy configuration mode, enter the **coap proxy** command in global configuration mode.

Example

This example shows how to start CoAP on the switch.

```
Device(config)# coap proxy
Device(config-coap-proxy)# start
```

stop (coap-proxy configuration)

To stop CoAP on the switch, use the **stop** command in coap-proxy configuration mode.

stop

Command Modes

coap-proxy configuration (config-coap-proxy)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

To access coap-proxy configuration mode, enter the **coap proxy** command in global configuration mode.

Example

This example shows how to stop CoAP on the switch.

```
Device(config)# coap proxy
Device(config-coap-proxy)# stop
```

transport (coap-proxy configuration)

To configure transport protocol, use the **transport** command in coap-proxy configuration mode.

transport{**tcp** | **udp**}

Syntax Description

tcp	Specifies a TCP protocol.
udp	Specifies a UDP protocol.

Command Modes

coap-proxy configuration (config-coap-proxy)

Command History

Release	Modification
Cisco IOS XE Fuji 16.9.2	This command was introduced.

Usage Guidelines

To access coap-proxy configuration mode, enter the **coap proxy** command in global configuration mode.

Example

This is an example to configure tcp as transport protocol

```
Device(config)# coap proxy
Device(config-coap-proxy)# transport tcp
```