



## **Catalyst 2960-X Switch Auto Smartports Configuration Guide, Cisco IOS Release 15.0(2)EX**

July 2013

**Cisco Systems, Inc.**

[www.cisco.com](http://www.cisco.com)

Cisco has more than 200 offices worldwide.  
Addresses, phone numbers, and fax numbers  
are listed on the Cisco website at  
[www.cisco.com/go/offices](http://www.cisco.com/go/offices).

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: [www.cisco.com/go/trademarks](http://www.cisco.com/go/trademarks). Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

*Auto Smartports Configuration Guide*

© 2013 Cisco Systems, Inc. All rights reserved.



## **Preface**   vii

- Document Conventions   vii
- Related Documentation   ix
- Obtaining Documentation and Submitting a Service Request   ix

## **Using the Command-Line Interface**   1-1

- Understanding Command Modes   1-1
- Understanding the Help System   1-3
- Understanding Abbreviated Commands   1-3
- Understanding no and default Forms of Commands   1-4
- Understanding CLI Error Messages   1-4
- Using Configuration Logging   1-4
- Using Command History   1-5
  - Changing the Command History Buffer Size   1-5
  - Recalling Commands   1-5
  - Disabling the Command History Feature   1-6
- Using Editing Features   1-6
  - Enabling and Disabling Editing Features   1-6
  - Editing Commands through Keystrokes   1-6
  - Editing Command Lines that Wrap   1-8
- Searching and Filtering Output of show and more Commands   1-9
- Accessing the CLI   1-9
  - Accessing the CLI through a Console Connection or through Telnet   1-9

## **Auto Smartports and Static Smartports Macros**   2-1

- Auto Smartports Macros   2-1
- Static Smartports Macros   2-1
- Event Triggers   2-2
- User-Defined Files   2-2
- Macro Persistence   2-2
- Managing Macros   2-3
- Auto Smartports and Cisco Medianet   2-3
- Device Classifier   2-3
  - Device Visibility Mode   2-4

**Configuring Auto Smartports and Static Smartports Macros 3-1**

- Configuring Macros 3-1
  - Auto Smartports Configuration Guidelines 3-2
  - Enabling Auto Smartports Macros 3-3
  - Default Auto Smartports Configuration 3-4
  - Configuring Auto Smartports Parameter Values 3-7
  - Configuring MAC Address Groups 3-8
  - Configuring Macro Persistence 3-11
  - Configuring Built-In Macro Options 3-12
  - Creating User-Defined Event Triggers 3-16
  - Configuring Event Trigger Control 3-19
  - Configuring User-Defined Triggers for User-Defined Macros 3-23
  - Configuring User-Defined Macros 3-24
  - Applying Macros on a Switch 3-29
  - Default Static Smartports Configuration 3-33
  - Static Smartports Configuration Guidelines 3-33
  - NEAT Configuration 3-33
  - Applying Static Smartports Macros 3-34
- Displaying Macros 3-36

**Auto Smartports and Static Smartports Macros CLI Commands 4-1**

- macro 4-2
- macro auto (CLI) 4-5
- macro auto apply (Cisco IOS shell scripting capability) 4-8
- macro auto config (Cisco IOS shell scripting capability) 4-10
- macro auto control 4-11
- macro auto device 4-13
- macro auto execute 4-15
- macro auto global control 4-21
- macro auto global processing 4-23
- macro auto mac-address-group 4-25
- macro auto monitor 4-27
- macro auto processing 4-28
- macro auto sticky 4-30
- macro auto trigger 4-31
- macro description 4-33
- macro global 4-34
- macro global description 4-37

shell trigger	4-38
show macro auto	4-40
show macro auto monitor clients	4-43
show macro auto monitor device	4-45
show macro auto monitor type	4-47
show parser macro	4-50
show shell	4-53





## Preface

---

This document describes how to configure Auto and Static Smartports macros on the Catalyst 2960-X Switch.

For information about the Cisco IOS commands, see the Cisco IOS documentation set on Cisco.com.

This guide does not describe how to use the embedded device manager GUIs or Cisco Network Assistant (hereafter referred to as *Network Assistant*). For information about the device manager, see the switch online help. For information about Network Assistant, see *Getting Started with Cisco Network Assistant* on Cisco.com.

You can use CiscoWorks LAN Management Solution (LMS) to configure and manage Auto Smartports. For information about using CiscoWorks LMS and Auto Smartports:

[http://www.cisco.com/en/US/prod/collateral/netmgtsw/ps6504/ps6528/ps2425/white\\_paper\\_c11-54288\\_1\\_ps11200\\_Products\\_White\\_Paper.html#wp9000042](http://www.cisco.com/en/US/prod/collateral/netmgtsw/ps6504/ps6528/ps2425/white_paper_c11-54288_1_ps11200_Products_White_Paper.html#wp9000042)

## Document Conventions

This document uses the following conventions:

Convention	Description
^ or Ctrl	Both the ^ symbol and Ctrl represent the Control (Ctrl) key on a keyboard. For example, the key combination <b>^D</b> or <b>Ctrl-D</b> means that you hold down the Control key while you press the D key. (Keys are indicated in capital letters but are not case sensitive.)
<b>bold font</b>	Commands and keywords and user-entered text appear in <b>bold font</b> .
<i>italic font</i>	Document titles, new or emphasized terms, and arguments for which you supply values are in <i>italic font</i> .
<code>courier font</code>	Terminal sessions and information the system displays appear in <code>courier font</code> .
<b>Bold courier font</b>	Bold courier font indicates text that the user must enter.
[x]	Elements in square brackets are optional.
...	An ellipsis (three consecutive nonbolded periods without spaces) after a syntax element indicates that the element can be repeated.
	A vertical line, called a pipe, indicates a choice within a set of keywords or arguments.
{x   y}	Required alternative keywords are grouped in braces and separated by vertical bars.

[x   y]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
[x {y   z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
<code>courier font</code>	Terminal sessions and information the system displays appear in <code>courier font</code> .
< >	Nonprinting characters such as passwords are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

### Reader Alert Conventions

This document uses the following conventions for reader alerts:



#### Note

Means *reader take note*. Notes contain helpful suggestions or references to materials not contained in this manual.



#### Tip

Means *the following information will help you solve a problem*. The tips information might not be troubleshooting or even an action, but could be useful information, similar to a Timesaver.



#### Caution

Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.



#### Timesaver

Means *the described action saves time*. You can save time by performing the action described in the paragraph.



#### Warning

Means *reader be warned*. In this situation, you might perform an action that could result in bodily injury.



## Related Documentation

**Note**

---

Before installing or upgrading the switch, refer to the switch release notes.

---

- Catalyst 2960-X Switch documentation at:  
[http://www.cisco.com/en/US/products/ps12995/products\\_installation\\_and\\_configuration\\_guides\\_list.html](http://www.cisco.com/en/US/products/ps12995/products_installation_and_configuration_guides_list.html)
  - Cisco SFP and SFP+ modules documentation, including compatibility matrixes at:  
[http://www.cisco.com/en/US/products/hw/modules/ps5455/tsd\\_products\\_support\\_series\\_home.html](http://www.cisco.com/en/US/products/hw/modules/ps5455/tsd_products_support_series_home.html)
- 

## Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

<http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html>

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed, and set content to be delivered directly to your desktop by a reader application. The RSS feeds are a free service, and Cisco supports RSS version 2.0.





# Using the Command-Line Interface

---

This chapter describes the Cisco IOS command-line interface (CLI) and how to use it to configure your standalone switch or a switch stack, referred to as the *switch*. It contains these sections:

- [Understanding Command Modes, page 1-1](#)
- [Understanding the Help System, page 1-3](#)
- [Understanding Abbreviated Commands, page 1-3](#)
- [Understanding no and default Forms of Commands, page 1-4](#)
- [Understanding CLI Error Messages, page 1-4](#)
- [Using Configuration Logging, page 1-4](#)
- [Using Command History, page 1-5](#)
- [Using Editing Features, page 1-6](#)
- [Searching and Filtering Output of show and more Commands, page 1-9](#)
- [Accessing the CLI, page 1-9](#)

## Understanding Command Modes

The Cisco IOS user interface is divided into many different modes. The commands available to you depend on which mode you are currently in. Enter a question mark (?) at the system prompt to obtain a list of commands available for each command mode.

When you start a session on the switch, you begin in user mode, often called user EXEC mode. Only a limited subset of the commands are available in user EXEC mode. For example, most of the user EXEC commands are one-time commands, such as **show** commands, which show the current configuration status, and **clear** commands, which clear counters or interfaces. The user EXEC commands are not saved when the switch reboots.

To have access to all commands, you must enter privileged EXEC mode. Normally, you must enter a password to enter privileged EXEC mode. From this mode, you can enter any privileged EXEC command or enter global configuration mode.

Using the configuration modes (global, interface, and line), you can make changes to the running configuration. If you save the configuration, these commands are stored and used when the switch reboots. To access the various configuration modes, you must start at global configuration mode. From global configuration mode, you can enter interface configuration mode and line configuration mode.

Table 1-1 describes the main command modes, how to access each one, the prompt you see in that mode, and how to exit the mode. The examples in the table use the hostname *Switch*.

**Table 1-1 Command Mode Summary**

Mode	Access Method	Prompt	Exit Method	About This Mode
User EXEC	Begin a session with your switch.	Switch>	Enter <b>logout</b> or <b>quit</b> .	Use this mode to <ul style="list-style-type: none"> <li>• Change terminal settings.</li> <li>• Perform basic tests.</li> <li>• Display system information.</li> </ul>
Privileged EXEC	While in user EXEC mode, enter the <b>enable</b> command.	Switch#	Enter <b>disable</b> to exit.	Use this mode to verify commands that you have entered. Use a password to protect access to this mode.
Global configuration	While in privileged EXEC mode, enter the <b>configure</b> command.	Switch(config)#	To exit to privileged EXEC mode, enter <b>exit</b> or <b>end</b> , or press <b>Ctrl-Z</b> .	Use this mode to configure parameters that apply to the entire switch.
VLAN configuration	While in global configuration mode, enter the <b>vlan</b> <i>vlan-id</i> command.	Switch(config-vlan)#	To exit to global configuration mode, enter the <b>exit</b> command.  To return to privileged EXEC mode, press <b>Ctrl-Z</b> or enter <b>end</b> .	Use this mode to configure VLAN parameters. When VTP mode is transparent, you can create extended-range VLANs (VLAN IDs greater than 1005) and save configurations in the switch startup configuration file.
Interface configuration	While in global configuration mode, enter the <b>interface</b> command (with a specific interface).	Switch(config-if)#	To exit to global configuration mode, enter <b>exit</b> .  To return to privileged EXEC mode, press <b>Ctrl-Z</b> or enter <b>end</b> .	Use this mode to configure parameters for the Ethernet ports.
Line configuration	While in global configuration mode, specify a line with the <b>line vty</b> or <b>line console</b> command.	Switch(config-line)#	To exit to global configuration mode, enter <b>exit</b> .  To return to privileged EXEC mode, press <b>Ctrl-Z</b> or enter <b>end</b> .	Use this mode to configure parameters for the terminal line.

# Understanding the Help System

You can enter a question mark (?) at the system prompt to display a list of commands available for each command mode. You can also obtain a list of associated keywords and arguments for any command, as shown in [Table 1-2](#).

**Table 1-2** Help Summary

Command	Purpose
<b>help</b>	Obtain a brief description of the help system in any command mode.
<i>abbreviated-command-entry?</i>	Obtain a list of commands that begin with a particular character string.  For example:  Switch# <b>di?</b> dir disable disconnect
<i>abbreviated-command-entry&lt;Tab&gt;</i>	Complete a partial command name.  For example:  Switch# <b>sh conf</b> <tab> Switch# <b>show configuration</b>
<b>?</b>	List all commands available for a particular command mode.  For example:  Switch> <b>?</b>
<i>command ?</i>	List the associated keywords for a command.  For example:  Switch> <b>show ?</b>
<i>command keyword ?</i>	List the associated arguments for a keyword.  For example:  Switch(config)# <b>cdp holdtime ?</b> <10-255> Length of time (in sec) that receiver must keep this packet

## Understanding Abbreviated Commands

You need to enter only enough characters for the switch to recognize the command as unique.

This example shows how to enter the **show configuration** privileged EXEC command in an abbreviated form:

```
Switch# show conf
```

## Understanding no and default Forms of Commands

Almost every configuration command also has a **no** form. In general, use the **no** form to disable a feature or function or reverse the action of a command. For example, the **no shutdown** interface configuration command reverses the shutdown of an interface. Use the command without the keyword **no** to re-enable a disabled feature or to enable a feature that is disabled by default.

Configuration commands can also have a **default** form. The **default** form of a command returns the command setting to its default. Most commands are disabled by default, so the **default** form is the same as the **no** form. However, some commands are enabled by default and have variables set to certain default values. In these cases, the **default** command enables the command and sets variables to their default values.

## Understanding CLI Error Messages

Table 1-3 lists some error messages that you might encounter while using the CLI to configure your switch.

**Table 1-3** Common CLI Error Messages

Error Message	Meaning	How to Get Help
% Ambiguous command: "show con"	You did not enter enough characters for your switch to recognize the command.	Re-enter the command followed by a question mark (?) with a space between the command and the question mark.  The possible keywords that you can enter with the command appear.
% Incomplete command.	You did not enter all the keywords or values required by this command.	Re-enter the command followed by a question mark (?) with a space between the command and the question mark.  The possible keywords that you can enter with the command appear.
% Invalid input detected at '^' marker.	You entered the command incorrectly. The caret (^) marks the point of the error.	Enter a question mark (?) to display all the commands that are available in this command mode.  The possible keywords that you can enter with the command appear.

## Using Configuration Logging

You can log and view changes to the switch configuration. You can use the Configuration Change Logging and Notification feature to track changes on a per-session and per-user basis. The logger tracks each configuration command that is applied, the user who entered the command, the time that the command was entered, and the parser return code for the command. This feature includes a mechanism for asynchronous notification to registered applications whenever the configuration changes. You can choose to have the notifications sent to the syslog.



**Note**

Only CLI or HTTP changes are logged.

# Using Command History

The software provides a history or record of commands that you have entered. The command history feature is particularly useful for recalling long or complex commands or entries, including access lists. You can customize this feature to suit your needs as described in these sections:

- [Changing the Command History Buffer Size, page 1-5](#) (optional)
- [Recalling Commands, page 1-5](#) (optional)
- [Disabling the Command History Feature, page 1-6](#) (optional)

## Changing the Command History Buffer Size

By default, the switch records ten command lines in its history buffer. You can alter this number for a current terminal session or for all sessions on a particular line. These procedures are optional.

Beginning in privileged EXEC mode, enter this command to change the number of command lines that the switch records during the current terminal session:

```
Switch# terminal history [size number-of-lines]
```

The range is from 0 to 256.

Beginning in line configuration mode, enter this command to configure the number of command lines the switch records for all sessions on a particular line:

```
Switch(config-line)# history [size number-of-lines]
```

The range is from 0 to 256.

## Recalling Commands

To recall commands from the history buffer, perform one of the actions listed in [Table 1-4](#). These actions are optional.

**Table 1-4**      *Recalling Commands*

Action <sup>1</sup>	Result
Press <b>Ctrl-P</b> or the up arrow key.	Recall commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Press <b>Ctrl-N</b> or the down arrow key.	Return to more recent commands in the history buffer after recalling commands with <b>Ctrl-P</b> or the up arrow key. Repeat the key sequence to recall successively more recent commands.
<b>show history</b>	While in privileged EXEC mode, list the last several commands that you just entered. The number of commands that appear is controlled by the setting of the <b>terminal history</b> global configuration command and the <b>history</b> line configuration command.

1. The arrow keys function only on ANSI-compatible terminals such as VT100s.

## Disabling the Command History Feature

The command history feature is automatically enabled. You can disable it for the current terminal session or for the command line. These procedures are optional.

To disable the feature during the current terminal session, enter the **terminal no history** privileged EXEC command.

To disable command history for the line, enter the **no history** line configuration command.

## Using Editing Features

This section describes the editing features that can help you manipulate the command line. It contains these sections:

- [Enabling and Disabling Editing Features, page 1-6](#) (optional)
- [Editing Commands through Keystrokes, page 1-6](#) (optional)
- [Editing Command Lines that Wrap, page 1-8](#) (optional)

## Enabling and Disabling Editing Features

Although enhanced editing mode is automatically enabled, you can disable it, re-enable it, or configure a specific line to have enhanced editing. These procedures are optional.

To globally disable enhanced editing mode, enter this command in line configuration mode:

```
Switch (config-line)# no editing
```

To re-enable the enhanced editing mode for the current terminal session, enter this command in privileged EXEC mode:

```
Switch# terminal editing
```

To reconfigure a specific line to have enhanced editing mode, enter this command in line configuration mode:

```
Switch(config-line)# editing
```

## Editing Commands through Keystrokes

[Table 1-5](#) shows the keystrokes that you need to edit command lines. These keystrokes are optional.

**Table 1-5** *Editing Commands through Keystrokes*

Capability	Keystroke <sup>1</sup>	Purpose
Move around the command line to make changes or corrections.	Press <b>Ctrl-B</b> , or press the left arrow key.	Move the cursor back one character.



Table 1-5 Editing Commands through Keystrokes (continued)

Capability	Keystroke <sup>1</sup>	Purpose
	Press <b>Ctrl-F</b> , or press the right arrow key.	Move the cursor forward one character.
	Press <b>Ctrl-A</b> .	Move the cursor to the beginning of the command line.
	Press <b>Ctrl-E</b> .	Move the cursor to the end of the command line.
	Press <b>Esc B</b> .	Move the cursor back one word.
	Press <b>Esc F</b> .	Move the cursor forward one word.
	Press <b>Ctrl-T</b> .	Transpose the character to the left of the cursor with the character located at the cursor.
Recall commands from the buffer and paste them in the command line. The switch provides a buffer with the last ten items that you deleted.	Press <b>Ctrl-Y</b> .	Recall the most recent entry in the buffer.
	Press <b>Esc Y</b> .	Recall the next buffer entry. The buffer contains only the last 10 items that you have deleted or cut. If you press <b>Esc Y</b> more than ten times, you cycle to the first buffer entry.
Delete entries if you make a mistake or change your mind.	Press the <b>Delete</b> or <b>Backspace</b> key.	Erase the character to the left of the cursor.
	Press <b>Ctrl-D</b> .	Delete the character at the cursor.
	Press <b>Ctrl-K</b> .	Delete all characters from the cursor to the end of the command line.
	Press <b>Ctrl-U</b> or <b>Ctrl-X</b> .	Delete all characters from the cursor to the beginning of the command line.
	Press <b>Ctrl-W</b> .	Delete the word to the left of the cursor.
	Press <b>Esc D</b> .	Delete from the cursor to the end of the word.
Capitalize or lowercase words or capitalize a set of letters.	Press <b>Esc C</b> .	Capitalize at the cursor.
	Press <b>Esc L</b> .	Change the word at the cursor to lowercase.
	Press <b>Esc U</b> .	Capitalize letters from the cursor to the end of the word.
Designate a particular keystroke as an executable command, perhaps as a shortcut.	Press <b>Ctrl-V</b> or <b>Esc Q</b> .	

Table 1-5 Editing Commands through Keystrokes (continued)

Capability	Keystroke <sup>1</sup>	Purpose
Scroll down a line or screen on displays that are longer than the terminal screen can display. <b>Note</b> The More prompt is used for any output that has more lines than can be displayed on the terminal screen, including <b>show</b> command output. You can use the <b>Return</b> and <b>Space</b> bar keystrokes whenever you see the More prompt.	Press the <b>Return</b> key.	Scroll down one line.
	Press the <b>Space</b> bar.	Scroll down one screen.
Redisplay the current command line if the switch suddenly sends a message to your screen.	Press <b>Ctrl-L</b> or <b>Ctrl-R</b> .	Redisplay the current command line.

1. The arrow keys function only on ANSI-compatible terminals such as VT100s.

## Editing Command Lines that Wrap

You can use a wraparound feature for commands that extend beyond a single line on the screen. When the cursor reaches the right margin, the command line shifts ten spaces to the left. You cannot see the first ten characters of the line, but you can scroll back and check the syntax at the beginning of the command. The keystroke actions are optional.

To scroll back to the beginning of the command entry, press **Ctrl-B** or the left arrow key repeatedly. You can also press **Ctrl-A** to immediately move to the beginning of the line.



### Note

The arrow keys function only on ANSI-compatible terminals such as VT100s.

In this example, the **access-list** global configuration command entry extends beyond one line. When the cursor first reaches the end of the line, the line is shifted ten spaces to the left and redisplayed. The dollar sign (\$) shows that the line has been scrolled to the left. Each time the cursor reaches the end of the line, the line is again shifted ten spaces to the left.

```
Switch(config)# access-list 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1
Switch(config)# $ 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1.20 255.25
Switch(config)# $t tcp 131.108.2.5 255.255.255.0 131.108.1.20 255.255.255.0 eq
Switch(config)# $108.2.5 255.255.255.0 131.108.1.20 255.255.255.0 eq 45
```

After you complete the entry, press **Ctrl-A** to check the complete syntax before pressing the **Return** key to execute the command. The dollar sign (\$) appears at the end of the line to show that the line has been scrolled to the right:

```
Switch(config)# access-list 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1$
```

The software assumes you have a terminal screen that is 80 columns wide. If you have a width other than that, use the **terminal width** privileged EXEC command to set the width of your terminal.

Use line wrapping with the command history feature to recall and modify previous complex command entries.

## Searching and Filtering Output of show and more Commands

You can search and filter the output for **show** and **more** commands. This is useful when you need to sort through large amounts of output or if you want to exclude output that you do not need to see. Using these commands is optional.

To use this functionality, enter a **show** or **more** command followed by the *pipe* character (`|`), one of the keywords **begin**, **include**, or **exclude**, and an expression that you want to search for or filter out:

```
command | {begin | include | exclude} regular-expression
```

Expressions are case sensitive. For example, if you enter `| exclude output`, the lines that contain *output* are not displayed, but the lines that contain *Output* appear.

This example shows how to include in the output display only lines where the expression *protocol* appears:

```
Switch# show interfaces | include protocol
Vlan1 is up, line protocol is up
Vlan10 is up, line protocol is down
GigabitEthernet1/0/1 is up, line protocol is down
GigabitEthernet1/0/2 is up, line protocol is up
```

## Accessing the CLI

You can access the CLI through a console connection, through Telnet, or by using the browser.

You manage the switch stack and the stack member interfaces through the stack master. You cannot manage stack members on an individual switch basis. You can connect to the stack master through the console port or the Ethernet management port of one or more stack members. Be careful with using multiple CLI sessions to the stack master. Commands you enter in one session are not displayed in the other sessions. Therefore, it is possible to lose track of the session from which you entered commands.



### Note

We recommend using one CLI session when managing the switch stack.

If you want to configure a specific stack member port, you must include the stack member number in the CLI command interface notation.

To debug a specific stack member, you can access it from the stack master by using the **session stack-member-number** privileged EXEC command. The stack member number is appended to the system prompt. For example, `Switch-2#` is the prompt in privileged EXEC mode for stack member 2, and where the system prompt for the stack master is `Switch`. Only the **show** and **debug** commands are available in a CLI session to a specific stack member.

## Accessing the CLI through a Console Connection or through Telnet

Before you can access the CLI, you must connect a terminal or a PC to the switch console or connect a PC to the Ethernet management port and then power on the switch, as described in the hardware installation guide that shipped with your switch.

If your switch is already configured, you can access the CLI through a local console connection or through a remote Telnet session, but your switch must first be configured for this type of access.

You can use one of these methods to establish a connection with the switch:

- Connect the switch console port to a management station or dial-up modem, or connect the Ethernet management port to a PC. For information about connecting to the console or Ethernet management port, see the switch hardware installation guide.
- Use any Telnet TCP/IP or encrypted Secure Shell (SSH) package from a remote management station. The switch must have network connectivity with the Telnet or SSH client, and the switch must have an enable secret password configured.
  - The switch supports up to 16 simultaneous Telnet sessions. Changes made by one Telnet user are reflected in all other Telnet sessions.
  - The switch supports up to five simultaneous secure SSH sessions.

After you connect through the console port, through the Ethernet management port, through a Telnet session or through an SSH session, the user EXEC prompt appears on the management station.



## Auto Smartports and Static Smartports Macros

---

- [Auto Smartports Macros, page 2-1](#)
- [Static Smartports Macros, page 2-1](#)
- [Event Triggers, page 2-2](#)
- [User-Defined Files, page 2-2](#)
- [Macro Persistence, page 2-2](#)
- [Auto Smartports and Cisco Medianet, page 2-3](#)
- [Device Classifier, page 2-3](#)

### Auto Smartports Macros

Auto Smartports macros dynamically configure ports based on the device type detected on the port. When the switch detects a new device on a port, it applies the appropriate macro. When there is a link-down event, the switch removes the macro. For example, when you connect a Cisco IP phone to a port, Auto Smartports applies the IP phone macro. The IP phone macro enables quality of service (QoS), security features, and a dedicated voice VLAN to ensure proper treatment of delay-sensitive voice traffic. Auto Smartports uses event triggers to map devices to port macros.

You can also manually configure and apply global macros.

The macros embedded in the switch software are groups of CLI commands.

You can also create user-defined macros by using the Cisco IOS Shell scripting capability, which is a BASH-like language syntax for command automation and variable replacement.

For information, see Chapter 2, “Configuring Auto Smartports and Static Smartports Macros.”

### Static Smartports Macros

Static Smartports macros provide port configurations that you apply manually based on the device connected to the port. When you apply a static macro, the macro CLI commands are added to the existing port configuration. When there is a link-down event on the port, the switch does not remove the static macro configuration.

## Event Triggers

Auto Smartports uses event triggers to map macros to the source port of the event. The most common triggers are based on Cisco Discovery Protocol (CDP) messages received from another device. A CDP event trigger occurs when these devices are detected:

- Cisco switch
- Cisco router
- Cisco IP Phone
- Cisco Wireless Access Points, including autonomous and lightweight access points
- Cisco IP video surveillance camera
- Cisco digital media player

Additional event triggers for Cisco and third-party devices are user-defined MAC address groups, MAC authentication bypass (MAB) messages, 802.1x authentication messages, and Link Layer Discovery Protocol (LLDP) messages.

LLDP supports a set of attributes used to discover neighbor devices. These type, length, and value attributes and descriptions are referred to as TLVs. LLDP-supported devices use TLVs to receive and send information. This protocol advertises details such as device configuration information, capabilities, and identity. Auto Smartports uses the LLDP *system capabilities* TLV as the event trigger. Use the event trigger control feature to specify if the switch applies a macro based on the detection method, device type, or configured trigger.

For more information about configuring the LLDP system capabilities TLV attributes for Auto Smartports, see the “Configuring LLDP, LLDP-MED, and Wired Location Service” chapter in the switch-specific software configuration guides.

Auto Smartports can detect devices based on Dynamic Host Control Protocol (DHCP) options. There are predefined and customizable macros for devices detected through DHCP options.

For devices that do not support CDP, MAB, or 802.1x authentication, such as network printers, LLDP, or legacy Cisco Digital Media Players, you can configure a MAC address group with a MAC operationally unique identifier (OUI)-based trigger. You map the MAC address to a built-in or user-defined macro that has the desired configuration.

## User-Defined Files

You can designate a remote server location for user-defined macro files. You can then update and maintain one set of macro files for use by multiple switches across the network.

## Macro Persistence

The macro persistence feature causes macro configurations to remain enabled on the switch ports regardless of a link-down event. This eliminates multiple system log and configuration change notifications when the switch has link-up and link-down events or is a domain member or an end point in an EnergyWise network.

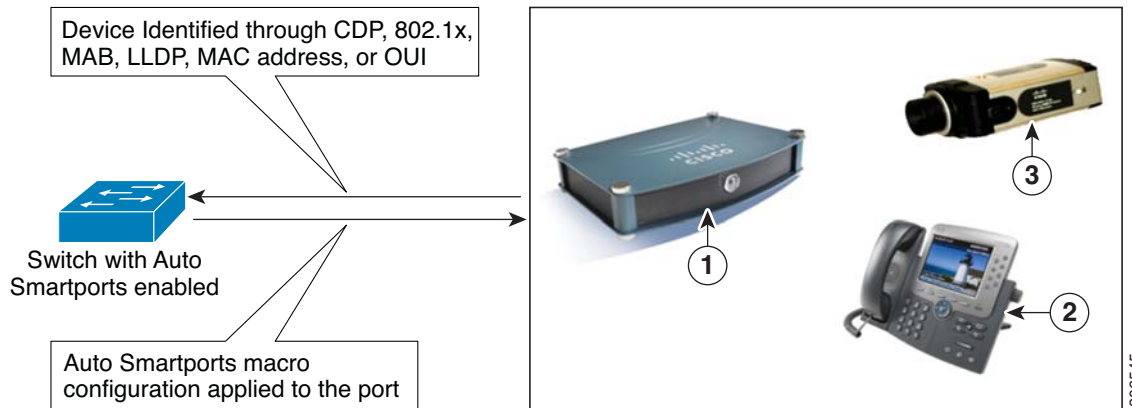
# Managing Macros

You can use the Cisco LAN management system (LMS) tools to deploy macros to switches. When you define a custom macro, you must also configure and associate a trigger for it. For information on configuring user-defined triggers, see the “[Configuring User-Defined Triggers for User-Defined Macros](#)” section on page 3-23.

## Auto Smartports and Cisco Medianet

Cisco Medianet enables intelligent services in the network infrastructure for a variety of video applications. A service of Medianet is autoprovisioning for Cisco Digital Media Players and Cisco IP video surveillance cameras through Auto Smartports. The switch identifies Cisco and third-party video devices by using CDP, 802.1x, MAB, LLDP, and MAC addresses (Figure 2-1). The switch applies the applicable macro to enable the appropriate VLAN, standard quality of service (QoS), and auto-QoS settings for the device. The switch also uses a built-in MAC address group to detect the legacy Cisco digital media player (DMP), based on an OUI of of4400 or 23ac00. You can also create custom user-defined macros for any video device.

**Figure 2-1 Cisco Medianet Deployment Example**



<b>1</b>	Digital media player	<b>3</b>	Cisco IP video surveillance camera
<b>2</b>	Cisco IP phone		

## Device Classifier

The device classifier (DC) feature is enabled by default on the switch.

The DC collects information from MAC-OUI and protocols such as CDP, LLDP, and DHCP to identify devices. You must enable CDP and LLDP on the switch. To make DHCP options information available to the DC, you must enable the DHCP snooping feature on the switch. The device attributes collected from these protocols are evaluated against a set of profiles available to the DC to find the best match. The best-matched profile is used for device identification.

Device-classifier uses profile definitions—built-in and default profiles. The built-in profiles contain the device profiles that are known to the Auto Smartports module, comprising a limited set of Cisco devices. They are built into Cisco IOS and cannot be changed. The default profiles are stored as a text file in nonvolatile storage and allow the DC to identify a much larger set of devices. The default profiles are updated as part of the Cisco IOS archive download.

When a new device is detected, the corresponding shell trigger executes the Auto Smartports configuration macro. Auto Smartports has built-in mappings for a large set of devices. You can use the commands described in the [“Configuring User-Defined Triggers for User-Defined Macros” section on page 3-23](#) to create new mappings. You can create the trigger mappings based on the profile name or device name that is provided by the DC.

For switch stacks, the DC runs only on the stack master. All the attributes for devices connected to stack master and stack member switch ports are available to the DC on the stack master. The device classification results are available only on the stack master. During a stack master failover event, the new stack master reclassifies all the devices. The device attributes for the existing and any new devices are available to the DC on the new stack master.

## Device Visibility Mode

The DC function is enabled on the switch by default. You can disable it by using the **no macro auto monitor** global configuration command. The DC feature provides **show** commands to display the devices that are connected to the switch. It also provides information about the physical port to which the device is connected, along with device MAC address and other vendor information. Only directly connected devices, such as another Layer 2 switch, are classified on nonaccess ports. On access ports that are connected to hubs, device classification is limited to 32 devices.

When you enable Auto Smartports, the DC is automatically enabled.





# Configuring Auto Smartports and Static Smartports Macros

---

- [Configuring Macros, page 3-1](#)
- [Displaying Macros, page 3-36](#)

## Configuring Macros

- [Auto Smartports Configuration Guidelines, page 3-2](#)
- [Enabling Auto Smartports Macros, page 3-3](#)
- [Default Auto Smartports Configuration, page 3-4](#)
- [Configuring Auto Smartports Parameter Values, page 3-7](#)
- [Configuring MAC Address Groups, page 3-8](#)
- [Configuring Macro Persistence, page 3-11](#)
- [Configuring Built-In Macro Options, page 3-12](#)
- [Creating User-Defined Event Triggers, page 3-16](#)
- [Configuring Event Trigger Control, page 3-19](#)
- [Configuring User-Defined Triggers for User-Defined Macros, page 3-23](#)
- [Configuring User-Defined Macros, page 3-24](#)
- [Applying Macros on a Switch, page 3-29](#)
- [Default Static Smartports Configuration, page 3-33](#)
- [Static Smartports Configuration Guidelines, page 3-33](#)
- [NEAT Configuration, page 3-33](#)
- [Applying Static Smartports Macros, page 3-34](#)

## Auto Smartports Configuration Guidelines

- You cannot 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** global configuration 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. Be sure to enter the commands that remove port security and Bridge Protocol Data Unit (BPDU) guard features *before* you enable Auto Smartports macros on a port.
- Do not configure port security when you enable device-specific Auto Smartports on the switch. The switch applies the appropriate port-based commands.
- 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. If you use EtherChannels, disable Auto Smartports on the EtherChannel interface by using the **no macro auto processing** interface configuration command.
- The built-in-macro default data VLAN is VLAN 1. The built-in macro 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** global configuration commands to configure the values.
- If you use the VLAN name in a macro, it must be the same name that is in the VLAN database for all switches in the VLAN Trunking Protocol (VTP) domain.
- Use the **show macro auto device** privileged EXEC command to display the default macros with the default parameter values, current values, and the configurable parameter list for each macro. You can also use the **show shell functions** privileged EXEC command to see the built-in-macro default values.
- To use 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**.
- For stationary devices that do not support CDP, MAB, or 802.1x authentication, such as network printers, configure a MAC address group with a MAC OUI-based trigger and map it to a user-defined macro with the desired configuration.
- An 802.1x-authentication-based trigger takes precedence over all other event triggers, such as Cisco Discovery Protocol (CDP) messages, Link Layer Discovery Protocol (LLDP) messages, or user-defined MAC address groups.
- The switch supports Auto Smartports macros only on directly connected devices. If multiple devices are connected, (for example, through a hub) the applied macro is associated with the first detected device.
- If authentication is enabled on a port, the switch ignores a MAC address trigger if authentication fails.

- When using MAC-address-based detection, ensure that Auto Smartports is enabled *only* on ports facing access devices and not on ports that face the network or an intermediate gateway switch.
- The order of CLI commands within the macro and the corresponding antimacro can be different.
- When the device identity is configured and the device is authenticated on a switch port, these RADIUS attributes could be downloaded:
  - VLAN ID and switch ACL name or number from the Cisco access control server (ACS)
  - ASP trigger name in an attribute-value (AV) pair.

After the AV pair is downloaded, the switch applies the macro on the port.

The downloaded VLAN ID or ACL name could conflict with the settings in the user-defined or builtin macro applied by the switch.

- Auto Smartports does not support lightweight access points in the Remote Edge Access Point (REAP) or Hybrid Remote Edge Access Point (HREAP) mode.

## Enabling Auto Smartports Macros

Follow this required procedure to enable macros globally on the switch.

Beginning in privileged EXEC mode:

	Command	Purpose
Step 1	<b>configure terminal</b>  <b>Example:</b> Switch# configure terminal	Enters global configuration mode.
Step 2	<b>macro auto global processing</b>  <b>Example:</b> Switch(config)# macro auto global processing	Globally enables macros on the switch.
Step 3	<b>end</b>  <b>Example:</b> Switch(config)# end	Returns to privileged EXEC mode.
Step 4	<b>show running-config</b>  <b>Example:</b> Switch# show running-config	Verifies that Auto Smartports is enabled.
Step 5	<b>copy running-config startup-config</b>  <b>Example:</b> Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

To return to the default setting, use the **no macro auto global processing** global configuration command.

To disable macros on a specific port, use the **no macro auto processing** interface configuration command.

You can use the **show macro auto device**, the **show shell functions**, and the **show shell triggers** privileged EXEC commands to display the event triggers and the built-in macros.

This example shows how to enable macros on the switch and then how to disable macros on a specific interface:

```
Switch(config)# macro auto global processing
Switch(config)# interface interface_id
Switch(config-if)# no macro auto processing
```

## Default Auto Smartports Configuration

- Auto Smartports is globally disabled and is enabled per interface.
- Macro persistence is globally disabled and is enabled per interface.
- Cisco IOS shell is enabled.
- The switch uses these built-in macros (the defaults) when Auto Smartports is enabled for the specific devices.

**Table 3-1** Device-Specific Built-In Macros

Macro Name	Description
CISCO_AP_AUTO_SMARTPORT	This macro applies the wireless access point macro for Cisco access points. It enables standard QoS, auto-QoS, and 802.1q encapsulated trunking. It configures the native VLAN on the interface. It also enables macro persistence so that the macro remains active after a link-down event.
CISCO_DMP_AUTO_SMARTPORT	This macro applies the digital media player macro for Cisco digital media players. It enables QoS trust, auto-QoS, port security, and spanning-tree protection. It configures the access VLAN for the interface and provides network protection from unknown unicast packets.  <b>Note</b> If you enter the <b>auto qos video media-player</b> interface configuration command, the switch automatically uses the CDP to detect the presence or absence of a Cisco digital media player.
CISCO_IPVSC_AUTO_SMARTPORT	This macro applies the IP camera macro for Cisco IP video surveillance cameras. It enables QoS trust, auto-QoS, port security, and spanning-tree protection. It configures the access VLAN for the interface and provides network protection from unknown unicast packets.
CISCO_LWAP_AUTO_SMARTPORT	This macro applies the lightweight wireless access point macro for Cisco lightweight wireless access points. It enables QoS, port security, storm control, DHCP snooping, and spanning-tree protection. It configures the access VLAN for the interface and provides network protection from unknown unicast packets.
CISCO_PHONE_AUTO_SMARTPORT	This macro applies the IP phone macro for Cisco IP phones. It enables QoS, port security, storm control, DHCP snooping, and spanning-tree protection. It also configures the access and voice VLANs for that interface.

**Table 3-1** Device-Specific Built-In Macros (continued)

Macro Name	Description
CISCO_ROUTER_AUTO_SMARTPORT	This macro applies the router macro for Cisco routers. It enables QoS and trunking with 802.1Q encapsulation and spanning-tree bridge protocol data unit (BPDU) protection.
CISCO_SWITCH_AUTO_SMARTPORT	This macro applies the switch macro for Cisco switches. It enables QoS and trunking with 802.1q encapsulation. It also configures the native VLAN on the interface.

Use these macros when the switch does not use device-specific macros.

**Table 3-2** Global and Custom Macros

Macro Name	Description
CISCO_CUSTOM_AUTO_SMARTPORT	This macro applies the per-port user-defined settings after the antimacro <sup>1</sup> is applied on a switch port. You specify the settings in the macro.
CISCO_LAST_RESORT_AUTO_SMARTPORT	This macro applies a per-port device-specific macro when the switch does not have built-in macro for the device. It has a basic configuration with a data VLAN.
CISCO_SWITCH_AAA_ACCOUNTING	This macro applies the authentication, authorization, and accounting (AAA) accounting settings.
CISCO_SWITCH_AAA_AUTHENTICATION	This macro applies the authentication, authorization, and accounting (AAA) authentication settings.
CISCO_SWITCH_AAA_AUTHORIZATION	This macro applies the authentication, authorization, and accounting (AAA) authorization settings.
CISCO_SWITCH_AUTO_IP_CONFIG	This macro applies the IP settings
CISCO_SWITCH_AUTO_PCI_CONFIG	This macro applies Payment Card Industry (PCI)-compliant settings.
CISCO_SWITCH_DOMAIN_NAME_CONFIG	This macro applies the domain name.
CISCO_SWITCH_ETHERCHANNEL_CONFIG	This macro applies the EtherChannel settings.
CISCO_SWITCH_HOSTNAME_CONFIG	This macro applies the hostname.
CISCO_SWITCH_HTTP_SERVER_CONFIG	This macro applies the HTTP server settings.
CISCO_SWITCH_LOGGING_SERVER_CONFIG	This macro applies the logging server settings.
CISCO_SWITCH_MGMT_VLAN_CONFIG	This macro applies the management VLAN settings.
CISCO_SWITCH_NAME_SERVER_CONFIG	This macro applies the name server settings.
CISCO_SWITCH_NTP_SERVER_CONFIG	This macro applies the Network Time Protocol (NTP) server settings.  <b>Note</b> If the Virtual Private Network (VPN) routing/forwarding instance (VRF) name is not configured, the <b>ntp server</b> global configuration command is not applied.

Table 3-2 Global and Custom Macros (continued)

Macro Name	Description
CISCO_SWITCH_RADIUS_SERVER_CONFIG	This macro applies the RADIUS server settings.
CISCO_SWITCH_SETUP_SNMP_TRAPS	This macro applies the Simple Network Management Protocol (SNMP) trap settings.
CISCO_SWITCH_SETUP_USR_CONFIG	This macro applies user settings.
CISCO_SWITCH_SNMP_SOURCE_CONFIG	This macro applies the SNMP source interface settings.
CISCO_SWITCH_TACACS_SERVER_CONFIG	This macro applies the TACACS server settings.
CISCO_SWITCH_USER_PASS_CONFIG	This macro applies the username and password settings.

1. The antimacro is the portion of the applied macro that removes the macro at a link-down event.

- The switch applies the CISCO\_PHONE\_AUTO\_SMARTPORT macro to Cisco IP phones.
- The access point macros have these enhancements:
  - The switch determines the access point type (autonomous or lightweight) and then applies the appropriate macro.
  - To reduce overrun errors at the ingress interface on an access point Ethernet receiver, the switch adds the QoS bandwidth setting to the access point macros when it receives a CDP message with the auto-QoS type, length, and value attributes (TLVs). QoS derives the bandwidth value from the auto-QoS TLVs.

If the CDP messages does not have the auto-QoS TLVs, the switch does not add the bandwidth setting to the macros.



**Note**

If you do not upgrade the access point image to one that has the auto-QoS TLVs, the switch does not add the bandwidth setting to the access point macros. When you configure the bandwidth before the link to the receiver goes down, the setting is removed when the link comes up.

If you add a macro command that sets the QoS bandwidth and the switch applies the macro to an access point that does not support the auto-QoS TLVs, the command is not applied to the access point. We recommend that you create a user-defined macro without that command.

- When a Catalyst 3750-E and 3560-E switch is connected to a Cisco Aironet 1250 access point, the switch applies a power setting to allocate up to 20 W.

When a switch running Cisco IOS Release 15.0(2)EX applies the CISCO\_DMP\_AUTO\_SMARTPORT macro to a CDP-capable digital media player, it generates an auto-QoS configuration for the digital media player.

The access point macros support the Control and Provisioning of Wireless Access Points (CAPWAP) protocol.

## Configuring Auto Smartports Parameter Values

The switch automatically maps from event triggers to built-in device-specific macros. You can follow this optional procedure to replace macro default parameter values with values that are specific to your switch.

Beginning in privileged EXEC mode:

	Command	Purpose
Step 1	<b>show macro auto device</b>  <b>Example:</b> Switch# show macro auto device	Displays the macro default parameter values.
Step 2	<b>configure terminal</b>  <b>Example:</b> Switch# configure terminal	Enters global configuration mode.
Step 3	<b>macro auto device {access-point   ip-camera   lightweight-ap   media-player   phone   router   switch} [parameter=value]</b>  <b>Example:</b> Switch(config)# macro auto device router	Replaces the specified macro default parameter values.  Enter new values in the form of a name-value pair separated by spaces: [ <b>&lt;name1&gt;=&lt;value1&gt; &lt;name2&gt;=&lt;value2&gt;...</b> ].  You can enter the VLAN ID or the VLAN name when specifying VLAN parameter values.  Default values are shown for each macro default parameter value. <ul style="list-style-type: none"> <li>• <b>access-point</b> <i>NATIVE_VLAN=1</i></li> <li>• <b>ip-camera</b> <i>ACCESS_VLAN=1</i></li> <li>• <b>lightweight-ap</b> <i>ACCESS_VLAN=1</i></li> <li>• <b>media-player</b> <i>ACCESS_VLAN=1</i></li> <li>• <b>phone</b> <i>ACCESS_VLAN=1 VOICE_VLAN=2</i></li> <li>• <b>router</b> <i>NATIVE_VLAN=1</i></li> <li>• <b>switch</b> <i>NATIVE_VLAN=1</i></li> </ul> <b>Note</b> You must enter the correct parameter name (for example, <i>VOICE_VLAN</i> ) because this text string must match the text string in the built-in macro definition.
Step 4	<b>end</b>  <b>Example:</b> Switch(config)# end	Returns to privileged EXEC mode.

	Command	Purpose
Step 5	<b>show macro auto device</b>  <b>Example:</b> Switch# show macro auto device	Verifies your entries.
Step 6	<b>copy running-config startup-config</b>  <b>Example:</b> Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

This example shows how to see the IP phone macro parameter values and how to change the default voice VLAN to 20. When you change the default values, they are not immediately applied on the interfaces with existing applied macros. The configured values are applied at the next link-up event. Note that the exact text string was used for VOICE\_VLAN. The entry is case sensitive.

```
Switch# show macro auto device phone
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

Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# macro auto device phone VOICE_VLAN=20
Switch(config)# end
Switch# show macro auto device phone
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:voice_vlan=20
```

## Configuring MAC Address Groups

For devices such as printers that do not support neighbor discovery protocols such as CDP or LLDP, use the MAC-address-based trigger configurations. This optional procedure requires these steps:

1. Configure a MAC-address-based trigger by using the **macro auto mac-address** global configuration command.
2. Associate the MAC address trigger to a built-in or a user-defined macro by using the **macro auto execute** global configuration command.



### Note

A switch running Cisco IOS Release 12.2(58)SE or later applies the macro as soon as it learns a MAC address in the MAC address group.

A switch running a release earlier than Cisco IOS Release 12.2(58)SE applies the macro after it learns a MAC address in the MAC address group and waits 60 seconds.



Beginning in privileged EXEC mode:

	Command	Purpose
Step 1	<b>configure terminal</b>  <b>Example:</b> Switch# configure terminal	Enters global configuration mode.
Step 2	<b>macro auto mac-address-group name</b>  <b>Example:</b> Switch(config)# macro auto mac-address-group address_trigger	Specifies the group name, and enter MAC address configuration mode.
Step 3	<b>[mac-address list list]   [oui [list list   range start-value size number]]</b>  <b>Example:</b> Switch(config-addr-grp-mac)# mac-address list 2222.3333.3334 22.33.44 a.b.c	Configures a list of MAC addresses separated by spaces.  Specify an operationally unique identifier (OUI) <b>list</b> or <b>range</b> . The OUI is the first three bytes of the MAC address and identifies the manufacturer of the product. Specifying the OUI allows devices that do not support neighbor discovery protocols to be recognized. <ul style="list-style-type: none"> <li>• <b>list</b>—Enter an OUI list in hexadecimal format separated by spaces.</li> <li>• <b>range</b>—Enter the starting OUI hexadecimal value (<i>start-value</i>).</li> <li>• <b>size</b>—Enter the length of the range (<i>number</i>) from 1 to 5 to create a list of sequential addresses.</li> </ul>
Step 4	<b>exit</b>  <b>Example:</b> Switch(mac-adress-config) exit	Returns to configuration mode.
Step 5	<b>macro auto execute address_trigger built-in macro name</b>  <b>Example:</b> Switch(config)# macro auto execute address_trigger builtin CISCO_PHONE_AUTO_SMARTPORT	Maps the MAC address-group trigger to a built-in or user-defined macro.  The MAC address trigger is applied to an interface after 65 seconds. The switch uses this hold time to apply a CDP- or LLDP-based event trigger instead of the MAC address trigger.
Step 6	<b>end</b>  <b>Example:</b> Switch(config)# end	Returns to privileged EXEC mode.

	Command	Purpose
Step 7	<b>show macro auto address-group <i>name</i></b>  <b>Example:</b> Switch# show macro auto address-group group2	Verifies your entries.
Step 8	<b>copy running-config startup-config</b>  <b>Example:</b> Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

This example shows how to create a MAC-address-group event trigger called *address\_trigger*, map it to the built in phone macro, and verify your entries:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# macro auto mac-address-group address_trigger
Switch(config-addr-grp-mac)# mac-address list 2222.3333.3334 22.33.44 a.b.c
Switch(config-addr-grp-mac)# oui list 455555 233244
Switch(config-addr-grp-mac)# oui range 333333 size 2
Switch(config-addr-grp-mac)# exit
Switch(config)# macro auto execute address_trigger builtin CISCO_PHONE_AUTO_SMARTPORT
Switch(config)# end
Switch# show running configuration | include macro
macro auto mac-address-group address_trigger
mac auto mac-address-group hel
mac auto execute address_trigger builtin CISCO_PHONE_AUTO_SMARTPORT
macro description CISCO_DMP_EVENT
mac description CISCO_SWITCH_EVENT
!
<output truncated>
```

The example shows how to create an OUI list with five sequential addresses starting with 00000A and how to verify your entries:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# macro auto mac-address-group size5ouilist
Switch(config-addr-grp-mac)# oui range 00000A size 5
Switch(config-addr-grp-mac)# exit
Switch(config)# mac auto execute size5-ouilist builtin macro
Switch(config)# macro auto execute address_trigger builtin CISCO_PHONE_AUTO_SMARTPORT
Switch(config)# end
Switch# show running configuration | include oui
oui list 00000E
oui list 00000D
oui list 00000C
oui list 00000B
oui list 00000A
```

## Configuring Macro Persistence

When you enable Auto Smartports on the switch, by default the macro configuration is applied at a link-up event and removed at a link-down event. When you enable macro persistence, the configuration is applied at link-up and is not removed at link-down. The applied configuration remains. Macro persistence remains configured after a reboot if you have saved the running configuration file.

Follow this optional procedure so that enable macros remain active on the switch after a link-down event.

Beginning in privileged EXEC mode:

	Command	Purpose
Step 1	<b>configure terminal</b>  <b>Example:</b> Switch# <code>configure terminal</code>	Enters global configuration mode.
Step 2	<b>interface <i>interface-id</i></b>  <b>Example:</b> Switch(config)# <code>interface gigabitethernet 2/0/1</code>	Specifies an interface and enters interface configuration mode.
Step 3	<b>macro auto sticky</b>  <b>Example:</b> Switch(config-if)# <code>macro auto port sticky</code>	Enables macros to remain active on the interface after a link-down event.
Step 4	<b>end</b>  <b>Example:</b> Switch(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	<b>show running-config interface <i>interface-id</i></b>  <b>Example:</b> Switch# <code>show running-config interface gigabit ethernet 2/0/1</code>	Verifies your entries.
Step 6	<b>copy running-config startup-config</b>  <b>Example:</b> Switch# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

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

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet 2/0/1
Switch(config-if)# macro auto port sticky
Switch(config-if)# exit
Switch(config)# end
```

```
Switch# show running-config interface gigabitethernet 2/0/1
Building configuration...

Current configuration : 243 bytes
!
<output truncated>
!
interface GigabitEthernet2/0/1
 srr-queue bandwidth share 1 30 35 5
 queue-set 2
 priority-queue out
 mls qos trust device cisco-phone
 mls qos trust cos
 macro auto port sticky
  service-policy input AUTOQOS-ENHANCED-CISCOPHONE-POLICY
end

<output truncated>

Switch#
```

## Configuring Built-In Macro Options

Use this procedure to map event triggers to built-in macros and to replace the built-in macro default parameters with values that are specific to your switch. If you need to *replace* default parameters values in a macro, use the **macro auto device** global configuration command. All commands in this procedure are optional.

Beginning in privileged EXEC mode:

	Command	Purpose
Step 1	<b>configure terminal</b>	Enters global configuration mode.
	<b>Example:</b> Switch# configure terminal	
Step 2	<b>macro auto execute event trigger builtin built-in macro name [parameter=value] [parameter=value]</b>	Defines mapping from an event trigger to a built-in macro. Specify an <i>event trigger</i> : <ul style="list-style-type: none"> <li>• CISCO_CUSTOM_EVENT</li> <li>• CISCO_DMP_EVENT</li> <li>• CISCO_IPVSC_EVENT</li> <li>• CISCO_LAST_RESORT_EVENT</li> <li>• CISCO_PHONE_EVENT</li> <li>• CISCO_ROUTER_EVENT</li> <li>• CISCO_SWITCH_EVENT</li> <li>• CISCO_WIRELESS_AP_EVENT</li> <li>• CISCO_WIRELESS_LIGHTWEIGHT_AP_EVENT</li> <li>• WORD—Apply a user-defined event trigger.</li> </ul> Specify a <b>builtin</b> <i>built-in macro name</i> : Enter new values in the form of <i>name value pair</i> separated by spaces: [ <i>&lt;name1&gt;=&lt;value1&gt; &lt;name2&gt;=&lt;value2&gt;...</i> ]. Default values are shown exactly as they should be entered. <ul style="list-style-type: none"> <li>• CISCO_AP_AUTO_SMARTPORT Specify the parameter value: <i>NATIVE_VLAN=1</i>.</li> <li>• CISCO_DMP_AUTO_SMARTPORT Specify the parameter value: <i>ACCESS_VLAN=1</i>.</li> <li>• CISCO_IPVSC_AUTO_SMARTPORT Specify the parameter value: <i>ACCESS_VLAN=1</i>.</li> <li>• CISCO_LWAP_AUTO_SMARTPORT Specify the parameter value: <i>ACCESS_VLAN=1</i>.</li> <li>• CISCO_PHONE_AUTO_SMARTPORT Specify the parameter values: <i>ACCESS_VLAN=1</i> and <i>VOICE_VLAN=2</i>.</li> <li>• CISCO_ROUTER_AUTO_SMARTPORT Specify the parameter value: <i>NATIVE_VLAN=1</i>.</li> <li>• CISCO_SWITCH_AUTO_SMARTPORT Specify the parameter value: <i>NATIVE_VLAN=1</i>.</li> </ul>
	<b>Example:</b> Switch(config)# macro auto execute CISCO_PHONE_EVENT builtin CISCO_PHONE_AUTO_SMARTPORT ACCESS_VLAN=10 VOICE_VLAN=20	

	Command	Purpose
Step 3	<b>remote</b> <i>url</i>  <b>Example:</b> Switch(config)# remote nvram://user:password@/C/macros	Specifies a remote server location for the remote macro file: <ul style="list-style-type: none"> <li>The syntax for the local flash file system on the standalone switch or the stack master: <b>flash:</b></li> <li>The syntax for the local flash file system on a stack member: <b>flash member number:</b></li> <li>The syntax for the FTP: <b>ftp:</b>[[//username[:password]@location]/directory]/filename</li> <li>The syntax for an HTTP server: <b>http:</b>[[username:password]@]{hostname   host-ip}/[directory]/filename</li> <li>The syntax for a secure HTTP server: <b>https:</b>[[username:password]@]{hostname   host-ip}/[directory]/filename</li> <li>The syntax for NVRAM: <b>nvram:</b>[[username:password]@]/[directory]/filename</li> <li>The syntax for the Remote Copy Protocol (RCP): <b>rcp:</b>[[//username@location]/directory]/filename</li> <li>The syntax for the Secure Copy Protocol (SCP): <b>scp:</b>[[//username@location]/directory]/filename</li> <li>The syntax for the TFTP: <b>tftp:</b>[[//location]/directory]/filename</li> </ul>
Step 4	<b>end</b>  <b>Example:</b> Switch(config)# end	Returns to privileged EXEC mode.
Step 5	<b>show running-config</b>  <b>Example:</b> Switch# show running-config	Verifies that Auto Smartports is enabled.
Step 6	<b>copy running-config startup-config</b>  <b>Example:</b> Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

This example shows how to use two built-in macros to connect 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:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#!!! the next command modifies the access and voice vlans
Switch(config)#!!! for the built in Cisco IP phone auto smartport macro
Switch(config)# macro auto execute CISCO_PHONE_EVENT builtin CISCO_PHONE_AUTO_SMARTPORT
ACCESS_VLAN=10 VOICE_VLAN=20
Switch(config)#
Switch(config)#!!! the next command modifies the Native vlan used for inter switch trunks
```

```

Switch(config)# macro auto execute CISCO_SWITCH_EVENT builtin CISCO_SWITCH_AUTO_SMARTPORT
NATIVE_VLAN=10
Switch(config)#
Switch(config)#! the next command enables auto smart ports globally
Switch(config)# macro auto global processing
Switch(config)#
Switch(config)# exit

Switch# !!! here is the running configuration of the interface connected
Switch# !!! to another Cisco Switch after the Macro is applied
Switch#
Switch# 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 configure the remote macro for native VLAN 5.

- a. Configure the remote macro in the macro.txt file.
- b. Use the **macro auto execute** configuration command to specify the remote location for the macro file.

#### Macro.txt file

```

if [[ $LINKUP -eq YES ]]; then
  conf t
    interface $INTERFACE
      macro description $TRIGGER
      auto qos voip trust
      switchport trunk encapsulation dot1q
      switchport trunk native vlan $NATIVE_VLAN
      switchport trunk allowed vlan ALL
      switchport mode trunk
    exit
  end
else
  conf t
    interface $INTERFACE
      no macro description
      no auto qos voip trust
      no switchport mode trunk
      no switchport trunk encapsulation dot1q
      no switchport trunk native vlan $NATIVE_VLAN
      no switchport trunk allowed vlan ALL
    exit
  end

Switch(config)# macro auto execute CISCO_SWITCH_EVENT remote tftp://<ip_address>/macro.txt
NATIVE_VLAN=5

Switch# show running configuration | include macro
macro auto execute CISCO_SWITCH_EVENT remote tftp://<ip_address>/macro.txt

```

```
NATIVE_VLAN=5
Switch#
```

## Creating User-Defined Event Triggers

When using MAB or 802.1x authentication as an event trigger, create a trigger that corresponds to the Cisco attribute-value pair (*auto-smart-port=event trigger*) sent by the RADIUS server. This procedure is optional.

Beginning in privileged EXEC mode:

	Command	Purpose
Step 1	<b>configure terminal</b>  <b>Example:</b> Switch# configure terminal	Enters global configuration mode.
Step 2	<b>shell trigger <i>identifier description</i></b>  <b>Example:</b> Switch(config)# shell trigger RADIUS_MAB_EVENT MAC_AuthBypass Event	Specifies the event trigger identifier and description. The identifier should have no spaces or hyphens between words.
Step 3	<b>end</b>  <b>Example:</b> Switch(config)# end	Returns to privileged EXEC mode.
Step 4	<b>show shell triggers</b>  <b>Example:</b> Switch# show shell triggers	Displays the event triggers on the switch.
Step 5	<b>copy running-config startup-config</b>  <b>Example:</b> Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

This example shows how to map a user-defined event trigger called RADIUS\_MAB\_EVENT to the built-in macro CISCO\_AP\_AUTO\_SMARTPORT, to replace the default VLAN with VLAN 10, and how to verify the entries.

- a. Connect the device to a MAB-enabled switch port.
- b. On the RADIUS server, set the attribute-value pair to **auto-smart-port=RADIUS\_MAB\_EVENT**.
- c. On the switch, create the event trigger RADIUS\_MAB\_EVENT.
- d. The switch recognizes the attribute-value pair=RADIUS\_MAB\_EVENT response from the RADIUS server and applies the macro CISCO\_AP\_AUTO\_SMARTPORT.

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
```



```

Switch(config)# !!! create a user defined trigger and map
Switch(config)# !!! a system defined macro to it
Switch(config)# !!! first create the trigger event
Switch(config)# shell trigger RADIUS_MAB_EVENT MAC_AuthBypass Event
Switch(config)#
Switch(config)#!!! map a system defined macro to the trigger event
Switch(config)# macro auto execute RADIUS_MAB_EVENT builtin ?
    CISCO_AP_AUTO_SMARTPORT          Configure native vlan and trust cos
    CISCO_CUSTOM_AUTOSMARTPORT       Configure user defined parameters
    CISCO_DMP_AUTO_SMARTPORT         Configure access vlan, qos and port-security
    CISCO_IP_CAMERA_AUTO_SMARTPORT   Configure access vlan, qos and port-security
    CISCO_LAST_RESORT_SMARTPORT      Configure access vlan
    CISCO_LWAP_AUTO_SMARTPORT        Configure native vlan, qos, port-security and
                                      storm-control
    CISCO_PHONE_AUTO_SMARTPORT       Configure access vlan, voice vlan, trust
                                      device, interface bandwidth, port-security
    CISCO_ROUTER_AUTO_SMARTPORT      Configure native vlan, spanning tree
                                      port-fast,trunk mode and trust dscp
    CISCO_SWITCH_AUTO_SMARTPORT      Configure native vlan, trunk mode
Switch(config)# macro auto execute RADIUS_MAB_EVENT builtin CISCO_AP_AUTO_SMARTPORT
ACCESS_VLAN=10
Switch(config)# exit
Switch# show shell triggers
User defined triggers
-----
Trigger Id: RADIUS_MAB_EVENT
Trigger description: MAC_AuthBypass Event
Trigger environment:
Trigger mapping function: CISCO_AP_SMARTPORT
<output truncated>

```

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

```

Switch# show shell triggers
User defined triggers
-----
Built-in triggers
-----
Trigger Id: CISCO_AUTO_CONSOLE_EVENT
Trigger namespace: ASP_TRIG_GLOBAL_FUTURE
Trigger description: Console password
Trigger mapping function:
Parameters:
Current version: 1
Negotiated version: 1
Mapped Function: CISCO_AUTO_CONSOLE_PASSWD

Trigger Id: CISCO_AUTO_TIMEZONE_CONFIG
Trigger namespace: ASP_TRIG_GLOBAL_USR
Trigger description: timezone parameters
Trigger mapping function:
Parameters:
Current version: 1
Negotiated version: 1
Mapped Function: CISCO_AUTO_TIMEZONE

Trigger Id: CISCO_CUSTOM_EVENT
Trigger namespace: ASP_TRIG
Trigger description: Custom macro event to apply user defined configuration
Trigger mapping function:
Parameters:
Current version: 1
Negotiated version: 1

```

```

Mapped Function: CISCO_CUSTOM_AUTOSMARTPORT

Trigger Id: CISCO_DMP_EVENT
Trigger namespace: ASP_TRIG
Trigger description: Digital media-player device event to apply port configuration
on
Trigger mapping function:
Parameters: ACCESS_VLAN=1
Current version: 1
Negotiated version: 1
Mapped Function: CISCO_DMP_AUTO_SMARTPORT

Trigger Id: CISCO_IPVSC_EVENT
Trigger namespace: ASP_TRIG
Trigger description: IP-camera device event to apply port configuration
Trigger mapping function:
Parameters: ACCESS_VLAN=1
Current version: 1
Negotiated version: 1
Mapped Function: CISCO_IP_CAMERA_AUTO_SMARTPORT

Trigger Id: CISCO_LAST_RESORT_EVENT
Trigger namespace: ASP_TRIG
Trigger description: Last resort event to apply port configuration
Trigger mapping function:
Parameters: ACCESS_VLAN=1
Current version: 1
Negotiated version: 1
Mapped Function: CISCO_LAST_RESORT_SMARTPORT

Trigger Id: CISCO_PHONE_EVENT
Trigger namespace: ASP_TRIG
Trigger description: IP-phone device event to apply port configuration
Trigger mapping function:
Parameters: ACCESS_VLAN=1 VOICE_VLAN=2
Current version: 1
Negotiated version: 1
Mapped Function: CISCO_PHONE_AUTO_SMARTPORT

```

<output truncated>

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

```

Switch# show shell functions
#User defined functions:

#Built-in functions:
function CISCO_AP_AUTO_SMARTPORT () {
    if [[ $LINKUP -eq 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 -eq 0 ]]; then
                default srr-queue bandwidth limit
            else
                srr-queue bandwidth limit $LIMIT
            fi
        fi
    fi
}

```

```

        fi
        if [[ $SW_POE -eq YES ]]; then
            if [[ $AP125X -eq AP125X ]]; then
                macro description AP125X
                macro auto port sticky
                power inline port maximum 20000
            fi
        fi
    fi
    exit
end
fi
if [[ $LINKUP -eq 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 -eq NO ]]; then
                no switchport mode
                no switchport trunk encapsulation
            fi
            if [[ $STICKY -eq YES ]]; then
                if [[ $SW_POE -eq YES ]]; then
                    if [[ $AP125X -eq AP125X ]]; then
                        no macro auto port sticky
                        no power inline port maximum
                    fi
                fi
            fi
        fi
    fi
    exit
end
fi
}
<output truncated>

```

## Configuring Event Trigger Control

Use event trigger control to specify when the switch applies macros. By default, the switch maps built-in and user-defined macros to these triggers:

- Detection method (for example, MAC address groups, MAB messages, 802.1x authentication messages, and LLDP messages)
- Device type (for example, Cisco switches, routers, and IP phones)
- Configured triggers

To configure a MAC address-based trigger, use the **macro auto global control device** global or interface configuration command.

When you select triggers, the switch applies macros only when those triggers map to macros.

## On a Switch

Beginning in privileged EXEC mode:

	Command	Purpose
Step 1	<b>configure terminal</b>  <b>Example:</b> Switch# configure terminal	Enters global configuration mode.
Step 2	<b>macro auto global control { device [access-point] [ip-camera] [lightweight-ap] [mac-address] [media-player] [phone] [router] [switch]   trigger [last-resort] }</b>  <b>Example:</b> Switch(config)# macro auto global control mac-address	Specifies when the switch applies a macro based on the detection method, device type, or trigger. <ul style="list-style-type: none"> <li>• <b>device</b>—Uses one or more of these devices as an event trigger: <ul style="list-style-type: none"> <li>– (Optional) <b>access-point</b>—Autonomous access point</li> <li>– (Optional) <b>ip-camera</b>—Cisco IP video surveillance camera</li> <li>– (Optional) <b>lightweight-ap</b>—Lightweight access point</li> <li>– (Optional) <b>mac-address</b>—Device MAC address</li> <li>– (Optional) <b>media-player</b>—Digital media player</li> <li>– (Optional) <b>phone</b>—Cisco IP phone</li> <li>– (Optional) <b>router</b>—Cisco router</li> <li>– (Optional) <b>switch</b>—Cisco switch</li> </ul> </li> <li>• <b>trigger</b>—Uses a specific event trigger. <ul style="list-style-type: none"> <li>– (Optional) <b>last-resort</b>—Last-resort trigger.</li> </ul> </li> </ul> <p>By 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.</p>
Step 3	<b>end</b>  <b>Example:</b> Switch(config)# end	Returns to privileged EXEC mode.
Step 4	<b>show running-config</b>  <b>Example:</b> Switch# show running-config	Verifies that Auto Smartports is enabled.
Step 5	<b>copy running-config startup-config</b>  <b>Example:</b> Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

To configure the switch to apply the CISCO\_AP\_AUTO\_SMARTPORT macro only when it detects an autonomous access point:

```
Switch# configure terminal
```

```
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# macro auto global control device access-point
Switch(config)# end
```

To configure the switch to apply the CISCO\_AP\_AUTO\_SMARTPORT macro only when it detects an autonomous access point or a Cisco IP phone:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# macro auto global control device access-point phone
Switch(config)# end
```

## On an Interface

Beginning in privileged EXEC mode:

	Command	Purpose
Step 1	<b>configure terminal</b>  <b>Example:</b> Switch# configure terminal	Enters global configuration mode.
Step 2	<b>interface</b> <i>interface_id</i>  <b>Example:</b> Switch(config)# interface gigabitethernet 2/0/2	Specifies an interface and enters interface configuration mode.
Step 3	<b>macro auto control</b> { <b>device</b> [ <b>access-point</b> ] [ <b>ip-camera</b> ] [ <b>lightweight-ap</b> ] [ <b>mac-address</b> ] [ <b>media-player</b> ] [ <b>phone</b> ] [ <b>router</b> ] [ <b>switch</b> ]   <b>trigger</b> [ <b>last-resort</b> ] }  <b>Example:</b> Switch(config)# macro auto global control mac-address	Specifies when the switch applies a macro based on the detection method, device type, or trigger. <ul style="list-style-type: none"> <li>• <b>device</b>—Use one or more of these devices as an event trigger: <ul style="list-style-type: none"> <li>– (Optional) <b>access-point</b>—Autonomous access point</li> <li>– (Optional) <b>ip-camera</b>—Cisco IP video surveillance camera</li> <li>– (Optional) <b>lightweight-ap</b>—Lightweight access point</li> <li>– (Optional) <b>mac-address</b>—Device MAC address</li> <li>– (Optional) <b>media-player</b>—Digital media player</li> <li>– (Optional) <b>phone</b>—Cisco IP phone</li> <li>– (Optional) <b>router</b>—Cisco router</li> <li>– (Optional) <b>switch</b>—Cisco switch</li> </ul> </li> <li>• <b>trigger</b>—Use a specific event trigger. <ul style="list-style-type: none"> <li>– (Optional) <b>last-resort</b>—Last-resort trigger.</li> </ul> </li> </ul> <p>By 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.</p>

	Command	Purpose
Step 4	<b>exit</b>  <b>Example:</b> Switch(config-if)# exit	Returns to global configuration mode.
Step 5	<b>end</b>  <b>Example:</b> Switch(config)# end	Returns to privileged EXEC mode.
Step 6	<b>show macro auto</b>  <b>Example:</b> Switch# show macro auto	Verifies your entries.
Step 7	<b>copy running-config startup-config</b>  <b>Example:</b> Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

To configure the switch to apply the CISCO\_AP\_AUTO\_SMARTPORT or CISCO\_PHONE\_AUTO\_SMARTPORT macro only when it detects a lightweight access point or a Cisco IP phone:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet 5/0/1
Switch(config-if)# macro auto control device lightweight-ap phone
Switch(config-if)# exit
Switch(config)# end
```

## Configuring User-Defined Triggers for User-Defined Macros

Beginning in privileged EXEC mode, follow these steps to configure a user-defined event trigger for a user-defined macro.

	Command	Purpose
Step 1	<b>configure terminal</b>  <b>Example:</b> Switch# configure terminal	Enters global configuration mode.
Step 2	<b>macro auto trigger <i>trigger_name</i></b> <b>[device   exit   no   profile]</b>  <b>Example:</b> Switch(config)# macro auto trigger DMP Switch(config-macro-trigger)# profile mediaplayer-DMP	Enters macro trigger configuration mode. In that mode, you can specify a user-defined event trigger that maps to a user-defined macro. <ul style="list-style-type: none"> <li>• <b>device</b>—Specifies a device name to map to the named trigger.</li> <li>• <b>exit</b>—Exits device group configuration mode.</li> <li>• <b>no</b>—Removes any configured device.</li> <li>• <b>profile</b>—Specifies a profile name to map to the named trigger.</li> </ul>
Step 3	<b>end</b>  <b>Example:</b> Switch(config)# end	Returns to privileged EXEC mode.
Step 4	<b>show running-config</b>  <b>Example:</b> Switch# show running-config	Verifies that Auto Smartports is enabled.
Step 5	<b>copy running-config startup-config</b>  <b>Example:</b> Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

### Example: User-Defined Event Trigger

This example shows how to configure a user-defined event trigger called DMP\_EVENT:

```
Switch(config)# macro auto trigger DMP_EVENT mediaplayer
```

## Configuring User-Defined Macros

The Cisco IOS shell has basic scripting capabilities for configuring user-defined macros. These macros can contain multiple lines and can include any CLI command. You can also define variable-substitution, conditionals, functions, and triggers within the macro. This procedure is optional.

**Note**

---

When configuring macros, you must enter a description. If the link is down (command **\$LINKUP -eq NO**), you must enter the **no macro description** command. These commands are mandatory for Auto Smartports to work.

---

Beginning in privileged EXEC mode, follow these steps to map a user-defined event trigger to a user-defined macro.



	Command	Purpose
Step 1	<b>configure terminal</b>  <b>Example:</b> Switch# configure terminal	Enters global configuration mode.
Step 2	<b>macro auto execute event trigger</b> <i>[parameter=value] {function contents}</i>  <b>Example:</b> <pre>Switch(config)# macro auto execute DMP_EVENT { if [[ \$LINKUP -eq 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 -eq NO ]]; then conf t interface \$INTERFACE no macro description no switchport access vlan 1 if [[ \$AUTH_ENABLED -eq 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 }</pre>	Specifies a user-defined macro that maps to an event trigger.  <i>{function contents}</i> Specify 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.  (Optional) <i>parameter=value</i> —Replace default values that begin with \$, and enter new values in the form of name value pair separated by spaces: [ <i>&lt;name1&gt;=&lt;value1&gt; &lt;name2&gt;=&lt;value2&gt;...</i> ].

	Command	Purpose
Step 3	<b>end</b>  <b>Example:</b> Switch(config)# end	Returns to privileged EXEC mode.
Step 4	<b>show running-config</b>  <b>Example:</b> Switch# show running-config	Verifies that Auto Smartports is enabled.
Step 5	<b>copy running-config startup-config</b>  <b>Example:</b> Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

### Example: User-Defined Event Trigger and Macro

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 in the CLI example.
4. The switch recognizes the attribute-value pair=DMP\_EVENT response from the RADIUS server and applies the macro associated with this event trigger.

```
Switch(config)# shell trigger DMP_EVENT mediaplayer
Switch(config)# macro auto execute DMP_EVENT {
if [[ $LINKUP -eq 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 -eq NO ]]; then
conf t
interface $INTERFACE
no macro description
no switchport access vlan 1
if [[ $AUTH_ENABLED -eq 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
}
Switch(config)# end

```

## Example: Last-Resort Event Trigger and Macro

The last-resort event trigger is applied when a device is classified by Device Classifier using CDP, LLDP or DHCP, but has no built-in macro.

To map the CISCO\_LAST\_RESORT\_AUTO\_SMARTPORT macro to the last-resort trigger:

```
Switch(config)# macro auto global control trigger last-resort
```

CISCO\_LAST\_RESORT\_AUTO\_SMARTPORT macro:

```

function CISCO_LAST_RESORT_SMARTPORT () {
  if [[ $LINKUP -eq YES ]]; then
    conf t
      interface $INTERFACE
        macro description $TRIGGER
        switchport access vlan $ACCESS_VLAN
        switchport mode access
        load-interval 60
        no shutdown
      exit
    end
  fi
  if [[ $LINKUP -eq NO ]]; then
    conf t
      interface $INTERFACE
        no macro description
        no switchport access vlan $ACCESS_VLAN
        no switchport mode access
        no load-interval 60
      exit
    end
  fi
}

```

Example of user-defined mapping of a MAC address trigger to map a last-resort macro:

```

Switch(config)#macro auto mac
Switch(config)#macro auto mac-address-group Laptop
Switch(config-addr-grp-mac)#mac-address list 0000.0011.2233
Switch(config-addr-grp-mac)#exit
Switch(config)#macro auto execute laptop builtin CISCO_LAST_RESORT_SMARTPORT
ACCESS_VLAN=10
Switch(config)#end
Switch#

```

## Example: Custom Event Trigger and CISCO\_CUSTOM\_AUTO\_SMARTPORT Macro

Default CISCO\_CUSTOM\_AUTO\_SMARTPORT macro:

```

if [[ $LINKUP -eq YES ]]; then
  conf t

```

```

        interface $INTERFACE
        exit
    end
fi
if [[ $LINKUP -eq NO ]]; then
    conf t
        interface $INTERFACE
        exit
    end
fi

```

To create a user-defined macro with the same name as the custom macro, override the CISCO\_CUSTOM\_AUTO\_SMARTPORT macro, and set the parameters for your switch, including the mapping from an event trigger to the macro.

```

Config# macro auto execute CISCO_CUSTOM_EVENT {
    if [[ $LINKUP -eq YES ]]; then
        conf t
            interface $INTERFACE
            description asp3-link-UP i.e. Custom Macro OFF
            no macro description
            switchport
            switchport mode access
            switchport access vlan $ACCESS_VLAN
            spanning-tree portfast
            exit
        end
    fi
    if [[ $LINKUP -eq NO ]]; then
        conf t
            interface $INTERFACE
            macro description $TRIGGER
            switchport access vlan $ACCESS_VLAN
            description asp3-link-DOWN i.e. Custom Macro ON
            exit
        end
    fi
}

```

**Table 3-3 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.
-eq	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.

**Table 3-3 Supported Cisco IOS Shell Keywords (continued)**

Command	Description
\$	Variables that begin with the \$ character are replaced with a parameter value.
#	Use the # character to enter comment text.

**Table 3-4 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.
time	Pipeline.
until	Looping construct.
while	Looping construct.

## Applying Macros on a Switch

You can use the CLI or the Cisco IOS shell scripting capability to set the macro parameters and to apply the macro.

### Using the CLI

Beginning in privileged EXEC mode:

	Command	Purpose
Step 1	<b>macro auto config ?</b>	(Optional) Displays the global macros.
	<b>Example:</b> Switch# macro auto config ?	

	Command	Purpose
Step 2	<b>macro auto config</b> <i>global macro</i>  <b>Example:</b> Switch# macro auto config CISCO_SWITCH_HOSTNAME_CONFIG	Sets the macro parameters.  Follow the prompts in the CLI.
Step 3	<b>copy running-config startup-config</b>  <b>Example:</b> Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

### Example: One Macro

To display the global macros:

```
Switch# 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

Switch# 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
```

```
Switch# macro auto config CISCO_SWITCH_HOSTNAME_CONFIG
Enter system's network name: CISCO
Do you want to apply the parameters? [yes/no]: yes
Enter configuration commands, one per line. End with CNTL/Z.
Switch# macro auto apply CISCO_SWITCH_HOSTNAME_CONFIG
Enter configuration commands, one per line. End with CNTL/Z.
CISCO#
```

**Example: Combined Macros**

```
Switch# macro auto config CISCO_SWITCH_AUTO_IP_CONFIG
Do you want to configure default domain name? [yes/no]: yes
Enter the domain name: cisco.com
Do you want to configure Name server ipv4 address? [yes/no]: yes
Enter the IPv4 address[a.b.c.d]: 10.77.11.34
Enter IP address of the logging host: 10.77.11.36
Do you want to configure VPN Routing/Forwarding Instance name? [yes/no]: no
Enter the ip address of NTP server[a.b.c.d]: 10.77.11.37
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.
Switch# macro auto apply CISCO_SWITCH_AUTO_IP_CONFIG
Enter configuration commands, one per line. End with CNTL/Z.
Switch#
```

**Using the Cisco IOS Shell**

Beginning in privileged EXEC mode:

	Command	Purpose
Step 1	Use one of the following <ul style="list-style-type: none"> <li>macro auto config ?</li> <li>macro auto apply ?</li> </ul> <p><b>Example:</b> Switch# macro auto config ? Switch# macro auto apply ?</p>	(Optional) Displays the global macros.
Step 2	<b>macro auto config</b> <i>macro-name</i> <i>parameter=value</i> [ <i>parameter=value</i> ] ...	Sets the macro parameters. Follow the prompts in the CLI.
	<p><b>Example:</b> Switch# macro auto config CISCO_SWITCH_HOSTNAME_CONFIG HOSTNAME=CISCO</p>	
Step 3	<b>macro auto apply</b> <i>macro-name</i>	Applies the macro to the switch.
	<p><b>Example:</b> Switch# macro auto apply CISCO_SWITCH_HOSTNAME_CONFIG</p>	

	Command	Purpose
Step 4	<b>show macro auto</b>  <b>Example:</b> Switch# show macro auto	Verifies your entries.  The user-defined values appear only in the <b>show</b> command output.
Step 5	<b>copy running-config startup-config</b>  <b>Example:</b> Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

### Example: One Single Shell Parameter for One Macro

```
Switch# macro auto config CISCO_SWITCH_HOSTNAME_CONFIG HOSTNAME=CISCO
Switch# macro auto apply CISCO_SWITCH_HOSTNAME_CONFIG
Enter configuration commands, one per line. End with CNTL/Z.
CISCO#
```

### Example: Multiple Shell Parameters and Values for One Macro

```
Switch# macro auto config CISCO_SWITCH_ETHERCHANNEL_CONFIG PORT_CH_ID=1 PORT_CH_TYPE=2
EC_PROTO=Y EC_PROTO_TYPE=PAGP NO_OF_INT=3 MODE=AUTO INTERFACE=Gig2/0/1,Gig2/0/2,Gig2/0/3
NON_SILENT=Y EC_APPLY=YES
Switch# macro auto apply CISCO_SWITCH_ETHERCHANNEL_CONFIG
Enter configuration commands, one per line. End with CNTL/Z.
Switch#
```

### Example: Combined Macros

```
Switch# macro auto config CISCO_SWITCH_AUTO_IP_CONFIG CISCO_SWITCH_DOMAIN_NAME_CONFIG
DOMAIN_NAME=cisco.com
Switch# macro auto config CISCO_SWITCH_AUTO_IP_CONFIG CISCO_SWITCH_LOGGING_SERVER_CONFIG
HOST_IP=10.77.11.36
Switch# macro auto config CISCO_SWITCH_AUTO_IP_CONFIG CISCO_SWITCH_NAME_SERVER_CONFIG
IP_V4_ADDR=10.77.11.37
Switch# macro auto config CISCO_SWITCH_AUTO_IP_CONFIG CISCO_SWITCH_NTP_SERVER_CONFIG
IP_ADDRESS=10.77.11.38 VRF=NO
Switch# macro auto apply CISCO_SWITCH_AUTO_IP_CONFIG
Enter configuration commands, one per line. End with CNTL/Z.
Switch#
```



## Default Static Smartports Configuration

There are no static Smartports macros enabled on the switch.

**Table 3-5** *Default Static Smartports Macros*

Macro Name <sup>1</sup>	Description
<b>cisco-global</b>	Use this global configuration macro to enable rapid per-VLAN spanning-tree plus (PVST+), loop guard, and dynamic port-error recovery for link state failures.
<b>cisco-desktop</b>	Use this interface configuration macro for increased network security and reliability when connecting a desktop device, such as a PC, to a switch port.
<b>cisco-phone</b>	Use this interface configuration macro when connecting a desktop device such as a PC with a Cisco IP Phone to a switch port. This macro is an extension of the <b>cisco-desktop</b> macro and provides the same security and resiliency feature and also dedicated voice VLANs to ensure proper treatment of delay-sensitive voice traffic.
<b>cisco-switch</b>	Use this interface configuration macro when connecting an access switch and a distribution switch or between access switches connected through small form-factor pluggable (SFP) modules.
<b>cisco-router</b>	Use this interface configuration macro when connecting the switch and a WAN router.
<b>cisco-wireless</b>	Use this interface configuration macro when connecting the switch and a wireless access point.

1. Cisco-default Smartports macros vary, depending on the software version running on your switch.

## Static Smartports Configuration Guidelines

- When a macro is applied globally to a switch or to a switch interface, the existing configuration on the interface is retained. This is helpful when applying an incremental configuration.
- If a command fails because of a syntax or a configuration error, the macro continues to apply the remaining commands. You can use the **macro global trace macro-name** global configuration command or the **macro trace macro-name** interface configuration command to apply and then debug the macro to find any syntax or configuration errors.
- Some CLI commands are specific to certain interface types. If you apply a macro to an interface that does not accept the configuration, the macro fails the syntax or the configuration check, and the switch returns an error message.
- Applying a macro to an interface range is the same as applying a macro 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.
- When you apply a macro to a switch or a switch interface, the macro name is automatically added to the switch or interface. You can display the macro names and applied commands using the **show running-config** user EXEC command.

## NEAT Configuration

The Network Edge Access Topology (NEAT) feature extends identity to areas outside the wiring closet (such as conference rooms).

In a NEAT scenario, when 802.1x authentication is successful and an ASP macro is sent from the Access Control Server (ACS) to the switch, you must make one of the following configurations:

- Change the host mode to multi-host.
- Enable trunk configuration on the authenticator switch by configuring the cisco-av-pair as *device-traffic-class=switch* at the ACS.

## Applying Static Smartports Macros

Beginning in privileged EXEC mode, follow these steps to apply a static Smartports macro:

	Command	Purpose
Step 1	<b>show parser macro</b>  <b>Example:</b> Switch# show parser macro	Displays the Cisco-default static Smartports macros embedded in the switch software.
Step 2	<b>show parser macro name</b> <i>macro-name</i>  <b>Example:</b> Switch# show parser macro name cisco-desktop	Displays the specific macro that you want to apply.
Step 3	<b>configure terminal</b>  <b>Example:</b> Switch# configure terminal	Enters global configuration mode.
Step 4	<b>macro global {apply   trace}</b> <i>macro-name</i> [ <b>parameter</b> { <i>value</i> }] [ <b>parameter</b> { <i>value</i> }] [ <b>parameter</b> { <i>value</i> }]  <b>Example:</b> Switch(config)# macro global apply cisco-desktop \$access_vlan 25	Applies a macro on the switch: <ul style="list-style-type: none"> <li>• To only apply each individual macro command, use the <b>macro global apply</b> <i>macro-name</i> command.</li> <li>• To apply and then debug a macro to find any syntax or configuration errors, use the <b>macro global trace</b> <i>macro-name</i> command.</li> </ul> Append the macro with the required values by using the <b>parameter value</b> keywords. Keywords that begin with \$ require a unique parameter value.  You can use the <b>macro global apply</b> <i>macro-name</i> ? command to display a list of any required values for the macro. If you apply a macro without entering the keyword values, the commands are invalid and are not applied.  (Optional) Specify unique parameter values that are specific to the switch. You can enter up to three keyword-value pairs. Parameter keyword matching is case sensitive. The corresponding value replaces all matching occurrences of the keyword.
Step 5	<b>interface</b> <i>interface-id</i>  <b>Example:</b> Switch(config)# interface gigabitethernet 2/0/5	(Optional) Specifies an interface and enters interface configuration mode.

	Command	Purpose
Step 6	<b>default interface</b> <i>interface-id</i>  <b>Example:</b> Switch(config)# default interface 2/1/4	(Optional) Clears all configuration from the specified interface.
Step 7	<b>macro</b> { <b>apply</b>   <b>trace</b> } <i>macro-name</i> [ <b>parameter</b> { <i>value</i> }] [ <b>parameter</b> { <i>value</i> }] [ <b>parameter</b> { <i>value</i> }]  <b>Example:</b> Switch(config-if)# macro apply cisco-desktop \$access_vlan 25	Applies a macro on the interface: <ul style="list-style-type: none"> <li>To only apply each individual macro command, use the <b>macro apply</b> <i>macro-name</i> command.</li> <li>To apply and then debug a macro to find any syntax or configuration errors, use the <b>macro trace</b> <i>macro-name</i> command.</li> </ul> Append the macro with the required values by using the <b>parameter value</b> keywords. Keywords that begin with \$ require a unique parameter value.  You can use the <b>macro global apply</b> <i>macro-name</i> ? command to display a list of any required values for the macro. If you apply a macro without entering the keyword values, the commands are invalid and are not applied.  (Optional) Specify unique parameter values that are specific to the switch. You can enter up to three keyword-value pairs. Parameter keyword matching is case sensitive. The corresponding value replaces all matching occurrences of the keyword.
Step 8	<b>end</b>  <b>Example:</b> Switch(config)# end	Returns to privileged EXEC mode.
Step 9	<b>show running-config interface</b> <i>interface-id</i>  <b>Example:</b> Switch# show running-config interface gigabit ethernet 1/0/4	Verifies that Auto Smartports is enabled.
Step 10	<b>copy running-config startup-config</b>  <b>Example:</b> Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

You can only delete a global macro-applied configuration on a switch by entering the **no** version of each command in the macro. You can delete a macro-applied configuration on a port by entering the **default interface** *interface-id* interface configuration command.

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

```
Switch# show parser macro name cisco-desktop
-----
Macro name : cisco-desktop
Macro type : default interface
```

```

# macro keywords $access_vlan
# Basic interface - Enable data VLAN only
# Recommended value for access vlan should not be 1
switchport access vlan $access_vlan
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
-----
Switch#
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet1/0/4
Switch(config-if)# macro apply cisco-desktop $access_vlan 25

```

## Displaying Macros

**Table 3-6** Commands for Displaying Auto Smartports and Static Smartports Macros

Command	Purpose
<b>show macro auto ?</b>	Displays information about Auto Smartports macros. <ul style="list-style-type: none"> <li>• device: Displays device macro information</li> <li>• event: Displays macro event-related commands</li> <li>• global: Displays global macro information</li> <li>• interface: Displays interface Auto Smartports status</li> </ul>
<b>show parser macro</b>	Displays all static smartports macros.
<b>show parser macro name</b> <i>macro-name</i>	Displays a specific static Smartports macro.
<b>show parser macro brief</b>	Displays the static Smartports macro names.

Table 3-6 Commands for Displaying Auto Smartports and Static Smartports Macros (continued)

Command	Purpose
<b>show parser macro description</b> [interface <i>interface-id</i> ]	Displays the static Smartports macro description for all interfaces or for a specified interface.
<b>show shell ?</b>	<p>Displays information about Auto Smartports event triggers and macros.</p> <ul style="list-style-type: none"> <li>• data-path: Displays data paths for <i>fetch</i></li> <li>• environment: Displays shell environment information</li> <li>• functions: Displays shell functions information</li> <li>• triggers: Displays shell triggers information</li> </ul> <p><b>Note</b> The <b>show shell</b> command is a feature at the Cisco IOS level. You may first have to enable Cisco IOS Shell by entering the <b>terminal shell</b> command before you can enter the <b>show shell</b> command. For more information, see the Cisco IOS Shell configuration guide on Cisco.com:  <a href="http://www.cisco.com/en/US/docs/ios/netmgmt/configuration/guide/nm_ios_shell.pdf">http://www.cisco.com/en/US/docs/ios/netmgmt/configuration/guide/nm_ios_shell.pdf</a></p>





## Auto Smartports and Static Smartports Macros CLI Commands

---

- [macro](#), page 4-2
- [macro auto \(CLI\)](#), page 4-5
- [macro auto control](#), page 4-11
- [macro auto device](#), page 4-13
- [macro auto execute](#), page 4-15
- [macro auto global control](#), page 4-21
- [macro auto global processing](#), page 4-23
- [macro auto mac-address-group](#), page 4-25
- [macro auto monitor](#), page 4-27
- [macro auto processing](#), page 4-28
- [macro auto sticky](#), page 4-30
- [macro auto trigger](#), page 4-31
- [macro description](#), page 4-33
- [macro global](#), page 4-34
- [macro global description](#), page 4-37
- [shell trigger](#), page 4-38
- [show macro auto](#), page 4-40
- [show macro auto monitor clients](#), page 4-43
- [show macro auto monitor device](#), page 4-45
- [show macro auto monitor type](#), page 4-47
- [show parser macro](#), page 4-50
- [show shell](#), page 4-53

# macro

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

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

## Syntax Description

<b>apply</b>	Applies a macro to an interface.
<b>trace</b>	Applies a macro to an interface and then debugs it.
<i>macro-name</i>	Specifies the name of the macro.
<b>parameter</b> <i>value</i>	(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

## Command History

Release	Modification
15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

## Usage Guidelines

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

You can use the **macro trace** *macro-name* interface configuration 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** user EXEC command.



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** user EXEC command. Display the contents of a specific macro by using the **show parser macro name** *macro-name* user EXEC 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 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* user EXEC command.

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* interface configuration command.

## Examples

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

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

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

```
Switch(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:

```
Switch# 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
```

```
-----
```

```
Switch#
```

```
Switch# configure terminal
```

```
Enter configuration commands, one per line. End with CNTL/Z.
```

```
Switch(config)# interface gigabitethernet1/0/4
```

```
Switch(config-if)# macro apply cisco-desktop $AVID 25
```

#### Related Commands

Command	Description
<b>macro description</b>	Adds a description about the macros that are applied to an interface. This is a mandatory command for Auto Smartports to work.
<b>macro global</b>	Applies a macro on a switch or applies and traces a macro on a switch.
<b>macro global description</b>	Adds a description about the macros that are applied to the switch.
<b>show parser macro</b>	Displays the macro definition for all macros or for the specified macro.

## macro auto (CLI)

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

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

Syntax Description		
	<b>apply</b>	Applies the macro.
	<b>config</b>	Enters the macro parameters.
	<i>macro-name</i>	Specifies the macro name.

**Command Default** No macros are applied to the switch.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

**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.

**Examples** To display global macros:

```
Switch# 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

Switch# 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

```

To display the parameters for a specific macro:

```

Switch# 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>

Switch# 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>

Switch# 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>

Switch# 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>

```

To set macro parameters and apply the macro using the CLI:

```

Switch# 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.
Switch# macro auto apply CISCO_SWITCH_ETHERCHANNEL_CONFIG
Enter configuration commands, one per line. End with CNTL/Z.
Switch#

```

**Related Commands**

<b>Command</b>	<b>Description</b>
<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
<b>show macro auto</b>	Displays information about macros.
<b>show shell</b>	Displays information about event triggers and macros.

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

**macro auto apply** *macro-name*

Syntax Description	apply	Applies the macro.
	<i>macro-name</i>	Specifies the macro name.

**Command Default** No macros are applied to the switch.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

**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.

**Examples** To display global macros:

```
Switch# 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
```

**Related Commands**

<b>Command</b>	<b>Description</b>
<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
<b>show macro auto</b>	Displays information about macros.
<b>show shell</b>	Displays information about event triggers and macros.

# macro auto config (Cisco IOS shell scripting capability)

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

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

## Syntax Description

<b>config</b>	Enters the macro parameters.
<i>macro-name</i>	Specifies the macro name.
<i>parameter=value</i>	<i>parameter=value</i> —Replaces values for global macro parameter values.
[ <i>parameter=value</i> ] ...	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
15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

## 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.

For more information, see the “Configuring and Applying Global Macros” section in the “Configuring Auto Smartports and Static Smartports Macros” chapter.

## Related Commands

Command	Description
<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
<b>show macro auto</b>	Displays information about macros.
<b>show shell</b>	Displays information about event triggers and macros.



# 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** interface configuration command. 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 [access-point] [ip-camera]
[lightweight-ap] [media-player] [phone] [router] [switch] | trigger [last-resort]}
```

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

Syntax Description	
<b>detection</b> [cdp] [lldp] [mac-address]	<b>detection</b> —Sets one or more of these as an event trigger: <ul style="list-style-type: none"> <li>(Optional) <b>cdp</b>—CDP messages</li> <li>(Optional) <b>lldp</b>—LLDP messages</li> <li>(Optional) <b>mac-address</b>—User-defined MAC address groups</li> </ul>
<b>device</b> [access-point] [ip-camera] [lightweight-ap] [media-player] [phone] [router] [switch]	<b>device</b> —Sets one or more of these devices as an event trigger: <ul style="list-style-type: none"> <li>(Optional) <b>access-point</b>—Autonomous access point</li> <li>(Optional) <b>ip-camera</b>—Cisco IP video surveillance camera</li> <li>(Optional) <b>lightweight-ap</b>—Lightweight access point</li> <li>(Optional) <b>media-player</b>—Digital media player</li> <li>(Optional) <b>phone</b>—Cisco IP phone</li> <li>(Optional) <b>router</b>—Cisco router</li> <li>(Optional) <b>switch</b>—Cisco switch</li> </ul>
<b>trigger</b> [last-resort]	<b>trigger</b> —Sets a specific event trigger. <ul style="list-style-type: none"> <li>(Optional) <b>last-resort</b>—Last-resort trigger.</li> </ul>

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

Command History	Release	Modification
	15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

**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** user EXEC command.

**Examples**

To set LLDP messages and MAC address groups as event triggers:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet 5/0/2
Switch(config-if)# macro auto control detection lldp mac-address
Switch(config-if)# exit
Switch(config)# end
```

To set access points, video surveillance cameras, and digital media players as event triggers:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet 5/0/1
Switch(config-if)# macro auto control device access-point ip-camera media-player
Switch(config-if)# exit
Switch(config)# end
```

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

**Related Commands**

Command	Description
<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
<b>macro auto mac-address-group</b>	Configures MAC address groups.
<b>macro auto sticky</b>	Configures macro persistence.
<b>shell trigger</b>	Creates event triggers.
<b>show macro auto</b>	Displays information about macros.
<b>show shell</b>	Displays information about event triggers and macros.

# macro auto device

To replace macro default parameter values with values that are specific to your switch, use the **macro auto device** global configuration command. Use the **no** form of this command to remove the parameter values.

```
macro auto device {access-point | ip-camera | lightweight-ap | media-player | phone | router |
switch} [parameter=value]
```

```
no macro auto device {access-point | ip-camera | lightweight-ap | media-player | phone | router
| switch} [parameter=value]
```

## Syntax Description

<b>access-point</b>	Replaces the access-point default parameter value: NATIVE_VLAN=1
<b>ip-camera</b>	Replaces the IP video surveillance camera default parameter value: ACCESS_VLAN=1
<b>lightweight-ap</b>	Replaces the lightweight access point default parameter value: ACCESS_VLAN=1
<b>media-player</b>	Replaces the digital media player default parameter value: ACCESS_VLAN=1
<b>phone</b>	Replaces the IP phone default parameter values: ACCESS_VLAN=1 VOICE_VLAN=2
<b>router</b>	Replaces the router default parameter value: NATIVE_VLAN=1
<b>switch</b>	Replaces the switch default parameter value: NATIVE_VLAN=1
<i>parameter=value</i>	(Optional) Replaces the macro default parameter values. Enter new values in the form of name value pair separated by spaces: [<name1>=<value1> <name2>=<value2>...]

## Command Default

Macro default parameter values are defined previously.

## Command Modes

Global configuration

## Command History

Release	Modification
15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

## Usage Guidelines

Use the **macro auto device** global configuration command to replace macro default parameter values with values that are specific to your switch. Use the **no** form of this command to remove the parameter values.

Use the **show macro device** privileged EXEC command to display the contents of the macros. Use the *parameter=value* keywords to replace default parameter values within a specific macro.

You can also use the **macro auto execute** global configuration command to specify default parameter values. This command also requires that you specify an event trigger and a built-in or 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 is applied to the switch. Only one command is active on the switch.

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

### Examples

This example shows how to display the IP phone macro parameter values, enable the IP phone macro, and change the default voice VLAN to 20:

```
Switch# show macro auto device phone
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

Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# macro auto device phone VOICE_VLAN=20
Switch(config)# end
Switch# show macro auto device phone
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:VOICE_VLAN=20
```

### Related Commands

Command	Description
<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
<b>macro auto mac-address-group</b>	Configures MAC address groups.
<b>macro auto sticky</b>	Configures macro persistence.
<b>shell trigger</b>	Creates event triggers.
<b>show macro auto</b>	Displays information about macros.
<b>show shell</b>	Displays information about event triggers and macros.

## 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** global configuration command.

```
macro auto execute event trigger {[builtin built-in macro name] | [remote url]} [parameter=value]
[function contents]
```

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

Syntax Description	
<i>event trigger</i>	<p>Defines mapping from an event trigger to a built-in macro.</p> <p>Specifies an <i>event trigger</i>:</p> <ul style="list-style-type: none"> <li>• CISCO_CUSTOM_EVENT</li> <li>• CISCO_DMP_EVENT</li> <li>• CISCO_IPVSC_EVENT</li> <li>• CISCO_LAST_RESORT_EVENT</li> <li>• CISCO_PHONE_EVENT</li> <li>• CISCO_ROUTER_EVENT</li> <li>• CISCO_SWITCH_EVENT</li> <li>• CISCO_WIRELESS_AP_EVENT</li> <li>• CISCO_WIRELESS_LIGHTWEIGHT_AP_EVENT</li> <li>• WORD—Apply a user-defined event trigger such as a MAC address group.</li> </ul>
<b>builtin</b> <i>built-in macro name</i>	<p>(Optional) Specifies a <b>builtin</b> <i>built-in macro name</i>:</p> <ul style="list-style-type: none"> <li>• CISCO_AP_AUTO_SMARTPORT Specify the parameter value: NATIVE_VLAN=1.</li> <li>• CISCO_DMP_AUTO_SMARTPORT Specify the parameter value: ACCESS_VLAN=1.</li> <li>• CISCO_IPVSC_AUTO_SMARTPORT Specify the parameter value: ACCESS_VLAN=1.</li> <li>• CISCO_LWAP_AUTO_SMARTPORT Specify the parameter value: ACCESS_VLAN=1.</li> <li>• CISCO_PHONE_AUTO_SMARTPORT Specify the parameter values: ACCESS_VLAN=1 and VOICE_VLAN=2.</li> <li>• CISCO_ROUTER_AUTO_SMARTPORT Specify the parameter value: NATIVE_VLAN=1.</li> <li>• CISCO_SWITCH_AUTO_SMARTPORT Specify the parameter value: NATIVE_VLAN=1.</li> </ul>
<i>parameter=value</i>	<p>(Optional) <i>parameter=value</i>—Replaces default values for parameter values shown for the <i>builtin-macro name</i>, for example, ACCESS_VLAN=1. Enter new values in the form of name value pair separated by a space: [&lt;name1&gt;=&lt;value1&gt; &lt;name2&gt;=&lt;value2&gt;...].</p>

<i>{function contents}</i>	(Optional) <i>{function contents}</i> 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.
<b>remote url</b>	(Optional) Specifies a remote server location: <ul style="list-style-type: none"> <li>The syntax for the local flash file system on the standalone switch or the stack master: <b>flash:</b></li> <li>The syntax for the local flash file system on a stack member: <b>flash member number:</b></li> <li>The syntax for the FTP: <b>ftp:[[/username[:password]]@location]/directory]/filename</b></li> <li>The syntax for an HTTP server: <b>http://[[username:password]@]{hostname   host-ip}[/directory]/filename</b></li> <li>The syntax for a secure HTTP server: <b>https://[[username:password]@]{hostname   host-ip}[/directory]/filename</b></li> <li>The syntax for the NVRAM: <b>nvrasm://[[username:password]@]/[directory]/filename</b></li> <li>The syntax for the Remote Copy Protocol (RCP): <b>rctp:[[/username@location]/directory]/filename</b></li> <li>The syntax for the Secure Copy Protocol (SCP): <b>scp:[[/username@location]/directory]/filename</b></li> <li>The syntax for the TFTP: <b>tftp:[[/location]/directory]/filename</b></li> </ul>

**Command Default** This command has no default setting.

**Command Modes** Global configuration

Command History	Release	Modification
	15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

**Usage Guidelines** Use the **macro auto execute** global configuration 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** global configuration commands. Use the **show shell triggers** privileged EXEC command to display the contents of the user-defined triggers and macros.

You can use the **macro auto mac-address-group** global configuration command 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** global configuration 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** global configuration 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.

## Examples

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:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#!!! the next command modifies the access and voice vlans
Switch(config)#!!! for the built in Cisco IP phone auto smartport macro
Switch(config)# macro auto execute CISCO_PHONE_EVENT builtin CISCO_PHONE_AUTO_SMARTPORT
ACCESS_VLAN=10 VOICE_VLAN=20
Switch(config)#
Switch(config)#!!! the next command modifies the Native vlan used for inter switch trunks
Switch(config)# macro auto execute CISCO_SWITCH_EVENT builtin CISCO_SWITCH_AUTO_SMARTPORT
NATIVE_VLAN=10
```

```

Switch(config)#
Switch(config)#! the next command enables auto smart ports globally
Switch(config)# macro auto global processing
Switch(config)#
Switch(config)# exit

Switch# !!! here is the running configuration of the interface connected
Switch# !!! to another Cisco Switch after the Macro is applied
Switch#
Switch# 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.

```

Switch(config)# shell trigger DMP_EVENT mediaplayer
Switch(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 4-1** 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 4-2** *Unsupported Cisco IOS Shell Reserved Keywords*

<b>Command</b>	<b>Description</b>
	Pipeline.
case	Conditional construct.
esac	Conditional construct.
for	Looping construct.
function	Shell function.
in	Conditional construct.
select	Conditional construct.
time	Pipeline.
until	Looping construct.
while	Looping construct.

**Related Commands**

<b>Command</b>	<b>Description</b>
<b>macro auto device</b>	Configures macro default parameter values.
<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
<b>macro auto mac-address-group</b>	Configures MAC address groups.
<b>macro auto sticky</b>	Configures macro persistence.
<b>shell trigger</b>	Creates event triggers.
<b>show macro auto</b>	Displays information about macros.
<b>show shell</b>	Displays information about event triggers and macros.

# 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** global configuration command. Use the **no** form of this command to disable trigger-to-macro mapping.

```
macro auto global control { device [access-point] [ip-camera] [lightweight-ap] [mac-address]
[media-player] [phone] [router] [switch] | trigger [last-resort] }
```

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

## Syntax Description

<b>device</b> [access-point] [ip-camera] [lightweight-ap] [mac-address] [media-player] [phone] [router] [switch]	<p><b>device</b>—Sets one or more of these devices as an event trigger:</p> <ul style="list-style-type: none"> <li>(Optional) <b>access-point</b>—Autonomous access point</li> <li>(Optional) <b>ip-camera</b>—Cisco IP video surveillance camera</li> <li>(Optional) <b>lightweight-ap</b>—Lightweight access point</li> <li>(Optional) <b>mac-address</b>—User-defined MAC address groups</li> <li>(Optional) <b>media-player</b>—Digital media player</li> <li>(Optional) <b>phone</b>—Cisco IP phone</li> <li>(Optional) <b>router</b>—Cisco router</li> <li>(Optional) <b>switch</b>—Cisco switch</li> </ul>
<b>trigger</b> [last-resort]	<p><b>trigger</b>—Sets a specific event trigger.</p> <ul style="list-style-type: none"> <li>(Optional) <b>last-resort</b>—Last-resort trigger</li> </ul>

## 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

## Command History

Release	Modification
15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

## 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** user EXEC command.

**Examples**

To set CDP messages, LLDP messages and MAC address groups as event triggers:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# macro auto global control detection cdp lldp mac-address
Switch(config)# end
```

To set autonomous access points, lightweight access points, and IP phones:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# macro auto global control device access-point lightweight-ap phone
Switch(config)# end
```

**Related Commands**

<b>Command</b>	<b>Description</b>
<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
<b>macro auto mac-address-group</b>	Configures MAC address groups.
<b>macro auto sticky</b>	Configures macro persistence.
<b>shell trigger</b>	Creates event triggers.
<b>show macro auto</b>	Displays information about macros.
<b>show shell</b>	Displays information about event triggers and macros.

# macro auto global processing

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

**macro auto global processing**

**no macro auto global processing**

**Syntax Description** This command has no arguments or keywords.

**Command Default** Auto Smartports is disabled.

**Command Modes** Global configuration

Command History	Release	Modification
	15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

**Usage Guidelines** Use the **macro auto global processing** global configuration 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** privileged EXEC command.

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

```
Switch(config)# macro auto global processing
Switch(config)# interface interface_id
Switch(config-if)# no macro auto processing
```

Related Commands	Command	Description
	<b>macro auto device</b>	Configures macro default parameter values.
	<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.

<b>Command</b>	<b>Description</b>
<b>macro auto mac-address-group</b>	Configures MAC address groups.
<b>macro auto sticky</b>	Configures macro persistence.
<b>shell trigger</b>	Creates event triggers.
<b>show macro auto</b>	Displays information about macros.
<b>show shell</b>	Displays information about event triggers and macros.

# 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** global configuration command. 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

<i>name</i>	Specifies the group name.
<b>oui</b>	(Optional) Specifies an operationally unique identifier (OUI) <b>list</b> or <b>range</b> . <ul style="list-style-type: none"> <li><b>list</b>—Enter an OUI list in hexadecimal format separated by spaces.</li> <li><b>range</b>—Enter the starting OUI hexadecimal value (<i>start-value</i>).</li> <li><b>size</b>—Enter the length of the range (<i>number</i>) from 1 to 5 to create a list of sequential addresses.</li> </ul>
<b>mac-address list</b> <i>list</i>	(Optional) Configures a list of MAC addresses separated by a space.

## Command Default

No groups are defined.

## Command Modes

Group configuration

## Command History

Release	Modification
15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

## Usage Guidelines

Use the **macro auto mac-address-group** global configuration 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** global configuration 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.

## Examples

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

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# macro auto mac-address-group address_trigger
Switch(config-addr-grp-mac)# mac-address list 2222.3333.3334 22.33.44 a.b.c
Switch(config-addr-grp-mac)# oui list 455555 233244
```

## macro auto mac-address-group

```

Switch(config-addr-grp-mac)# oui range 333333 size 2
Switch(config-addr-grp-mac)# exit
Switch(config)# end
Switch# 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>

```

## Related Commands

Command	Description
<b>macro auto device</b>	Configures macro default parameter values.
<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
<b>macro auto sticky</b>	Configures macro persistence.
<b>shell trigger</b>	Creates event triggers.
<b>show macro auto</b>	Displays information about macros.
<b>show shell</b>	Displays information about event triggers and macros.



# macro auto monitor

To enable the device classifier, use the **macro auto monitor** global configuration command. Use the **no** form of this command to disable the device classifier.

**macro auto monitor**

**no macro auto monitor**

**Syntax Description** This command has no arguments or keywords.

**Command Default** Device classifier is enabled.

**Command Modes** Global configuration

Command History	Release	Modification
	15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

**Usage Guidelines** Use the **no macro auto monitor** global configuration command to disable the device classifier. You cannot disable the device classifier while it is being used by features such as ASP.

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

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# macro auto monitor
Switch(config)# end
```

Related Commands	Command	Description
	<b>show macro auto monitor clients</b>	Displays the clients using the device classifier facility on the switch.
	<b>show macro auto monitor device</b>	Displays the devices connected to a switch, along with their properties and classifications.
	<b>show macro auto monitor type</b>	Displays all the device types known to the device classification agent.
	<b>show shell</b>	Displays information about event triggers and macros.

# macro auto processing

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

**macro auto processing**

**no macro auto processing**

**Syntax Description** This command has no arguments or keywords.

**Command Default** Auto Smartports is disabled.

**Command Modes** Interface configuration

Command History	Release	Modification
	15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

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

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** interface configuration 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** privileged EXEC command.

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

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

Related Commands	Command	Description
	<b>macro auto device</b>	Configures macro default parameter values.
	<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
	<b>macro auto mac-address-group</b>	Configures MAC address groups.

<b>Command</b>	<b>Description</b>
<b>macro auto sticky</b>	Configures macro persistence.
<b>shell trigger</b>	Creates event triggers.
<b>show macro auto</b>	Displays information about macros.
<b>show shell</b>	Displays information about event triggers and macros.

# macro auto sticky

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

**macro auto sticky**

**no macro auto sticky**

**Syntax Description** This command has no arguments or keywords.

**Command Default** Macro persistence is disabled.

**Command Modes** Global configuration

Command History	Release	Modification
	15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

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

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

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet 2/0/1
Switch(config-if)# macro auto port sticky
Switch(config-if)# exit
Switch(config)# end
```

Related Commands	Command	Description
	<b>macro auto device</b>	Configures macro default parameter values.
	<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
	<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
	<b>macro auto mac-address-group</b>	Configures MAC address groups.
	<b>shell trigger</b>	Creates event triggers.
	<b>show macro auto</b>	Displays information about macros.
	<b>show shell</b>	Displays information about event triggers and macros.

# 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** global configuration command. 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

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

## Command Default

No user-defined triggers are configured.

## Command Modes

Global configuration

## Command History

Release	Modification
15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

## 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** global configuration command 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.

Only the full *profile\_name* and *device\_name* string match is supported in Cisco IOS Release 15.0(1)SE. After the device is profiled, you must add the complete string to the device-group database.

## Examples

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:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# macro auto trigger DMP
Switch(config-macro-trigger)# profile mediaplayer-DMP
Switch(config-macro-trigger)# exit
```

## macro auto trigger

```
Switch(config)# end
```

### Related Commands

Command	Description
<b>macro auto device</b>	Configures macro default parameter values.
<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
<b>macro auto mac-address-group</b>	Configures MAC address groups.
<b>shell trigger</b>	Creates event triggers.
<b>show macro auto</b>	Displays information about macros.
<b>show macro auto monitor device</b>	Displays the devices connected to a switch, along with their properties and classifications.
<b>show macro auto monitor type</b>	Displays all the device types known to the device classification agent.
<b>show shell</b>	Displays information about event triggers and macros.

# macro description

To enter a description about which macros are applied to an interface, use the **macro description** interface configuration command. 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

## Command History

Release	Modification
15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

## 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** privileged EXEC command.

## Examples

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

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

## Related Commands

Command	Description
<b>macro apply</b>	Applies a macro on an interface.
<b>macro global</b>	Applies a macro on a switch or applies and traces a macro on a switch
<b>macro global description</b>	Adds a description about the macros that are applied to the switch.
<b>macro trace</b>	Applies and traces a macro on an interface.
<b>show parser macro</b>	Displays the macro definition for all macros or for the specified macro.

# macro global

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

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

## Syntax Description

<b>apply</b>	Applies a macro to the switch.
<b>trace</b>	Applies a macro to a switch and debugs the macro.
<i>macro-name</i>	Specifies the name of the macro.
<b>parameter value</b>	(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

## Command History

Release	Modification
15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

## Usage Guidelines

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

Use the **macro global trace** *macro-name* global configuration 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** user EXEC command.



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** user EXEC command. Display the contents of a specific macro by using the **show parser macro name** *macro-name* user EXEC 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** user EXEC command.

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

### Examples

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

```
Switch# 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** global configuration 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:

```
Switch(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'
```

### Related Commands

Command	Description
<b>macro apply</b>	Applies a macro on an interface.
<b>macro description</b>	Adds a description about the macros that are applied to an interface.
<b>macro global description</b>	Adds a description about the macros that are applied to the switch.

<b>Command</b>	<b>Description</b>
<b>macro trace</b>	Applies and traces a macro on an interface.
<b>show parser macro</b>	Displays the macro definition for all macros or for the specified macro.

# macro global description

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

**macro global description** *text*

**no macro global description** *text*

<b>Syntax Description</b>	<b>description</b> <i>text</i> Enters a description about the macros that are applied to the switch.
---------------------------	------------------------------------------------------------------------------------------------------

<b>Command Default</b>	This command has no default setting.
------------------------	--------------------------------------

<b>Command Modes</b>	Global configuration
----------------------	----------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

<b>Usage Guidelines</b>	<p>Use the <b>description</b> 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.</p> <p>You can verify your settings by entering the <b>show parser macro description</b> privileged EXEC command.</p>
-------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>Examples</b>	<p>This example shows how to add a description to a switch:</p> <pre>Switch(config)# macro global description uddld aggressive mode enabled</pre>
-----------------	---------------------------------------------------------------------------------------------------------------------------------------------------

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>macro apply</b>	Applies a macro on an interface.
	<b>macro description</b>	Adds a description about the macros that are applied to an interface.
	<b>macro global</b>	Applies a macro on a switch or applies and traces a macro on a switch.
	<b>macro trace</b>	Applies and debugs a macro on an interface.
	<b>show parser macro</b>	Displays the macro definition for all macros or for the specified macro.

# shell trigger

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

**shell trigger** *identifier description*

**no shell trigger** *identifier description*

## Syntax Description

<i>identifier</i>	Specifies the event trigger identifier. The identifier should have no spaces or hyphens between words.
<i>description</i>	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

## Command History

Release	Modification
15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

## Usage Guidelines

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

**Examples**

This example shows how to create a user-defined event trigger called RADIUS\_MAB\_EVENT:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# shell trigger RADIUS_MAB_EVENT MAC_AuthBypass Event
Switch(config)# end
```

**Related Commands**

<b>Command</b>	<b>Description</b>
<b>macro auto device</b>	Configures macro default parameter values.
<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
<b>macro auto mac-address-group</b>	Configures MAC address groups.
<b>macro auto sticky</b>	Configures macro persistence.
<b>shell trigger</b>	Creates event triggers.
<b>show macro auto</b>	Displays information about macros.
<b>show shell</b>	Displays information about event triggers and macros.

## show macro auto

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

```
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

<b>address-group</b> [ <i>address-group-name</i> ]	Displays address-group information.  (Optional) <i>address-group-name</i> —Displays information for the specified address group.
<b>device</b> [access-point] [ip-camera] [lightweight-ap] [media-player] [phone] [router] [switch]	Displays device information about one or more devices. <ul style="list-style-type: none"> <li>(Optional) <b>access-point</b>—Autonomous access point</li> <li>(Optional) <b>ip-camera</b>—Cisco IP video surveillance camera</li> <li>(Optional) <b>lightweight-ap</b>—Lightweight access point</li> <li>(Optional) <b>media-player</b>—Digital media player</li> <li>(Optional) <b>phone</b>—Cisco IP phone</li> <li>(Optional) <b>router</b>—Cisco router</li> <li>(Optional) <b>switch</b>—Cisco switch</li> </ul>
<b>global</b> [ <i>event_trigger</i> ]	Displays Auto Smartports information about the switch.  (Optional) <i>event_trigger</i> —Displays information about the specified event trigger.
<b>interface</b> [ <i>interface_id</i> ]	Displays interface status.  (Optional) <i>interface_id</i> —Displays information about the specified interface.

### Command Modes

User EXEC  
Privileged EXEC

### Command History

Release	Modification
15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

### Usage Guidelines

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

### Examples

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

```
Switch# 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* privileged EXEC command to view the TEST3 address group configuration on the switch:

```
Switch# show macro auto address-group TEST3
MAC Address Group Configuration:
```

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

#### Related Commands

Command	Description
<b>macro auto device</b>	Configures macro default parameter values.
<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
<b>macro auto mac-address-group</b>	Configures MAC address groups.
<b>macro auto sticky</b>	Configures macro persistence.
<b>shell trigger</b>	Creates event triggers.
<b>show shell</b>	Displays information about event triggers and macros.



# show macro auto monitor clients

To display the clients using the device classifier facility on the switch, use the **show macro auto monitor clients** user EXEC command.

**show macro auto monitor clients**

## Syntax Description

This command has no arguments or keywords.

## Command Default

User EXEC  
Privileged EXEC

## Command History

Release	Modification
15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

## 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 macro auto monitor 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 macro auto monitor** command. If you attempt to disable the DC while a client is using it, an error message appears.

## Examples

This example shows how to use the **show macro auto monitor clients** privileged EXEC command to view the clients using the DC on the switch:

```
Switch# show macro auto monitor 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 macro auto monitor
These subsystems should be disabled before disabling Device classifier
Auto Smart Ports

% Error - device classifier is not disabled
```

## Related Commands

Command	Description
<b>macro auto device</b>	Configures macro default parameter values.
<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
<b>macro auto mac-address-group</b>	Configures MAC address groups.

<b>Command</b>	<b>Description</b>
<b>macro auto sticky</b>	Configures macro persistence.
<b>shell trigger</b>	Creates event triggers.
<b>show macro auto monitor type</b>	Displays all the device types recognized by the device classifier.
<b>show shell</b>	Displays information about event triggers and macros.

# show macro auto monitor device

To display the devices connected to a switch and their associated properties, use the **show macro auto monitor device** user EXEC command.

```
show macro auto monitor device [detail | filter string | interface interface_id | mac-address
                               mac_address]
```

Syntax Description		
<b>detail</b>		Displays detailed device classifier information.
<b>filter</b> <i>string</i>		Displays information for devices that match the filter.
<b>interface</b> <i>interface_id</i>		Displays information about devices attached to the specified interface.
<b>mac</b> <i>mac_address</i>		Displays device information for the specified endpoint.

Command Modes	
	User EXEC Privileged EXEC

Command History	Release	Modification
	15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

Usage Guidelines	
	Use this command to display the devices connected to a switch. Use the <b>show macro auto device</b> privileged EXEC command to display the configurable parameters for a device.

Examples	
	This example shows how to use the <b>show macro auto monitor device</b> privileged EXEC command with no optional keywords to view the devices connected to the switch:

```
Switch# show macro auto monitor device
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 macro auto monitor device** privileged EXEC command with the optional **mac-address** keyword to view summary information about the connected device with the specified MAC address:

```
Switch# show macro auto monitor device 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 macro auto monitor device** privileged EXEC command with the optional **mac-address** and **detail** keywords to view detailed information about the connected device with the specified MAC address:

```
Switch# show macro auto monitor device mac-address 001f.9e90.1250 detail
```

### show macro auto monitor device

```

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 macro auto monitor device** privileged EXEC command with the optional **interface** keyword to view summary information about the device connected to the specified interface:

```

Switch# show macro auto monitor device 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 macro auto monitor device** privileged EXEC command with the optional **interface** and **detail** keywords to view detailed information about the device connected to the specified interface:

```

Switch# show macro auto monitor device 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
=====

```

### Related Commands

Command	Description
<b>macro auto device</b>	Configures macro default parameter values.
<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
<b>macro auto mac-address-group</b>	Configures MAC address groups.
<b>macro auto sticky</b>	Configures macro persistence.
<b>shell trigger</b>	Creates event triggers.
<b>show macro auto monitor type</b>	Displays all the device types recognized by the device classifier.
<b>show shell</b>	Displays information about event triggers and macros.

# show macro auto monitor type

To display all the device types recognized by the device classifier, use the **show macro auto monitor type** user EXEC command.

**show macro auto monitor type** [**table** [*built-in* | *default*] | **string** *filter\_string*]

Syntax Description	table	Displays device classification in a table.
	<i>built-in</i>	Displays device classification information from the built-in device table.
	<i>default</i>	Displays device classification information from the default device table.
	<b>filter</b> <i>string</i>	Displays information for devices that match the filter.

Command Modes	User EXEC Privileged EXEC
---------------	------------------------------

Command History	Release	Modification
	15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

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 <b>filter</b> keyword to limit the command output.
------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Examples	This example shows how to use the <b>show macro auto monitor type</b> privileged EXEC command with no optional keywords to view the devices recognized by the device classifier:
----------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

```
Switch# show macro auto monitor 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
```

## show macro auto monitor type

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

**Related Commands**

<b>Command</b>	<b>Description</b>
<b>macro auto device</b>	Configures macro default parameter values.
<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
<b>macro auto mac-address-group</b>	Configures MAC address groups.
<b>macro auto sticky</b>	Configures macro persistence.
<b>macro auto trigger</b>	Specifies a trigger to be associated with the device type or profile name.
<b>shell trigger</b>	Creates event triggers.
<b>show macro auto monitor device</b>	Displays information about the devices connected to the switch.
<b>show shell</b>	Displays information about event triggers and macros.

# show parser macro

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

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

Syntax Description	Parameter	Description
	<b>brief</b>	(Optional) Displays the name of each macro.
	<b>description [interface <i>interface-id</i>]</b>	(Optional) Displays all macro descriptions or the description of a specific interface.
	<b>name <i>macro-name</i></b>	(Optional) Displays information about a single macro identified by the macro name.

Command Modes	Mode
	User EXEC Privileged EXEC

Command History	Release	Modification
	15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

## Examples

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:

```
Switch# 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 is an example of output from the **show parser macro name** command:

```
Switch# 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 is an example of output from the **show parser macro brief** command:

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

## show parser macro

```
customizable      : snmp
```

This is an example of output from the **show parser macro description** command:

```
Switch# 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 is an example of output from the **show parser macro description interface** command:

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

### Related Commands

Command	Description
<b>macro apply</b>	Applies a macro on an interface or applies and traces a macro on an interface.
<b>macro description</b>	Adds a description about the macros that are applied to an interface.
<b>macro global</b>	Applies a macro on a switch or applies and traces a macro on a switch.
<b>macro global description</b>	Adds a description about the macros that are applied to the switch.
<b>show running-config</b>	Displays the operating configuration.

# show shell

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

**show shell** [**environment** | **functions** [**brief** | *shell\_function*] | **triggers**]

Syntax Description	environment	(Optional) Displays shell environment information.
	<b>functions</b> [ <b>brief</b>   <i>shell_function</i> ]	(Optional) Displays macro information. <ul style="list-style-type: none"> <li>• <b>brief</b>—Names of the shell functions.</li> <li>• <i>shell_function</i>—Name of a shell function.</li> </ul>
	<b>triggers</b>	(Optional) Displays event trigger information.

Command Modes	User EXEC Privileged EXEC
---------------	------------------------------

Command History	Release	Modification
	15.0(2)EX	This command was introduced on Catalyst 2960-X switches.

**Usage Guidelines** Use this command to display the shell information for the switch.

The **show shell** command is a feature at the Cisco IOS level. You may first have to enable Cisco IOS Shell by entering the **terminal shell** command before you can enter the **show shell** command. For more information, see the Cisco IOS Shell configuration guide on Cisco.com:  
[http://www.cisco.com/en/US/docs/ios/netmgmt/configuration/guide/nm\\_ios\\_shell.pdf](http://www.cisco.com/en/US/docs/ios/netmgmt/configuration/guide/nm_ios_shell.pdf)

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

```
Switch# 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=(),  
 \$SEC\_MODE=(), \$SEC\_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** privileged EXEC command to view the built-in macros in the switch software:

```
Switch# 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
            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
        fi
    exit
end
fi
}
<output truncated>
```

**show shell**

<b>Related Commands</b>	<b>Command</b>	<b>Description</b>
	<b>macro auto device</b>	Configures macro default parameter values.
	<b>macro auto execute</b>	Configures mapping from an event trigger to a built-in macro.
	<b>macro auto global processing</b>	Enables Auto Smartports on a switch.
	<b>macro auto mac-address-group</b>	Configures MAC address groups.
	<b>macro auto sticky</b>	Configures macro persistence.
	<b>shell trigger</b>	Creates event triggers.
	<b>show macro auto</b>	Displays information about macros.