



Configuring SmartPort Macros

This chapter describes how to configure and apply SmartPort macros on your switch.



Note

For complete syntax and usage information for the switch commands used in this chapter, refer to the *Catalyst 4500 Series Switch Cisco IOS Command Reference* and related publications at <http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/index.htm>.

This chapter consists of these sections:

- [Understanding SmartPort Macros, page 12-1](#)
- [Configuring Smart-Port Macros, page 12-2](#)
- [Displaying SmartPort Macros, page 12-8](#)

Understanding SmartPort Macros

SmartPort macros provide a convenient way to save and share common configurations. You can use SmartPort macros to enable features and settings based on the location of a switch in the network and for mass configuration deployments across the network.

Each SmartPort macro is a set of CLI commands that you define. SmartPort macro sets do not contain new CLI commands; Each SmartPort macro is a group of existing CLI commands.

When you apply a SmartPort macro on an interface, the CLI commands contained within the macro are configured on the interface. When the macro is applied to an interface, the existing interface configurations are not lost. The new commands are added to interface and are saved in the running configuration file.

Configuring Smart-Port Macros

You can create a new SmartPort macro or use an existing macro as a template to create a new macro that is specific to your application. After you create the macro, you can apply it to an interface or a range of interfaces.

This section includes information about these topics:

- [Default SmartPort Macro Configuration, page 12-2](#)
- [SmartPort Macro Configuration Guidelines, page 12-4](#)
- [Creating and Applying SmartPort Macros, page 12-4](#)

Default SmartPort Macro Configuration

This section illustrates the default configurations for the four supported macros. These macros can only be viewed and applied; they cannot be modified by the user.

- [cisco-desktop, page 12-2](#)
- [cisco-phone, page 12-2](#)
- [cisco-switch, page 12-3](#)
- [cisco-router, page 12-3](#)

cisco-desktop

This is the example for the cisco-desktop macro:

```
# 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
# Ensure port-security age is greater than one minute
# and use inactivity timer
# "Port-security maximum 1" is the default and will not
# Show up in the config
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
```

cisco-phone

This is the example for the cisco-phone macro:

```
# 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
# Update the Voice VLAN (VVID) value which should be
# different from data VLAN
```

```

# Recommended value for voice vlan (VVID) should not be 1
switchport voice vlan $VVID
# Enable port security limiting port to a 3 MAC
# addressess -- One for desktop and two for phone
switchport port-security
switchport port-security maximum 3
# 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
# Enable auto-qos to extend trust to attached Cisco phone
auto qos voip cisco-phone
# Configure port as an edge network port
spanning-tree portfast
spanning-tree bpduguard enable@

```

cisco-switch

This is the example for the cisco-switch macro:

```

# Access Uplink to Distribution
switchport trunk encapsulation dot1q
# Define unique Native VLAN on trunk ports
# Recommended value for native vlan (NVID) should not be 1
switchport trunk native vlan $NVID
# Update the allowed VLAN range (VRANGE) such that it
# includes data, voice and native VLANs
# switchport trunk allowed vlan $VRANGE
# Hardcode trunk and disable negotiation to
# speed up convergence
switchport mode trunk
switchport nonegotiate
# Configure qos to trust this interface
auto qos voip trust
# 802.1w defines the link as pt-pt for rapid convergence
spanning-tree link-type point-to-point

```

cisco-router

This is the example for the cisco-router macro:

```

# Access Uplink to Distribution
switchport trunk encapsulation dot1q
# Define unique Native VLAN on trunk ports
# Recommended value for native vlan (NVID) should not be 1
switchport trunk native vlan $NVID
# Update the allowed VLAN range (VRANGE) such that it
# includes data, voice and native VLANs
# switchport trunk allowed vlan $VRANGE
# Hardcode trunk and disable negotiation to
# speed up convergence
# Hardcode speed and duplex to router
switchport mode trunk
switchport nonegotiate
speed 100
duplex full
# Configure qos to trust this interface
auto qos voip trust
qos trust dscp
# Ensure fast access to the network when enabling the interface.
# Ensure that switch devices cannot become active on the interface.

```

```
spanning-tree portfast
spanning-tree bpduguard enable
```

SmartPort Macro Configuration Guidelines

Follow these guidelines when configuring macros on your switch:

- Do not use **exit** or **end** commands when creating a macro. This action could cause commands that follow **exit** or **end** to execute in a different command mode.
- When creating a macro, all CLI commands should be interface configuration mode commands.
- Some CLI commands are specific to certain interface types. The macro will fail the syntax check or the configuration check, and the switch will return an error message if it is applied to an interface that does not accept the configuration.
- When a macro is applied to an interface, all existing configuration on the interface is retained. This is helpful when applying an incremental configuration to an interface.
- If you modify a macro definition by adding or deleting commands, the changes are not reflected on the interface where the original macro was applied. You need to reapply the updated macro on the interface to apply the new or changed commands.
- You can use the **macro trace** *macro-name* interface configuration command to show what macros are running on an interface or to debug the macro to determine any syntax or configuration errors.
- If a command fails when you apply a macro, either due to a syntax error or to a configuration error, the macro continues to apply the remaining commands to the interface.
- 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 individual interface within the range. If a macro command fails on one interface, it is still applied to the remaining interfaces.
- Specific keywords are required when you apply the system-defined macros (**cisco-desktop**, **cisco-phone**, **cisco-switch**, and **cisco-router**) on an interface.
- At most, 3 keyword-value pairs are allowed per system-defined macro.

Creating and Applying SmartPort Macros

To create and apply a SmartPort macro, perform the following task:

| | Command | Purpose |
|--------|--|--|
| Step 1 | Switch # configure terminal | Enters global configuration mode. |
| Step 2 | Switch(config)# macro name <i>macro-name</i> | Creates a macro definition, and enters a macro name. A macro definition can contain up to 3000 characters. Enters the macro commands with one command per line. Use the @ character to end the macro. Use the # character at the beginning of a line to enter comment text within the macro. Do not use the exit or end commands in a macro. This action could cause any commands following exit or end to execute in a different command mode. For best results, all commands in a macro should be interface configuration mode commands. |

| | Command | Purpose |
|--------|--|---|
| Step 3 | Switch(config)# interface <i>interface-id</i> | Enters interface configuration mode and specifies the interface on which to apply the macro. |
| Step 4 | Switch(config-if)# macro { apply trace } <i>macro-name</i> [<i>keyword</i>] [<i>value</i>] [<i>keyword</i>] [<i>value</i>] [<i>keyword</i>] [<i>value</i>] | Applies each command defined in the macro to the interface. As of Cisco IOS Release 12.2(20)EWA, the system defined macros are cisco-desktop , cisco-phone , cisco-switch , and cisco-router . |
| Step 5 | Switch(config-if)# macro description <i>text</i> | (Optional) Enters a description about the macro that is applied to the interface. |
| Step 6 | Switch(config-if)# end | Returns to privileged EXEC mode. |
| Step 7 | Switch# show parser macro | Verifies that the macro was created. |
| Step 8 | Switch# show running-config interface <i>interface-id</i> | Verifies that the macro is applied to an interface. |
| Step 9 | Switch# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

The **no** form of the **macro name** global configuration command only deletes the macro definition. It does not affect the configuration of those interfaces on which the macro is already applied. You can delete a macro-applied configuration on an interface by entering the **default interface** *interface-id* interface configuration command. Alternatively, you can create an *anti-macro* for an existing macro that contains the **no** form of all the corresponding commands in the original macro. Then apply the anti-macro to the interface.

The following sections illustrate how to apply and display the attachments on each of the supported macros:

- [cisco-desktop](#), page 12-5
- [cisco-phone](#), page 12-6
- [cisco-switch](#), page 12-7
- [cisco-router](#), page 12-7

cisco-desktop

This example shows how to use the system-defined macro **cisco-desktop** to assign a value of 35 to the access VLAN of the Fast Ethernet interface 2/9.



Note

This macro requires the **\$AVID** keyword, which is the access VLAN of the port.

```
Switch(config)# interface fastethernet2/9
Switch(config-if)# macro apply cisco-desktop $AVID 35
Switch(config-if)# end
Switch# show parser macro name cisco-desktop
Macro name : cisco-desktop
Macro type : customizable

# Basic interface - Enable data VLAN only
# Recommended value for access vlan (AVID) should not be 1
switchport access vlan $AVID [access_vlan_id]
switchport mode access
# Enable port security limiting port to a single
# MAC address -- that of desktop
```

```

switchport port-security
# Ensure port-security age is greater than one minute
# and use inactivity timer
# "Port-security maximum 1" is the default and will not
# Show up in the config
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# show parser macro description
Interface      Macro Description
-----
Fa2/9          cisco-desktop
-----

```

cisco-phone

This example shows how to use the system-defined macro **cisco-phone** to assign a value of 35 to the access VLAN and 56 to the voice VLAN on the Fast Ethernet interface 2/9.



Note

This macro requires the **\$AVID** and **\$VVID** keywords, which are the access and voice VLANs of the port.

```

Switch(config)# interface fastethernet2/9
Switch(config-if)# macro apply cisco-phone
Switch(config-if)# macro description cisco-phone $AVID 35 $VVID 56
Switch(config-if)# end
Switch# show parser macro name cisco-phone
Macro name : cisco-phone
Macro type : customizable

# VoIP enabled interface - Enable data VLAN
# and voice VLAN (VVID)
# Recommended value for access vlan (AVID) should not be 1\
switchport access vlan $AVID [access_vlan_id]
switchport mode access
# Update the Voice VLAN (VVID) value which should be
# different from data VLAN
# Recommended value for voice vlan (VVID) should not be 1
switchport voice vlan $VVID [voice_vlan_id]
# Enable port security limiting port to a 3 MAC
# addresses -- One for desktop and two for phone
switchport port-security
switchport port-security maximum 3
# 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
# Enable auto-qos to extend trust to attached Cisco phone
auto qos voip cisco-phone
# Configure port as an edge network port
spanning-tree portfast
spanning-tree bpduguard enable@

Switch# show parser macro description
Interface      Macro Description
-----

```

```
Fa2/9          cisco-phone
-----
```

cisco-switch

This example shows how to use the system-defined macro **cisco-switch** to assign a value of 38 to the native VLAN on the Fast Ethernet interface 2/9.



Note

This macro requires the **\$NVID** keyword, which is the native VLANs of the port.

```
Switch(config)# interface fastethernet2/9
Switch(config-if)# macro apply cisco-switch
Switch(config-if)# macro description cisco-switch $NVID 38
Switch(config-if)# end
Switch# show parser macro name cisco-switch
Macro name : cisco-switch
Macro type : customizable
```

```
# Access Uplink to Distribution
switchport trunk encapsulation dot1q
# Define unique Native VLAN on trunk ports
# Recommended value for native vlan (NVID) should not be 1
switchport trunk native vlan $NVID [native_vlan_id]
# Update the allowed VLAN range (VRANGE) such that it
# includes data, voice and native VLANs
# switchport trunk allowed vlan $VRANGE [vlan_range]
# Hardcode trunk and disable negotiation to
# speed up convergence
switchport mode trunk
switchport nonegotiate
# Configure qos to trust this interface
auto qos voip trust
# 802.1w defines the link as pt-pt for rapid convergence
spanning-tree link-type point-to-point
```

```
Switch# show parser macro description
Interface      Macro Description
-----
Fa2/9          cisco-switch
-----
```

cisco-router

This example shows how to use the system-defined macro **cisco-router** to assign a value of 451 to the native VLAN on the Fast Ethernet interface 2/9.



Note

This macro requires the **\$NVID** keyword, which is the native VLANs of the port.

```
Switch(config)# interface fastethernet2/9
Switch(config-if)# macro apply cisco-router
Switch(config-if)# macro description cisco-router $NVID 451
Switch(config-if)# end
Switch# show parser macro name cisco-router
Macro name : cisco-router
Macro type : customizable
```

```
# Access Uplink to Distribution
```

```

switchport trunk encapsulation dot1q
# Define unique Native VLAN on trunk ports
# Recommended value for native vlan (NVID) should not be 1
switchport trunk native vlan $NVID [native_vlan_id]
# Update the allowed VLAN range (VRANGE) such that it
# includes data, voice and native VLANs
# switchport trunk allowed vlan $VRANGE [vlan_range]
# Hardcode trunk and disable negotiation to
# speed up convergence
# Hardcode speed and duplex to router
switchport mode trunk
switchport nonegotiate
speed 100
duplex full
# Configure qos to trust this interface
auto qos voip trust
qos trust dscp
# Ensure fast access to the network when enabling the interface.
# Ensure that switch devices cannot become active on the interface.
spanning-tree portfast
spanning-tree bpduguard enable

Switch# show parser macro description
Interface      Macro Description
-----
Fa2/9          cisco-router
-----

```

Displaying SmartPort Macros

To display the SmartPort macros, use one or more of the privileged EXEC commands in [Table 12-1](#).

Table 12-1 Commands for Displaying SmartPort Macros

| Command | Purpose |
|---|---|
| show parser macro | Displays all configured macros. |
| show parser macro name <i>macro-name</i> | Displays a specific macro. |
| show parser macro brief | Displays the configured macro names. |
| show parser macro description [interface <i>interface-id</i>] | Displays the macro description for all interfaces or for a specified interface. |