



Configuring Gigabit Ethernet SPAs

- [Default Configuration Values, page 1](#)
- [Specifying the Interface Address on a SPA, page 2](#)
- [Configuring Gigabit Ethernet SPAs, page 2](#)
- [Modifying the MAC Address on the Interface, page 3](#)
- [Gathering MAC Address Accounting Statistics, page 4](#)
- [Configuring Autonegotiation on an Interface, page 5](#)
- [Configuring Ethernet Flow Control, page 5](#)
- [Modifying the Interface MTU Size, page 6](#)
- [Configuring the Encapsulation Type and VLANs, page 7](#)
- [Configuring the Hold Queue, page 7](#)
- [Configuring EtherChannels, page 8](#)
- [Configuring a Priority Queue on Gigabit Ethernet SPAs, page 8](#)
- [Configuring the Interface for Internal Loopback, page 14](#)
- [Preprovisioning the SIPs and Gigabit Ethernet SPAs, page 14](#)
- [Gigabit Ethernet SPAs Configuration Examples, page 15](#)

Default Configuration Values

The table below lists the default configuration parameters when an interface is enabled on a Gigabit Ethernet SPA.

Table 1: Gigabit Ethernet SPA Default Configuration Values

Parameter	Default Value
Autonegotiation	Enabled (supported only on Cisco 5-Port Gigabit Ethernet SPA)

Parameter	Default Value
Flow control	Enabled
Hold queue	Disabled
IEEE 802.1Q encapsulation	Disabled
MAC accounting	Disabled
MAC address	Hardware burned in address (BIA)
MTU	1514 for normal interfaces, 1518 for 802.1Q VLAN

Specifying the Interface Address on a SPA

Gigabit Ethernet SPAs on the Cisco uBR10012 router use an addressing format that specifies the physical location of the SIP, SPA, and interface. The interface address format is *slot/subslot/port*, where:

- *slot*—Specifies the slot number in the Cisco uBR10012 router where the SIP is installed.
- *subslot*—Specifies the secondary slot of the SIP where the SPA is installed.
- *port*—Specifies the number of the individual interface port on a SPA.

The following example shows how to specify the second interface (1) on a Gigabit Ethernet SPA installed in the first subslot of a SIP (0) installed in chassis slot 3:

```
Router(config)# interface gigabitethernet 3/0/1
```

Configuring Gigabit Ethernet SPAs

To configure a Gigabit Ethernet SPA, follow this procedure:



Note

Perform either

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example: Router> enable</p>	<p>Enables privileged EXEC mode.</p> <p>Enter your password if prompted.</p>

	Command or Action	Purpose
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	Use the appropriate command based on the Gigabit Ethernet SPA in use: <ul style="list-style-type: none"> • interface gigabitethernet <i>slot/subslot/port[.subinterface-number]</i> • interface tengigabitethernet <i>slot/subslot/port[.subinterface-number]</i> Example: Router(config)# interface gigabitethernet 5/0/0	Specifies the Gigabit Ethernet or the 10-Gigabit Ethernet interface to enter interface configuration mode. <ul style="list-style-type: none"> • <i>slot/subslot/port</i>—Specifies the location of the interface. • <i>subinterface-number</i>—(Optional) Specifies a secondary interface (subinterface) number.
Step 4	Use the appropriate command: <ul style="list-style-type: none"> • ip address <i>ip-address mask</i> • ipv6 address <i>ipv6-prefix/prefix-length</i> Example: Router(config-if)# ip address 172.18.189.38 255.255.255.224 or Router(config-if)# ipv6 address 3000:1116::1:3:300:1/112	Assigns an IP address to the interface. <ul style="list-style-type: none"> • <i>ip-address mask</i>—Specifies an IP address and subnet mask. • <i>ipv6-prefix/prefix-length</i>—Specifies an IPv6 network address and prefix length.
Step 5	no shutdown Example: Router(config-if)# no shutdown	Removes the shutdown configuration, which forces an interface administratively down.
Step 6	end Example: Router# end	Returns to privileged EXEC mode.

Modifying the MAC Address on the Interface

The Gigabit Ethernet SPAs use a default MAC address for each port that is derived from the base address that is stored in the electrically erasable programmable read-only memory (EEPROM) on the backplane of the Cisco uBR10012 router.

To modify the default MAC address of an interface to some user-defined address, do the following:

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	mac-address <i>ieee-address</i> Example: Router(config-if) # mac-address <i>ieee-address</i>	Modifies the default MAC address of an interface to a user-defined address. <i>ieee-address</i> —Specifies the 48-bit Institute of Electrical and Electronics Engineers (IEEE) MAC address written as a dotted triple of four-digit hexadecimal numbers (<i>xxxx.yyyy.zzzz</i>). To return to the default MAC address on the interface, use the no form of the command.
Step 4	end Example: Router# end	Returns to privileged EXEC mode.

Gathering MAC Address Accounting Statistics

MAC address accounting provides accounting information for IP traffic based on the source and destination MAC addresses of the LAN interfaces. MAC address accounting calculates the total packet and byte counts for a LAN interface that receives or sends IP packets to or from a unique MAC address. It also records a time stamp for the last packet received or sent. With MAC address accounting, you can determine how much traffic is being sent to or received from various peers.

To enable MAC address accounting on an interface, do the following:

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Router# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ip accounting mac-address [input output] Example: Router(config-if)# <code>ip accounting mac-address input</code>	Enables MAC address accounting on an interface. <ul style="list-style-type: none"> • input—Performs accounting based on the source MAC address on received packets. • output—Performs accounting based on the destination MAC address on transmitted packets.
Step 4	end Example: Router# <code>end</code>	Returns to privileged EXEC mode.

Configuring Autonegotiation on an Interface

Autonegotiation is enabled by default and can be disabled on the 5-Port Gigabit Ethernet SPA. During autonegotiation, advertisement for flow control, speed, and duplex occurs. If autonegotiation is disabled on one end of a link, it must be disabled on the other end of the link. If one end of a link has autonegotiation disabled while the other end of the link does not, the link does not come up properly on both ends. Flow control is always negotiated when autonegotiation is enabled.



Note

Autonegotiation is not supported on the 1-Port 10-Gigabit Ethernet SPA.

To disable autonegotiation on the 5-Port Gigabit Ethernet SPA, use the **no negotiation auto** command in interface configuration mode.

```
Router(config-if)# no negotiation auto
```

No advertisement of flow control occurs.

To re-enable autonegotiation on a Gigabit Ethernet interface, use the **negotiation auto** command in interface configuration mode.

```
Router(config-if)# negotiation auto
```

Advertisement of flow control occurs.

Configuring Ethernet Flow Control

The Ethernet flow control feature is a mechanism for temporarily stopping the transmission of data between two peers to prevent packet drops in the event of data overflow. The overwhelmed network element will send a PAUSE frame, which halts transmission from the sending node for a specified period of time. Flow control is enabled by default on the 1-Port 10-Gigabit Ethernet SPA.

**Note**

Flow control is always negotiated when autonegotiation is enabled.

To disable flow control on the 1-Port 10-Gigabit Ethernet SPA, use the **flowcontrol receive off** command in interface configuration mode.

```
Router(config-if)# flowcontrol receive off
```

When the flow control is disabled, the PAUSE frames received from its peer are ignored and transmission continues.

To re-enable flow control on a Gigabit Ethernet interface, use the **no flowcontrol receive off** command in interface configuration mode.

```
Router(config-if)# no flowcontrol receive off
```

Modifying the Interface MTU Size

The Cisco IOS software supports three different types of configurable maximum transmission unit (MTU) options at different levels of the protocol stack:

- **Interface MTU**—Checked by the SPA on traffic coming in from the network. Different interface types support different interface MTU sizes and defaults. The interface MTU defines the maximum packet size allowable (in bytes) for an interface before drops occur. If the frame is smaller than the interface MTU size, but is not smaller than the minimum frame size for the interface type (such as 64 bytes for Ethernet), then the frame continues to process.
- **IP MTU**—Can be configured on an interface or a subinterface and is used by the Cisco IOS software to determine whether fragmentation of a packet takes place. If an IP packet exceeds the IP MTU size, then the packet is fragmented.
- **Tag or Multiprotocol Label Switching (MPLS) MTU**—Can be configured on an interface or a subinterface and allows up to six different labels, or tag headers, to be attached to a packet. The maximum number of labels is dependent on your Cisco IOS software release.

**Note**

For the Gigabit Ethernet SPAs on the Cisco uBR10012 router, the default MTU size is 1500 bytes. When the interface is being used as a Layer 2 port, the maximum configurable MTU is 9000 bytes. The SPA automatically adds an additional 22 bytes to the configured MTU size to accommodate some of the additional overhead. However, when the Gigabit Ethernet SPAs are configured in the interface tunnel, the configurable MTU is 9180 bytes; but it is highly recommended to use a maximum packet size of 9132 bytes.

Interface MTU Configuration Guidelines

When configuring the interface MTU size on a Gigabit Ethernet SPA on a Cisco uBR10012 router, consider the following guidelines:

- The default interface MTU size accommodates a 1500-byte packet, plus 22 additional bytes to cover the following additional overhead:
 - Layer 2 header—14 bytes

- Dot1Q header—4 bytes
- CRC—4 bytes



Note Depending on your Cisco IOS software release, a certain maximum number of MPLS labels are supported. If you need to support more than two MPLS labels, then you must increase the default interface MTU size.

- If you are using MPLS, be sure that the **mpls mtu** command is configured for a value less than or equal to the interface MTU.
- If you are using MPLS labels, then you should increase the default interface MTU size to accommodate the number of MPLS labels. Each MPLS label adds 4 bytes of overhead to a packet.

Interface MTU Configuration Task

To modify the MTU size on an interface, use the **mtu bytes** command in interface configuration mode.

```
Router(config-if)# mtu 9128
```

The default is 1500 bytes. The maximum configurable MTU is 9129 bytes.

bytes—Specifies the maximum number of bytes for a packet.

To return to the default MTU size, use the **no** form of the command.

Configuring the Encapsulation Type and VLANs

By default, the interfaces on the Gigabit Ethernet SPAs support Advanced Research Projects Agency (ARPA) encapsulation. They do not support configuration of service access point or SNAP encapsulation for transmission of frames; however, the interfaces will properly receive frames that use service access point and SNAP encapsulation.

The Cisco uBR10012 router supports virtual local area networks (VLANs) to separate a service provider's subscriber traffic. Each Gigabit Ethernet SPA supports up to 8000 VLANs.

The other encapsulation supported by the SPA interfaces is IEEE 802.1Q encapsulation for VLANs. To create a subinterface on a SPA interface port and configure the subinterface on a VLAN using IEEE 802.1Q encapsulation, use the **encapsulation dot1q vlan-id** command in subinterface configuration mode.

Configuring the Hold Queue

You can limit the size of the input queue on a Gigabit Ethernet SPA in units of packets. The input hold queue prevents a single interface from flooding the network server with too many input packets. Further input packets are discarded if the interface has too many input packets outstanding in the system. The default input hold queue limit is 75 packets.

To modify the input hold queue on a Gigabit Ethernet SPA, use the **hold-queue length in** command in interface configuration mode.

The *length* specifies the maximum number of packets in the queue. The valid values are from 0 to 4096.

Configuring EtherChannels

An EtherChannel bundles individual Gigabit Ethernet links into a single logical link that provides the aggregate bandwidth of up to four physical links. This feature helps improve the cost effectiveness of a device by increasing cumulative bandwidth without requiring hardware upgrades. In addition, IEEE 802.3ad link bundling provides a capability to dynamically provision, manage, and monitor various aggregated links and enables interoperability between various Cisco devices and the devices of third-party vendors.



Note Gigabit Ethernet SPAs on the Cisco uBR10012 routers do not support Link Aggregation Control Protocol (LACP).

Configuring a Priority Queue on Gigabit Ethernet SPAs

The Gigabit Ethernet SPAs provide the ability to separate high-priority traffic from low-priority traffic and places the traffic in the appropriate interface queue. Priority and nonpriority traffic are separated at the SIP to prevent the dropping of high-priority traffic in an oversubscription case.



Note Each SPA supports one priority queue.

The following classification types are available to prioritize ingress traffic on the Gigabit Ethernet SPAs:

- VLAN 802.1Q priority bits
- IP DSCP bits
- IP precedence bits
- IPv6 traffic-class bits
- MPLS EXP bits

Classifying Ingress VLAN Traffic

To classify ingress VLAN traffic based on the 802.1Q priority bits, use the following commands in interface and subinterface configuration mode:

Command	Purpose
<code>plim qos input map cos enable</code>	Enables classification of ingress VLAN traffic according to the 802.1Q priority bits. Note This command can only be applied to VLAN interfaces.

Command	Purpose
plim qos input map cos <i>cos-value</i> queue low-latency	<p>(Optional) Classifies incoming VLAN traffic on a subinterface according to the 802.1Q priority bits and places the traffic into the appropriate queue.</p> <ul style="list-style-type: none"> • <i>cos-value</i>—Specifies an IEEE 802.1Q/ISL CoS value from 0 to 7. <p>Note When you configure a class of service (CoS) value on a QinQ subinterface, the CoS value applies to all QinQ subinterfaces with the same outer VLAN ID.</p> <ul style="list-style-type: none"> • low-latency—Specifies the high-priority queue.

Classifying Ingress IP Traffic According to DSCP Bits

To classify ingress IP traffic based on the value of the DSCP bits, use the following commands in interface configuration mode:

Command	Purpose
plim qos input map ip dscp-based	<p>Enables the classification of incoming IP traffic according to the value of the DSCP bits.</p> <p>Note This command only applies to physical interfaces.</p>

Command	Purpose
<p>plim qos input map ip dscp <i>dscp-value</i> queue low-latency</p>	<p>Classifies incoming IP traffic according to the value of the DSCP bits and places the traffic into the appropriate queue. By default, IP traffic with the DSCP bits will use the low-latency queue, and traffic with any other DSCP value will use the low-priority queue.</p> <ul style="list-style-type: none"> • <i>dscp-value</i>—Value of the DSCP bits. Values can be one of the following: <ul style="list-style-type: none"> 0 to 63—Differentiated services code point value af11—001010 af12—001100 af13—001110 af21—010010 af22—010100 af23—010110 af31—011010 af32—011100 af33—011110 af41—100010 af42—100100 af43—100110 cs1—Precedence 1 (001000) cs2—Precedence 2 (010000) cs3—Precedence 3 (011000) cs4—Precedence 4 (100000) cs5—Precedence 5 (101000) cs6—Precedence 6 (110000) cs7—Precedence 7 (111000) default—000000 ef—101110 <p>A range of values can be specified separated by a dash (-), or a list of values can be specified.</p> <ul style="list-style-type: none"> • low-latency—Specifies the high-priority queue.

Classifying Ingress IP Traffic According to IP Precedence Bits

To classify ingress IP traffic based on the value of the IP precedence bits, use the following commands in interface configuration mode:

Command	Purpose
plim qos input map ip precedence-based	Enables the classification of incoming IP traffic according to the IP precedence value. Note This command only applies to physical interfaces.
plim qos input map ip precedence <i>precedence-value</i> queue low-latency	Classifies incoming IP traffic according to the value of the IP precedence bits and places the traffic into the appropriate queue. <i>precedence-value</i> —Value of the IP precedence bits (0 to 7). A range of values can be specified separated by a dash (-), or a list of values can be specified. low-latency —Specifies the high-priority queue.

Classifying Ingress IPv6 Traffic According to Traffic-Class Bits

To classify ingress IPv6 traffic based on the value of the traffic-class bits, use the following command in interface configuration mode:

Command	Purpose
plim qos input map ipv6 tc <i>tc-value</i> queue low-latency	

Command	Purpose
	<p>Classifies ingress IPv6 traffic based on the value of the traffic-class bits and places the traffic into the appropriate queue.</p> <p>By default, IPv6 traffic with a traffic-class value uses the high-priority queue, and all other traffic will use the low-priority queue. Only the most significant six bits of the traffic-class octet is used for the classification.</p> <p>Note This command only applies to physical interfaces.</p> <ul style="list-style-type: none"> • tc-value—Value of the traffic-class bits. Values can be one of the following: <ul style="list-style-type: none"> 0 to 63—Differentiated services code point value af11—001010 af12—001100 af13—001110 af21—010010 af22—010100 af23—010110 af31—011010 af32—011100 af33—011110 af41—100010 af42—100100 af43—100110 cs1—Precedence 1 (001000) cs2—Precedence 2 (010000) cs3—Precedence 3 (011000) cs4—Precedence 4 (100000) cs5—Precedence 5 (101000) cs6—Precedence 6 (110000) cs7—Precedence 7 (111000) default—000000 ef—101110 <p>A range of values can be specified separated by a dash (-), or a list of values can be specified.</p>

Command	Purpose
	<ul style="list-style-type: none"> • low-latency—Specifies the high-priority queue.

Classifying Ingress MPLS Traffic According to EXP Bits

To classify ingress MPLS traffic based on the value of the EXP bits, use the following commands in interface configuration mode:

Command	Purpose
<pre>plim qos input map mpls exp <i>exp-value</i> queue low-latency</pre>	<p>Classifies incoming MPLS traffic according to the value of the EXP bits and places the traffic into the appropriate queue.</p> <p>Note This command only applies to physical interfaces.</p> <ul style="list-style-type: none"> • <i>exp-value</i>—Value of the EXP bits (0 to 7). A range of values can be specified separated by a dash (-), or a list of values can be specified. • low-latency—Specifies the high-priority queue.

Configuring the Interface for Internal Loopback

Loopback support is useful for testing the interface without connectivity to the network, or for diagnosing equipment malfunctions between the interface and a device. The Gigabit Ethernet SPAs support internal loopback mode.

To enable internal loopback at the MAC device on a Gigabit Ethernet SPA, use the **loopback mac** command in interface configuration mode.

Preprovisioning the SIPs and Gigabit Ethernet SPAs

Preprovisioning is an optional configuration task for the Cisco 10000 Series SPA Interface Processor-600 and the Gigabit Ethernet SPAs. It allows you to preprovision a line card slot in the Cisco uBR10012 router to accept a particular SIP, and to preconfigure the SPA interfaces without the SIP being physically present in the chassis. This feature allows planning for future configurations.

For information on preprovisioning the Gigabit Ethernet SPAs and Cisco 10000 Series SPA Interface Processor-600, see the [Optional Configuration Tasks](#) section.

For preprovisioning a SPA, the subslot must be physically empty or gracefully deactivated. For information on gracefully deactivating a SPA, see the [Deactivating a SPA](#) section.

**Tip**

To display the slots, if any, that are preprovisioned for a card type, use the **show running-config | include card** command.

Removing Preprovisioning on a SIP and SPA

To remove a preprovisioning configuration from a line card slot, use the **no card** command. This removes all configuration information for that subslot, as well as any information in the SNMP MIB database about the card and its card slot.

To remove preprovisioning configuration information, the SIP slot or SPA subslot must be physically empty or gracefully deactivated. For information on deactivating SIPs and SPAs, see the [Preparing for Online Removal of a SIP](#) and [Preparing for Online Removal of a SPA](#) sections.

Gigabit Ethernet SPAs Configuration Examples

Basic Interface Configuration Example

The following example shows how to specify the interface that you want to configure, configure an IP address for the interface, and save the configuration. This example configures interface port 0 on the SPA that is located in subslot 3 of the SIP, which is installed in slot 1 of the Cisco uBR10012 router.

```
! Enter global configuration mode.
!
Router# configure terminal
! Enter configuration commands, one per line. End with CNTL/Z.
!
! Specify the interface address.
!
Router(config)# interface gigabitethernet 1/3/0
!
! Configure an IP address.
!
Router(config-if)# ip address 192.168.50.1 255.255.255.0
!
! Start the interface.
!
Router(config-if)# no shut
!
! Save the configuration to NVRAM.
!
Router(config-if)# exit
Router# copy running-config startup-config
```

MAC Address Configuration Example

The following example changes the default MAC address on the interface to 1111.2222.3333:

```
! Enter global configuration mode.
!
Router# configure terminal
! Enter configuration commands, one per line. End with CNTL/Z.
!
! Specify the interface address
!
Router(config)# interface gigabitethernet 1/3/0
!
! Modify the MAC address.
```

```
!
Router(config-if)# mac-address 1111.2222.3333
```

MAC Address Accounting Configuration Example

The following example enables MAC Address Accounting:

```
! Enter global configuration mode.
!
Router# configure terminal
! Enter configuration commands, one per line. End with CNTL/Z.
!
! Enable MAC address accounting
Router(config)# ip accounting mac-address {input | output}
Router(config-if)# ip accounting ?
access-violations Account for IP packets violating access lists on this interface
mac-address Account for MAC addresses seen on this interface
output-packets Account for IP packets output on this interface
precedence Count packets by IP precedence on this interface
Router(config-if)# ip accounting mac
Router(config-if)# ip accounting mac-address ?
input Source MAC address on received packets
output Destination MAC address on transmitted packets
Router(config-if)# ip accounting mac-address ip
Router(config-if)# ip accounting mac-address input ?
! Specify MAC address accounting for traffic entering the interface.
!
Router(config-if)# ip accounting mac-address input
! Specify MAC address accounting for traffic leaving the interface.
!
Router(config-if)# ip accounting mac-address output
Router(config-if)# end
! Verify the MAC Address on the interface.
!
Router# show interfaces GigabitEthernet 1/3/0 mac-accounting
GigabitEthernet1/3/0
Input (511 free)
000f.f7b0.5200(26 ): 124174 packets, 7450440 bytes, last: 1884ms ago
Total: 124174 packets, 7450440 bytes
Output (511 free)
000f.f7b0.5200(26 ): 135157 packets, 8109420 bytes, last: 1884ms ago
Total: 135157 packets, 8109420 bytes
```

MTU Configuration Example

The following example sets the interface MTU to 9180 bytes:

```
! Enter global configuration mode.
!
Router# configure terminal
! Enter configuration commands, one per line. End with CNTL/Z.
!
! Specify the interface address
!
Router(config)# interface gigabitethernet 1/3/0
!
! Configure the interface MTU.
!
Router(config-if)# mtu 9180
```

VLAN Configuration Example

The following example creates subinterface number 268 on the SPA interface port, and configures the subinterface on the VLAN with ID number 268 using IEEE 802.1Q encapsulation:

```
! Enter global configuration mode.
!
```

```

Router# configure terminal
! Enter configuration commands, one per line. End with CNTL/Z.
!
! Specify the interface address
!
Router(config)# interface gigabitethernet 1/3/0.268
!
! Configure dot1q encapsulation and specify the VLAN ID.
!
Router(config-subif)# encapsulation dot1q 268

```

Priority Queue Classification Configuration Example

The following example enables DSCP-based classification on the SPA that is located in subslot 3 of the SIP in slot 1 of the Cisco uBR10012 router:

```

! Enter global configuration mode.
!
Router# configure terminal
! Enter configuration commands, one per line. End with CNTL/Z.
!
! Specify the interface address
!
Router(config)# interface gigabitethernet 1/3/0
!
! Configure the priority queue classification type.
Router(config-if)# plim qos input map ip dscp-based

```

Configuring any Keyword on Subinterfaces for PPPoE-QinQ Support

The following example configures seven subinterfaces with various outer and inner VLAN IDs:

```

! Enter global configuration mode.
!
Router# configure terminal
! Enter configuration commands, one per line. End with CNTL/Z.
Router# interface GigabitEthernet3/0/0.1
Router(config-if)# encapsulation dot1q 100 second-dot1q 100
Router(config-if)# exit
Router# interface GigabitEthernet3/0/0.2
Router(config-if)# encapsulation dot1q 100 second-dot1q 200
Router(config-if)# exit
Router# interface GigabitEthernet3/0/0.3
Router(config-if)# encapsulation dot1q 100 second-dot1q 300-400,500-600
Router(config-if)# exit
Router# interface GigabitEthernet3/0/0.4
Router(config-if)# encapsulation dot1q 100 second-dot1q any
Router(config-if)# exit
Router# interface GigabitEthernet3/0/0.5
Router(config-if)# encapsulation dot1q 200 second-dot1q 50
Router(config-if)# exit
Router# interface GigabitEthernet3/0/0.6
Router(config-if)# encapsulation dot1q 200 second-dot1q 1000-2000,3000-4000
Router(config-if)# exit
Router# interface GigabitEthernet3/0/0.7
Router(config-if)# encapsulation dot1q 200 second-dot1q any

```

