



# Transparent Bridging Commands

---

Use the commands in this chapter to configure and monitor transparent bridging networks. For transparent bridging configuration information and examples, refer to the “Configuring Transparent Bridging” chapter in the *Bridging and IBM Networking Configuration Guide*.

## access-list (extended)

Use the **access-list** global configuration command to provide extended access lists that allow more detailed access lists. These lists allow you to specify both source and destination addresses and arbitrary bytes in the packet.

```
access-list access-list-number {permit | deny} source source-mask destination  
destination-mask offset size operator operand
```

### Syntax Description

<i>access-list-number</i>	Integer from 1100 to 1199 that you assign to identify one or more <b>permit/deny</b> conditions as an extended access list. Note that a list number in the range 1100 to 1199 distinguishes an extended access list from other access lists.
<b>permit</b>	Allows a connection when a packet matches an access condition. The Cisco IOS software stops checking the extended access list after a match occurs. All conditions must be met to make a match.
<b>deny</b>	Disallows a connection when a packet matches an access condition. The software stops checking the extended access list after a match occurs. All conditions must be met to make a match.
<i>source</i>	Media Access Control (MAC) Ethernet address in the form <i>xxxx.xxxx.xxxx</i> .
<i>source-mask</i>	Mask of MAC Ethernet source address bits to be ignored. The software uses the <i>source</i> and <i>source-mask</i> arguments to match the source address of a packet.
<i>destination</i>	MAC Ethernet value used for matching the destination address of a packet.
<i>destination-mask</i>	Mask of MAC Ethernet destination address bits to be ignored. The software uses the <i>destination</i> and <i>destination mask</i> arguments to match the destination address of a packet.
<i>offset</i>	Range of values that must be satisfied in the access list. Specified in decimal or in hexadecimal format in the form <i>0xnn</i> . The offset is the number of bytes from the destination address field; it is not an offset from the start of the packet. The number of bytes you need to offset from the destination address varies depending on the media encapsulation type you are using.
<i>size</i>	Range of values that must be satisfied in the access list. Must be an integer 1 to 4.

<i>operator</i>	Compares arbitrary bytes within the packet. Can be one of the following keywords: <b>lt</b> —less than <b>gt</b> —greater than <b>eq</b> —equal <b>neq</b> —not equal <b>and</b> —bitwise and <b>xor</b> —bitwise exclusive or <b>nop</b> —address match only
<i>operand</i>	Compares arbitrary bytes within the packet. The value to be compared to or masked against.

### Default

No extended access lists are established.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

After an access list is initially created, any subsequent additions (possibly entered from the terminal) are placed at the *end* of the list. In other words, you cannot selectively add or remove access list command lines from a specific access list.

An extended access list should not be used on FDDI interfaces that provide transit bridging.

---

**Note** Due to their complexity, extended access lists should only be used by those who are very familiar with the Cisco IOS software. For example, to use extended access lists, it is important to understand how different encapsulations on different media would generally require different offset values to access particular fields.

---



**Caution** Do not specify offsets into a packet that are greater than the size of the packet.

### Examples

The following example permits packets from MAC addresses 000c.1bxx.xxxx to any MAC address if the packet contains a value less than 0x55AA in the 2 bytes that begin 0x1e bytes into the packet:

```
interface ethernet 0
  bridge-group 3 output-pattern 1102
  access-list 1102 permit 000c.1b00.0000 0000.00ff.ffff
  0000.0000.0000 ffff.ffff.ffff 0x1e 2 lt 0x55aa
```

The following example permits an NOP operation:

```
interface ethernet 0
  bridge-group 3 output-pattern 1102
  access-list 1101 permit 0000.0000.0000 ffff.ffff.ffff 0000.0000.0000 ffff.ffff.ffff
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**access-list (standard)**

**access-list (type-code)**

**bridge-group output-pattern-list**

## access-list (standard)

Use the **access-list** global configuration command to establish MAC address access lists. Use the **no** form of this command to remove a single access-list entry.

```
access-list access-list-number {permit | deny} address mask
no access-list access-list-number
```

### Syntax Description

<i>access-list-number</i>	Integer from 700 to 799 that you select for the list.
<b>permit</b>	Permits the frame.
<b>deny</b>	Denies the frame.
<i>address mask</i>	48-bit MAC addresses written in dotted triplet form. The ones bits in the <i>mask</i> argument are the bits to be ignored in <i>address</i> .

### Default

No MAC address access lists are established.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Usage Guidelines

Configuring bridging access lists of type 700 may cause a momentary interruption of traffic flow.

### Example

The following example assumes that you want to disallow the bridging of Ethernet packets of all Sun workstations on Ethernet interface 1. Software assumes that all such hosts have Ethernet addresses with the vendor code 0800.2000.0000. The first line of the access list denies access to all Sun workstations, while the second line permits everything else. You then assign the access list to the input side of Ethernet interface 1.

```
access-list 700 deny 0800.2000.0000 0000.00FF.FFFF
access-list 700 permit 0000.0000.0000 FFFF.FFFF.FFFF
interface ethernet 1
 bridge-group 1 input-address-list 700
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**access-list (extended)**

**access-list (type-code)**

## access-list (type-code)

Use the **access-list** global configuration command to build type-code access lists. Use the **no** form of this command to remove a single access list entry.

```
access-list access-list-number {permit | deny} type-code wild-mask  
no access-list access-list-number
```

### Syntax Description

<i>access-list-number</i>	User-selectable number between 200 and 299 that identifies the list.
<b>permit</b>	Permits the frame.
<b>deny</b>	Denies the frame.
<i>type-code</i>	16-bit hexadecimal number written with a leading “0x”; for example, 0x6000. You can specify either an Ethernet type code for Ethernet-encapsulated packets, or a DSAP/SSAP pair for 802.3 or 802.5-encapsulated packets. Ethernet type codes are listed in the appendix “Ethernet Type Codes.”
<i>wild-mask</i>	16-bit hexadecimal number whose ones bits correspond to bits in the <i>type-code</i> argument that should be ignored when making a comparison. (A mask for a DSAP/SSAP pair should always be at least 0x0101. This is because these two bits are used for purposes other than identifying the SAP codes.)

### Default

No type-code access lists are built.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Type-code access lists can have an impact on system performance; therefore, keep the lists as short as possible and use wildcard bit masks whenever possible.

Access lists are evaluated according to the following algorithm:

- If the packet is Ethernet Type II or SNAP, the type-code field is used.
- Other packet type, then the LSAP is used.

Packets are treated according to the following algorithm:

- If the length/type field is greater than 1500, the packet is treated as an ARPA packet.
- If the length/type field is less than or equal to 1500, and the DSAP and SSAP fields are AAAA, the packet is treated as an LSAP packet.

- If the length/type field is less than or equal to 1500, and the DSAP and SSAP fields are *not* AAAA, the packet is treated using type-code filtering.

If the LSAP-code filtering is used, all SNAP and Ethernet Type II packets are bridged without obstruction. If type-code filtering is used, all LSAP packets are bridged without obstruction.

If you have both Ethernet Type II and LSAP packets on your network, you should set up access lists for both.

## Examples

The following example permits only LAT frames (type 0x6004) and filters out all other frame types:

```
access-list 201 permit 0x6004 0x0000
```

The following example filters out only type codes assigned to Digital (0x6000 to 0x600F) and lets all other types pass:

```
access-list 202 deny 0x6000 0x600F
access-list 202 permit 0x0000 0xFFFF
```

Use the last item of an access list to specify a default action; for example, permit everything else or deny everything else. If nothing else in the access list matches, the default action is normally to deny access; that is, filter out all other type codes.

## Related Commands

You can use the master indexes or search online to find documentation of related commands.

**access-list (extended)**

**access-list (standard)**

## bridge acquire

Use the **bridge acquire** global configuration command to forward any frames for stations that the system has learned about dynamically. Use the **no** form of this command to disable the behavior.

**bridge** *bridge-group* **acquire**  
**no bridge** *bridge-group* **acquire**

### Syntax Description

*bridge-group* Bridge group number specified in the **bridge protocol** command.

### Default

Enabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

When using the command default, the Cisco IOS software forwards any frames from stations that it has learned about dynamically. If you use the **no** form of this command, the bridge stops forwarding frames to stations it has dynamically learned about through the discovery process and limits frame forwarding to statically configured stations. That is, the bridge filters out all frames except those whose sourced-by or destined-to addresses have been statically configured into the forwarding cache. The **no** form of this command prevents the forwarding of a dynamically learned address.

### Example

The following example prevents the forwarding of dynamically determined source and destination addresses:

```
no bridge 1 acquire
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge address**  
**bridge protocol**

---

## bridge address

Use the **bridge address** global configuration command to filter frames with a particular MAC-layer station source or destination address. Use the **no** form of this command to disable the forwarding ability.

```
bridge bridge-group address mac-address {forward | discard} [interface]  
no bridge bridge-group address mac-address
```

### Syntax Description

<i>bridge-group</i>	Bridge group number. It must be the same number specified in the <b>bridge protocol</b> command.
<i>mac-address</i>	48-bit dotted-triplet hardware address such as that displayed by the EXEC <b>show arp</b> command, for example, 0800.cb00.45e9. It is either a station address, the broadcast address, or a multicast destination address.
<b>forward</b>	Frame sent from or destined to the specified address is forwarded as appropriate.
<b>discard</b>	Frame sent from or destined to the specified address is discarded without further processing.
<i>interface</i>	(Optional) Interface specification, such as Ethernet 0. It is added after the <b>forward</b> or <b>discard</b> keyword to indicate the interface on which that address can be reached.

### Default

Disabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Any number of addresses can be configured into the system without a performance penalty.

---

**Note** MAC addresses on Ethernets are “bit swapped” when compared with MAC addresses on Token Ring and FDDI. For example, address 0110.2222.3333 on Ethernet is 8008.4444.CCCC on Token Ring and FDDI. Access lists always use the canonical Ethernet representation. When using different media and building access lists to filter on MAC addresses, keep this point in mind. Note that when a bridged packet traverses a serial link, it has an Ethernet-style address.

---

### Examples

The following example enables frame filtering with MAC address 0800.cb00.45e9. The frame is forwarded through Ethernet interface 1:

```
bridge 1 address 0800.cb00.45e9 forward ethernet 1
```

The following example disables the ability to forward frames with MAC address 0800.cb00.45e9:

```
no bridge 1 address 0800.cb00.45e9
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge acquire**  
**bridge-group input-address-list**  
**bridge-group output-address-list**  
**bridge protocol**

## bridge bridge

Use the **bridge bridge** global configuration command to enable the bridging of a specified protocol in a specified bridge group. Use the **no** form of this command to disable the bridging of a specified protocol in a specified bridge group.

```
bridge bridge-group bridge protocol  
no bridge bridge-group bridge protocol
```

### Syntax Description

<i>bridge-group</i>	Bridge-group number. It must be the same number specified in the <b>bridge protocol</b> command.
<i>protocol</i>	Any of the supported routing protocols. The default is to bridge all of these protocols.

### Default

Bridge every protocol.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

When IRB is enabled, the default route/bridge behavior in a bridge group is to bridge all protocols. You do not have to use the **bridge bridge** command to enable bridging.

You can use the **no bridge bridge** command to disable bridging in a bridge group so that it does not bridge a particular protocol. When you disable bridging for a protocol in a bridge group, routable packets of this protocol are routed when the bridge is explicitly configured to route this protocol, and nonroutable packets are dropped because bridging is disabled for this protocol.

---

**Note** Packets of nonroutable protocols such as LAT are only bridged. You cannot disable bridging for the nonroutable traffic.

---

### Example

The following example disables bridging of IP in bridge group 1:

```
no bridge 1 bridge ip
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

```
bridge irb  
bridge protocol  
bridge route
```

## bridge circuit-group pause

Use the **bridge circuit-group pause** global configuration command to configure the interval during which transmission is suspended in a circuit group after circuit group changes take place.

**bridge** *bridge-group* **circuit-group** *circuit-group* **pause** *milliseconds*

### Syntax Description

<i>bridge-group</i>	Bridge group number specified in the <b>bridge protocol</b> command.
<i>circuit-group</i>	Number of the circuit group to which the interface belongs.
<i>milliseconds</i>	Forward delay interval. It must be a value in the range 0 to 10000 ms.

### Default

0 ms pause

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

Circuit-group changes include the addition or deletion of an interface and interface state changes.

### Example

The following example sets the circuit group pause to 5000 ms:

```
bridge 1 circuit-group 1 pause 5000
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge circuit-group source-based**  
**bridge-group circuit-group**  
**bridge protocol**  
**show bridge circuit-group**

## bridge circuit-group source-based

Use the **bridge circuit-group source-based** global configuration command to use just the source MAC address for selecting the output interface. Use the **no** form of this command to remove the interface from the bridge group.

**bridge** *bridge-group* **circuit-group** *circuit-group* **source-based**  
**no bridge** *bridge-group* **circuit-group** *circuit-group* **source-based**

### Syntax Description

*bridge-group* Bridge group number specified in the **bridge protocol** command.

*circuit-group* Number of the circuit group to which the interface belongs.

### Default

No bridge-group interface is assigned.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

For applications that depend on the ordering of mixed unicast and multicast traffic from a given source, load distribution must be based on the source MAC address only. The **bridge circuit-group source-based** command modifies the load distribution strategy to accommodate such applications.

### Example

The following example uses the source MAC address for selecting the output interface to a bridge group:

```
bridge 1 circuit-group 1 source-based
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge circuit-group pause**  
**bridge-group circuit-group**  
**bridge protocol**  
**show bridge circuit-group**

## bridge cmf

Use the **bridge cmf** global configuration command to enable constrained multicast flooding (CMF) for all configured bridge groups. Use the **no** form of this command to disable constrained multicast flooding.

**bridge cmf**  
**no bridge cmf**

### Syntax Description

This command has no arguments or keywords.

### Default

Constrained multicast flooding is disabled.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

### Example

The following example enables constrained multicast flooding for all configured bridge groups:

```
bridge cmf
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**clear bridge multicast**  
**show bridge multicast**

## bridge crb

Use the **bridge crb** global configuration command to enable the Cisco IOS software to both route and bridge a given protocol on separate interfaces within a single router. Use the **no** form of this command to disable the feature.

**bridge crb**  
**no bridge crb**

### Syntax Description

This command has no arguments or keywords.

### Defaults

Concurrent routing and bridging is disabled.

When concurrent routing and bridging has been enabled, the default behavior is to bridge all protocols that are not explicitly routed in a bridge group.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.0.

When concurrent routing and bridging is first enabled in the presence of existing bridge groups, it generates a **bridge route** configuration command for any protocol for which any interface in the bridge group is configured for routing. This is a precaution that applies only when concurrent routing and bridging is not already enabled, bridge groups exist, and the **bridge crb** command is encountered.

Once concurrent routing and bridging has been enabled, you must configure an explicit **bridge route** command for any protocol that is to be routed on interfaces in a bridge group (in addition to any required protocol-specific interface configuration).

### Example

The following command enables concurrent routing and bridging:

```
bridge crb
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge route**

## bridge domain

Use the **bridge domain** global configuration command to establish a domain by assigning it a decimal value between 1 and 10. Use the **no** form of this command to return it to a single bridge domain by choosing domain zero (0).

**bridge** *bridge-group* **domain** *domain-number*  
**no bridge** *bridge-group* **domain**

### Syntax Description

*bridge-group* Bridge group number specified in the **bridge protocol ieee** command. The **dec** keyword is not valid for this command.

*domain-number* Domain ID number you choose. The default domain number is zero; this is the domain number required when communicating to IEEE bridges that do not support this domain extension.

### Default

Single bridge domain

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Cisco has implemented a proprietary extension to the IEEE spanning-tree software in order to support multiple spanning-tree domains. You can place any number of routers within the domain. The routers in the domain, and only those routers, will then share spanning-tree information.

Use this feature when multiple routers share the same cable, and you wish to use only certain discrete subsets of these routers to share spanning-tree information with each other. This function is most useful when running other applications, such as IP UDP flooding, that use the IEEE Spanning-Tree Protocol. It can also be used to reduce the number of global reconfigurations in large bridged networks.



**Caution** Use multiple spanning-tree domains with care. Because bridges in different domains do not share spanning-tree information, bridge loops can be created if the domains are not carefully planned.

---

**Note** This command works only when the bridge group is running the IEEE Spanning-Tree Protocol.

---

### Example

The following example places bridge group 1 in bridging domain 3. Only other routers that are in domain 3 will accept spanning-tree information from this router.

```
bridge 1 domain 3
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

### **bridge protocol**

## bridge forward-time

Use the **bridge forward-time** global configuration command to specify the forward delay interval for the Cisco IOS software. Use the **no** form of this command to return the default interval.

**bridge** *bridge-group* **forward-time** *seconds*  
**no bridge** *bridge-group* **forward-time** *seconds*

### Syntax Description

*bridge-group*            Bridge group number specified in the **bridge protocol** command.

*seconds*                Forward delay interval. It must be a value in the range 10 to 200 seconds.

### Default

30-second delay

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The forward delay interval is the amount of time the software spends listening for topology change information after an interface has been activated for bridging and before forwarding actually begins.

Each bridge in a spanning tree adopts the **hello-time**, **forward-time**, and **max-age** parameters of the root bridge, regardless of what its individual configuration might be.

### Example

The following example sets the forward delay interval to 60 seconds:

```
bridge 1 forward-time 60
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge hello-time**  
**bridge max-age**  
**bridge protocol**

## bridge-group

Use the **bridge-group** interface configuration command to assign each network interface to a bridge group. Use the **no** form of this command to remove the interface from the bridge group.

```
bridge-group bridge-group  
no bridge-group bridge-group
```

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
---------------------	--

### Default

No bridge group interface is assigned.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

You can bridge on any interface, including any serial interface, regardless of encapsulation. Bridging can be configured between interfaces on different cards, although the performance is lower compared with interfaces on the same card. Also note that serial interfaces must be running with HDLC, X.25, or Frame Relay encapsulation.

---

**Note** Several modifications to interfaces in bridge groups, including adding interfaces to bridge groups, will result in any Token Ring or FDDI interfaces in that bridge group being reinitialized.

---

### Example

In the following example, Ethernet interface 0 is assigned to bridge-group 1, and bridging is enabled on this interface:

```
interface ethernet 0  
  bridge-group 1
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

```
bridge-group cbus-bridging  
bridge-group circuit-group  
bridge-group input-pattern-list  
bridge-group output-pattern-list  
bridge-group spanning-disabled
```

## bridge-group aging-time

Use the **bridge-group aging-time** global configuration command to set the length of time that a dynamic entry can remain in the bridge table from the time the entry was created or last updated. Use the **no** form of this command to return to the default aging-time interval.

**bridge-group** *bridge-group* **aging-time** *seconds*  
**no bridge-group** *bridge-group* **aging-time**

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>seconds</i>	Aging time, in the range 0 to 1000000 seconds. The default is 300 seconds.

### Default

300 seconds

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

If hosts on a bridged network are likely to move, decrease the aging-time to enable the bridge to adapt quickly to the change. If hosts do not transmit continuously, increase the aging time to record the dynamic entries for a longer time and thus reduce the possibility of flooding when the hosts transmit again.

### Example

The following example sets the aging time to 200 seconds:

```
bridge-group 1 aging-time 200
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge-group**

## bridge-group cbus-bridging

Use the **bridge-group cbus-bridging** interface configuration command to enable autonomous bridging on a ciscoBus2 controller. Use the **no** form of this command to disable autonomous bridging.

**bridge-group** *bridge-group* **cbus-bridging**  
**no bridge-group** *bridge-group* **cbus-bridging**

### Syntax Description

*bridge-group*                      Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.

### Default

Autonomous bridging is disabled.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Normally, bridging takes place on the processor card at interrupt level. When autonomous bridging is enabled, bridging takes place entirely on the ciscoBus2 controller, significantly improving performance.

You can enable autonomous bridging on Ethernet, FDDI (FCIT) and HSSI interfaces that reside on a ciscoBus2 controller. Autonomous bridging is not supported on Token Ring interfaces, regardless of the type of bus in use.

To enable autonomous bridging on an interface, that interface must first be defined as part of a bridge group. When a bridge group includes both autonomously and normally bridged interfaces, packets are autonomously bridged in some cases, but bridged normally in others. For example, when packets are forwarded between two autonomously bridged interfaces, those packets are autonomously bridged. But when packets are forwarded between an autonomously bridged interface and one that is not, the packet must be normally bridged. When a packet is flooded, the packet is autonomously bridged on autonomously bridged interfaces, but must be normally bridged on any others.

---

**Note** In order to maximize performance when using a ciscoBus2 controller, use the **bridge-group cbus-bridging** command to enable autonomous bridging on any Ethernet, FDDI, or HSSI interface.

---

---

**Note** You can only filter by MAC-level address on an interface when autonomous bridging is enabled on that interface; autonomous bridging disables all other filtering, as well as priority queuing.

---

### Example

In the following example, autonomous bridging is enabled on Ethernet interface 0:

```
interface ethernet 0
  bridge-group 1
  bridge-group 1 cbus-bridging
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

### **bridge-group**

## bridge-group circuit-group

Use the **bridge-group circuit-group** interface configuration command to assign each network interface to a bridge group. Use the **no** form of this command to remove the interface from the bridge group.

```
bridge-group bridge-group circuit-group circuit-group  
no bridge-group bridge-group circuit-group circuit-group
```

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>circuit-group</i>	Circuit group number. The range is 1 to 9.

### Default

No bridge group interface is assigned.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

Circuit groups are primarily intended for use with HDLC-encapsulated serial interfaces. They are not supported for packet-switched networks such as X.25 or Frame Relay. Circuit groups are best applied to groups of serial lines of equal bandwidth, but can accommodate mixed bandwidths as well.

---

**Note** You must configure bridging before you configure a circuit group on an interface.

---

### Example

In the following example, Ethernet interface 0 is assigned to circuit group 1 of bridge group 1:

```
interface ethernet 0  
  bridge-group 1 circuit-group 1
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

```
bridge circuit-group pause  
bridge circuit-group source-based  
show bridge circuit-group
```

## bridge-group input-address-list

Use the **bridge-group input-address-list** interface configuration command to assign an access list to a particular interface. This access list is used to filter packets received on that interface based on their MAC source addresses. Use the **no** form of this command to remove an access list from an interface.

```
bridge-group bridge-group input-address-list access-list-number  
no bridge-group bridge-group input-address-list access-list-number
```

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>access-list-number</i>	Access list number you assigned with the <b>access-list</b> command. It must be in the range 700 to 799.

### Default

No access list is assigned.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example assumes you want to disallow the bridging of Ethernet packets of all Sun workstations on Ethernet interface 1. Software assumes that all such hosts have Ethernet addresses with the vendor code 0800.2000.0000. The first line of the access list denies access to all Sun workstations, while the second line permits everything else. You then assign the access list to the input side of Ethernet interface 1.

```
access-list 700 deny 0800.2000.0000 0000.00FF.FFFF  
access-list 700 permit 0000.0000.0000 FFFF.FFFF.FFFF  
interface ethernet 1  
  bridge-group 1 input-address-list 700
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**access-list (extended)**  
**access-list (standard)**  
**bridge address**  
**bridge-group output-address-list**

## bridge-group input-lat-service-deny

Use the **bridge-group input-lat-service-deny** interface configuration command to specify the group codes by which to deny access upon input. Use the **no** form of this command to remove this access condition.

```
bridge-group bridge-group input-lat-service-deny group-list  
no bridge-group bridge-group input-lat-service-deny group-list
```

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>group-list</i>	List of LAT service groups. Single numbers and ranges are permitted. Specify a zero (0) to disable the LAT group code for the bridge group.

### Default

No group codes are specified.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Autonomous bridging must be disabled to use this command.

This command prevents the system from bridging any LAT service advertisement that has any of the specified groups set.

### Example

The following example causes any advertisements with groups 6, 8, and 14 through 20 to be dropped:

```
interface ethernet 0  
  bridge-group 1 input-lat-service-deny 6 8 14-20
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

```
bridge-group  
bridge-group input-lat-service-permit  
bridge-group output-lat-service-deny
```

## bridge-group input-lat-service-permit

Use the **bridge-group input-lat-service-permit** interface configuration command to specify the group codes by which to permit access upon input. Use the **no** form of this command to remove this access condition.

**bridge-group** *bridge-group* **input-lat-service-permit** *group-list*  
**no bridge-group** *bridge-group* **input-lat-service-permit** *group-list*

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>group-list</i>	LAT service groups. Single numbers and ranges are permitted. Specify a zero (0) to disable the LAT group code for the bridge group.

### Default

No group codes are specified.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Autonomous bridging must be disabled to use this command.

This command causes the system to bridge only those service advertisements that match at least one group in the group list specified by the *group-list* argument.

If a message specifies group codes in both the deny and permit list, the message is not bridged.

### Example

The following example bridges any advertisements from groups 1, 5, and 12 through 14:

```
interface ethernet 1
  bridge-group 1 input-lat-service-permit 1 5 12-14
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge-group input-lat-service-deny**  
**bridge-group output-lat-service-permit**

## bridge-group input-lsap-list

Use the **bridge-group input-lsap-list** interface configuration command to filter IEEE 802.2-encapsulated packets on input. Use the **no** form of this command to disable this capability.

```
bridge-group bridge-group input-lsap-list access-list-number  
no bridge-group bridge-group input-lsap-list access-list-number
```

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>access-list-number</i>	Access list number you assigned with the standard <b>access-list</b> command. Specify a zero (0) to disable the application of the access list on the bridge group.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Autonomous bridging must be disabled to use this command.

This access list is applied to all IEEE 802.2 frames received on that interface prior to the bridge-learning process. SNAP frames must also pass any applicable Ethernet type-code access list.

### Example

The following example specifies access list 203 on Ethernet interface 1:

```
interface ethernet 1  
  bridge-group 3 input-lsap-list 203
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**access-list (extended)**

**access-list (standard)**

**bridge-group**

**bridge-group output-lsap-list**

## bridge-group input-pattern-list

Use the **bridge-group input-pattern-list** interface configuration command to associate an extended access list with a particular interface in a particular bridge group. Use the **no** form of this command to disable this capability.

**bridge-group** *bridge-group* **input-pattern-list** *access-list-number*  
**no** **bridge-group** *bridge-group* **input-pattern-list** *access-list-number*

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>access-list-number</i>	Access list number you assigned using the extended <b>access-list</b> command. Specify a zero (0) to disable the application of the access list on the interface.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Autonomous bridging must be disabled to use this command.

### Example

The following command applies access list 1101 to bridge group 3 using the filter defined in group 1:

```
interface ethernet 0
  bridge-group 3 input-pattern-list 1101
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**access-list (extended)**  
**access-list (standard)**  
**bridge-group**  
**bridge-group output-pattern-list**

## bridge-group input-type-list

Use the **bridge-group input-type-list** interface configuration command to filter Ethernet- and SNAP-encapsulated packets on input. Use the **no** form of this command to disable this capability.

```
bridge-group bridge-group input-type-list access-list-number  
no bridge-group bridge-group input-type-list access-list-number
```

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>access-list-number</i>	Access list number you assigned with the standard <b>access-list</b> command. Specify a zero (0) to disable the application of the access list on the bridge group.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Autonomous bridging must be disabled to use this command.

For SNAP-encapsulated frames, the access list is applied against the 2-byte TYPE field given after the DSAP/SSAP/OUI fields in the frame.

This access list is applied to all Ethernet and SNAP frames received on that interface prior to the bridge learning process. SNAP frames must also pass any applicable IEEE 802 DSAP/SSAP access lists.

### Example

The following example shows how to configure a Token Ring interface with an access list that allows only the LAT protocol to be bridged:

```
interface tokenring 0  
  ip address 131.108.1.1 255.255.255.0  
  bridge-group 1  
  bridge-group 1 input-type-list 201
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**access-list (extended)**

**access-list (standard)**

**bridge-group**

**bridge-group output-type-list**

## bridge-group lat-compression

Use the **bridge-group lat-compression** interface configuration command to reduce the amount of bandwidth that LAT traffic consumes on the serial interface by specifying a LAT-specific form of compression. Use the **no** form of this command to disable LAT compression on the bridge group.

**bridge-group** *bridge-group* **lat-compression**  
**no bridge-group** *bridge-group* **lat-compression**

### Syntax Description

*bridge-group*                      Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Autonomous bridging must be disabled to use this command.

Compression is applied to LAT frames being sent out the router through the interface in question.

LAT compression can be specified only for serial interfaces. For the most common LAT operations (user keystrokes and acknowledgment packets), LAT compression reduces LAT's bandwidth requirements by nearly a factor of two.

### Example

The following example compresses LAT frames on the bridge assigned to group 1:

```
bridge-group 1 lat-compression
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge-group**

## bridge-group output-address-list

Use the **bridge-group output-address-list** interface configuration command to assign an access list to a particular interface for filtering the MAC destination addresses of packets that would ordinarily be forwarded out that interface. Use the **no** form of this command to remove an access list from an interface.

```
bridge-group bridge-group output-address-list access-list-number  
no bridge-group bridge-group output-address-list access-list-number
```

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>access-list-number</i>	Access list number you assigned with the standard <b>access-list</b> command.

### Default

No access list is assigned.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example assigns access list 703 to Ethernet interface 3:

```
interface ethernet 3  
  bridge-group 5 output-address-list 703
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**access-list (extended)**  
**access-list (standard)**  
**bridge address**  
**bridge-group**  
**bridge-group input-address-list**

## bridge-group output-lat-service-deny

Use the **bridge-group output-lat-service-deny** interface configuration command to specify the group codes by which to deny access upon output. Use the **no** form of this command to cancel the specified group codes.

**bridge-group** *bridge-group* **output-lat-service-deny** *group-list*  
**no bridge-group** *bridge-group* **output-lat-service-deny** *group-list*

### Syntax Description

*bridge-group*                      Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.

*group-list*                        List of LAT groups. Single numbers and ranges are permitted.

### Default

No group codes are assigned.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Autonomous bridging must be disabled to use this command.

This command causes the system to not bridge onto this output interface any service advertisements that contain groups matching any of those in the group list.

### Example

The following example prevents bridging of LAT service announcements from groups 12 through 20:

```
interface ethernet 0
  bridge-group 1
  bridge-group 1 output-lat-service-deny 12-20
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**access-list (extended)**

**access-list (standard)**

**bridge-group**

**bridge-group input-lat-service-deny**

**bridge-group output-lat-service-permit**

## bridge-group output-lat-service-permit

Use the **bridge-group output-lat-service-permit** interface configuration command to specify the group codes by which to permit access upon output. Use the **no** form of this command to cancel specified group codes.

```
bridge-group bridge-group output-lat-service-permit group-list  
no bridge-group bridge-group output-lat-service-permit group-list
```

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>group-list</i>	LAT service advertisements.

### Default

No group codes are specified.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Autonomous bridging must be disabled to use this command.

This command causes the system to bridge onto this output interface only those service advertisements that match at least one group in the specified group code list.

---

**Note** If a message matches both a deny and a permit condition, it will not be bridged.

---

### Example

The following example allows only LAT service announcements from groups 5, 12, and 20 on this bridge:

```
interface ethernet 0  
  bridge-group 1 output-lat-service-permit 5 12 20
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge-group input-lat-service-permit**  
**bridge-group output-lat-service-deny**

## bridge-group output-lsap-list

Use the **bridge-group output-lsap-list** interface configuration command to filter IEEE 802-encapsulated packets on output. Use the **no** form of this command to disable this capability.

**bridge-group** *bridge-group* **output-lsap-list** *access-list-number*  
**no bridge-group** *bridge-group* **output-lsap-list** *access-list-number*

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>access-list-number</i>	Access list number you assigned with the standard <b>access-list</b> command. Specify a zero (0) to disable the application of the access list on the bridge group.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Autonomous bridging must be disabled to use this command.

SNAP frames must also pass any applicable Ethernet type-code access list. This access list is applied just before sending out a frame to an interface.

For performance reasons, specify both input and output type code filtering on the same interface.

Access lists for Ethernet- and IEEE 802-encapsulated packets affect only bridging functions. It is not possible to use such access lists to block frames with protocols that are being routed.

Packets bearing an 802.2 LSAP of 0xAAAA qualify for LSAP filtering since they are inherently in 802.3 format. However, because they also carry a Type field, they are matched against any Type filters. Therefore, if you use LSAP filters on an interface that may bear SNAP encapsulated packets you must explicitly permit 0xAAAA.

### Example

The following example specifies access list 204 on Ethernet interface 0:

```
interface ethernet 0
  bridge-group 4 output-lsap-list 204
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**access-list (extended)**

**access-list (standard)**

**bridge-group**

**bridge-group input-lsap-list**

## bridge-group output-pattern-list

Use the **bridge-group output-pattern-list** interface configuration command to associate an extended access list with a particular interface. Use the **no** form of this command to disable this capability.

**bridge-group** *bridge-group* **output-pattern-list** *access-list-number*  
**no bridge-group** *bridge-group* **output-pattern-list** *access-list-number*

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>access-list-number</i>	Extended access list number you assigned using the extended <b>access-list</b> command. Specify a zero (0) to disable the application of the access list on the interface.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Autonomous bridging must be disabled to use this command.

### Example

The following example filters all packets sent by bridge group 3 using the filter defined in access-list 1102:

```
interface ethernet 0
  bridge-group 3 output-pattern-list 1102
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**access-list (extended)**  
**bridge-group**  
**bridge-group input-pattern-list**

## bridge-group output-type-list

Use the **bridge-group output-type-list** interface configuration command to filter Ethernet- and SNAP-encapsulated packets on output. Use the **no** form of this command to disable this capability.

**bridge-group** *bridge-group* **output-type-list** *access-list-number*  
**no bridge-group** *bridge-group* **output-type-list** *access-list-number*

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>access-list-number</i>	Access list number you assigned with the standard <b>access-list</b> command. Specify a zero (0) to disable the application of the access list on the bridge group. This access list is applied just before sending out a frame to an interface.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Autonomous bridging must be disabled to use this command.

### Example

The following example specifies access-list 202 on Ethernet interface 0:

```
interface ethernet 0
  bridge-group 2 output-type-list 202
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**access-list (extended)**  
**access-list (standard)**  
**bridge-group**  
**bridge-group input-type-list**

## bridge-group path-cost

Use the **bridge-group path-cost** interface configuration command to set a different path cost. Use the **no** form of this command to choose the default path cost for the interface.

**bridge-group** *bridge-group* **path-cost** *cost*  
**no bridge-group** *bridge-group* **path-cost** *cost*

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>cost</i>	Path cost can range from 1 to 65535, with higher values indicating higher costs. This range applies regardless of whether the IEEE or Digital Spanning-Tree Protocol has been specified.

### Defaults

The default path cost is computed from the interface's bandwidth setting. The following are IEEE default path cost values. The Digital path cost default values are different.

Ethernet—100  
16-Mb Token Ring—62  
FDDI—10  
HSSI—647  
MCI/SCI Serial—647

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

By convention, the path cost is 10000/data rate of the attached LAN (IEEE), or 100000/data rate of the attached LAN (Digital), in megabits per second.

### Example

The following example changes the default path cost for Ethernet interface 0:

```
interface ethernet 0  
  bridge-group 1 path-cost 250
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge-group**

## bridge-group priority

Use the **bridge-group priority** interface configuration command to set an interface priority. The interface priority is used to select the designated port for this bridge-group on the connected media. One designated port on each media is needed to compute the spanning tree.

**bridge-group** *bridge-group* **priority** *number*

### Syntax Description

<i>bridge-group</i>	Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.
<i>number</i>	Priority number ranging from 0 to 255 (Digital), or 0 to 64000 (IEEE).

### Defaults

When the IEEE Spanning-Tree Protocol is enabled on the router: 32768

When the Digital Spanning-Tree Protocol is enabled on the router: 128

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The lower the number, the more likely it is that the bridge on the interface will be chosen as the root.

### Example

The following example increases the likelihood that the root bridge will be the one on Ethernet interface 0 in bridge group 1:

```
interface ethernet 0
  bridge-group 1 priority 0
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge-group**  
**bridge priority**

## bridge-group spanning-disabled

Use the **bridge-group spanning-disabled** interface configuration command to disable the spanning tree on a given interface.

**bridge-group** *bridge-group* **spanning-disabled**  
**no bridge-group** *bridge-group* **spanning-disabled**

### Syntax Description

*bridge-group*                      Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.

### Default

Spanning tree enabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

To enable transparent bridging on an interface, use the **bridge protocol** command to specify the type of Spanning-Tree Protocol to be used. The **bridge-group spanning-disabled** command can be used to disable that spanning tree on that interface.

When a *loop-free* path exists between any two bridged subnetworks, you can prevent BPDUs generated in one transparent bridging subnetwork from impacting nodes in the other transparent bridging subnetwork, yet still permit bridging throughout the bridged network as a whole.

For example, when transparently bridged LAN subnetworks are separated by a WAN, you can use this command to prevent BPDUs from traveling across the WAN link. You would apply this command to the serial interfaces connecting to the WAN in order to prevent BPDUs generated in one domain from impacting nodes in the remote domain. Because these BPDUs are prevented from traveling across the WAN link, using this command also has the secondary advantage of reducing traffic across the WAN link.

---

**Note** In order to disable the spanning tree, you must make sure that no parallel paths exist between transparently bridged interfaces in the network.

---

### Example

In the following example, the spanning tree for the serial interface 0 is disabled:

```
interface serial 0
  bridge-group 1 spanning-disabled
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge-group**

**bridge protocol**

## bridge-group sse

Use the **bridge-group sse** interface configuration command to enable Cisco's silicon switching engine (SSE) switching function. Use the **no** form of this command to disable SSE switching.

**bridge-group** *bridge-group sse*  
**no bridge-group** *bridge-group sse*

### Syntax Description

*bridge-group*                      Number of the bridge group to which the interface belongs. It must be a number in the range 1 to 63.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

### Example

The following example enables SSE switching:

```
bridge-group 1 sse
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**source-bridge**

## bridge hello-time

Use the **bridge hello-time** global configuration command to specify the interval between hello bridge protocol data units (BPDUs). Use the **no** form of this command to return the default interval.

```
bridge bridge-group hello-time seconds  
no bridge bridge-group hello-time
```

### Syntax Description

<i>bridge-group</i>	Bridge group number. It must be the same number specified in the <b>bridge protocol</b> command.
<i>seconds</i>	Interval between 1 and 10 seconds.

### Default

1 second

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Each bridge in a spanning tree adopts the **hello-time**, **forward-time**, and **max-age** parameters of the root bridge, regardless of what its individual configuration might be.

### Example

The following example sets the interval to 5 seconds:

```
bridge 1 hello-time 5
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

```
bridge forward-time  
bridge max-age  
bridge protocol
```

## bridge irb

Use the **bridge irb** global configuration command to enable the Cisco IOS software to route a given protocol between routed interfaces and bridge groups or to route a given protocol between bridge groups. Use the **no** form of this command to disable the feature.

**bridge irb**  
**no bridge irb**

### Syntax Description

This command has no arguments or keywords.

### Default

Integrated routing and bridging (IRB) is disabled.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

IRB is supported for transparent bridging, but not for source-route bridging. IRB is supported on all interface media types except X.25 and ISDN bridged interfaces.

### Example

The following example command enables integrated routing and bridging:

```
bridge irb
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge bridge**  
**bridge route**  
**interface bvi**  
**show interfaces irb**

## bridge lat-service-filtering

Use the **bridge lat-service-filtering** global configuration command to specify LAT group-code filtering. Use the **no** form of this command to disable the use of LAT service filtering on the bridge group.

```
bridge bridge-group lat-service-filtering  
no bridge bridge-group lat-service-filtering
```

### Syntax Description

*bridge-group* Bridge group number specified in the **bridge protocol** command.

### Default

LAT service filtering is disabled.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

This command informs the system that LAT service advertisements require special processing.

### Example

The following example specifies that LAT service announcements traveling across bridge group 1 require some special processing:

```
bridge 1 lat-service-filtering
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

### **bridge protocol**

## bridge max-age

Use the **bridge max-age** global configuration command to change the interval the bridge will wait to hear BPDUs from the root bridge. If a bridge does not hear BPDUs from the root bridge within this specified interval, it assumes that the network has changed and will recompute the spanning-tree topology. Use the **no** form of this command to return the default interval.

**bridge** *bridge-group* **max-age** *seconds*  
**no bridge** *bridge-group* **max-age**

### Syntax Description

<i>bridge-group</i>	Bridge group number specified in the <b>bridge protocol</b> command.
<i>seconds</i>	Interval the bridge will wait to hear BPDUs from the root bridge. It must be a value in the range 10 to 200 seconds.

### Default

15 seconds

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Each bridge in a spanning tree adopts the **hello-time**, **forward-time**, and **max-age** parameters of the root bridge, regardless of what its individual configuration might be.

### Example

The following example increases the maximum idle interval to 20 seconds:

```
bridge 1 max-age 20
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge forward-time**  
**bridge hello-time**  
**bridge protocol**

## bridge multicast-source

Use the **bridge multicast-source** global configuration command to configure bridging support to allow the forwarding, but not the learning, of frames received with multicast source addresses. Use the **no** form of this command to disable this function on the bridge.

**bridge** *bridge-group* **multicast-source**  
**no bridge** *bridge-group* **multicast-source**

### Syntax Description

*bridge-group* Bridge group number specified in the **bridge protocol** command.

### Default

Disabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

If you need to bridge Token Ring over other medium, RSRB is recommended.

### Example

The following example allows the forwarding, but not the learning, of frames received with multicast source addresses:

```
bridge 2 multicast-source
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge protocol**

## bridge priority

Use the **bridge priority** global configuration command to configure the priority of an individual bridge, or the likelihood that it will be selected as the root bridge.

**bridge** *bridge-group* **priority** *number*

### Syntax Description

<i>bridge-group</i>	Bridge group number specified in the <b>bridge protocol</b> command.
<i>number</i>	The lower the number, the more likely the bridge will be chosen as root. When the IEEE Spanning-Tree Protocol is enabled, <i>number</i> ranges from 0 to 65535 (default is 32768). When the Digital Spanning-Tree Protocol is enabled, <i>number</i> ranges from 0 to 255 (default is 128).

### Defaults

When the IEEE Spanning-Tree Protocol is enabled on the router: 32768

When the Digital Spanning-Tree Protocol is enabled on the router: 128

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

When two bridges tie for position as the root bridge, an interface priority determines which bridge will serve as the root bridge. Use the **bridge-group priority** interface configuration command to control an interface priority.

### Example

The following example establishes this bridge as a likely candidate to be the root bridge:

```
bridge 1 priority 100
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge-group priority**  
**bridge protocol**

---

## bridge protocol

Use the **bridge protocol** global configuration command to define the type of Spanning-Tree Protocol. Use the **no** form of this command, with the appropriate keywords and arguments, to delete the bridge group.

```
bridge bridge-group protocol { ieee | dec }  
no bridge bridge-group protocol { ieee | dec }
```

### Syntax Description

<i>bridge-group</i>	Number in the range 1 to 63 that you choose to refer to a particular set of bridged interfaces. Frames are bridged only among interfaces in the same group. You will use the group number you assign in subsequent bridge configuration commands.
<b>dec</b>	Digital Spanning-Tree Protocol.
<b>ieee</b>	IEEE Ethernet Spanning-Tree Protocol.

### Default

No Spanning-Tree Protocol is defined.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The routers support two Spanning-Tree Protocols: the IEEE 802.1 standard and the earlier Digital Spanning-Tree Protocol upon which the IEEE standard is based. Multiple domains are supported for the IEEE 802.1 Spanning-Tree Protocol.

---

**Note** The IEEE 802.1D Spanning-Tree Protocol is the preferred way of running the bridge. Use the Digital Spanning-Tree Protocol only for backward compatibility.

---

### Example

The following example shows bridge 1 as using the Digital Spanning-Tree Protocol:

```
bridge 1 protocol dec
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge domain**

**bridge-group**

## bridge route

Use the **bridge route** global configuration command to enable the routing of a specified protocol in a specified bridge group. Use the **no** form of this command to disable the routing of a specified protocol in a specified bridge group.

**bridge** *bridge-group* **route** *protocol*  
**no bridge** *bridge-group* **route** *protocol*

### Syntax Description

<i>bridge-group</i>	Bridge-group number. It must be the same number specified in the <b>bridge protocol</b> command.
<i>protocol</i>	One of the following protocols: <b>apollo</b> , <b>appletalk</b> , <b>clns</b> , <b>decnet</b> , <b>ip</b> , <b>ipx</b> , <b>vines</b> , <b>xns</b> .

### Default

No default bridge group or protocol is specified.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

### Example

In the following example, AppleTalk and IP are routed on bridge group 1:

```
bridge crb
bridge 1 protocol ieee
bridge 1 route appletalk
bridge 1 route ip
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge crb**  
**bridge protocol**

## clear bridge

Use the **clear bridge** privileged EXEC command to remove any learned entries from the forwarding database and to clear the transmit and receive counts for any statically or system configured entries.

**clear bridge** *bridge-group*

### Syntax Description

*bridge-group*            Bridge group number specified in the **bridge protocol** command.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example shows the use of the **clear bridge** command:

```
clear bridge 1
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge address**

**bridge protocol**

## clear bridge multicast

Use the **clear bridge multicast** EXEC command to clear transparent bridging multicast state information.

```
clear bridge [bridge-group] multicast [router-ports | groups | counts] [group-address]  
[interface-unit] [counts]
```

### Syntax Description

<i>bridge-group</i>	(Optional) Bridge group number specified in the <b>bridge protocol</b> command.
<b>router-ports</b>	(Optional) Clear multicast router ports.
<b>groups</b>	(Optional) Clear multicast groups.
<b>counts</b>	(Optional) Clear RX and TX counts.
<i>group-address</i>	(Optional) Multicast IP address associated with a specific multicast group.
<i>interface-unit</i>	(Optional) Specific interface, such as Ethernet 0.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

If you do not specify arguments or keywords as part of the command, the command clears router ports, group ports, and counts for all configured bridge groups.

Use the **show bridge multicast** command to list transparent bridging multicast state information, then use specific pieces of state information in the **clear bridge multicast** command.

### Examples

The following example command clears router ports, group ports, and counts for bridge group 1:

```
clear bridge 1 multicast
```

The following example command clears the group and count information for the group identified as 235.145.145.223, interface Ethernet 0/3 for bridge group 1:

```
clear bridge 1 multicast groups 235.145.145.223 Ethernet0/3 counts
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge cmf**  
**show bridge multicast**

## clear sse

Use the **clear sse** privileged EXEC command to reinitialize the Silicon Switch Processor (SSP) on the Cisco 7000 series routers with RSP7000.

**clear sse**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

### Example

The following example reinitializes the SSP:

```
clear sse
```

## clear vlan statistics

Use the **clear vlan statistics** privileged EXEC command to remove virtual LAN statistics from any statically or system configured entries.

**clear vlan statistics**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

### Example

The following example clears VLAN statistics:

```
clear vlan statistics
```

## encapsulation isl

Use the **encapsulation isl** subinterface configuration command to enable the Inter-Switch Link (ISL), a Cisco proprietary protocol for interconnecting multiple switches and maintaining VLAN information as traffic goes between switches.

**encapsulation isl** *domain*

### Syntax Description

*domain* VLAN domain number.

### Default

Disabled

### Command Mode

Subinterface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.1.

ISL encapsulation adds a 30-byte header to the beginning of the Ethernet frame. The header contains a 2-byte VLAN identifier that maintains VLAN identities between switches.

### Example

The following example enables ISL on FDDI subinterface 2/1.20:

```
interface FastEthernet 2/1.20.  
 ip address 171.69.2.2 255.255.255.0  
 encapsulation isl 400  
 bridge-group 50
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge-group**  
**show bridge vlan**  
**show interfaces**  
**show span**

## encapsulation sde

Use the **encapsulation sde** subinterface configuration command to enable IEEE 802.10 Secure Data Exchange (SDE) encapsulation of transparently bridged traffic on a specified interface within an assigned bridge group.

**encapsulation sde** *said*

### Syntax Description

*said* Security association identifier. The valid range is 0 through 0xFFF.

### Default

Disabled

### Command Mode

Subinterface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

SDE encapsulation is only applicable to transparently bridged traffic, and is configurable on the following interface types:

- Ethernet
- Nonencapsulated FDDI
- Token Ring (except MultiBus Token Ring)
- HDLC serial

---

**Note** The current implementation of SDE encapsulation is not recommended for serial or Ethernet media.

---

### Example

The following example enables SDE on FDDI subinterface 2/0.1 and assigns a security association identifier of 9999:

```
interface fddi 2/0.1
  encapsulation sde 9999
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge-group**  
**show bridge vlan**  
**show interfaces**  
**show span**

## ethernet-transit-oui

Use the **ethernet-transit-oui** interface configuration command to choose the Organizational Unique Identifier (OUI) code to be used in the encapsulation of Ethernet Type II frames across Token Ring backbone networks. Various versions of this OUI code are used by Ethernet/Token Ring translational bridges. The default OUI form is **90-compatible**, which can be chosen with the **no** form of this command.

```
ethernet-transit-oui [90-compatible | standard | cisco]
no ethernet-transit-oui
```

### Syntax Description

<b>90-compatible</b>	(Optional) Default OUI form.
<b>standard</b>	(Optional) Standard OUI form.
<b>cisco</b>	(Optional) Cisco's OUI form.

### Default

**90-compatible**

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

This command replaces and extends the **bridge old-oui** command in Software Release 9.0.

The actual OUI codes that are used, when they are used, and how they compare to Software Release 9.0-equivalent commands is shown in Table 1.

**Table 1 Bridge OUI Codes**

<b>Keyword</b>	<b>OUI Used</b>	<b>When Used/Benefits</b>	<b>9.0 Command Equivalent</b>
<b>90-compatible</b>	0000F8	By default, when talking to other Cisco routers. Provides the most flexibility.	<b>no bridge old-oui</b>
<b>cisco</b>	00000C	Provided for compatibility with future equipment.	None
<b>standard</b>	000000	When talking to IBM 8209 bridges and other vendor equipment. Does not provide for as much flexibility as the other two choices.	<b>bridge old-oui</b>

Do not use the keyword **standard** unless you are forced to interoperate with other vendor equipment, such as the IBM 8209, in providing Ethernet and Token Ring mixed media bridged connectivity. The use of the **standard** OUI of 000000 in the encapsulation of Ethernet Type II frames creates encapsulated frames on Token Rings that have formats identical to SNAP-encapsulated frames. The router receiving such a frame on a Token Ring for delivery on the Ethernet cannot distinguish between the two, and therefore must make an arbitrary choice between presenting the frame on the Ethernet as a SNAP-encapsulated frame or as an Ethernet Type II frame. The choice has been made to present all such frames as Ethernet Type II. Therefore, it is impossible to use the **standard** keyword if you wish to bridge SNAP-encapsulated frames between Token Rings and Ethernets. Using either the **cisco** or **90-compatible** keywords does not present such a restriction, because SNAP frames and Ethernet Type II-encapsulated frames have different OUI codes on Token Ring networks.

---

**Note** Prior to IOS Software Release 11.1, the default OUI for SNAP on Token Ring for high-end routers was hexadecimal 000F8, and for low-end routers was hexadecimal 00000. If you combine IOS Software Release 11.1 or later with any earlier high-end release on adjacent routers that are bridging over a token ring, you must use the **ethernet-transit-oui standard** command to bring the two software versions into agreement.

---

### Example

The following example specifies Cisco's OUI form:

```
interface tokenring 0
  ethernet-transit-oui cisco
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**bridge-group**

**bridge protocol**

## frame-relay map bridge broadcast

Use the **frame-relay map bridge broadcast** interface configuration command to bridge over a Frame Relay network. Use the **no** form of this command to delete the mapping entry.

**frame-relay map bridge *dldi* broadcast**  
**no frame-relay map bridge *dldi***

### Syntax Description

*dldi*    DLCI number. The valid range is 16 to 1007.

### Default

No mapping entry is established.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Bridging over a Frame Relay network is supported both on networks that support a multicast facility and those that do not.

### Example

The following example allows bridging over a Frame Relay network:

```
frame-relay map bridge 144 broadcast
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**encapsulation frame-relay**

## interface bvi

Use the **interface bvi** interface configuration command to create the bridge-group virtual interface (BVI) that represents the specified bridge group to the routed world and links the corresponding bridge group to the other routed interfaces. Use the **no** form of this command to delete the BVI.

```
interface bvi bridge-group  
no interface bvi bridge-group
```

### Syntax Description

*bridge-group*                      Bridge-group number. It must be the same number specified in the **bridge protocol** command.

### Default

No BVI is created.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

You must enable IRB before attempting to create a BVI.

When you intend to bridge and route a given protocol in the same bridge group, you must configure the network-layer attributes of the protocol on the BVI. Do not configure protocol attributes on the bridged interfaces. No bridging attributes can be configured on the BVI.

### Example

The following example creates a bridge-group virtual interface and associates it with bridge group 1:

```
interface bvi 1
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

### **bridge irb**

## ip routing

Use the **ip routing** command to enable IP routing. Use the **no** form of this command to disable IP routing so that you can then bridge IP.

**ip routing**  
**no ip routing**

### Syntax Description

This command has no arguments or keywords.

### Default

IP routing is enabled.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

All protocols except IP are bridged by a router unless their routing is explicitly enabled. Refer to the “IP Commands” chapter of the *Network Protocols Command Reference, Part 1* for the procedures to enable routing of individual protocols. IP is normally routed by the router.

Also note that bridging and routing are done on a per-system basis. If a protocol is being routed, it must be routed on all interfaces that are handling that protocol. This is similar for bridging. You cannot route IP on one interface and bridge it on another interface.

Assign the *same* IP address to all network interfaces to manage the system with Telnet, TFTP, SNMP, ICMP (ping), and so forth. Once bridging is enabled, all IP and ARP frames are forwarded or flooded by the router according to standard bridging and spanning-tree rules. IP routing processes such as IGRP or RIP must not be running.

### Example

The following example disables IP routing:

```
no ip routing
```

## show bridge

Use the **show bridge** privileged EXEC command to view classes of entries in the bridge forwarding database.

```
show bridge [bridge-group] [interface] [address [mask]] [verbose]
```

### Syntax Description

<i>bridge-group</i>	(Optional) Number that specifies a particular spanning tree.
<i>interface</i>	(Optional) Specific interface, such as Ethernet 0.
<i>address</i>	(Optional) 48-bit canonical (Ethernet ordered) MAC address. This may be entered with an optional mask of bits to be ignored in the address, which is specified with the <i>mask</i> argument.
<i>mask</i>	(Optional) Bits to be ignored in the address. You must specify the <i>address</i> argument if you want to specify a mask.
<b>verbose</b>	(Optional) Shows additional detail, including any Frame Relay DLCI associated with a station address.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0. The **verbose** keyword first appeared in Cisco IOS Release 11.0.

The following are possible variations of the **show bridge** command:

```
show bridge ethernet 0
show bridge 0000.0c00.0000 0000.00FF.FFFF
show bridge 0000.0c00.0e1a
show bridge
show bridge verbose
```

In the sample output, the first command would display all entries for hosts reachable via Ethernet interface 0, the second command would display all entries with the vendor code of 0000.0c00.0000, and the third command would display the entry for address 0000.0c00.0e1a. In the fourth command, all entries in the forwarding database would be displayed. The fifth command provides additional detail. In all five lines, the bridge-group number has been omitted.

### Sample Displays

The following is sample output of the **show bridge** command. The second display is output from the **show bridge** command with the **verbose** argument.

```
Router# show bridge

Total of 300 station blocks, 280 free
Codes: P - permanent, S - self

Bridge Group 32:Bridge Group 32:
```

Address	Action	Interface	Age	RX count	TX count
0180.c200.0000	receive	-	S	0	0
ffff.ffff.ffff	receive	-	S	0	0
0900.2b01.0001	receive	-	S	0	0
0300.0c00.0001	receive	-	S	0	0
0000.0c05.1000	forward	Ethernet0/1	4	1	0
0000.0c04.4b5b	receive	-	S	0	0
0000.0c04.4b5e	receive	-	S	0	0
0000.0c04.4b5d	receive	-	S	0	0
0000.0c04.4b5c	receive	-	S	0	0
0000.0c05.4a62	forward	Ethernet0/1	4	1	0
aa00.0400.2108	forward	Ethernet0/1	0	42	0
0000.0c12.b888	forward	Ethernet0/2	4	1	0
0000.0c12.b886	forward	Ethernet0/1	4	1	0
aa00.0400.4d09	forward	Ethernet0/1	4	1	0
0000.0c06.fb9a	forward	Ethernet0/1	4	1	0
0000.0c04.b039	forward	Ethernet0/1	4	1	0

```
router# show bridge verbose
```

Total of 300 station blocks, 287 free

Codes: P - permanent, S - self

BG Hash	Address	Action	Interface	DLCI	Age	RX count	TX count
32 00/0	0180.c200.0000	receive	-	-	S	0	0
32 00/1	ffff.ffff.ffff	receive	-	-	S	0	0
32 01/0	0900.2b01.0001	receive	-	-	S	0	0
32 01/1	0300.0c00.0001	receive	-	-	S	0	0
32 10/0	0000.0c04.4b5b	receive	-	-	S	0	0
32 15/0	0000.0c04.4b5e	receive	-	-	S	0	0
32 16/0	0000.0c04.4b5d	receive	-	-	S	0	0
32 17/0	0000.0c04.4b5c	receive	-	-	S	0	0
32 29/0	aa00.0400.2108	forward	Ethernet0/1	-	0	48	0
32 30/0	0000.0c12.b888	forward	Ethernet0/2	-	0	1	0
32 A4/0	0800.2002.ff5b	forward	Ethernet0/1	-	0	6	0
32 E2/0	aa00.0400.e90b	forward	Ethernet0/1	-	0	65	0
32 F2/0	0000.0c04.b042	forward	Ethernet0/2	-	3	2	0

Table 2 describes significant fields shown in the display.

**Table 2 Show Bridge Field Descriptions**

Field	Description
Total of 300 station blocks	Total number of forwarding database elements in the system. The memory to hold bridge entries is allocated in blocks of memory sufficient to hold 300 individual entries. When the number of free entries falls below 25, another block of memory sufficient to hold another 300 entries is allocated. Therefore, the size of the bridge forwarding database is limited to the amount of free memory in the router.
295 free	Number in the free list of forwarding database elements in the system. The total number of forwarding elements is expanded dynamically, as needed.
BG	Bridging group to which the address belongs.
Hash	Hash key/relative position in the keyed list.
Address	Canonical (Ethernet ordered) MAC address.

**Table 2 Show Bridge Field Descriptions (Continued)**

<b>Field</b>	<b>Description</b>
Action	Action to be taken when that address is looked up; choices are to discard or forward the datagram.
Interface	Interface, if any, on which that address was seen.
Age	Number of minutes since a frame was received from or sent to that address. The letter "P" indicates a permanent entry. The letter "S" indicates the system as recorded by the router. On the modular systems, this is typically the broadcast address and the router's own hardware address; on the IGS, this field will also include certain multicast addresses.
RX count	Number of frames received from that address.
TX count	Number of frames forwarded to that address.

## show bridge circuit-group

Use the **show bridge circuit-group** EXEC command to display the interfaces configured in each circuit group and show whether they are currently participating in load distribution.

```
show bridge [bridge-group] circuit-group [circuit-group] [src-mac-address]
                [dst-mac-address]
```

### Syntax Description

<i>bridge-group</i>	(Optional) Number that specifies a particular bridge group.
<i>circuit-group</i>	(Optional) Number that specifies a particular circuit group.
<i>src-mac-address</i>	(Optional) 48-bit canonical (Ethernet ordered) source MAC address.
<i>dst-mac-address</i>	(Optional) 48-bit canonical (Ethernet ordered) destination MAC address.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

### Sample Display

The following is sample output of various **show bridge circuit-group** command strings:

```
RouterA> show bridge circuit-group

Bridge group 1 Circuit group 1:
Interface Serial0 : inserted, learning, forwarding
Interface Serial3 : inserted, learning, forwarding
Bridge group 1 Circuit group 2:
Interface Serial2 : inserted, learning, forwarding

RouterA> show bridge 1 circuit-group 1

Bridge group 1 Circuit group 1:
Interface Serial0 : inserted, learning, forwarding
Interface Serial3 : inserted, learning, forwarding

RouterA> show bridge 1 circuit-group 2

Bridge group 1 Circuit group 2:
Interface Serial2 : inserted, learning, forwarding

RouterA> show bridge 1 circuit-group 1 0000.6502.23EA 0000.1234.4567

Output circuit group interface is Serial3

RouterA> show bridge 1 circuit-group 1 0000.6502.23EA

%Destination MAC address required
```

## show bridge circuit-group

---

```
RouterB> show bridge 1 circuit-group 1

Bridge group 1 Circuit group 1:
Transmission pause interval is 250ms
Output interface selection is source-based
Interface Serial0 : inserted, learning, forwarding
Interface Serial3 : inserted, learning, forwarding
Interface Serial2 is unavailable

RouterB> show bridge 1 circuit-group 1 0000.6502.23EA 0000.1234.4567

%Please enter source MAC address only
```

Table 3 describes significant fields shown in the display.

**Table 3 Show Bridge Circuit-Group Field Descriptions**

Field	Description
inserted/not inserted	Indicates whether interface is included or not included in circuit-group operation. If the interface is administratively down, or if line protocol is not up, the interface is not included in the circuit-group operation.
learning/not learning	Indicates whether this interface is in Spanning-Tree Protocol (IEEE or Digital) learning or not learning state.
forwarding/not forwarding	Indicates whether this port is in Spanning-Tree Protocol (IEEE or Digital) forwarding or not forwarding state.

## show bridge group

Use the **show bridge group** privileged EXEC command to display the status of each bridge group.

```
show bridge group [verbose]
```

### Syntax Description

**verbose** (Optional) Displays detailed information.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

### Sample Display

```
Router# show bridge group

Bridge Group 1 is running the DEC compatible Spanning Tree protocol

Port 7 (ATM0.1 LANE Ethernet) of bridge group 1 is down
Port 4 (TokenRing0) of bridge group 1 is forwarding
```

“Forwarding” and “down” indicate the port state as determined by the spanning-tree algorithm or via configuration.

## show bridge multicast

Use the **show bridge multicast** EXEC command to display transparent bridging multicast state information.

**show bridge** [*bridge-group*] **multicast** [**router-ports** | **groups**] [*group-address*]

### Syntax Description

<i>bridge-group</i>	(Optional) Bridge group number specified in the <b>bridge protocol</b> command.
<b>router-ports</b>	(Optional) Display information for multicast router ports.
<b>groups</b>	(Optional) Display information for multicast groups.
<i>group-address</i>	(Optional) Multicast IP address associated with a specific multicast group.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

### Sample Display

The following is sample output for the **show bridge multicast** command:

```
Router# show bridge multicast

Multicast router ports for bridge group 1:

 2 multicast router ports
Fddi2/0      R
Ethernet0/4  R

Multicast groups for bridge group 1:

235.145.145.223      RX count  TX count
Fddi2/0      R          0          2
Ethernet0/4  R          0          3
Ethernet0/3  G          1          0

235.5.5.5           RX count  TX count
Fddi2/0      R          0          2
Ethernet0/4  R          0          3
Ethernet0/3  G          1          0

235.4.4.4           RX count  TX count
Fddi2/0      R          0          2
Ethernet0/4  R          0          3
Ethernet0/3  G          1          0

Router#
```

---

Table 4 describes significant fields shown in the display.

**Table 4 Show Bridge Multicast Field Descriptions**

<b>Field</b>	<b>Description</b>
Multicast router ports for...	List of the multicast router ports by bridge group. Within the bridge group cluster, the display lists the number of multicast router ports and then lists the ports by interface.
Multicast groups for...	List of the multicast groups by bridge group. Within each multicast group, identified by a unique address, the display lists each port by interface name and indicates whether that port is a group member ("G"), a multicast router port ("R"), or both. The RX and TX counts show the number of multicast packets that have been constrained to the multicast group by the bridge.

## show bridge vlan

Use the **show bridge vlan** privileged EXEC command to view virtual LAN subinterfaces.

### show bridge vlan

#### Syntax Description

This command has no arguments or keywords.

#### Command Mode

Privileged EXEC

#### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

#### Sample Display

The following is sample output from the **show bridge vlan** command:

```
Router# show bridge vlan

Bridge Group: 50

Virtual LAN Trunking Interface(s):  vLAN Protocol:      vLAN ID:  State
Fddi2/0.1000                       IEEE 802.10      1000      forwarding
FastEthernet4/0.500                 Inter Switch Link 500        listening

Virtual LAN Native Interface(s):   State
Ethernet0/1                         forwarding
Serial1/1                           down
```

Table 5 describes the fields shown in the display.

**Table 5 Show Bridge VLAN Field Description**

Field	Description
Bridge Group	Bridge group to which these interfaces belong.
Virtual LAN Trunking Interface(s)	VLAN interface.
vLAN Protocol)	IEEE 802.10 or Cisco ISL encapsulation.
vLAN ID	VLAN identifier that maintains VLAN identities between switches.
State	Spanning-tree port state of the interface.
Virtual LAN Native Interface(s):	Interfaces whose transparently bridged traffic will be propagated only to other LAN segments within the same virtual LAN.

## show interfaces crb

Use the **show interfaces crb** privileged EXEC command to display the configuration for each interface that has been configured for routing or bridging.

### show interfaces crb

### Syntax Description

This command has no arguments or keywords.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.0.

### Sample Display

The following is sample output for the **show interfaces crb** command:

```
Router# show interfaces crb

Ethernet0/0

Routed protocols on Ethernet0/0:
appletalk decnet ip novell

Ethernet0/1

Routed protocols on Ethernet0/1:
appletalk decnet ip novell

Ethernet0/2

Routed protocols on Ethernet0/2:
appletalk ip

Bridged protocols on Ethernet0/2:
clns decnet vines apollo
novell xns

Software MAC address filter on Ethernet0/2
Hash Len  Address          Matches  Act  Type
0x00: 0   ffff.ffff.ffff  0       RCV  Physical broadcast
0x00: 1   ffff.ffff.ffff  0       RCV  Appletalk zone
0x2A: 0   0900.2b01.0001  0       RCV  DEC spanning tree
0x49: 0   0000.0c36.7a45  0       RCV  Interface MAC address
0xc0: 0   0100.0ccc.cccc  20      RCV  CDP
0xc2: 0   0180.c200.0000  0       RCV  IEEE spanning tree
0xF8: 0   0900.07ff.ffff  0       RCV  Appletalk broadcast

Ethernet0/3

Routed protocols on Ethernet0/3:
appletalk ip
```

## show interfaces crb

---

```
Bridged protocols on Ethernet0/3:
clns decnet vines apollo
novell xns

Software MAC address filter on Ethernet0/3
Hash Len  Address           Matches  Act  Type
0x00: 0   ffff.ffff.ffff    0        RCV  Physical broadcast
0x00: 1   ffff.ffff.ffff    0        RCV  Appletalk zone
0x2A: 0   0900.2b01.0001    0        RCV  DEC spanning tree
0x49: 0   0000.0c36.7a45    0        RCV  Interface MAC address
0xc0: 0   0100.0ccc.cccc    48       RCV  CDP
0xc2: 0   0180.c200.0000    0        RCV  IEEE spanning tree
0xF8: 0   0900.07ff.ffff    0        RCV  Appletalk broadcast

Router#
```

Table 6 describes significant fields shown in the display.

**Table 6 Show Interfaces CRB Field Descriptions**

Field	Description
Routed protocols on...	List of the routed protocols configured for the specified interface.
Bridged protocols on...	List of the bridged protocols configured for the specified interface.
Software MAC address filter on...	Table of software MAC address filter information for the specified interface.
Hash	Hash key/relative position in the keyed list for this MAC-address entry.
Len	Length of this entry to the beginning element of this hash chain.
Address	Canonical (Ethernet ordered) MAC address.
Matches	Number of received packets matched to this MAC address.
Act	Action to be taken when that address is looked up; choices are to receive or discard the packet.
Type	MAC address type.

## show interfaces irb

Use the **show interfaces irb** privileged EXEC command to display the configuration for each interface that has been configured for integrated routing or bridging.

**show interfaces** [*interface*] **irb**

### Syntax Description

*interface* (Optional) Specific interface, such as Ethernet 0.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

### Sample Display

The following is sample output for the **show interfaces irb** command:

```
Router# show interfaces ethernet 2 irb

Ethernet 2

Routed protocols on Ethernet 2:
appletalk ip

Bridged protocols on Ethernet 2:
appletalk  clns  decnet  vines
apollo     ipx   xns

Software MAC address filter on Ethernet 2
Hash Len  Address           Matches  Act   Type
0x00: 0   ffff.ffff.ffff    4886    RCV   Physical broadcast
0x1F: 0   0060.3e2b.a221    7521    RCV   Appletalk zone
0x1F: 1   0060.3e2b.a221    0       RCV   Bridge-group Virtual Interface
0x2A: 0   0900.2b01.0001    0       RCV   DEC spanning tree
0x05: 0   0900.0700.00a2    0       RCV   Appletalk zone
0xC2: 0   0180.c200.0000    0       RCV   IEEE spanning tree
0xF8: 0   0900.07ff.ffff    2110    RCV   Appletalk broadcast
```

Table 7 describes significant fields shown in the display.

**Table 7 Show Interfaces IRB Field Descriptions**

<b>Field</b>	<b>Description</b>
Routed protocols on...	List of the routed protocols configured for the specified interface.
Bridged protocols on...	List of the bridged protocols configured for the specified interface.
Software MAC address filter on...	Table of software MAC address filter information for the specified interface.
Hash	Hash key/relative position in the keyed list for this MAC-address entry.
Len	Length of this entry to the beginning element of this hash chain.
Address	Canonical (Ethernet ordered) MAC address.
Matches	Number of received packets matched to this MAC address.
Act	Action to be taken when that address is looked up; choices are to receive or discard the packet.
Type	MAC address type.

---

## show span

Use the **show span** privileged EXEC command to display the spanning-tree topology known to the router. The display indicates whether LAT group code filtering is in effect.

**show span**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Sample Display

The following is sample output for the **show span** command:

```
RouterA# show span

Bridge Group 1 is executing the DEC compatible Spanning Tree protocol
  Bridge Identifier has priority 128, address 0000.304c.f686
  Configured hello time 1, max age 15, forward delay 30
  We are the root of the spanning tree
  Topology change flag not set, detected flag not set
  Times: hold 1, topology change 30, notification 30
         hello 1, max age 15, forward delay 30, aging 300
  Timers: hello 1, topology change 0, notification 0

Port 7 (ATM0.1 LANE Ethernet) of bridge group 1 is down
  Port path cost 0, Port priority 128
  Designated root has priority 128, address 0000.304c.f686
  Designated bridge has priority 128, address 0000.304c.f686
  Designated port is 7, path cost 0
  Timers: message age 0, forward delay 0, hold 0
```

## show sse summary

Use the **show sse summary** EXEC command to display a summary of Silicon Switch Processor (SSP) statistics:

**show sse summary**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

### Sample Display

The following is sample output from the **show sse summary** command:

```
Router# show sse summary

SSE utilization statistics

      Program words  Rewrite bytes  Internal nodes  Depth
Overhead           499             1             8
IP                  0             0             0    0
IPX                 0             0             0    0
SRB                 0             0             0    0
CLNP                0             0             0    0
IP access lists     0             0             0
Total used          499             1             8
Total free          65037          262143
Total available     65536          262144

Free program memory
[499..65535]
Free rewrite memory
[1..262143]

Internals
75032 internal nodes allocated, 75024 freed
SSE manager process enabled, microcode enabled, 0 hangs
Longest cache computation 4ms, longest quantum 160ms at 0x53AC8
```

## show vlans

Use the **show vlans** privileged EXEC command to view virtual LAN subinterfaces.

### show vlans

### Syntax Description

This command has no arguments or keywords.

### Command Mode

Privileged EXEC

### Usage Guide

This command first appeared in Cisco IOS Release 11.0.

### Sample Display

The following is sample output from the **show vlans** command:

```
RouterC7xxx# show vlans

Virtual LAN ID:300 (IEEE 802.10 Encapsulation)

    vLAN Trunk Interface: FDDI 1/1.10

    Protocols Configured:  Address:          Received:      Transmitted:
    IP 31.108.1.1 642645

Virtual LAN ID:400 (ISL Encapsulation)

    vLAN Trunk Interface: FastEthernet 2/1.20

    Protocols Configured:  Address:          Received:      Transmitted:
    IP 171.69.2.2 123456654321
    Bridge Group50 51908234

Virtual LAN ID:500 (ISL Encapsulation)

    vLAN Trunk Interface: FastEthernet 2/1.30

    Protocols Configured:  Address:          Received:      Transmitted:
    IPX 1000 987654456789

Virtual LAN ID:600 (ISL Encapsulation)

    vLAN Trunk Interface: FastEthernet 2/1.30

    Protocols Configured:  Address:          Received:      Transmitted:
    IP198.92.3.381144508
    IPX10012 3
    Bridge Group5082345190
```

Table 8 describes the fields shown in the display.

**Table 8 Show VLAN Field Description**

<b>Field</b>	<b>Description</b>
Virtual LAN ID	The domain number of the virtual LAN.
vLAN Trunk Interface	The subinterface that carries the VLAN traffic.
Protocols Configured	The protocols configured on the VLAN.
Address	The network address.
Received	Packets received.
Transmitted	Packets transmitted.

## x25 map bridge

Use the **x25 map bridge** interface configuration command to configure the bridging of packets in X.25 frames. Use the **no** form of this command to disable the Internet-to-X.121 mapping.

```
x25 map bridge x.121-address broadcast [options-keywords]  
no x25 map bridge
```

### Syntax Description

<i>x.121-address</i>	The X.121 address.
<b>broadcast</b>	Required keyword for bridging over X.25.
<i>options-keywords</i>	(Optional) Additional functionality that can be specified for originated calls. Can be any of the options listed in Table 9.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The X.25 bridging software uses the same spanning-tree algorithm as the other bridging functions, but allows packets to be encapsulated in X.25 frames and transmitted across X.25 media. This command specifies IP-to-X.121 address mapping and maintains a table of both the Ethernet and X.121 addresses.

The X.25 bridging implementation supports the map options listed in Table 9.

Table 9 X.25 Map Options

Option	Description
<b>compress</b>	Specifies that X.25 payload compression be used for mapping the traffic to this host. Each virtual circuit established for compressed traffic uses a significant amount of memory (for a table of learned data patterns) and for computation (for compression and decompression of all data). Cisco recommends that compression be used with careful consideration to its impact on overall performance.
<b>method { cisco   ietf   snap   multi }</b>	Specifies the encapsulation method. The choices are as follows: <ul style="list-style-type: none"> <li>• <b>cisco</b>—Cisco’s proprietary encapsulation; not available if more than one protocol is to be carried.</li> <li>• <b>ietf</b>—Default RFC 1356 operation: protocol identification of single-protocol virtual circuits and protocol identification within multiprotocol virtual circuits uses the standard encoding, which is compatible with RFC 877. Multiprotocol virtual circuits are used only if needed.</li> <li>• <b>snap</b>—RFC 1356 operation where IP is identified with SNAP rather than the standard IETF method (the standard method is compatible with RFC 877).</li> <li>• <b>multi</b>—Forces a map that specifies a single protocol to set up a multiprotocol virtual circuit when a call is originated; also forces a single-protocol PVC to use multiprotocol data identification methods for all datagrams sent and received.</li> </ul>
<b>no-incoming</b>	Use the map only to originate calls.
<b>no-outgoing</b>	Do not originate calls when using the map.
<b>idle minutes</b>	Specifies an idle timeout for calls other than the interface default; 0 minutes disables the idle timeout.
<b>reverse</b>	Specifies reverse charging for outgoing calls.
<b>accept-reverse</b>	Causes the Cisco IOS software to accept incoming reverse-charged calls. If this option is not present, the Cisco IOS software clears reverse-charged calls unless the interface accepts all reverse-charged calls.
<b>broadcast</b>	Causes the Cisco IOS software to direct any broadcasts sent through this interface to the specified X.121 address. This option also simplifies the configuration of OSPF; see “Usage Guidelines” for more detail.
<b>cug group-number</b>	Specifies a closed user group number (from 1 to 99) for the mapping in an outgoing call.
<b>nvc count</b>	Sets the maximum number of virtual circuits for this map or host. The default <i>count</i> is the <b>x25 nvc</b> setting of the interface. A maximum number of eight virtual circuits can be configured for each map. Compressed TCP may use only 1 virtual circuit.
<b>packetsize in-size out-size</b>	Proposes maximum input packet size ( <i>in-size</i> ) and maximum output packet size ( <i>out-size</i> ) for an outgoing call. Both values typically are the same and must be one of the following values: 16, 32, 64, 128, 256, 512, 1024, 2048, or 4096.
<b>window-size in-size out-size</b>	Proposes the packet count for input window ( <i>in-size</i> ) and output window ( <i>out-size</i> ) for an outgoing call. Both values typically are the same, must be in the range 1 to 127, and must be less than the value set by the <b>x25 modulo</b> command.

**Table 9 X.25 Map Options (Continued)**

<b>Option</b>	<b>Description</b>
<b>throughput</b> <i>in out</i>	Sets the requested throughput class values for input ( <i>in</i> ) and output ( <i>out</i> ) throughput across the network for an outgoing call. Values for <i>in</i> and <i>out</i> are in bits per second (bps) and range from 75 to 48000 bps.
<b>transit-delay</b> <i>milliseconds</i>	Specifies the transit delay value in milliseconds (0 to 65534) for an outgoing call, for networks that support transit delay.
<b>nuid</b> <i>username password</i>	Specifies that a network user ID (NUID) facility be sent in the outgoing call with the specified Terminal Access Controller Access Control System (TACACS) username and password (in a format defined by Cisco). This option should be used only when connecting to another Cisco router. The combined length of the username and password should not exceed 127 characters.
<b>nudata</b> <i>string</i>	Specifies the network user identification in a format determined by the network administrator (as allowed by the standards). This option is provided for connecting to non-Cisco equipment that requires an NUID facility. The string should not exceed 130 characters and must be enclosed in quotation marks (“ ”) if there are any spaces present.
<b>rpoa</b> <i>name</i>	Specifies the name defined by the <b>x25 roa</b> command for a list of transit Recognized Operating Agencies (ROAs) to use in outgoing Call Request packets.
<b>passive</b>	Specifies that the X.25 interface should send compressed outgoing TCP datagrams only if they were already compressed when they were received. This option is available only for compressed TCP maps.

### Example

The following example allows bridging over an X.25 network:

```
x25 map bridge 31370054065 broadcast
```

### Related Commands

You can use the master indexes or search online to find documentation of related commands.

**x25 address**

**x25 map**

