# Bridging Commands

# access-expression

To define an access expression, use the **access-expression** command in interface configuration mode. To remove the access expression from the given interface, use the **no** form of this command.

> **access-expression** {**in** | **out**} *expression*

> **no access-expression** {**in** | **out**} *expression*

| Syntax Description | **in** | **out** | Either **in** or **out** is specified to indicate whether the access expression is applied to packets entering or leaving this interface. You can specify both an input and an output access expression for an interface, but only one of each. |
| :--- | :--- | :--- |
| | *expression* | Boolean access list expression, built as explained in the "Usage Guidelines" section. |

**Defaults**　　No access expression is defined.

**Command Modes**　　Interface configuration

**Command History**

| Release | Modification |
| :--- | :--- |
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**　　Use this command in conjunction with the **access-list** command in interface configuration mode.

An access expression consists of a list of terms, separated by Boolean operators, and optionally grouped in parentheses.

An access expression term specifies a type of access list, followed by its name or number. The result of the term is either true or false, depending on whether the access list specified in the term permits or denies the frame. Table 8 describes the terms that can be used.

*Table 8*　　　*Access Expression Terms*

| Access Expression Term | Definition |
| :--- | :--- |
| lsap(2nn) | Subnetwork Access Protocol access list to be evaluated for this frame (Cisco 200 series). |
| type(2nn) | Subnetwork Access Protocol (SNAP) type access list to be evaluated for this frame (Cisco 200 series). |
| smac(7nn) | Access list to match the source MAC address of the frame (Cisco 700 series). |

*Table 8        Access Expression Terms (continued)*

| Access Expression Term | Definition |
|---|---|
| dmac(7nn) | Access list to match the destination MAC address of the frame (Cisco 700 series). |
| netbios-host(name) | NetBIOS-host access list to be applied on NetBIOS frames traversing the interface. |
| netbios-bytes(name) | NetBIOS-bytes access list to be applied on NetBIOS frames traversing the interface. |

Access expression terms are separated by Boolean operators as listed in Table 9.

*Table 9        Boolean Operators for Access Expression Terms*

| Boolean Operators | Definitions |
|---|---|
| ~ (called "not") | Negates, or reverses, the result of the term or group of terms immediately to the right of the ~.<br><br>Example: "~lsap (201)" returns FALSE if "lsap (201)" itself were TRUE. |
| & (called "and") | Returns TRUE if the terms or parenthetical expressions to the left and right of the & both return TRUE.<br><br>Example: "lsap (201) & dmac (701)" returns TRUE if both the lsap (201) and dmac (701) terms return TRUE. |
| \| (called "or") | Returns TRUE if the terms or parenthetical expressions either to the left or to the right of the \| or both return TRUE.<br><br>Example: "lsap (201) \| dmac (701)" returns TRUE if either the lsap (201) or dmac (701) terms return TRUE, or if both return TRUE. |

Terms can be grouped in parenthetical expressions. Any of the terms and operators can be placed in parentheses, similar to what is done in arithmetic expressions, to affect order of evaluation.

An "access-expression" type filter cannot exist with a "source-bridge" type filter on the same interface. The two types of filters are mutually exclusive.

**Note**     The incorrect use of parentheses can drastically affect the result of an operation because the expression is read from left to right.

**Related Commands**

| Command | Description |
|---|---|
| **access-list** | Configures the access list mechanism for filtering frames by protocol type or vendor code. |

# access-list

To configure the access list mechanism for filtering frames by protocol type or vendor code, use the **access-list** command in global configuration mode. To remove the single specified entry from the access list, use the **no** form of this command.

**access-list** *access-list-number* {**permit** | **deny**} {*type-code wild-mask* | *address mask*}

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

**Syntax Description**

| | |
|---|---|
| *access-list-number* | Integer that identifies the access list. If the *type-code* and *wild-mask* arguments are included, this integer ranges from 200 to 299, indicating that filtering is by protocol type. If the *address* and *mask* arguments are included, this integer ranges from 700 to 799, indicating that filtering is by vendor code. |
| **permit** | Permits the frame. |
| **deny** | Denies the frame. |
| *type-code* | 16-bit hexadecimal number written with a leading 0x; for example, 0x6000. Specify either a Link Service Access Point (LSAP) type code for 802-encapsulated packets or a Subnetwork Access Protocol (SNAP) type code for SNAP-encapsulated packets. (LSAP, sometimes called SAP, refers to the type codes found in the DSAP and SSAP fields of the 802 header.) |
| *wild-mask* | 16-bit hexadecimal number whose ones bits correspond to bits in the *type-code* argument. The *wild-mask* argument indicates which bits in the *type-code* argument should be ignored when making a comparison. (A mask for a DSAP/SSAP pair should always be 0x0101 because these two bits are used for purposes other than identifying the SAP code.) |
| *address* | 48-bit Token Ring address written as a dotted triple of four-digit hexadecimal numbers. This field is used for filtering by vendor code. |
| *mask* | 48-bit Token Ring address written as a dotted triple of four-digit hexadecimal numbers. The ones bits in *mask* are the bits to be ignored in *address*. This field is used for filtering by vendor code. For source address filtering, the mask always should have the high-order bit set. This is because the IEEE 802 standard uses this bit to indicate whether a Routing Information Field (RIF) is present, not as part of the source address. |

**Defaults**

No access list is configured.

**Command Modes**

Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |

| Release | Modification |
|---------|--------------|
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**     For a list of type codes, see Appendix: Ethernet Type Codes.

**Examples**     In the following example, the access list permits only Novell frames (LSAP 0xE0E0) and filters out all other frame types. This set of access lists would be applied to an interface via the **source-bridge input-lsap list** or **source-bridge input-lsap list** command (described later in this chapter).

```
access-list 201 permit 0xE0E0 0x0101
access-list 201 deny 0x0000 0xFFFF
```

Combine the DSAP/LSAP fields into one number to do LSAP filtering; for example, 0xE0E0—not 0xE0. Note that the deny condition specified in the preceding example is not required; access lists have an implicit deny as the last statement. Adding this statement can serve as a useful reminder, however.

The following access list filters out only SNAP type codes assigned to Digital Equipment Corporation (DEC) (0x6000 to 0x6007) and lets all other types pass. This set of access lists would be applied to an interface using the **source-bridge input-type-list** or **source-bridge output-type-list** command (described later in this chapter).

```
access-list 202 deny 0x6000 0x0007
access-list 202 permit 0x0000 0xFFFF
```

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

Type code access lists will negatively affect system performance by greater than 30 percent. Therefore, we recommend that you keep the lists as short as possible and use wildcard bit masks whenever possible.

**Related Commands**

| Command | Description |
|---------|-------------|
| **access-expression** | Defines an access expression. |
| **source-bridge input-address-list** | Applies an access list to an interface configured for source-route bridging, and filters source-routed packets received from the router interface based on the source MAC address. |
| **source-bridge input-lsap-list** | Filters, on input, FDDI and IEEE 802-encapsulated packets that include the DSAP and SSAP fields in their frame formats. |
| **source-bridge input-type-list** | Filters SNAP-encapsulated packets on input. |

| Command | Description |
|---------|-------------|
| **source-bridge output-address-list** | Applies an access list to an interface configured for SRB, and filters source-routed packets sent to the router interface based on the destination MAC address. |
| **source-bridge output-lsap-list** | Filters, on output, FDDI and IEEE 802-encapsulated packets that have DSAP and SSAP fields in their frame formats. |
| **source-bridge output-type-list** | Filters SNAP-encapsulated frames by type code on output. |

# access-list (extended-ibm)

To provide extended access lists that allow more detailed access lists, use the **access-list** command in global configuration mode. 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**

| | |
|---|---|
| *access-list-number* | Integer from 1100 to 1199 that you assign to identify one or more **permit/deny** conditions as an extended access list. Note that a list number in the range from 1100 to 1199 distinguishes an extended access list from other access lists. |
| **permit** | 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. |
| **deny** | 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. |
| *source* | MAC Ethernet address in the form *xxxx.xxxx.xxxx*. |
| *source-mask* | Mask of MAC Ethernet source address bits to be ignored. The software uses the *source* and *source-mask* arguments to match the source address of a packet. |
| *destination* | MAC Ethernet value used for matching the destination address of a packet. |
| *destination-mask* | Mask of MAC Ethernet destination address bits to be ignored. The software uses the *destination* and *destination mask* arguments to match the destination address of a packet. |
| *offset* | Range of values that must be satisfied in the access list. Specified in decimal or in hexadecimal format in the form 0x*nn*. 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. |
| *size* | Range of values that must be satisfied in the access list. Must be an integer from 1 to 4. |

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

**Defaults**   No extended access lists are established.

**Command Modes**   Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**   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.

There is not a **no** form for this command.

**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 shows an extended access list. The first **access-list** command permits packets from MAC addresses 000c.1b*xx.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. The seconds **access-list** command permits an NOP operation:

```
access-list 1102 permit 000c.1b00.0000 0000.00ff.ffff 0000.0000.0000
    ffff.ffff.ffff 0x1e 2 lt 0x55aa
access-list 1101 permit 0000.0000.0000 ffff.ffff.ffff 0000.0000.0000
    ffff.ffff.ffff
!
interface ethernet 0
 bridge-group 3 output-pattern 1102
```

The following is sample output from the **show interfaces crb** command for the access list configured above:

```
Router# 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
```

Table 10 describes significant fields shown in the display.

*Table 10          show interfaces crb Field Descriptions*

| Field | Description |
|---|---|
| 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. |

**Related Commands**

| Command | Description |
|---|---|
| **access-list (standard-ibm)** | Establishes MAC address access lists. |
| **access-list (type-code-ibm)** | Builds type-code access lists. |
| **bridge-group output-pattern-list** | Associates an extended access list with a particular interface. |

# access-list (standard-ibm)

To establish a MAC address access list, use the **access-list** command in global configuration mode. To remove access list, use the **no** form of this command.

>   **access-list** *access-list-number* {**permit** | **deny**} *address mask*

>   **no access-list** *access-list-number*

**Syntax Description**

| | |
|---|---|
| *access-list-number* | Integer from 700 to 799 that you select for the list. |
| **permit** | Permits the frame. |
| **deny** | Denies the frame. |
| *address mask* | 48-bit MAC addresses written as a dotted triple of four-digit hexadecimal numbers. The ones bits in the *mask* argument are the bits to be ignored in *address*. |

**Defaults**      No MAC address access lists are established.

**Command Modes**      Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

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

**Examples**      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, and 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**

| Command | Description |
|---|---|
| **access-list (type-code-ibm)** | Builds type-code access lists. |

# access-list (type-code-ibm)

To build type-code access lists, use the **access-list** command in global configuration mode. To remove the access list, use the **no** form of this command.

> **access-list** *access-list-number* {**permit** | **deny**} *type-code wild-mask*

> **no access-list** *access-list-number*

**Syntax Description**

| | |
|---|---|
| *access-list-number* | User-selectable number from 200 to 299 that identifies the list. |
| **permit** | Permits the frame. |
| **deny** | Denies the frame. |
| *type-code* | 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 destination service access point (DSAP)/source service access point (SSAP) pair for 802.3 or 802.5-encapsulated packets. Ethernet type codes are listed in the appendix "Ethernet Type Codes." |
| *wild-mask* | 16-bit hexadecimal number whose ones bits correspond to bits in the *type-code* argument that should be ignored when making a comparison. (A mask for a DSAP/SSAP pair should always be at least 0x0101 because these two bits are used for purposes other than identifying the SAP codes.) |

**Defaults**  No type-code access lists are built.

**Command Modes**  Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**  Type-code access lists can have negatively affect 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.
- If the packet is another 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 Advanced Research Projects Agency (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 using type-code filtering.

- 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 Link Service Access Point (LSAP) 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 shows how to permit only local-area transport (LAT) frames (type 0x6004) and filters out all other frame types:

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

The following example shows how to filter out only type codes assigned to Digital Equipment Corporation (DEC) (0x6000 to 0x600F) and lets all other types pass:

```
access-list 202 deny 0x6000 0x000F
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**

| Command | Description |
|---|---|
| **access-list (standard-ibm)** | Establishes MAC address access lists. |

# bridge acquire

To forward any frames for stations that the system has learned about dynamically, use the **bridge acquire** command in global configuration mode. To disable the behavior, use the **no** form of this command.

**bridge** *bridge-group* **acquire**

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

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

**Defaults**    Enabled

**Command Modes**    Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**    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.

**Examples**    The following example shows how to prevent the forwarding of dynamically determined source and destination addresses:

```
no bridge 1 acquire
```

**Related Commands**

| Command | Description |
|---|---|
| **bridge address** | Filters frames with a particular MAC-layer station source or destination address. |
| **bridge protocol** | Defines the type of Spanning Tree Protocol. |

# bridge address

To filter frames with a particular MAC-layer station source or destination address, use the **bridge address** in global configuration mode. To disable the filtering of frames, use the **no** form of this command.

> **bridge** *bridge-group* **address** *mac-address* {**forward** | **discard**} [*interface*]

> **no bridge** *bridge-group* **address** *mac-address*

**Syntax Description**

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

**Defaults**    Disabled

**Command Modes**    Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

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

**Note**    MAC addresses on Ethernet 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, remember this point. Note that when a bridged packet traverses a serial link, it has an Ethernet-style address.

**Examples**     The following example shows how to enable 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 shows how to disable the ability to forward frames with MAC address 0800.cb00.45e9:

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

**Related Commands**

| Command | Description |
|---|---|
| **bridge acquire** | Forwards any frames for stations that the system has learned about dynamically. |
| **bridge-group input-address-list** | Assigns an access list to a particular interface. |
| **bridge-group output-address-list** | Assigns an access list to a particular interface for filtering the MAC destination addresses of packets that would ordinarily be forwarded out that interface. |
| **bridge protocol** | Defines the type of Spanning Tree Protocol. |

# bridge bitswap-layer3-addresses

To enable transparent bridging or source-route translational bridging or IP Advanced Research Projects Agency (ARPA) between canonical and noncanonical media types, use the **bridge bitswap-layer3-addresses** command in global configuration mode. To revert to the default setting, use the **no** form of this command.

> **bridge** *bridge-group* **bitswap-layer3-addresses**

> **no bridge** *bridge-group* **bitswap-layer3-addresses**

**Syntax Description**

| | |
|---|---|
| *bridge-group* | Bridge group number. |

**Defaults**     Disabled

**Command Modes**     Global configuration

**Command History**

| Release | Modification |
|---|---|
| 11.3(5) T | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**     This command "bit-swaps" (to and from noncanonical format) the hardware addresses that are embedded in layer 3 of ARP and Reverse Address Resolution Protocol (RARP) frames. This function enables IP communication between Token Ring and non-Token Ring media in a transparent-bridging environment. Because transparent bridging views the source-route bridge domain as a Token Ring media, enabling this command for a transparent bridge group also enables this function for source-route translational bridging (SR/TLB).

The user must ensure the frames are small enough to be sent on all media types because there is no end to end bridging protocol to negotiate the largest frame size.

There is no attempt to reformat ARP frames between ARP and Subnetwork Access Protocol (SNAP) formats.

**Examples**     The following example shows how to enable bit-swapping of addresses to and from noncanonical form in a transparent-bridged environment:

```
no ip routing
!
interface ethernet 0
 bridge-group 1
!
```

```
interface token-ring 0
 bridge-group 1
!
!
bridge 1 protocol ieee
bridge 1 bitswap-layer3-addresses
```

# bridge bridge

To enable the bridging of a specified protocol in a specified bridge group, use the **bridge bridge** command in global configuration mode. To disable the bridging of a specified protocol in a specified bridge group, use the **no** form of this command.

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

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

| Syntax Description | | |
|---|---|---|
| | *bridge-group* | Bridge group number specified in the **bridge protocol** command. |
| | *protocol* | Any of the supported routing protocols. The default is to bridge all of these protocols. |

**Defaults**  Bridge every protocol.

**Command Modes**  Global configuration

| Command History | Release | Modification |
|---|---|---|
| | 11.2 | This command was introduced. |
| | 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| | 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**  When integrated routing and bridging (IRB) is enabled, the default route/bridge behavior in a bridge group is to bridge all protocols. You need not 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 local-area transport (LAT), are bridged only. You cannot disable bridging for the nonroutable traffic.

**Examples**  The following example shows how to disable bridging of IP in bridge group 1:

```
no bridge 1 bridge ip
```

| | Command | Description |
|---|---|---|
| **Related Commands** | **bridge irb** | Enables the Cisco IOS software to route a given protocol between routed interfaces and bridge groups or to route a given protocol between bridge groups. |
| | **bridge protocol** | Defines the type of Spanning Tree Protocol. |
| | **bridge route** | Enables the routing of a specified protocol in a specified bridge group. |

# bridge circuit-group pause

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

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

**Syntax Description**

| | |
|---|---|
| *bridge-group* | Bridge group number specified in the **bridge protocol** command argument. |
| *circuit-group* | Number of the circuit group to which the interface belongs. |
| *milliseconds* | Forward delay interval. It must be a value in the range from 0 to 10000 ms. |

**Defaults**      The default forward delay interval is 0.

**Command Modes**      Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.3 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

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

There is not a **no** form for this command.

**Examples**      The following example shows how to set the circuit group pause to 5000 ms:

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

**Related Commands**

| Command | Description |
|---|---|
| **bridge circuit-group source-based** | Uses just the source MAC address for selecting the output interface. |
| **bridge-group circuit-group** | Assigns each network interface to a bridge group. |
| **bridge protocol** | Defines the type of Spanning Tree Protocol. |

# bridge circuit-group source-based

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

> **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. |

**Defaults**  No bridge-group interface is assigned.

**Command Modes**  Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.3 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**  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.

**Examples**  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**

| Command | Description |
|---|---|
| **bridge circuit-group pause** | Configures the interval during which transmission is suspended in a circuit group after circuit group changes take place. |
| **bridge-group circuit-group** | Assigns each network interface to a bridge group. |
| **bridge protocol** | Defines the type of Spanning Tree Protocol. |

# bridge cmf

To enable constrained multicast flooding (CMF) for all configured bridge groups, use the **bridge cmf** command in global configuration mode. To disable constrained multicast flooding, use the **no** form of this command.

**bridge cmf**

**no bridge cmf**

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

**Defaults**    CMF is disabled.

**Command Modes**    Global configuration

**Command History**

| Release | Modification |
|---------|-------------|
| 11.2 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Examples**    The following example shows how to enable CMF for all configured bridge groups:

```
bridge cmf
```

**Related Commands**

| Command | Description |
|---------|-------------|
| **clear bridge multicast** | Clears transparent bridging multicast state information. |
| **show bridge multicast** | Displays transparent bridging multicast state information. |

# bridge crb

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

> **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 Modes**    Global configuration

**Command History**

| Release | Modification |
|---|---|
| 11.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**    When concurrent routing and bridging is first enabled in the presence of existing bridge groups, it command generates a **bridge route** configuration command for any protocol for which any interface in the bridge group is configured for routing. This precaution 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).

**Examples**    The following command shows how to enable concurrent routing and bridging:

```
bridge crb
```

**Related Commands**

| Command | Description |
|---|---|
| **bridge route** | Enables the routing of a specified protocol in a specified bridge group. |

# bridge domain

To establish a domain by assigning it a decimal value from 1 and 10, use the **bridge domain** command in global configuration mode. To return to a single bridge domain by choosing domain zero (0), use the **no** form of this command.

> **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. |

**Defaults**

Single bridge domain. The default domain number is 0.

**Command Modes**

Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**

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 want 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 User Datagram Protocol (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.

**Examples**    The following example shows how to place 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**

| Command | Description |
|---|---|
| **bridge protocol** | Defines the type of Spanning Tree Protocol. |

# bridge forward-time

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

**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 from 10 to 200 seconds. The default is 30 seconds. |

**Defaults**    30-second delay

**Command Modes**    Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**    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 its individual configuration.

**Examples**    The following example shows how to set the forward delay interval to 60 seconds:

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

**Related Commands**

| Command | Description |
|---|---|
| **bridge-group subscriber-trunk** | Specifies that an interface is at the upstream point of traffic flow. |
| **bridge max-age** | Changes the interval the bridge will wait to hear BPDUs from the root bridge. |
| **bridge protocol** | Defines the type of Spanning Tree Protocol. |

# bridge hello-time

To specify the interval between hello bridge protocol data units (BPDUs), use the **bridge hello-time** command in global configuration mode. To return the default interval, use the **no** form of this command.

**bridge** *bridge-group* **hello-time** *seconds*

**no bridge** *bridge-group* **hello-time**

**Syntax Description**

| | |
|---|---|
| *bridge-group* | Bridge group number specified in the **bridge protocol** command. |
| *seconds* | Interval from 1 to 10 seconds. The default is 1 second. |

**Defaults**

1 second

**Command Modes**

Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**

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

**Examples**

The following example shows how to set the interval to 5 seconds:

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

**Related Commands**

| Command | Description |
|---|---|
| **bridge forward-time** | Specifies the forward delay interval for the Cisco IOS software. |
| **bridge max-age** | Changes the interval the bridge will wait to hear BPDUs from the root bridge. |
| **bridge protocol** | Defines the type of Spanning Tree Protocol. |

# bridge irb

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 **bridge irb** command in global configuration mode. To disable the feature, use the **no** form of this command.

> **bridge irb**

> **no bridge irb**

## Syntax Description

This command has no arguments or keywords.

## Defaults

Integrated routing and bridging (IRB) is disabled.

## Command Modes

Global configuration

## Command History

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

## Usage Guidelines

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.

## Examples

The following shows how to enable integrated routing and bridging:

```
bridge irb
```

## Related Commands

| Command | Description |
|---|---|
| **bridge bitswap-layer3-addresses** | Enables the bridging of a specified protocol in a specified bridge group. |
| **bridge route** | Enables the routing of a specified protocol in a specified bridge group. |
| **interface bvi** | Creates the BVI that represents the specified bridge group to the routed world and links the corresponding bridge group to the other routed interfaces. |
| **show interfaces irb** | Displays the configuration for each interface that has been configured for integrated routing or bridging. |

# bridge lat-service-filtering

To specify local-area transport (LAT) group-code filtering, use the **bridge lat-service-filtering** command in global configuration mode. To disable the use of LAT service filtering on the bridge group, use the **no** form of this command.

> **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. |

**Defaults**

LAT service filtering is disabled.

**Command Modes**

Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**

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

**Examples**

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

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

**Related Commands**

| Command | Description |
|---|---|
| **bridge protocol** | Defines the type of Spanning Tree Protocol. |

# bridge max-age

To change the interval the bridge will wait to hear Bridge Protocol Data Unit (BPDU) from the root bridge, use the **bridge max-age** command in global configuration mode. To return to the default interval, use the **no** form of this command.

**bridge** *bridge-group* **max-age** *seconds*

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

| Syntax Description | | |
|---|---|---|
| *bridge-group* | Bridge group number specified in the **bridge protocol** command. | |
| *seconds* | Interval the bridge will wait to hear BPDUs from the root bridge. It must be a value in the range from 10 to 200 seconds. The default is 15 seconds. | |

**Defaults**  15 seconds

**Command Modes**  Global configuration

| Command History | Release | Modification |
|---|---|---|
| | 10.0 | This command was introduced. |
| | 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| | 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**  Each bridge in a spanning tree adopts the **hello-time**, **forward-time**, and **max-age** parameters of the root bridge, regardless of its individual configuration. If a bridge does not receive BPDUs from the root bridge within this specified interval, it considers the network to be changed and will recompute the spanning-tree topology.

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

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

| Related Commands | Command | Description |
|---|---|---|
| | **bridge forward-time** | Specifies the forward delay interval for the Cisco IOS software. |
| | **bridge-group subscriber-trunk** | Specifies that an interface is at the upstream point of traffic flow. |
| | **bridge protocol** | Defines the type of Spanning Tree Protocol. |

# bridge multicast-source

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

> **bridge** *bridge-group* **multicast-source**

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

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

**Defaults**      Disabled

**Command Modes**      Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**      If you need to bridge Token Ring over another medium, remote source-route bridging (RSRB) is recommended.

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

```
bridge 2 multicast-source
```

**Related Commands**

| Command | Description |
|---|---|
| **bridge protocol** | Defines the type of Spanning Tree Protocol. |

# bridge priority

To configure the priority of an individual bridge, or the likelihood that it will be selected as the root bridge, use the **bridge priority** command in global configuration mode.

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

| Syntax Description | | |
|---|---|---|
| | *bridge-group* | Bridge group number specified in the **bridge protocol** command. |
| | *number* | The lower the number, the more likely the bridge will be chosen as root. When the IEEE Spanning Tree Protocol is enabled, the *number* argument ranges from 0 to 65535 (default is 32768). When the Digital Spanning Tree Protocol is enabled, the *number* argument 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 Modes**

Global configuration

| Command History | Release | Modification |
|---|---|---|
| | 10.0 | This command was introduced. |
| | 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| | 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**

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** command in interface configuration mode to control an interface priority.

There is not a **no** form for this command.

**Examples**

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

```
bridge 1 priority 100
```

| Related Commands | Command | Description |
|---|---|---|
| | **bridge-group priority** | Sets an interface priority. |
| | **bridge protocol** | Defines the type of Spanning Tree Protocol. |

# bridge protocol

To define the type of Spanning Tree Protocol, use the **bridge protocol** command in global configuration mode. To delete the bridge group, use the **no** form of this command with the appropriate keywords and arguments.

**bridge** *bridge-group* **protocol** {**dec** | **ibm** | **ieee** | **vlan-bridge**}

**no bridge** *bridge-group* **protocol** {**dec** | **ibm** | **ieee** | **vlan-bridge**}

**Syntax Description**

| | |
|---|---|
| *bridge-group* | Number in the range from 1 to 255 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. |
| **dec** | Digital Spanning Tree Protocol. |
| **ibm** | IBM Spanning Tree Protocol. |
| **ieee** | IEEE Ethernet Spanning Tree Protocol. |
| **vlan-bridge** | VLAN-Bridge Spanning Tree Protocol. |

**Defaults**

No Spanning Tree Protocol is defined.

**Command Modes**

Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.0(1)T | The **ibm** and **vlan-bridge** keywords were added. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**

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.

**Examples**

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

```
bridge 1 protocol dec
```

| Related Commands | Command | Description |
|---|---|---|
| | **bridge domain** | Establishes a domain by assigning it a decimal value from 1 to 10. |
| | **bridge-group** | Assigns each network interface to a bridge group. |

# bridge protocol ibm

To create a bridge group that runs the automatic spanning-tree function, use the **bridge protocol ibm** command in global configuration mode. To cancel the previous assignment, use the **no** form of this command.

**bridge** *bridge-group* **protocol ibm**

**no bridge** *bridge-group* **protocol ibm**

| Syntax Description | | |
|---|---|---|
| | *bridge-group* | Number in the range from 1 to 9 that refers to a particular set of bridged interfaces. |

**Defaults**　　No bridge group is defined.

**Command Modes**　　Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.3 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Examples**　　The following example specifies bridge 1 to use the automatic spanning-tree function:

```
bridge 1 protocol ibm
```

**Related Commands**

| Command | Description |
|---|---|
| **show source-bridge** | Displays the current source bridge configuration and miscellaneous statistics. |
| **source-bridge spanning (automatic)** | Enables the automatic spanning-tree function for a specified group of bridged interfaces. |
| **source-bridge spanning (manual)** | Enables use of spanning explorers. |

# bridge route

To enable the routing of a specified protocol in a specified bridge group, use the **bridge route** command in global configuration mode. To disable the routing of a specified protocol in a specified bridge group, use the **no** form of this command.

**bridge** *bridge-group* **route** *protocol*

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

**Syntax Description**

| | |
|---|---|
| *bridge-group* | Bridge group number specified in the **bridge protocol** command. |
| *protocol* | One of the following protocols: <br> • **appletalk** <br> • clns <br> • decnet <br> • ip <br> • ipx |

**Defaults**    No default bridge group or protocol is specified.

**Command Modes**    Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.3 | This command was introduced. |
| 12.2(13)T | The following values for the *protocol* argument were removed: <br> • **apollo** <br> • **vines** <br> • **xns** |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Examples**    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 | Command | Description |
|---|---|---|
| | **bridge crb** | Enables the Cisco IOS software to both route and bridge a given protocol on separate interfaces within a single router. |
| | **bridge protocol** | Defines the type of Spanning Tree Protocol. |

# bridge subscriber-policy

To bind a bridge group with a subscriber policy, use the **bridge subscriber-policy** command in global configuration mode. To disable the subscriber bridge group feature, use the **no** form of this command.

**bridge** *bridge-group* **subscriber-policy** *policy*

**no bridge** *bridge-group* **subscriber-policy** *policy*

| | |
|---|---|
| **Syntax Description** | |

| *bridge-group* | Bridge group number, in the range from 1 to 256, specified in the **bridge protocol** command. |
|---|---|
| *policy* | Subscriber policy number in the range from 1 to 100. |

**Defaults**     Table 11 shows the default values that are applied if no forward or filter decisions have been specified for the subscriber policy:

*Table 11          Packet Default Values*

| **Packet** | **Upstream** |
|---|---|
| ARP | Permit |
| Broadcast | Deny |
| CDP | Deny/Disable |
| Multicast | Permit |
| Spanning Tree Protocol | Deny/Disable |
| Unknown Unicast | Deny |

**Command Modes**     Global configuration

**Command History**

| **Release** | **Modification** |
|---|---|
| 11.3 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**     Standard access lists can coexist with the subscriber policy. However, subscriber policy will take precedence over the access list by being checked first. A packet permitted by the subscriber policy will be checked against the access list if it is specified. A packet denied by subscriber policy will be dropped with no further access list checking.

**Examples**     The following example forms a subscriber bridge group using policy 1:

```
bridge 1 subscriber-policy 1
```

**Related Commands**

| Command | Description |
| --- | --- |
| **bridge protocol** | Defines the type of Spanning Tree Protocol. |
| **show subscriber-policy** | Displays the details of a subscriber policy. |
| **subscriber-policy** | Defines or modifies the forward and filter decisions of the subscriber policy. |

# bridge-group

To assign each network interface to a bridge group, use the **bridge-group** command in interface configuration mode. To remove the interface from the bridge group, use the **no** form of this command.

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

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

**Syntax Description**

| | |
|---|---|
| *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |

**Defaults**

No bridge group interface is assigned.

**Command Modes**

Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**

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 high-level data link control (HLDC), 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 re initialized.

**Examples**

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**

| Command | Description |
|---|---|
| **bridge-group cbus-bridging** | Enables autonomous bridging on a ciscoBus2 controller. |
| **bridge-group circuit-group** | Assigns each network interface to a bridge group. |

**Cisco IOS Bridging Command Reference**

| Command | Description |
|---------|-------------|
| **bridge-group input-pattern-list** | Associates an extended access list with a particular interface in a particular bridge group. |
| **bridge-group output-pattern-list** | Associates an extended access list with a particular interface. |
| **bridge-group spanning-disabled** | Disables the spanning tree on a given interface. |

# bridge-group aging-time

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 **bridge-group aging-time** command in global configuration mode. To return to the default aging-time interval, use the **no** form of this command.

**bridge-group** *bridge-group* **aging-time** *seconds*

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

| Syntax Description | | |
|---|---|---|
| *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. | |
| *seconds* | Aging time, in the range from 10 to 1000000 seconds. The default is 300 seconds. | |

**Defaults**    300 seconds

**Command Modes**    Global configuration

**Command History**

| Release | Modification |
|---|---|
| 10.3 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**    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 send continuously, increase the aging time to record the dynamic entries for a longer time and thus reduce the possibility of flooding when the hosts send again.

**Examples**    The following example sets the aging time to 200 seconds:

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

**Related Commands**

| Command | Description |
|---|---|
| **bridge-group** | Assigns each network interface to a bridge group. |

**Cisco IOS Bridging Command Reference** ■

# bridge-group cbus-bridging

To enable autonomous bridging on a ciscoBus2 controller, use the **bridge-group cbus-bridging** command in interface configuration mode. To disable autonomous bridging, use the **no** form of this command.

> **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 from 1 to 255. |

**Defaults**    Autonomous bridging is disabled.

**Command Modes**    Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

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

You can enable autonomous bridging on Ethernet, FDDI (FCIT) and High-Speed Serial Interface (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, first define that interface 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 filter by MAC-level address on an interface only when autonomous bridging is enabled on that interface; autonomous bridging disables all other filtering and priority queueing.

**Examples**

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**

| Command | Description |
|---|---|
| **bridge-group** | Assigns each network interface to a bridge group. |

# bridge-group circuit-group

To assign each network interface to a bridge group, use the **bridge-group circuit-group** command in interface configuration mode. To remove the interface from the bridge group, use the **no** form of this command.

> **bridge-group** *bridge-group* **circuit-group** *circuit-group*

> **no bridge-group** *bridge-group* **circuit-group** *circuit-group*

| Syntax Description | | |
|---|---|---|
| | *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |
| | *circuit-group* | Circuit group number. The range is from 1 to 9. |

**Defaults**     No bridge group interface is assigned.

**Command Modes**     Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.3 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**     Circuit groups are primarily intended for use with High-Speed Serial Interface (HSSI)-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.

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

**Examples**     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**

| Command | Description |
|---|---|
| **bridge circuit-group pause** | Configures the interval during which transmission is suspended in a circuit group after circuit group changes take place. |
| **bridge circuit-group source-based** | Uses just the source MAC address for selecting the output interface. |

# bridge-group input-address-list

To assign an access list to a particular interface, use the **bridge-group input-address-list** command in interface configuration mode. To remove an access list from an interface, use the **no** form of this command.

**bridge-group** *bridge-group* **input-address-list** *access-list-number*

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

| Syntax Description | | |
|---|---|---|
| | *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |
| | *access-list-number* | Access list number you assigned with the **access-list** command. It must be in the range from 700 to 799. |

**Defaults**          No access list is assigned.

**Command Modes**          Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**          Use an access list to filter packets received on a particular interface, based on the MAC source addresses (of the packets).

**Examples**          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, and 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 | Command | Description |
|---|---|---|
| | **access-list (standard-ibm)** | Establishes MAC address access lists. |
| | **bridge address** | Filters frames with a particular MAC-layer station source or destination address. |
| | **bridge-group output-address-list** | Assigns an access list to a particular interface for filtering the MAC destination addresses of packets that would ordinarily be forwarded out that interface. |

# bridge-group input-lat-service-deny

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

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

> **no bridge-group** *bridge-group* **input-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 from 1 to 255. |
| | *group-list* | List of local-area transport (LAT) service groups. Single numbers and ranges are permitted. Ranges are specified with a dash between the first and last group numbers in the range. Specify a zero (0) to disable the LAT group code for the bridge group. |

**Defaults**  No group codes are specified.

**Command Modes**  Interface configuration

| Command History | Release | Modification |
|---|---|---|
| | 10.0 | This command was introduced. |
| | 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| | 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**  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.

**Examples**  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**

| Command | Description |
|---|---|
| **bridge-group** | Assigns each network interface to a bridge group. |
| **bridge-group input-lat-service-permit** | Specifies the group codes by which to permit access upon input. |
| **bridge-group output-lat-service-deny** | Specifies the group codes by which to deny access upon output. |

# bridge-group input-lat-service-permit

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

> **bridge-group** *bridge-group* **input-lat-service-permit** *group-list*

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

**Syntax Description**

| | |
|---|---|
| *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |
| *group-list* | local-area transport (LAT) service groups. Single numbers and ranges are permitted. Specify a zero (0) to disable the LAT group code for the bridge group. |

**Defaults**    No group codes are specified.

**Command Modes**    Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**    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.

**Examples**    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 | Command | Description |
|---|---|---|
| | **bridge-group input-lat-service-deny** | Specifies the group codes by which to deny access upon input. |
| | **bridge-group output-lat-service-permit** | Specifies the group codes by which to permit access upon output. |

# bridge-group input-lsap-list

To filter IEEE 802.2-encapsulated packets on input, use the **bridge-group input-lsap-list** command in interface configuration mode. To disable this capability, use the **no** form of this command.

**bridge-group** *bridge-group* **input-lsap-list** *access-list-number*

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

**Syntax Description**

| | |
|---|---|
| *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |
| *access-list-number* | Access list number you assigned with the standard **access-list** command. Specify a zero (0) to disable the application of the access list on the bridge group. |

**Defaults**      Disabled

**Command Modes**      Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**      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. Subnetwork Access Protocol (SNAP) frames must also pass any applicable Ethernet type-code access list.

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

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

**Related Commands**

| Command | Description |
|---|---|
| **access-list (standard-ibm)** | Establishes MAC address access lists. |
| **bridge-group** | Assigns each network interface to a bridge group. |
| **bridge-group output-lsap-list** | Filters IEEE 802-encapsulated packets on output. |

# bridge-group input-pattern-list

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

> **bridge-group** *bridge-group* **input-pattern-list** *access-list-number*

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

| *Syntax Description* | *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |
|---|---|---|
| | *access-list-number* | Access list number you assigned using the extended **access-list** command. Specify a zero (0) to disable the application of the access list on the interface. |

**Defaults**    Disabled

**Command Modes**    Interface configuration

| **Command History** | **Release** | **Modification** |
|---|---|---|
| | 10.0 | This command was introduced. |
| | 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| | 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**    Autonomous bridging must be disabled to use this command.

**Examples**    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** | **Command** | **Description** |
|---|---|---|
| | **access-list (standard-ibm)** | Establishes MAC address access lists. |
| | **bridge-group** | Assigns each network interface to a bridge group. |
| | **bridge-group output-pattern-list** | Associates an extended access list with a particular interface. |

# bridge-group input-type-list

To filter Ethernet- and Subnetwork Access Protocol (SNAP)-encapsulated packets on input, use the **bridge-group input-type-list** command in interface configuration mode. To disable this capability, use the **no** form of this command.

> **bridge-group** *bridge-group* **input-type-list** *access-list-number*

> **no bridge-group** *bridge-group* **input-type-list** *access-list-number*

| | |
|---|---|
| ***Syntax Description*** | |

| | |
|---|---|
| *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |
| *access-list-number* | Access list number you assigned with the standard **access-list** command. Specify a zero (0) to disable the application of the access list on the bridge group. |

**Defaults**    Disabled

**Command Modes**    Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**    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 destination service access point (DSAP)/source service access point (SSAP)/Organizationally Unique Identifier (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.

**Examples**    The following example shows how to configure a Token Ring interface with an access list that allows only the local-area transport (LAT) protocol to be bridged:

```
interface tokenring 0
 ip address 172.16.0.0 255.255.255.0
 bridge-group 1
 bridge-group 1 input-type-list 201
```

| Related Commands | Command | Description |
|---|---|---|
| | **access-list (standard-ibm)** | Establishes MAC address access lists. |
| | **bridge-group** | Assigns each network interface to a bridge group. |
| | **bridge-group output-type-list** | Filters Ethernet- and SNAP-encapsulated packets on output. |

# bridge-group lat-compression

To reduce the amount of bandwidth that local-area transport (LAT) traffic consumes on the serial interface by specifying a LAT-specific form of compression, use the **bridge-group lat-compression** command in interface configuration mode. To disable LAT compression on the bridge group, use the **no** form of this command.

**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 from 1 to 255. | |

**Defaults**   Disabled

**Command Modes**   Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**   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.

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

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

**Related Commands**

| Command | Description |
|---|---|
| **bridge-group** | Assigns each network interface to a bridge group. |

# bridge-group output-address-list

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 **bridge-group output-address-list** command in interface configuration mode. To remove an access list from an interface, use the **no** form of this command.

> **bridge-group** *bridge-group* **output-address-list** *access-list-number*

> **no bridge-group** *bridge-group* **output-address-list** *access-list-number*

**Syntax Description**

| | |
|---|---|
| *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |
| *access-list-number* | Access list number you assigned with the standard **access-list** command. |

**Defaults**     No access list is assigned.

**Command Modes**     Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

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

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

**Related Commands**

| Command | Description |
|---|---|
| **access-list (standard-ibm)** | Establishes MAC address access lists. |
| **bridge address** | Filters frames with a particular MAC-layer station source or destination address. |
| **bridge-group** | Assigns each network interface to a bridge group. |
| **bridge-group input-address-list** | Assigns an access list to a particular interface. |

# bridge-group output-lat-service-deny

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

**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 from 1 to 255. |
| --- | --- | --- |
| | *group-list* | List of local-area transport (LAT) groups. Single numbers and ranges are permitted. |

**Defaults**       No group codes are assigned.

**Command Modes**    Interface configuration

| Command History | Release | Modification |
| --- | --- | --- |
| | 10.0 | This command was introduced. |
| | 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| | 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**   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.

**Examples**      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 | Command | Description |
| --- | --- | --- |
| | **access-list (standard-ibm)** | Establishes MAC address access lists. |
| | **bridge-group** | Assigns each network interface to a bridge group. |

| Command | Description |
|---|---|
| **bridge-group input-lat-service-deny** | Specifies the group codes by which to deny access upon input. |
| **bridge-group output-lat-service-permit** | Specifies the group codes by which to permit access upon output. |

# bridge-group output-lat-service-permit

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

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

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

| Syntax Description | | |
|---|---|
| *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |
| *group-list* | Local-area transport (LAT) service advertisements. |

**Defaults**          No group codes are specified.

**Command Modes**      Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**     Autonomous bridging must be disabled for you to use this command.

This command causes the system to bridge onto an 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.

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

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

| Related Commands | Command | Description |
|---|---|---|
| | **bridge-group input-lat-service-permit** | Specifies the group codes by which to permit access upon input. |
| | **bridge-group output-lat-service-deny** | Specifies the group codes by which to deny access upon output. |

# bridge-group output-lsap-list

To filter IEEE 802-encapsulated packets on output, use the **bridge-group output-lsap-list** command in interface configuration mode. To disable this capability, use the **no** form of this command.

**bridge-group** *bridge-group* **output-lsap-list** *access-list-number*

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

**Syntax Description**

| | |
|---|---|
| *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |
| *access-list-number* | Access list number you assigned with the standard **access-list** command. Specify a zero (0) to disable the application of the access list on the bridge group. |

**Defaults**  Disabled

**Command Modes**  Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**  Autonomous bridging must be disabled to use this command.

Subnetwork Access Protocol (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. Such access lists cannot be used to block frames with protocols that are being routed.

Packets bearing an 802.2 LSAP of 0xAAAA qualify for LSAP filtering because 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 Link Service Access Point (LSAP) filters on an interface that may bear SNAP-encapsulated packets, you must explicitly permit 0xAAAA.

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

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

**Related Commands**

| Command | Description |
|---|---|
| **access-list (standard-ibm)** | Establishes MAC address access lists. |
| **bridge-group** | Assigns each network interface to a bridge group. |
| **bridge-group input-lsap-list** | Filters IEEE 802.2-encapsulated packets on input. |

# bridge-group output-pattern-list

To associate an extended access list with a particular interface, use the **bridge-group output-pattern-list** command in interface configuration mode. To disable this capability, use the **no** form of this command.

> **bridge-group** *bridge-group* **output-pattern-list** *access-list-number*

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

**Syntax Description**

| | |
|---|---|
| *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |
| *access-list-number* | Extended access list number you assigned using the extended **access-list** command. Specify a zero (0) to disable the application of the access list on the interface. |

**Defaults**  Disabled

**Command Modes**  Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**  Autonomous bridging must be disabled to use this command.

**Examples**  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**

| Command | Description |
|---|---|
| **access-list (standard-ibm)** | Establishes MAC address access lists. |
| **bridge-group** | Assigns each network interface to a bridge group. |
| **bridge-group input-pattern-list** | Associates an extended access list with a particular interface in a particular bridge group. |

# bridge-group output-type-list

To filter Ethernet- and Subnetwork Access Protocol (SNAP)-encapsulated packets on output, use the **bridge-group output-type-list** command in interface configuration mode. To disable this capability, use the **no** form of this command.

> **bridge-group** *bridge-group* **output-type-list** *access-list-number*

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

| Syntax Description | | |
|---|---|---|
| | *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |
| | *access-list-number* | Access list number you assigned with the standard **access-list** 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. |

**Defaults**      Disabled

**Command Modes**      Interface configuration

| Command History | Release | Modification |
|---|---|---|
| | 10.0 | This command was introduced. |
| | 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| | 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**      Autonomous bridging must be disabled to use this command.

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

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

| Related Commands | Command | Description |
|---|---|---|
| | **access-list (standard-ibm)** | Establishes MAC address access lists. |
| | **bridge-group** | Assigns each network interface to a bridge group. |
| | **bridge-group input-type-list** | Filters Ethernet- and SNAP-encapsulated packets on input. |

# bridge-group path-cost

To set a different path cost, use the **bridge-group path-cost** command in interface configuration mode. To choose the default path cost for the interface, use the **no** form of this command.

**bridge-group** *bridge-group* **path-cost** *cost*

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

**Syntax Description**

| | |
|---|---|
| *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |
| *cost* | Relative cost of using the path. 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 Modes**

Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**

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.

**Examples**

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

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

| Related Commands | Command | Description |
|---|---|---|
| | **bridge-group** | Assigns each network interface to a bridge group. |

# bridge-group priority

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

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

**Syntax Description**

| | |
|---|---|
| *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |
| *number* | Priority number ranging from 0 to 255 (Digital), or 0 to 64000 (IEEE). The default is 32768 if IEEE Spanning Tree Protocol is enabled on the router or 128 if Digital Spanning Tree Protocol is enabled on the router. |

**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 Modes**

Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**

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

There is not a **no** form for this command.

**Examples**

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
```

The following example shows the **bridge-group priority** help information for 9-bit port number size:

```
Router(config-if)# bridge-group 1 priority ?
<0-255> increments of 2 for IEEE or vlan-bridge, others 1
```

The following example shows the **bridge-group priority** help information for 10-bit port number size:

```
Router(config-if)# bridge-group 1 priority ?
<0-255> increments of 4 for IEEE or vlan-bridge, others 1
```

| Related Commands | Command | Description |
|---|---|---|
| | **bridge-group** | Assigns each network interface to a bridge group. |
| | **bridge priority** | Configures the priority of an individual bridge, or the likelihood that it will be selected as the root bridge. |

# bridge-group spanning-disabled

To disable the spanning tree on a given interface, use the **bridge-group spanning-disabled** command in interface configuration mode. To enable the spanning tree on a given interface, use the no form of this command.

> **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 from of 1 to 255. |
|---|---|---|

**Defaults**    Spanning tree is enabled.

**Command Modes**    Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Usage Guidelines**    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 Bridge Protocol Data Unit (BPDU)s 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.

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

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

**Related Commands**

| Command | Description |
|---|---|
| **bridge-group** | Assigns each network interface to a bridge group. |
| **bridge protocol** | Defines the type of Spanning Tree Protocol. |

# bridge-group sse

To enable the Cisco silicon switching engine (SSE) switching function, use the **bridge-group sse** command in interface configuration mode. To disable SSE switching, use the **no** form of this command.

**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 from 1 to 255. |
|---|---|---|

**Defaults**  Disabled

**Command Modes**  Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 10.0 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Examples**  The following shows how to enable SSE switching:

```
bridge-group 1 sse
```

**Related Commands**

| Command | Description |
|---|---|
| **source-bridge** | Configures an interface for SRB. |

# bridge-group subscriber-loop-control

To enable loop control on virtual circuits associated with a bridge group, use the **bridge-group subscriber-loop-control** command in interface configuration mode. To disable loop control, use the **no** form of this command.

> **bridge-group** *bridge-group* **subscriber-loop-control**

> **no bridge-group** *bridge-group* **subscriber-loop-control**

**Syntax Description**

| | |
|---|---|
| *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. |

**Defaults**    Loop control is disabled.

**Command Modes**    Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 12.2 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Examples**    The following shows how to enable loop control on virtual circuits associated with bridge group 1:

```
bridge-group 1 subscriber-loop-control
```

**Related Commands**

| Command | Description |
|---|---|
| **bridge protocol** | Defines the type of Spanning Tree Protocol. |
| **bridge subscriber-policy** | Binds a bridge group with a subscriber policy. |
| **show subscriber-policy** | Displays the details of a subscriber policy. |
| **subscriber-policy** | Defines or modifies the forward and filter decisions of the subscriber policy. |

# bridge-group subscriber-trunk

To specify that an interface is at the upstream point of traffic flow, use the **bridge-group subscriber-trunk** command in interface configuration mode. To remove the specification and reset the interface to a nontrunking port, use the **no** form of this command.

> **bridge-group** *bridge-group* **subscriber-trunk**

> **no bridge-group** *bridge-group* **subscriber-trunk**

| Syntax Description | | |
|---|---|---|
| *bridge-group* | Number of the bridge group to which the interface belongs. It must be a number in the range from 1 to 255. | |

**Defaults**   The interface is set to a nontrunking port.

**Command Modes**   Interface configuration

**Command History**

| Release | Modification |
|---|---|
| 11.3 | This command was introduced. |
| 12.2(33)SRA | This command was integrated into Cisco IOS Release 12.2(33)SRA. |
| 12.2SX | This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware. |

**Examples**   The following example sets bridge group 1 as the upstream point of traffic flow:

```
bridge-group 1 subscriber-trunk
```

**Related Commands**

| Command | Description |
|---|---|
| **bridge protocol** | Defines the type of Spanning Tree Protocol. |
| **bridge subscriber-policy** | Binds a bridge group with a subscriber policy. |
| **show subscriber-policy** | Displays the details of a subscriber policy. |
| **subscriber-policy** | Defines or modifies the forward and filter decisions of the subscriber policy. |