



# Source-Route Bridging Commands

---

Use the commands in this chapter to configure and monitor source-route bridging (SRB) networks. For source-route bridging configuration information and examples, refer to the “Configuring Source-Route Bridging” chapter in the *Bridging and IBM Networking Configuration Guide*.

## access-expression

Use the **access-expression** interface configuration command to define an access expression. Use the **no** form of this command to remove the access expression from the given interface. You use this command in conjunction with the **access-list** interface configuration 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.

### Default

No access expression is defined.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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 10 describes the terms that can be used.

**Table 10 Access Expression Terms**

Access Expression Term	Definition
lsap(2nn)	The LSAP access list to be evaluated for this frame (200 series).
type(2nn)	The SNAP type access list to be evaluated for this frame (200 series).
smac(7nn)	The access list to match the source MAC address of the frame (700 series).
dmac(7nn)	The access list to match the destination MAC address of the frame (700 series).
netbios-host(name)	The netbios-host access list to be applied on NetBIOS frames traversing the interface.
netbios-bytes(name)	The netbios-bytes access list to be applied on NetBIOS frames traversing the interface.

---

**Note** The *netbios-host* and *netbios-bytes* access expression terms always will return FALSE for frames that are not NetBIOS frames.

---

Access expression terms are separated by Boolean operators as listed in Table 11.

**Table 11 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 ~. 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. 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 to the left or right of the   either or both of return TRUE. Example: “lsap (201)   dmac (701)” returns TRUE if either the lsap (201) or dmac (701) terms return TRUE, as well as 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 left to right.

---

Related Command  
**access-list**

## access-list

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

```
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

<i>access-list-number</i>	Integer that identifies the access list. If the <i>type-code wild-mask</i> arguments are included, this integer ranges from 200 to 299, indicating that filtering is by protocol type. If the <i>address</i> and <i>mask</i> arguments are included, this integer ranges from 700 to 799, indicating that filtering is by vendor code.
<b>permit</b>	Permits the frame.
<b>deny</b>	Denies the frame.
<i>type-code</i>	16-bit hexadecimal number written with a leading 0x; for example, 0x6000. Specify either a Link Service Access Point (LSAP) type code for 802-encapsulated packets or a 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.)
<i>wild-mask</i>	16-bit hexadecimal number whose ones bits correspond to bits in the <i>type-code</i> argument. The <i>wild-mask</i> indicates which bits in the <i>type-code</i> 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.)
<i>address</i>	48-bit Token Ring address written in dotted triplet form. This field is used for filtering by vendor code.
<i>mask</i>	48-bit Token Ring address written in dotted triplet form. The ones bits in <i>mask</i> are the bits to be ignored in <i>address</i> . This field is used for filtering by vendor code.

---

**Note** 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 RIF is present, not as part of the source address.

---

### Default

No access list is configured.

## Command Mode

Global configuration

## Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

For a list of type codes, refer to the “Ethernet Type Codes” appendix of this manual.

## 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** commands (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 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** commands (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, it is recommended that you keep the lists as short as possible and use wildcard bit masks whenever possible.

## Related Commands

**access-expression**

**source-bridge input-address-list**

**source-bridge input-lsap-list**

**source-bridge input-type-list**

**source-bridge output-address-list**

**source-bridge output-lsap-list**

**source-bridge output-type-list**

## bridge protocol ibm

Use the **bridge protocol ibm** global configuration command to create a bridge group that runs the automatic spanning-tree function. Use the **no** form of this command to cancel the previous assignment.

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

### Syntax Description

*bridge-group*      Number in the range 1 to 9 that you choose to refer to a particular set of bridged interfaces.

### Default

No bridge group is defined.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

### Example

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

```
bridge 1 protocol ibm
```

### Related Commands

**show source-bridge**  
**source-bridge spanning (automatic)**  
**source-bridge spanning (manual)**

## clear netbios-cache

Use the **clear netbios-cache** privileged EXEC command to clear the entries of all dynamically learned NetBIOS names. This command will not remove statically defined name cache entries.

**clear netbios-cache**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The Cisco IOS software automatically learns NetBIOS names. This command clears those entries.

### Example

The following example clears all dynamically learned NetBIOS names:

```
clear netbios-cache
```

### Related Commands

**netbios enable-name-cache**  
**netbios name-cache timeout**  
**show netbios-cache**

## clear rif-cache

Use the **clear rif-cache** privileged EXEC command to clear the entire RIF cache.

**clear rif-cache**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Some entries in the RIF cache are dynamically added and others are static.

### Example

The following example clears the entire RIF cache:

```
clear rif-cache
```

### Related Commands

**rif**  
**rif timeout**  
**show rif**

## clear source-bridge

Use the **clear source-bridge** privileged EXEC command to clear the source-bridge statistical counters.

**clear source-bridge**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example clears the source-bridge statistical counters:

```
clear source-bridge
```

### Related Command

A dagger (†) indicates that the command is documented outside this chapter.

**clear bridge** †

## clear sse

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

**clear sse**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

The silicon switching engine (SSE) is on the SSP board in the Cisco 7000 router.

### Example

The following example reinitializes the SSP:

```
clear sse
```

## ethernet-transit-oui

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

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

### Syntax Description

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

### Default

**90-compatible**

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

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

Before using this command, you must have completely configured your router using multiport source-bridging and transparent bridging.

The **standard** keyword is used when you are forced to interoperate with other vendor equipment, such as the IBM 8209, in providing Ethernet and Token Ring mixed media bridged connectivity.

The actual OUI codes that are used, when they are used, and how they compare to Software Release 9.0-equivalent commands are shown in Table 12.

**Table 12 Bridge OUI Codes**

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

Specify the **90-compatible** keyword when talking to our routers. This keyword provides the most flexibility. When **90-compatible** is specified or the default is used, Token Ring frames with an OUI of 0x0000F8 are translated into Ethernet Type II frames while Token Ring frames with the OUI of 0x000000 are translated into SNAP-encapsulated frames. Specify the **standard** keyword when talking to IBM 8209 bridges and other vendor equipment. This OUI does not provide for as much flexibility as the other two choices. The **cisco** OUI is provided for compatibility with future equipment.

Do not use the **standard** keyword unless you are forced to interoperate with other vendor equipment, such as the IBM 8209, in providing Ethernet and Token Ring mixed media bridged connectivity. Only use the **standard** keyword when you are transferring data between IBM 8209 Ethernet/Token Ring bridges and routers running the SR/TLB software (to create a Token Ring backbone to connect Ethernets).

Use of the **standard** keyword causes the OUI code in Token Ring frames to always be 0x000000. In the context of the **standard** keyword, an OUI of 0x000000 identifies the frame as an Ethernet Type II frame. (Compare with 90-compatible, where 0x000000 OUI means SNAP-encapsulated frames.)

If you use the **90-compatible** keyword, the router, acting as an SR/TLB, can distinguish immediately on Token Ring interfaces between frames that started on an Ethernet Type II frame and those that started on an Ethernet as a SNAP-encapsulated frame. The distinction is possible because the router uses the 0x0000F8 OUI when converting Ethernet Type II frames into Token Ring SNAP frames, and leaves the OUI as 0x000000 for Ethernet SNAP frames going to a Token Ring. This distinction in OUIs leads to efficiencies in the design and execution of the SR/TLB product; no tables need to be kept to know which Ethernet hosts use SNAP encapsulation and which hosts use Ethernet Type II.

The IBM 8209 bridges, however, by using the 0x000000 OUI for all the frames entering the Token Ring, must take extra measures to perform the translation. For every station on each Ethernet, the 8209 bridges attempt to remember the frame format used by each station, and assume that once a station sends out a frame using Ethernet Type II or 802.3, it will always continue to do so. It must do this because in using 0x000000 as an OUI, there is no way to distinguish between SNAP and Type II frame types. Because the SR/TLB router does not need to keep this database, when 8209 compatibility is enabled with the **standard** keyword, the SR/TLB chooses to translate all Token Ring SNAP frames into Ethernet Type II frames as described earlier in this discussion. Because every nonroutable protocol on Ethernet uses either non SNAP 802.3 (which traverses fully across a mixed IBM 8209/ router Token Ring backbone) or Ethernet Type II, this results in correct interconnectivity for virtually all applications.

Do not use the **standard** OUI if you want SR/TLB to output Ethernet SNAP frames. Using either the **90-compatible** or **cisco** OUI does not present such a restriction, because SNAP frames and Ethernet Type II-encapsulated frames have different OUI codes on Token Ring networks.

### Example

The following example specifies standard OUI form:

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

### Related Commands

A dagger (†) indicates that the command is documented outside this chapter.

**ethernet-transit-oui** †

**source-bridge transparent**

## Inm alternate

Use the **inm alternate** interface configuration command to specify the threshold reporting link number. In order for a LAN Reporting Manager (LRM) to change parameters, it must be attached to the reporting link with the lowest reporting link number, and that reporting link number must be lower than this threshold reporting link number. Use the **no** form of this command to restore the default of 0.

**inm alternate** *number*  
**no inm alternate**

### Syntax Description

*number* Threshold reporting link number. It must be in the range 0 to 3.

### Default

0

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

LAN Network Manager (LNM) employs the concepts of reporting links and reporting link numbers. A reporting link is simply a connection (or potential connection) between an LRM and a bridge. A reporting link number is a unique number used to identify a reporting link. An IBM bridge allows four simultaneous reporting links numbered 0 to 3. Only the LRM attached to the lowest number connection is allowed to change any parameters, and then only when that connection number falls below a certain configurable number. In the default configuration, the LRM connected through link 0 is the only LRM allowed to change parameters.

---

**Note** Setting the threshold reporting link number on one interface in a source-route bridge will cause it to appear on the other interface of the bridge, because the command applies to the bridge itself and not to either of the interfaces.

---

### Examples

The following example permits LRMs connected through links 0 and 1 to change parameters:

```
! provide appropriate global configuration command if not currently in your config.  
!  
! permit 0 and 1  
inm alternate 1
```

The following example permits all LRMs to change parameters in the Cisco IOS software:

```
! provide appropriate global configuration command if not currently in your config.  
!  
! permit 0, 1, 2, and 3  
inm alternate 3
```

Related Command  
**Inm password**

## Inm crs

Use the **lnm crs** interface configuration command to monitor the current logical configuration of a Token Ring. Use the **no** form of this command to disable this function.

**lnm crs**  
**no lnm crs**

### Syntax Description

This command has no arguments or keywords.

### Default

Enabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The Configuration Report Server (CRS) service keeps track of the current logical configuration of a Token Ring and reports any changes to LNM. It also reports on various other activities such as the change of the Active Monitor on a Token Ring.

For more information about the Active Monitor, refer to the *IBM Token Ring Architecture Reference Manual* or the IEEE 802.5 specification.

### Example

The following example disables monitoring of the current logical configuration of a Token Ring:

```
interface tokenring 0
no lnm crs
```

### Related Commands

**lnm rem**  
**lnm rps**

## Inm disabled

Use the **inm disabled** global configuration command to disable LAN Network Manager (LNM) functionality. Use the **no** form of this command to restore LNM functionality.

**inm disabled**  
**no inm disabled**

### Syntax Description

This command has no arguments or keywords.

### Default

Enabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

Under some circumstances, you can disable all LNM server functions on the router without having to determine whether to disable a specific server, such as the ring parameter server or the ring error monitor on a given interface.

This command can be used to terminate all LNM server input and reporting links. In normal circumstances, this command should not be necessary because it is a superset of the functions normally performed on individual interfaces by the **no inm rem** and **no inm rps** commands.

### Example

The following example disables LNM functionality:

```
inm disabled
```

### Related Commands

**inm pathtrace-disabled**  
**inm rem**  
**inm rps**  
**inm snmp-only**  
**show inm bridge**

## Inm express-buffer

Use the **inm express-buffer** interface configuration command to enable the LNM RPS express buffer function. Use the **no** form of this command to disable the function.

**inm express-buffer**

**no inm express-buffer**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

The RPS express buffer function allows the router to set the express buffer bit to ensure priority service for frames required for ring station initiation. When this function is enabled, the router sets the express buffer bit in its initialize ring station response. This allows Token Ring devices to insert into the ring during bursty conditions.

### Example

The following example enables the LNM RPS express buffer function:

```
inm express-buffer
```

## Inm loss-threshold

Use the **inm loss-threshold** interface configuration command to set the threshold at which the Cisco IOS software sends a message informing all attached LNMs that it is dropping frames. Use the **no** form of this command to return to the default value.

**inm loss-threshold** *number*  
**no inm loss-threshold**

### Syntax Description

*number* A single number expressing the percentage loss rate in hundredths of a percent. The valid range is 0 to 9999.

### Default

10 (.10 percent)

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The software sends a message to all attached LNMs whenever it begins to drop frames. The point at which this report is generated (threshold) is a percentage of the number of frames dropped compared with the number of frames forwarded.

When setting this value, remember that 9999 would mean 100 percent of your frames could be dropped before the message is sent. A value of 1000 would mean 10 percent of the frames could be dropped before sending the message. A value of 100 would mean 1 percent of the frames could be dropped before the message is sent.

### Example

In the following example, the loss threshold is set to 0.02 percent:

```
interface tokenring 0
 inm loss-threshold 2
```

## Inm password

Use the **inm password** interface configuration command to set the password for the reporting link. Use the **no** form of this command to return the password to its default value of 00000000.

**inm password** *number string*  
**no inm password** *number*

### Syntax Description

<i>number</i>	Number of the reporting link to which to apply the password. This value should be in the range 0 to 3.
<i>string</i>	Password you enter at the keyboard. In order to maintain compatibility with LNM, the parameter <i>string</i> should be a six- to eight-character string of the type listed in the “Usage Guidelines” section.

**Default**  
00000000

**Command Mode**  
Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

LAN Network Manager (LNM) employs the concepts of reporting links and reporting link numbers. A reporting link is simply a connection (or potential connection) between a LAN Reporting Manager (LRM) and a bridge. A reporting link number is a unique number used to identify a reporting link. An IBM bridge allows four simultaneous reporting links numbered 0 to 3. Only the LRM attached to the lowest number connection is allowed to change any parameters, and then only when that connection number falls below a certain configurable number. In the default configuration, the LRM connected through link 0 is the only LRM allowed to change parameters.

Each reporting link has its own password. Passwords are used not only to prevent unauthorized access from an LRM to a bridge, but to control access to the different reporting links. This is important because of the different abilities associated with the various reporting links.

Characters allowable in the *string* are the following:

- Letters
- Numbers
- Special characters @, #, \$, or %

Passwords are displayed only through use of the privileged EXEC **show running-config** command.

---

**Note** There are two parameters in an IBM bridge that have no corresponding parameter in the Cisco IOS software. This means that any attempt to modify these parameters from LNM will fail and display an error message. The LNM names of these two parameters are *route active status* and *single route broadcast mode*.

---

### Example

In the following example, the password *Zephyr@* is assigned to reporting link 2:

```
! provide appropriate global configuration command if not currently in your config.  
!  
inm password 2 Zephyr@
```

### Related Command

**Inm alternate**

## Inm pathtrace-disabled

Use the **inm pathtrace-disabled** global configuration command to disable pathtrace reporting to LAN Network Manager (LNM) stations. Use the **no** form of this command to restore pathtrace reporting functionality.

**inm pathtrace-disabled [all | origin]**  
**no inm pathtrace-disabled**

### Syntax Description

**all** Disable pathtrace reporting to the LNM and originating stations.

**origin** Disable pathtrace reporting to originating stations only.

### Default

Enabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

Under some circumstances, such as when new hardware has been introduced into the network and is causing problems, the automatic report path trace function can be disabled. The new hardware may be setting bit-fields B1 or B2 (or both) of the routing control field in the routing information field embedded in a source-route bridged frame. This condition may cause the network to be flooded by report path trace frames if the condition is persistent. The **inm pathtrace-disabled** command, along with its options, allows you to alleviate network congestion that may be occurring by disabling all or part of the automatic report path trace function within LNM.

### Example

The following example disables all pathtrace reporting:

```
inm pathtrace-disabled
```

### Related Commands

**inm disabled**  
**inm pathtrace-disabled**  
**show inm bridge**

## Inm rem

Use the **inm rem** interface configuration command to monitor errors reported by any station on the ring. Use the **no** form of this command to disable this function.

**inm rem**  
**no inm rem**

### Syntax Description

This command has no arguments or keywords.

### Default

Enabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The Ring Error Monitor (REM) service monitors errors reported by any station on the ring. It also monitors whether the ring is in a functional state or in a failure state.

### Example

The following example shows the use of the **inm rem** command:

```
interface tokenring 0
 inm rem
```

### Related Commands

**inm crs**  
**inm rps**

## Inm rps

Use the **lnm rps** interface configuration command to ensure that all stations on a ring are using a consistent set of reporting parameters. Use the **no** form of this command to disable this function.

**lnm rps**  
**no lnm rps**

### Syntax Description

This command has no arguments or keywords.

### Default

Enabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The Ring Parameter Server (RPS) service ensures that all stations on a ring are using a consistent set of reporting parameters and are reporting to LNM when any new station joins a Token Ring.

### Example

The following example shows the use of the **lnm rps** command:

```
interface tokenring 0
lnm rps
```

### Related Commands

**lnm crs**  
**lnm rem**

## Inm snmp-only

Use the **inm snmp-only** global configuration command to prevent any LNM stations from modifying parameters in the Cisco IOS software. Use the **no** form of this command to allow modifications.

**inm snmp-only**  
**no inm snmp-only**

### Syntax Description

This command has no arguments or keywords.

### Default

Enabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Configuring a router for LNM support is very simple. It happens automatically as a part of configuring the router to act as a source-route bridge. There are several commands available to modify the behavior of the LNM support, but none of them are necessary for it to function.

Because there is now more than one way to remotely change parameters in the Cisco IOS software, this command was developed to prevent them from detrimentally interacting with each other.

This command does not affect the ability of LNM to monitor events, only to modify parameters in the Cisco IOS software.

### Example

The following command prevents any LNM stations from modifying parameters in the software:

```
inm snmp-only
```

## Inm softerr

Use the **inm softerr** interface configuration command to set the time interval in which the Cisco IOS software will accumulate error messages before sending them. Use the **no** form of this command to return to the default value.

**inm softerr** *milliseconds*  
**no inm softerr**

### Syntax Description

*milliseconds* Time interval in tens of milliseconds between error messages. The valid range is 0 to 65535.

### Default

200 ms (2 seconds)

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

All stations on a Token Ring notify the Ring Error Monitor (REM) when they detect errors on the ring. In order to prevent excessive messages, error reports are not sent immediately, but are accumulated for a short period of time and then reported. A station learns this value from a router (configured as a source-route bridge) when it first enters the ring.

### Example

The following example changes the error-reporting frequency to once every 5 seconds:

```
inm softerr 500
```

### Related Command

**inm rem**

## mac-address

Use the **mac-address** interface configuration command to set the MAC layer address of the Cisco Token Ring.

```
mac-address ieee-address
```

### Syntax Description

*ieee-address* 48-bit IEEE MAC address written as a dotted triplet of four-digit hexadecimal numbers

### Default

No MAC layer address is set.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

There is a known defect in earlier forms of this command of the Texas Instruments (TI) Token Ring MAC firmware. This implementation is used by Proteon, Apollo, and IBM RTs. A host using a MAC address whose first two bytes are zeros (such as a Cisco router) will not properly communicate with hosts using that form of this command of TI firmware.

There are two solutions. The first involves installing a static RIF entry for every faulty node with which the router communicates. If there are many such nodes on the ring, this may not be practical. The second solution involves setting the MAC address of the Cisco Token Ring to a value that works around the problem.

This command forces the use of a different MAC address on the specified interface, thereby avoiding the TI MAC firmware problem. It is up to the network administrator to ensure that no other host on the network is using that MAC address.

### Example

The following example sets the MAC layer address, where *xx.xxxx* is an appropriate second half of the MAC address to use:

```
interface tokenring 0
mac-address 5000.5axx.xxxx
```

## multiring

Use the **multiring** interface configuration command to enable collection and use of RIF information. Use the **no** form of this command to disable the use of RIF information for the protocol specified.

```
multiring {protocol-keyword [all-routes | spanning] | all | other}
no multiring {protocol-keyword [all-routes | spanning] | all | other}
```

### Syntax Description

<i>protocol-keyword</i>	Specifies a protocol; see the keyword list under the “Usage Guidelines” section.
<b>all-routes</b>	Uses all-routes explorers.
<b>spanning</b>	Uses spanning-tree explorers.
<b>all</b>	Enables the multiring for <i>all</i> frames.
<b>other</b>	Enables the multiring for <i>any</i> routed frame not included in the previous list of supported protocols.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0. The keywords **all-routes** and **spanning** first appeared in Cisco Release 11.1.

Level 3 routers that use protocol-specific information (for example, Novell IPX or XNS headers) rather than MAC information to route datagrams also must be able to collect and use RIF information to ensure that they can transmit datagrams across a source-route bridge. The software default is to not collect and use RIF information for routed protocols. This allows operation with software that does not understand or properly use RIF information.

The current software allows you to specify a protocol. This is specified by the argument *protocol-keyword*. The protocols supported and the keywords you can enter include the following:

- **apollo**—Apollo Domain
- **appletalk**—AppleTalk Phase 1 and 2
- **clns**—ISO CLNS
- **decnet**—DECnet Phase IV
- **ip**—IP
- **ipx**—Novell IPX

- **vines**—Banyan VINES
- **xns**—XNS

---

**Note** When you are configuring DLSw+ over FDDI, the **multiring** command supports only IP and IPX.

---

The **multiring** command was extended in Software Release 8.3 to allow for per-protocol specification of the interface's ability to append RIFs to routed protocols. When it is enabled for a protocol, the router will source packets that include information used by source-route bridges. This allows a router with Token Ring interfaces, for the protocol or protocols specified, to connect to a source-bridged Token Ring network. If a protocol is not specified for multiring, the router can only route packets to nodes directly connected to its local Token Ring.

---

**Note** Previous to Software Release 8.3, the **multiring** command enabled multiring protocols, in particular, the use of explorers and RIFs, for *all* routable protocols. This sometimes caused problems when multiring-capable devices speaking one particular protocol were attached to the same ring as a nonmultiring-capable device speaking a different network protocol. If the earlier **multiring** command (pre-8.3 release) was not specified, nodes speaking one particular protocol would be able to communicate through the router, but nodes speaking other protocols could not. The reverse was true when the multiring capability was specified on the interface. In 8.3 or later releases of the software, the command **multiring all** is equivalent to the 8.2 and earlier forms of the **multiring** command.

---

### Example

These commands enable IP and Novell IPX bridging on a Token Ring interface. RIFs will be generated for IP frames, but not for the Novell IPX frames.

```
! commands that follow apply to interface token 0
interface tokenring 0
! generate RIFs for IP frames
multiring ip
! enable the Token Ring interface for IP
ip address 131.108.183.37 255.255.255.0
! enable the Token Ring interface for Novell IPX
novell network 33
```

### Related Commands

A dagger (†) indicates that the command is documented outside this chapter.

**clear rif-cache**

**rif**

**rif timeout**

**show rif**

**xns encapsulation** †

## netbios access-list bytes

Use the **netbios access-list bytes** global configuration command to define the offset and hexadecimal patterns with which to match byte offsets in NetBIOS packets. Use the **no** form of this command to remove an entire list or the entry specified with the *pattern* argument.

```
netbios access-list bytes name {permit | deny} offset pattern  
no netbios access-list bytes name {permit | deny} offset pattern
```

### Syntax Description

<i>name</i>	Name of the access list being defined.
<b>permit</b>	Permits the condition.
<b>deny</b>	Denies the condition.
<i>offset</i>	Decimal number indicating the number of bytes into the packet where the byte comparison should begin. An offset of zero points to the very beginning of the NetBIOS header. Therefore, the NetBIOS delimiter string (0xFFEF), for example, begins at offset 2.
<i>pattern</i>	Hexadecimal string of digits representing a byte pattern. The argument <i>pattern</i> must conform to certain conventions. These conventions are listed under the “Usage Guidelines” section.

### Default

No offset or pattern is defined.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

For offset pattern matching, the byte pattern must be an even number of hexadecimal digits in length.

The byte pattern must be no more than 16 bytes (32 hexadecimal digits) in length.

As with all access lists, the NetBIOS access lists are scanned in order.

You can specify a wildcard character in the byte string indicating that the value of that byte does not matter in the comparison. This is done by specifying two asterisks (\*\*) in place of digits for that byte. For example, the following command would match 0xababcd, 0xab00cd, and so on.

```
netbios access-list bytes marketing permit 3 0xab**cd
```

### Examples

The following example shows how to configure for offset pattern matching:

```
netbios access-list bytes marketing permit 3 0xabcd
```

In the following example, the byte pattern would not be accepted because it must be an even number of hexadecimal digits.:

```
netbios access-list bytes marketing permit 3 0xabc
```

In the following example, the byte pattern would not be permitted because the byte pattern is longer than 16 bytes in length:

```
netbios access-list bytes marketing permit 3 00112233445566778899aabbccddeeff00
```

The following example would match 0xabaacd, 0xab00cd, and so on:

```
netbios access-list bytes marketing permit 3 0xab**cd
```

The following example deletes the entire marketing NetBIOS access list named *marketing*:

```
no netbios access-list bytes marketing
```

The following example removes a single entry from the list:

```
no netbios access-list bytes marketing deny 3 0xab**cd
```

In the following example, the first line serves to deny all packets with a byte pattern starting in offset 3 of 0xab. However, this denial would also include the pattern 0xabcd because the entry permitting the pattern 0xabcd comes *after* the first entry:

```
netbios access-list bytes marketing deny 3 0xab
netbios access-list bytes marketing permit 3 0xabcd
```

## Related Commands

**netbios input-access-filter bytes**

**netbios output-access-filter bytes**

## netbios access-list host

Use the **netbios access-list host** global configuration command to assign the name of the access list to a station or set of stations on the network. The NetBIOS station access list contains the station name to match, along with a permit or deny condition. Use the **no** form of this command to remove either an entire list or just a single entry from a list, depending upon the argument given for *pattern*.

```
netbios access-list host name {permit | deny} pattern
no netbios access-list host name {permit | deny} pattern
```

### Syntax Description

<i>name</i>	Name of the access list being defined.
<b>permit</b>	Permits the condition.
<b>deny</b>	Denies the condition.
<i>pattern</i>	A set of characters. The characters can be the name of the station, or a combination of characters and pattern-matching symbols that establish a pattern for a set of NetBIOS station names. This combination can be especially useful when stations have names with the same characters, such as a prefix. The table in the “Usage Guidelines” section explains the pattern-matching symbols that can be used.

### Default

No access list is assigned.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Table 13 explains the pattern-matching characters that can be used.

**Table 13 Station Name Pattern-Matching Characters**

Character	Description
*	Used to match any character or string of characters. Can only be used as the last character in the name.
?	Matches any single character. If this wildcard is used as the first letter of the name, you must precede it with a CNTL-V key sequence. Otherwise it will be interpreted by the router as a request for help.

## Examples

The following example specifies a full station name to match:

```
netbios access-list host marketing permit ABCD
```

The following example specifies a prefix where the pattern matches any name beginning with the characters DEFG:

```
!The string DEFG itself is included in this condition.  
netbios access-list host marketing deny DEFG*
```

The following example permits any station name with the letter W as the first character and the letter Y as the third character in the name. The second and fourth character in the name can be any character. This example would allow stations named WXYZ and WAYB; however, stations named WY and WXY would not be allowed because the ? must match specific characters in the name.

```
netbios access-list host marketing permit W?Y?
```

The following example illustrates how to combine wildcard characters. In this example the marketing list denies any name beginning with AC that is not at least three characters in length (the ? would match any third character). The string ACBD and ACB would match, but the string AC would not:

```
netbios access-list host marketing deny AC?
```

In the following example, a single entry in the marketing NetBIOS access list is removed:

```
no netbios access-list host marketing deny AC?*
```

In the following example, the entire marketing NetBIOS access list is removed:

```
no netbios access-list host marketing
```

## Related Commands

**netbios input-access-filter host**

**netbios output-access-filter host**

## netbios enable-name-cache

Use the **netbios enable-name-cache** interface configuration command to enable NetBIOS name caching. Use the **no** form of this command to disable the name-cache behavior.

**netbios enable-name-cache**  
**no netbios enable-name-cache**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

This command enables the NetBIOS name cache on the specified interface. By default the name cache is disabled for the interface. Proxy explorers must be enabled on any interface that is using the NetBIOS name cache.

### Example

The following example enables NetBIOS name caching for Token Ring interface 0:

```
interface tokenring 0
source-bridge proxy-explorer
netbios enable-name-cache
```

### Related Commands

**clear netbios-cache**  
**netbios name-cache timeout**  
**show netbios-cache**

## netbios input-access-filter bytes

Use the **netbios input-access-filter bytes** interface configuration command to define a byte access list filter on incoming messages. The actual access filter byte offsets and patterns used are defined in one or more **netbios-access-list bytes** commands. Use the **no** form of this command with the appropriate name to remove the entire access list.

```
netbios input-access-filter bytes name  
no netbios input-access-filter bytes name
```

### Syntax Description

*name* Name of a NetBIOS access filter previously defined with one or more of the **netbios access-list bytes** global configuration commands.

### Default

No access list is defined.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example applies a previously-defined filter named marketing to packets coming into tokenring 1:

```
interface tokenring 1  
!  
netbios input-access-filter bytes marketing
```

### Related Commands

```
netbios access-list bytes  
netbios input-access-filter bytes
```

## netbios input-access-filter host

Use the **netbios input-access-filter host** interface configuration command to define a station access list filter on incoming messages. The access lists of station names are defined in **netbios access-list host** commands. Use the **no** form of this command with the appropriate argument to remove the entire access list.

**netbios input-access-filter host** *name*  
**no netbios input-access-filter host** *name*

### Syntax Description

*name* Name of a NetBIOS access filter previously defined with one or more of the **netbios access-list host** global configuration commands.

### Default

No access list is defined.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example filters packets coming into tokenring1 using the NetBIOS access list named *marketing*:

```
interface tokenring 1
netbios access-list host marketing permit W?Y?
netbios input-access-filter host marketing
```

### Related Commands

**netbios access-list host**  
**netbios output-access-filter host**

## netbios name-cache

Use the **netbios name-cache** global configuration command to define a static NetBIOS name cache entry, tying the server with the name *netbios-name* to the *mac-address*, and specifying that the server is accessible either locally via the *interface-name* specified, or remotely, via the **ring-group** *group-number* specified. Use the **no** form of this command to remove the entry.

```
netbios name-cache mac-address netbios-name {interface-name | ring-group group-number}  
no netbios name-cache mac-address netbios-name
```

### Syntax Description

<i>mac-address</i>	The MAC address.
<i>netbios-name</i>	Server name linked to the MAC address.
<i>interface-name</i>	Name of the interface by which the server is accessible locally.
<b>ring-group</b>	Specifies that the link is accessible remotely.
<i>group-number</i>	Number of the ring group by which the server is accessible remotely. This ring group number must match the number you have specified with the <b>source-bridge ring-group</b> command. The valid range is 1 to 4095.

### Default

No entry is defined.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

To specify an entry in the static name cache, first specify a Routing Information Field (RIF) that leads to the server's MAC address. The Cisco IOS software displays an error message if it cannot find a static RIF entry for the server when the NetBIOS name-cache entry is attempted or if the server's type conflicts with that given for the static RIF entry.

---

**Note** The names are case sensitive. Therefore "Cc" is not the same as "cC".

---

### Examples

The following example indicates the syntax usage of this command if the NetBIOS server is accessed locally:

```
source-bridge ring-group 2
rif 0220.3333.4444 00c8.042.0060 tokenring 0
netbios name-cache 0220.3333.4444 DEF tokenring 0
```

The following example indicates the syntax usage of this command if the NetBIOS server is accessed remotely:

```
source-bridge ring-group 2
rif 0110.2222.3333 0630.021.0030 ring group 2
netbios name-cache 0110.2222.3333 DEF ring-group 2
```

### Related Command

**show netbios-cache**

## netbios name-cache name-len

Use the **netbios name-cache name-len** global configuration command to specify how many characters of the NetBIOS type name the name cache will validate.

**netbios name-cache name-len** *length*

### Syntax Description

*length* Length of the NetBIOS type name. The range is 8 to 16 characters.

### Default

15 characters

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example specifies that the name cache will validate 16 characters of the NetBIOS type name:

```
netbios name-cache name-len 16
```

### Related Commands

**netbios enable-name-cache**

**netbios name-cache**

**netbios name-cache proxy-datagram**

**netbios name-cache query-timeout**

**netbios name-cache recognized-timeout**

**netbios name-cache timeout**

## netbios name-cache proxy-datagram

Use the **netbios name-cache proxy-datagram** global configuration command to enable the Cisco IOS software to act as a proxy and send NetBIOS datagram type frames.

**netbios name-cache proxy-datagram** *seconds*

### Syntax Description

<i>seconds</i>	Time interval, in seconds, that the software forwards a route broadcast datagram type packet. The valid range is any number greater than 0.
----------------	---

### Default

There is no default time interval.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example specifies that the software will forward a NetBIOS datagram type frame in 20-second intervals:

```
netbios name-cache proxy-datagram 20
```

### Related Commands

- netbios enable-name-cache**
- netbios name-cache**
- netbios name-cache query-timeout**
- netbios name-cache recognized-timeout**
- netbios name-cache timeout**

## netbios name-cache query-timeout

Use the **netbios name-cache query-timeout** global configuration command to specify the “dead” time, in seconds, that starts when a host sends any ADD\_NAME\_QUERY, ADD\_GROUP\_NAME, or STATUS\_QUERY frame. During this dead time, the Cisco IOS software drops any repeat, duplicate ADD\_NAME\_QUERY, ADD\_GROUP\_NAME, or STATUS\_QUERY frame sent by the same host. This timeout is only effective at the time of the login negotiation process. Use the **no** form of this command to bring the time back to the default of 6 seconds.

**netbios name-cache query-timeout** *seconds*  
**no netbios name-cache query-timeout**

### Syntax Description

*seconds* “Dead” time period in seconds. Default is 6 seconds.

### Default

6 seconds

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example sets the timeout to 15 seconds:

```
netbios name-cache query-timeout 15
```

### Related Command

**netbios name-cache recognized-timeout**

## netbios name-cache recognized-timeout

Use the **netbios name-cache recognized-timeout** global configuration command to specify the “dead” time, in seconds, that starts when a host sends any FIND\_NAME or NAME\_RECOGNIZED frame. During this dead time, the Cisco IOS software drops any repeat, duplicate FIND\_NAME or NAME\_RECOGNIZED frame sent by the same host. This timeout is only effective at the time of the login negotiation process. Use the **no** form of this command to bring the time back to the default of 6 seconds.

**netbios name-cache recognized-timeout** *seconds*  
**no netbios name-cache recognized-timeout**

### Syntax Description

*seconds* “Dead” time period in seconds. Default is 6 seconds.

### Default

6 seconds

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example sets the timeout to 15 seconds:

```
netbios name-cache recognized-timeout 15
```

### Related Command

**netbios name-cache query-timeout**

## netbios name-cache timeout

Use the **netbios name-cache timeout** global configuration command to enable NetBIOS name caching and to set the time that entries can remain in the NetBIOS name cache. Use the **no** form of this command to bring the time back to the default of 15 minutes.

**netbios name-cache timeout** *minutes*  
**no netbios name-cache timeout** *minutes*

### Syntax Description

*minutes* Time, in minutes, that entries can remain in the NetBIOS name cache. Once the time expires, the entry will be deleted from the cache. Default is 15 minutes.

### Default

15 minutes

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

This command allows you to establish NetBIOS name caching. NetBIOS name caching can be used only between routers that are running Software Release 9.1 or later. NetBIOS name-caching does not apply to static entries.

### Example

The following example sets the timeout to 10 minutes:

```
interface tokenring 0
 netbios name-cache timeout 10
```

### Related Command

**show netbios-cache**

## netbios output-access-filter bytes

Use the **netbios output-access-filter bytes** interface configuration command to define a byte access list filter on outgoing messages. Use the **no** form of this command to remove the entire access list.

**netbios output-access-filter bytes** *name*  
**no netbios output-access-filter bytes** *name*

### Syntax Description

*name* Name of a NetBIOS access filter previously defined with one or more of the **netbios access-list bytes** global configuration commands.

### Default

No access list is defined.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example filters packets leaving tokenring 1 using the NetBIOS access list named *engineering*:

```
interface tokenring 1
netbios access-list bytes engineering permit 3 0xabcd
netbios output-access-filter bytes engineering
```

### Related Commands

**netbios access-list bytes**  
**netbios input-access-filter bytes**

## netbios output-access-filter host

Use the **netbios output-access-filter host** interface configuration command to define a station access list filter on outgoing messages. Use the **no** form of this command to remove the entire access list.

```
netbios output-access-filter host name  
no netbios output-access-filter host name
```

### Syntax Description

*name* Name of a NetBIOS access filter previously defined with one or more of the **netbios access-list host** global configuration commands.

### Default

No access list filter is defined.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example filters packets leaving tokenring 1 using the NetBIOS access list named *engineering*:

```
interface tokenring 1  
netbios access-list host engineering permit W?Y?  
netbios output-access-filter host engineering
```

### Related Commands

```
netbios access-list host  
netbios input-access-filter host
```

## rif

Use the **rif** global configuration command to enter static source-route information into the Routing Information Field (RIF) cache. If a Token Ring host does not support the use of IEEE 802.2 TEST or XID datagrams as explorer packets, you may need to add static information to the RIF cache of the router. Use the **no** form of this command to remove an entry from the cache.

```
rif mac-address rif-string {interface-name | ring-group ring}
no rif mac-address {interface-name | ring-group ring}
```

### Syntax Description

<i>mac-address</i>	12-digit hexadecimal string written as a dotted triplet; for example, 0010.0a00.20a6.
<i>rif-string</i>	Series of 4-digit hexadecimal numbers separated by a period (.). This RIF string is inserted into the packets sent to the specified MAC address.
<i>interface-name</i>	Interface name (for example, tokenring 0) that indicates the origin of the RIF.
<b>ring-group</b>	Specifies the origin of the RIF is a ring group.
<i>ring</i>	Ring group number that indicates the origin of the RIF. This ring group number must match the number you have specified with the <b>source-bridge ring-group</b> command. The valid range is 1 to 4095.

### Default

No static source-route information is entered.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

You must specify either an interface name or a ring group number to indicate the origin of the RIF. You specify an interface name (for example, tokenring 0) with the *interface-name* argument, and you specify a ring group number with the **ring-group ring** argument. The ring group number must match the number you specified with the **source-bridge ring-group** command. Ring groups are explained in the “Configuring Source-Route Bridging” chapter of the *Bridging and IBM Networking Configuration Guide*.

Using the command **rif mac-address** without any other arguments puts an entry into the RIF cache indicating that packets for this MAC address should not have RIF information.

Do not configure a static RIF with any of the *all rings* type codes. Doing so causes traffic for the configured host to appear on more than one ring and leads to unnecessary congestion.

---

**Note** Input to the **source-bridge** interface configuration command is in decimal format. RIF displays and input are in hexadecimal format, and IBM source-route bridges use hexadecimal for input. It is essential that bridge and ring numbers are consistent for proper network operation. This means you must explicitly declare the numbers to be hexadecimal by preceding the number with 0x, or you must convert IBM hexadecimal numbers to a decimal equivalent when entering them. For example, IBM hexadecimal bridge number 10 would be entered as hexadecimal number 0x10 or decimal number 16 in the configuration commands. In the displays, these commands always will be in decimal.

---

### Example

The following example configuration sets up a static RIF:

```
! insert entry with MAC address 1000.5A12.3456 and RIF of
! 0630.0081.0090 into RIF cache
rif 1000.5A12.3456 0630.0081.0090 tokenring 0
```

### Related Commands

**multiring**

**source-bridge ring-group**

## rif timeout

Use the **rif timeout** global configuration command to determine the number of minutes an inactive Routing Information Field (RIF) entry is kept. RIF information is maintained in a cache whose entries are aged. Use the **no** form of this command to restore the default.

**rif timeout** *minutes*  
**no rif timeout**

### Syntax Description

*minutes*                      Number of minutes an inactive RIF entry is kept. The value must be greater than 0. Default is 15 minutes.

### Default

15 minutes

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

A RIF entry is cached based on the MAC address and the interface.

A RIF entry can be aged out even if there is active traffic, but the traffic is fast or autonomously switched.

A RIF entry is refreshed only if a RIF field of an incoming frame is identical to the RIF information of the RIF entry in the cache.

Until a RIF entry is removed from the cache, no new information is accepted for that RIF entry.

### Example

The following example changes the timeout period to 5 minutes:

```
rif timeout 5
```

### Related Commands

**clear rif-cache**  
**rif validate-enable**  
**show rif**

## **rif validate-age**

Use the **rif validate-age** global configuration command to define the validation time when the Cisco IOS software is acting as a proxy for NetBIOS NAME\_QUERY packet or for explorer frames.

**rif validate-age** *seconds*

### Syntax Description

*seconds*

Interval, in seconds, at which a proxy is sent. The valid range is any number greater than 0. Default is 2 seconds.

### Default

2 seconds

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

If the timer expires before the response is received, the RIF entry or the NetBIOS cache entry is marked as invalid and is flushed from the cache table when another explorer or NAME\_QUERY packet is received.

### Example

The following example specifies the interval at which a proxy is sent to be 3 seconds:

```
rif validate-age 3
```

### Related Commands

**rif**  
**rif timeout**

## rif validate-enable

Use the **rif validate-enable** global configuration command to enable RIF validation for entries learned on an interface (Token Ring or FDDI). Use the **no** form of this command to disable the specification.

**rif validate-enable**  
**no rif validate-enable**

### Syntax Description

This command has no arguments or keywords.

### Default

RIF validation is enabled.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.0.

A RIF validation algorithm is used for the following cases:

- To decrease convergence time to a new source route path when an intermediate bridge goes down.
- To keep a valid RIF entry in a RIF cache even if a RIF entry is not refreshed either because traffic is fast or autonomously switched, or because there is no traffic.

A directed IEEE TEST command is sent to the destination MAC address. If a response received in the time specified by **rif validate-age**, the entry is refreshed and is considered valid. Otherwise, the entry is removed from the cache. To prevent sending too many TEST commands, any entry that has been refreshed in less than 70 seconds is considered valid.

Validation is triggered as follows:

- When a RIF entry is found in the cache.
- When a RIF field of an incoming frame and the RIF information of the RIF entry is not identical. If, as the result of validation, the entry is removed from the cache, the RIF field of the next incoming frame with the same MAC address is cached.
- When the RIF entry is not refreshed for the time specified in the **rif timeout** command.

---

**Note** If the RIF entry has been in the RIF cache for 6 hours, and has not been refreshed for the time specified in the **rif timeout** command, the entry is removed unconditionally from the cache.

---

---

**Note** The **rif validate enable** commands have no effect on remote entries learned over RSRB.

---

### Example

The following example enables RIF validation:

```
rif validate-enable
```

### Related Commands

**rif timeout**

**rif validate-age**

**rif validate-enable-age**

**rif validate-enable-route-cache**

## **rif validate-enable-age**

Use the **rif validate-enable-age** global configuration command to enable RIF validation for stations on a source-route bridge network that do not respond to an IEEE TEST command. Use the **no** form of this command to disable the specification.

**rif validate-enable-age**  
**no rif validate-enable-age**

### Syntax Description

This command has no arguments or keywords.

### Default

RIF validation is enabled.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.0.

You must first issue the **rif validate-enable** command.

When this command is enabled, a RIF entry is not removed from the cache even if it becomes invalid. If the entry is refreshed, it becomes valid again.

If a RIF field of an incoming frame and the RIF information of the invalid RIF entry are not identical, the old RIF information is replaced by the new information.

---

**Note** The **rif validate enable** commands have no effect on remote entries learned over RSRB.

---

### Example

The following example enables RIF validation:

```
rif validate-enable-age
```

### Related Command

**rif validate-enable**

## **rif validate-enable-route-cache**

Use the **rif validate-enable-route-cache** global configuration command to enable synchronization of the RIF cache with the protocol route cache. Use the **no** form of this command to disable the specification.

**rif validate-enable-route-cache**  
**no rif validate-enable-route-cache**

### Syntax Description

This command has no arguments or keywords.

### Default

This command is disabled by default.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.0.

When a RIF entry is removed from the RIF cache, or the RIF information in the RIF entry is changed, the protocol route caches are synchronized with the RIF cache.

---

**Note** The rif validate enable commands have no effect on remote entries learned over RSRB.

---

### Example

The following example synchronizes the RIF cache with the protocol route cache:

```
rif validate-enable-route-cache
```

### Related Command

**rif validate-enable**

## show controllers token

Use the **show controllers token** privileged EXEC command to display information about memory management, error counters, and the board itself. Depending on the board being used, the output can vary. This command also displays proprietary information. Thus, the information that **show controllers token** displays is of primary use to our technical personnel. Information that is useful to users can be obtained with the **show interfaces tokenring** command, described later.

### show controllers token

#### Syntax Description

This command has no arguments or keywords.

#### Command Mode

Privileged EXEC

#### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

#### Sample Display

The following is sample output from the **show controllers token** command of a CSC-IR or CSC-2R card:

```
Router# show controllers token

TR Unit 0 is board 0 - ring 0

state 3, dev blk: 0x1D2EBC, mailbox: 0x2100010, sca: 0x2010000
  current address: 0000.3080.6f40, burned in address: 0000.3080.6f40
  current TX ptr: 0xBA8, current RX ptr: 0x800

Last Ring Status: none

Stats: soft:0/0, hard:0/0, sig loss:0/0
      tx beacon: 0/0, wire fault 0/0, recovery: 0/0
      only station: 0/0, remote removal: 0/0
Bridge: local 3330, bnum 1, target 3583
      max_hops 7, target idb: 0x0, not local
Interface failures: 0 -- Bkgnd Ints: 0
TX shorts 0, TX giants 0

Monitor state: (active)
  flags 0xC0, state 0x0, test 0x0, code 0x0, reason 0x0
f/w ver: 1.0, chip f/w: '000000.ME31100', [bridge capable]
SMT form of this command s: 1.01 kernel, 4.02 fastmac
ring mode: F00, internal enables: SRB REM RPS CRS/NetMgr
internal functional: 0000011A (0000011A), group: 00000000 (00000000)
if_state: 1, ints: 0/0, ghosts: 0/0, bad_states: 0/0
t2m fifo purges: 0/0
t2m fifo current: 0, t2m fifo max: 0/0, proto_errs: 0/0
ring: 3330, bridge num: 1, target: 3583, max hops: 7
```

```

Packet counts:
  receive total: 298/6197, small: 298/6197, large 0/0
    runts: 0/0, giants: 0/0
    local: 298/6197, bridged: 0/0, promis: 0/0
    bad rif: 0/0, multiframe: 0/0
  ring num mismatch 0/0, spanning violations 0
  transmit total: 1/25, small: 1/25, large 0/0
    runts: 0/0, giants: 0/0, errors 0/0
bad fs: 0/0, bad ac: 0
congested: 0/0, not present: 0/0
  Unexpected interrupts: 0/0, last unexp. int: 0

  Internal controller counts:
line errors: 0/0, internal errors: 0/0
burst errors: 0/0, ari/fci errors: 0/0
abort errors: 0/0, lost frame: 0/0
copy errors: 0/0, rcvr congestion: 0/0
token errors: 0/0, frequency errors: 0/0
dma bus errors: -/-, dma parity errors: -/-
  Internal controller smt state:
Adapter MAC: 0000.3080.6f40, Physical drop: 00000000
NAUN Address: 0000.a6e0.11a6, NAUN drop: 00000000
Last source: 0000.a6e0.11a6, Last poll: 0000.3080.6f40
Last MVID: 0006, Last attn code: 0006
Txmit priority: 0006, Auth Class: 7FFF
Monitor Error: 0000, Interface Errors: FFFF
Correlator: 0000, Soft Error Timer: 00C8
Local Ring: 0000, Ring Status: 0000
Beacon rcv type: 0000, Beacon txmit type: 0000
Beacon type: 0000, Beacon NAUN: 0000.a6e0.11a6

```

Table 14 describes the fields shown in the first line of sample output.

**Table 14 Show Controllers Token Field Descriptions—Part 1**

Field	Description
TR Unit 0	Unit number assigned to the Token Ring interface associated with this output.
is board 0	Board number assigned to the Token Ring controller board associated with this interface.
ring 0	Number of the Token Ring associated with this board.

In the following line, state 3 indicates the state of the board. The rest of this output line displays memory mapping that is of primary use to our engineers.

```
state 3, dev blk: 0x1D2EBC, mailbox: 0x2100010, sca: 0x2010000
```

The following line also appears in **show interface token** output as the address and burned in address (bia), respectively:

```
current address: 0000.3080.6f40, burned in address: 0000.3080.6f40
```

The following line displays buffer management pointers that change by board:

```
current TX ptr: 0xBA8, current RX ptr: 0x800
```

The following line indicates the ring status from the controller chip set. This information is used by LAN Network Manager:

```
Last Ring Status: none
```

The following line displays Token Ring statistics. See the Token Ring specification for more information:

```
Stats: soft:0/0, hard:0/0, sig loss:0/0
      tx beacon: 0/0, wire fault 0/0, recovery: 0/0
      only station: 0/0, remote removal: 0/0
```

The following line indicates that Token Ring communication has been enabled on the interface. If this line of output appears, the message “Source Route Bridge capable” should appear in the **show interfaces tokenring** display.

```
Bridge: local 3330, bnum 1, target 3583
```

Table 15 describes the fields shown in the following line of sample output:

```
max_hops 7, target idb: 0x0, not local
```

**Table 15 Show Controllers Token Field Descriptions—Part 2**

Field	Description
max_hops 7	Maximum number of bridges.
target idb: 0x0	Destination interface definition.
not local	Interface has been defined as a remote bridge.

The following line is specific to the hardware:

```
Interface failures: 0 -- Bkgnd Ints: 0
```

In the following line, TX shorts are the number of packets the interface transmits that are discarded because they are smaller than the medium’s minimum packet size. TX giants are the number of packets the interface transmits that are discarded because they exceed the medium’s maximum packet size.

```
TX shorts 0, TX giants 0
```

The following line indicates the state of the controller. Possible values include active, failure, inactive, and reset.

```
Monitor state: (active)
```

The following line displays detailed information relating to the monitor state shown in the previous line of output. This information relates to the firmware on the controller. This information is relevant to our engineers only if the monitor state is something other than active.

```
flags 0xC0, state 0x0, test 0x0, code 0x0, reason 0x0
```

Table 16 describes the fields in the following line or output:

```
f/w ver: 1.0 expr 0, chip f/w: '000000.ME31100', [bridge capable]
```

**Table 16 Show Controllers Token Field Descriptions—Part 3**

Field	Description
f/w ver: 1.0	Version of our firmware on the board.
chip f/w: '000000.ME31100'	Firmware on the chip set.
[bridge capable]	Interface has not been configured for bridging, but it has that capability.

The following line displays the version numbers for the kernel and the accelerator microcode of the Madge firmware on the board; this firmware is the LLC interface to the chip set:

```
SMT form of this command s: 1.01 kernel, 4.02 fastmac
```

The following line displays LAN Network Manager information that relates to ring status:

```
ring mode: F00, internal enables: SRB REM RPS CRS/NetMgr
```

The following line corresponds to the functional address and the group address shown in **show interfaces tokenring** output:

```
internal functional: 0000011A (0000011A), group: 00000000 (00000000)
```

The following line displays interface board state information that is proprietary:

```
if_state: 1, ints: 0/0, ghosts: 0/0, bad_states: 0/0
```

The following lines display information that is proprietary. Our engineers use this information for debugging purposes:

```
t2m fifo purges: 0/0
t2m fifo current: 0, t2m fifo max: 0/0, proto_errs: 0/0
```

Each of the fields in the following line maps to a field in the **show source bridge** display, as follows: ring maps to srn; bridge num maps to bn; target maps to trn; and max hops maps to max:

```
ring: 3330, bridge num: 1, target: 3583, max hops: 7
```

In the following lines of output, the number preceding the slash (/) indicates the count since the value was last displayed; the number following the slash (/) indicates count since the system was last booted:

```
Packet counts:
receive total: 298/6197, small: 298/6197, large 0/0
```

In the following line, the number preceding the slash (/) indicates the count since the value was last displayed; the number following the slash (/) indicates count since the system was last booted. The runts and giants values that appear here correspond to the runts and giants values that appear in **show interfaces tokenring** output:

```
runts: 0/0, giants: 0/0
```

The following lines are receiver-specific information that our engineers can use for debugging purposes:

```
local: 298/6197, bridged: 0/0, promis: 0/0
bad rif: 0/0, multiframe: 0/0
ring num mismatch 0/0, spanning violations 0
transmit total: 1/25, small: 1/25, large 0/0
runts: 0/0, giants: 0/0, errors 0/0
```

## show controllers token

---

The following lines include very specific statistics that are not relevant in most cases, but exist for historical purposes. In particular, the internal errors, burst errors, ari/fci, abort errors, copy errors, frequency errors, dma bus errors, and dma parity errors fields are not relevant.

```
Internal controller counts:
line errors: 0/0, internal errors: 0/0
burst errors: 0/0, ari/fci errors: 0/0
abort errors: 0/0, lost frame: 0/0
copy errors: 0/0, rcvr congestion: 0/0
token errors: 0/0, frequency errors: 0/0
dma bus errors: -/-, dma parity errors: -/-
```

The following lines are low-level Token Ring interface statistics relating to the state and status of the Token Ring with respect to all other Token Rings on the line:

```
Internal controller smt state:
Adapter MAC:      0000.3080.6f40, Physical drop:      00000000
NAUN Address:    0000.a6e0.11a6, NAUN drop:          00000000
Last source:     0000.a6e0.11a6, Last poll:          0000.3080.6f40
Last MVID:       0006, Last attn code:              0006
Txmit priority:  0006, Auth Class:                   7FFF
Monitor Error:   0000, Interface Errors:           FFFF
Correlator:      0000, Soft Error Timer:            00C8
Local Ring:      0000, Ring Status:              0000
Beacon rcv type: 0000, Beacon txmit type:          0000
```

## show interfaces tokenring

Use the **show interfaces tokenring** privileged EXEC command to display information about the Token Ring interface and the state of source-route bridging.

**show interfaces tokenring** [*number*]

### Syntax Description

*number* (Optional) Interface number. If you do not provide a value, the command will display statistics for all Token Ring interfaces.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Sample Display

The following is sample output from the **show interfaces tokenring** command:

```
Router# show interfaces tokenring

TokenRing 0 is up, line protocol is up
Hardware is 16/4 Token Ring, address is 5500.2000.dc27 (bia 0000.3000.072b)
  Internet address is 150.136.230.203, subnet mask is 255.255.255.0
  MTU 8136 bytes, BW 16000 Kbit, DLY 630 usec, rely 255/255, load 1/255
  Encapsulation SNAP, loopback not set, keepalive set (10 sec)
  ARP type: SNAP, ARP Timeout 4:00:00
  Ring speed: 16 Mbps
  Single ring node, Source Route Bridge capable
  Group Address: 0x00000000, Functional Address: 0x60840000
  Last input 0:00:01, output 0:00:01, output hang never
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  Five minute input rate 0 bits/sec, 0 packets/sec
  Five minute output rate 0 bits/sec, 0 packets/sec
  16339 packets input, 1496515 bytes, 0 no buffer
    Received 9895 broadcasts, 0 runts, 0 giants
      0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    32648 packets output, 9738303 bytes, 0 underruns
  0 output errors, 0 collisions, 2 interface resets, 0 restarts
  5 transitions
```

Table 17 describes significant fields shown in the display.

**Table 17 Show Interfaces Tokenring Field Descriptions**

Field	Description
Token Ring is up/down	Interface is currently active and inserted into ring (up) or inactive and not inserted (down).
Token Ring is Reset	Hardware error has occurred. This is not in the sample output; it is informational only.
Token Ring is Initializing	Hardware is up, in the process of inserting the ring. This is not in the sample output; it is informational only.
Token Ring is Administratively Down	Hardware has been taken down by an administrator. This is not in the sample output; it is informational only. "Disabled" indicates the Cisco IOS software has received over 5000 errors in a keepalive interval, which is 10 seconds by default.
line protocol is { up   down   administratively down }	Indicates whether the software processes that handle the line protocol believe the interface is usable (that is, whether keepalives are successful).
Hardware	Specifies the hardware type. "Hardware is ciscoBus Token Ring" indicates that the board is a CSC-C2CTR board. "Hardware is 16/4 Token Ring" indicates that the board is a CSC-1R, CSC-2R, or a CSC-R16M board. Also shows the address of the interface.
Internet address	Lists the Internet address followed by subnet mask.
MTU	Maximum transmission unit of the interface.
BW	Bandwidth of the interface in kilobits per second.
DLY	Delay of the interface in microseconds.
rely	Reliability of the interface as a fraction of 255 (255/255 is 100 percent reliability), calculated as an exponential average over 5 minutes.
load	Load on the interface as a fraction of 255 (255/255 is completely saturated), calculated as an exponential average over 5 minutes.
Encapsulation	Encapsulation method assigned to interface.
loopback	Indicates whether loopback is set or not.
keepalive	Indicates whether keepalives are set or not.
ARP type	Type of Address Resolution Protocol assigned.
Ring speed	Speed of Token Ring—4 or 16 Mbps.
{ Single ring   multiring node }	Indicates whether a node is enabled to collect and use source routing information (RIF) for routable Token Ring protocols.
Group Address	Interface's group address, if any. The group address is a multicast address; any number of interfaces on the ring may share the same group address. Each interface may have at most one group address.
Functional Address	
Last input	Number of hours, minutes, and seconds since the last packet was successfully received by an interface. Useful for knowing when a dead interface failed.

**Table 17 Show Interfaces Tokenring Field Descriptions (Continued)**

<b>Field</b>	<b>Description</b>
output hang	Number of hours, minutes, and seconds (or never) since the interface was last reset because of a transmission that took too long. When the number of hours in any of the “last” fields exceeds 24 hours, the number of days and hours is printed. If that field overflows, asterisks are printed.
Output queue, drops Input queue, drops	Number of packets in output and input queues. Each number is followed by a slash, the maximum size of the queue, and the number of packets dropped due to a full queue.
Five minute input rate, Five minute output rate	Average number of bits and packets transmitted per second in the last 5 minutes.
packets input	Total number of error-free packets received by the system.
broadcasts	Total number of broadcast or multicast packets received by the interface.
runts	Number of packets that are discarded because they are smaller than the medium’s minimum packet size.
giants	Number of packets that are discarded because they exceed the medium’s maximum packet size.
CRC	Cyclic Redundancy Checksum generated by the originating LAN station or far-end device does not match the checksum calculated from the data received. On a LAN, this usually indicates noise or transmission problems on the LAN interface or the LAN bus itself. A high number of CRCs is usually the result of a station transmitting bad data.
frame	Number of packets received incorrectly having a CRC error and a noninteger number of octets.
overrun	Number of times the serial receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver’s ability to handle the data.
ignored	Number of received packets ignored by the interface because the interface hardware ran low on internal buffers. These buffers are different than the system buffers mentioned previously in the buffer description. Broadcast storms and bursts of noise can cause the ignored count to be increased.
packets output	Total number of messages transmitted by the system.
bytes	Total number of bytes, including data and MAC encapsulation, transmitted by the system.
underruns	Number of times that the far-end transmitter has been running faster than the near-end router’s receiver can handle. This may never be reported on some interfaces.
output errors	Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, as some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories.
collisions	Since a Token Ring cannot have collisions, this statistic is nonzero only if an unusual event occurred when frames were being queued or dequeued by the system software.

**Table 17 Show Interfaces Tokenring Field Descriptions (Continued)**

<b>Field</b>	<b>Description</b>
interface resets	Number of times an interface has been reset. The interface may be reset by the administrator or automatically when an internal error occurs.
Restarts	Should always be zero for Token Ring interfaces.
transitions	Number of times the ring made a transition from up to down, or vice versa. A large number of transitions indicates a problem with the ring or the interface.

## show lnm bridge

Use the **show lnm bridge** privileged EXEC command to display all currently configured bridges and all parameters that are related to the bridge as a whole, not to one of its interfaces.

**show lnm bridge**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Sample Display

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

```
Router# show lnm bridge

Bridge 001-2-003, Ports 0000.3000.abc4, 0000.0028.abcd
Active Links: 0000.0000.0000 0000.0000.0000 0000.0000.0000 0000.0000.0000
Notification: 0 min, Threshold 00.10%
```

Table 18 describes significant fields shown in the display.

**Table 18 Show LNM Bridge Field Descriptions**

Field	Description
Bridge 001-2-003	Ring and bridge numbers of this bridge.
Ports 0000.3000.abc4....	MAC addresses of the two interfaces of this bridge.
Active Links:	Any LNM stations that are currently connected to this bridge. An entry preceded by an asterisk is the controlling LNM.
Notification: 0 min	Current counter notification interval in minutes.
Threshold 00.10%	Current loss threshold that will trigger a message to LNM.

## show lnm config

Use the **show lnm config** privileged EXEC command to display the logical configuration of all bridges configured in a router. This information is needed to configure an LNM Management Station to communicate with a router. This is especially important when the router is configured as a multiport bridge, thus employing the concept of a virtual ring.

### **show lnm config**

#### Syntax Description

This command has no arguments or keywords.

#### Command Mode

Privileged EXEC

#### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

#### Sample Displays

The following is sample output from the **show lnm config** command for a simple two-port bridge:

```
Router# show lnm config

Bridge(s) currently configured:

      From    ring 001, address 0000.3000.abc4
      Across  bridge 002
      To      ring 003, address 0000.0028.abcd
```

The following is sample output from the **show lnm config** command for a multiport bridge:

```
Router# show lnm config

Bridge(s) currently configured:

      From    ring 001, address 0000.0028.abc4
      Across  bridge 001
      To      ring 008, address 4000.0028.abcd

      From    ring 002, address 0000.3000.abc4
      Across  bridge 002
      To      ring 008, address 4000.3000.abcd

      From    ring 003, address 0000.3000.5735
      Across  bridge 003
      To      ring 008, address 4000.3000.5735
```

Table 19 describes significant fields shown in the display.

**Table 19 Show LNM Config Field Descriptions**

<b>Field</b>	<b>Description</b>
From ring 001	Ring number of the first interface in the two-port bridge.
address 0000.3000.abc4	MAC address of the first interface in the two-port bridge.
Across bridge 002	Bridge number assigned to this bridge.
To ring 003	Ring number of the second interface in the two-port bridge.
address 0000.0028.abcd	MAC address of the second interface in the two-port bridge.

## show lnm interface

Use the **show lnm interface** privileged EXEC command to display all LNM-related information about a specific interface or all interfaces.

**show lnm interface** [*type number*]

### Syntax Description

*type* (Optional) Interface type.

*number* (Optional) Interface number.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

This command is for all types of interfaces, including Token Ring interfaces. If you want information specific to Token Ring, use the **show lnm ring** command.

### Sample Display

The following is sample output from the **show lnm interface** command:

```
router# show lnm interface
nonisolating error counts
interface ring Active Monitor SET dec lost cong. fc freq.token
TokenRing1 0001* 1000.5a98.23a0 00200 00001 00000 00000 00000 0000000002

Notification flags: FE00, Ring Intensive: FFFF, Auto Intensive: FFFF
Active Servers: LRM LBS REM RPS CRS
Last NNIN: never, from 0000.0000.0000.
Last Claim: never, from 0000.0000.0000.
Last Purge: never, from 0000.0000.0000.
Last Beacon: never, 'none' from 0000.0000.0000.
Last MonErr: never, 'none' from 0000.0000.0000.

isolating error counts
station int ring loc. weight line inter burst ac abort
1000.5a98.23a0 T1 0001 0000 00 - N00000 00000 00000 00000 00000
1000.5a98.239e T1 0001 0000 00 - N00000 00000 00000 00000 00000
1000.5a6f.bc15 T1 0001 0000 00 - N00000 00000 00000 00000 00000
0000.3000.abc4 T1 0001 0000 00 - N00000 00000 00000 00000 00000
1000.5a98.239f T1 0001 0000 00 - N00000 00000 00000 00000 00000
```

Table 20 describes significant fields shown in the display. See the **show lnm station** command for a description of the fields in the bottom half of the sample output.

**Table 20 Show LNM Interface Field Descriptions**

Field	Description
interface	Interface about which information was requested.
ring	Number assigned to that Token Ring. An asterisk following the ring number indicates that there are stations with nonzero error counters present on that ring.
Active Monitor	Address of the station that is currently providing “Active Monitor” functions to the ring. The description of this server can be found in the <i>IBM Token Ring Architecture Reference Manual</i> .
SET	Current soft error reporting time for the ring in units of tens of milliseconds.
dec	Rate at which the various counters of nonisolating errors are being decreased. This number is in errors per 30 seconds.
other nonisolating error counts: lost, cong., fc, and freq.token	Current values of the five nonisolating error counters specified in the 802.5 specification. These are Lost Frame errors, Receiver Congestion errors, FC errors, Frequency errors, and Token errors.
Notification flags:	Representation of which types of ring errors are being reported to LNM. The description of this number can be found in the <i>IBM Token Ring Architecture Reference Manual</i> .
Ring Intensive:	Representation of which specific ring error messages are being reported to LNM when in the “Ring Intensive” reporting mode. The description of this number can be found in the <i>IBM Token Ring Architecture Reference Manual</i> .
Auto Intensive:	Representation of which specific ring error messages are being reported to LNM when in the “Auto Intensive” reporting mode. The description of this number can be found in the <i>IBM Token Ring Architecture Reference Manual</i> .
Active Servers:	A list of which servers are currently active on this Token Ring. The possible acronyms and their meanings are as follows: <ul style="list-style-type: none"> <li>• CRS—Configuration Report Server</li> <li>• LRM—LAN Reporting Manager</li> <li>• LBS—LAN Bridge Server</li> <li>• REM—Ring Error Monitor</li> <li>• RPS—Ring Parameter Server</li> </ul> The description of these servers can be found in the <i>IBM Token Ring Architecture Reference Manual</i> .
Last NNIN:	Time since the last “Neighbor Notification Incomplete” frame was received, and the station that sent this message.
Last Claim:	Time since the last “Claim Token” frame was received, and the station that sent this message.
Last Purge:	Time since the last “Purge Ring” frame was received, and the station that sent this message.
Last Beacon:	Time since the last “Beacon” frame was received, the type of the last beacon frame, and the station that sent this message.
Last Mon Err:	Time since the last “Report Active Monitor Error” frame was received, the type of the last monitor error frame, and the station that sent this message.

Related Commands

**show Inm ring**

**show Inm station**

## show lnm ring

Use the **show lnm ring** privileged EXEC command to display all LNM information about a specific Token Ring or all Token Rings. If a specific interface is requested, it also displays a list of all currently active stations on that interface.

```
show lnm ring [ring-number]
```

### Syntax Description

*ring-number* (Optional) Number of a specific Token Ring. It can be a value in the range 1 to 4095.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The output of this command is the same as the output of the **show lnm interface** command. See the **show lnm interface** and **show lnm station** commands for sample output and a description of the fields. The same information can be obtained by using the **show lnm interface** command, but instead of specifying an interface number, you specify a ring number as an argument.

### Related Commands

**show lnm interface**

**show lnm station**

## show lnm station

Use the **show lnm station** privileged EXEC command to display LNM-related information about a specific station or all known stations on all rings. If a specific station is requested, it also displays a detailed list of that station's current MAC-level parameters.

**show lnm station** [*address*]

### Syntax Description

*address* (Optional) Address of a specific LNM station.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.0.

10.0.

### Sample Display

The following is sample output from the **show lnm station** command when a particular address (in this case, 1000.5abc15) has been specified:

```
Router# show lnm station 1000.5a6f.bc15

station      int  ring  loc.  weight  isolating error counts
1000.5a6f.bc15  T1  0001  0000  00 - N  line inter burst  ac  abort

Unique ID: 0000.0000.0000      NAUN: 0000.3000.abc4
Functional: C000.0000.0000      Group: C000.0000.0000
Physical Location: 00000        Enabled Classes: 0000
Allowed Priority: 00000         Address Modifier: 0000
Product ID: 00000000.00000000.00000000.00000000.0000
Ucode Level: 00000000.00000000.0000
Station Status: 00000000.0000
Last transmit status: 00
```

Table 21 describes significant fields shown in the display.

**Table 21 Show LNM Station Field Descriptions**

Field	Description
station	MAC address of the given station on the Token Ring.
int	Interface used to reach the given station.
ring	Number of the Token Ring where the given station is located.
loc.	Physical location number of the given station.

**Table 21 Show LNM Station Field Descriptions (Continued)**

<b>Field</b>	<b>Description</b>
weight	Weighted accumulation of the errors of the given station, and of its NAUN. The three possible letters and their meanings are as follows: <sup>1</sup> <ul style="list-style-type: none"> <li>• N—not in a reported error condition.</li> <li>• P—in a “pre-weight” error condition.</li> <li>• W—in a “pre-weight” error condition.</li> </ul>
isolating error counts	Current values of the five isolating error counters specified in the 802.5 specification. These are Line errors, Internal errors, Burst errors, AC errors, and Abort errors.
<b>Values below this point will be zero unless LNM has previously requested this information.</b>	
Unique ID:	Uniquely assigned value for this station.
NAUN:	MAC address of this station’s “upstream” neighbor.
Functional:	MAC-level functional address currently in use by this station.
Group:	MAC-level group address currently in use by this station.
Physical Location:	Number assigned to this station as its “Physical Location” identifier.
Enabled Classes:	Functional classes that the station is allowed to transmit.
Allowed Priority:	Maximum access priority that the station may use when transmitting onto the Token Ring.
Address Modifier:	Reserved field.
Product ID:	Encoded 18-byte string used to identify what hardware/software combination is running on this station.
Ucode Level:	10-byte EBCDIC string indicating the microcode level of the station.
Station Status:	Implementation-dependent vector that is not specified anywhere.
Last transmit status:	Contains the strip status of the last “Report Transmit Forward” MAC frame forwarded by this interface.

1. The description of these error conditions can be found in the *IBM Architecture Reference Manual*.

## show netbios-cache

Use the **show netbios-cache** privileged EXEC command to display a list of NetBIOS cache entries.

**show netbios-cache**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Sample Display

The following is sample output from the **show netbios-cache** command:

```
Router# show netbios-cache

      HW Addr          Name           How      Idle      NetBIOS Packet Savings
1000.5a89.449a        IC6W06_B      TR1       6         0
1000.5a8b.14e5        IC_9Q07A      TR1       2         0
1000.5a25.1b12        IC9Q19_A      TR1       7         0
1000.5a25.1b12        IC9Q19_A      TR1      10         0
1000.5a8c.7bb1        BKELSA1       TR1       4         0
1000.5a8b.6c7c        ICELSB1       TR1       -         0
1000.5a31.df39        ICASC_01      TR1       -         0
1000.5ada.47af        BKELSA2       TR1      10         0
1000.5a8f.018a        ICELSC1       TR1       1         0
```

Table 22 describes significant fields shown in the display.

**Table 22 Show NetBIOS-Cache Field Descriptions**

Field	Description
HW Addr	MAC address mapped to the NetBIOS name in this entry.
Name	NetBIOS name mapped to the MAC address in this entry.
How	Interface through which this information was learned.
Idle	Period of time (in seconds) since this entry was last accessed. A hyphen in this column indicates it is a static entry in the NetBIOS name cache.
NetBIOS Packet Savings	Number of packets to which local replies were made (thus preventing transmission of these packets over the network).

### Related Commands

**netbios name-cache**  
**netbios name-cache timeout**

## show rif

Use the **show rif** privileged EXEC command to display the current contents of the RIF cache.

```
show rif
```

### Syntax Description

This command has no arguments or keywords.

### Command Mode

Privileged EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Sample Display

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

```
Router# show rif

Codes: * interface, - static, + remote
Hardware Addr How Idle (min) Routing Information Field
5C02.0001.4322 rg5 - 0630.0053.00B0
5A00.0000.2333 TR0 3 08B0.0101.2201.0FF0
5B01.0000.4444 - - -
0000.1403.4800 TR1 0 -
0000.2805.4C00 TR0 * -
0000.2807.4C00 TR1 * -
0000.28A8.4800 TR0 0 -
0077.2201.0001 rg5 10 0830.0052.2201.0FF0
```

In the display, entries marked with an asterisk (\*) are the router's interface addresses. Entries marked with a dash (-) are static entries. Entries with a number denote cached entries. If the RIF timeout is set to something other than the default of 15 minutes, the timeout is displayed at the top of the display.

Table 23 describes significant fields shown in the display.

**Table 23 Show RIF Field Descriptions**

Field	Description
Hardware Addr	Lists the MAC-level addresses.
How	Describes how the RIF has been learned. Possible values include a ring group (rg), or interface (TR).
Idle (min)	Indicates how long, in minutes, since the last response was received directly from this node.
Routing Information Field	Lists the RIF.

### Related Command

**multiring**

## show source-bridge

Use the **show source-bridge** privileged EXEC command to display the current source bridge configuration and miscellaneous statistics.

### show source-bridge

#### Syntax Description

This command has no arguments or keywords.

#### Command Mode

Privileged EXEC

#### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

#### Sample Display

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

```
Router# show source-bridge

Local Interfaces:max receive transmit
srn bn trn r p s n hp cnt:bytes cnt:bytes drops
TR0 5 1 10 * * 7 39:1002 23:62923

Ring Group 10:
This peer: TCP 150.136.92.92
Maximum output TCP queue length, per peer: 100
Peers:
TCP 150.136.92.92 - 2 0 0 0 0 0
TCP 150.136.93.93 open 2* 18 18 3 0 0
Rings:
bn: 1 rn: 5 local ma: 4000.3080.844b TokenRing0 fwd: 18
bn: 1 rn: 2 remote ma: 4000.3080.8473 TCP 150.136.93.93 fwd: 36

Explorers: ----- input ----- ----- output -----
spanning all-rings total spanning all-rings total
TR0 0 3 3 3 5 8
Router#
```

Table 24 describes significant fields shown in the display.

**Table 24 Show Source-Bridge Field Descriptions**

Field	Description
Local Interfaces:	Description of local interfaces.
max	Maximum routing descriptor length.
receive	Packets: bytes received on interface for source bridging.
transmit	Packets: bytes transmitted on interface for source bridging.
srn	Ring number of this Token Ring.
bn	Bridge number of this router, for this ring.

**Table 24 Show Source-Bridge Field Descriptions (Continued)**

Field	Description
trn	Group in which the interface is configured. (The target ring number, or virtual ring group.)
r	Ring group is assigned. An asterisk (*) in this field indicates that ring group has been assigned for this interface.
p	Interface can respond with proxy explorers. An asterisk (*) in this field indicates the interface can respond to proxy explorers.
s	Spanning-tree explorers enabled on the interface. An asterisk (*) indicates, that this interface will forward spanning-tree explorers.
n	Interface has NetBIOS name caching enabled. An asterisk (*) in this field indicates the interface has NetBIOS name caching enabled.
hp	Indicates hops.
Ring Group <i>n</i> :	Describes ring group <i>n</i> , where <i>n</i> is the number of the ring group.
This peer:	Address and address type of this peer.
Maximum output TCP queue length, per peer:	Maximum number of packets queued up on this peer before the Cisco IOS software starts dropping packets.
Peers:	Addresses and address types of the ring group peers.
state	Current state of the peer, open or closed. A hyphen indicates this router.
lv	Indicates form of this command of remote source-route bridge. The “l” indicates local acknowledgment.
pkts_rx	Lists the number of packets received.
pkts_tx	Lists the number of packets transmitted.
expl_gn	Lists the explorers generated.
drops	Lists the number of dropped packets.
TCPq	Lists the current TCP backup queue length.
Rings:	Describes the ring groups. Information displayed includes the bridge groups, ring groups, whether the group is local or remote, the MAC address, the network address or interface type, and the number of packets forwarded. A type shown as “locvrt” indicates a local virtual ring used by SDLLC or SR/TLB; a type shown as “remvrt” indicates a remote virtual ring used by SDLLC or SR/TLB.
Explorers:	This section describes the explorer packets that the Cisco IOS software has transmitted and received.
input	Explorers received by Cisco IOS software.
output	Explorers generated by Cisco IOS software.
TR0	Interface on which explorers were received.
spanning	Spanning-tree explorers.
all-rings	All-rings explored.
total	Summation of spanning and all-rings.

## show span

Use the **show span** EXEC command to display the spanning-tree topology known to the router.

**show span**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

### Sample Display

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

```
RouterA> show span

Bridge Group 1 is executing the IBM compatible spanning tree protocol
  Bridge Identifier has priority 32768, address 0000.0c0c.f68b
  Configured hello time 2, max age 6, forward delay 4
  Current root has priority 32768, address 0000.0c0c.f573
  Root port is 001A (TokenRing0/0), cost of root path is 16
  Topology change flag not set, detected flag not set
  Timers: hold 1, topology change 30, notification 30
          hello 2, max age 6, forward delay 4, aging 300
  Timers: hello 0, topology change 0, notification 0
Port 001A (TokenRing0/0) of bridge group 1 is forwarding. Path cost 16
  Designated root has priority 32768, address 0000.0c0c.f573
  Designated bridge has priority 32768, address 0000.0c0c.f573
  Designated port is 001B, path cost 0, peer 0
  Timers: message age 1, forward delay 0, hold 0
Port 002A (TokenRing0/1) of bridge group 1 is blocking. Path cost 16
  Designated root has priority 32768, address 0000.0c0c.f573
  Designated bridge has priority 32768, address 0000.0c0c.f573
  Designated port is 002B, path cost 0, peer 0
  Timers: message age 0, forward delay 0, hold 0
Port 064A (spanRSRB) of bridge group 1 is disabled. Path cost 250
  Designated root has priority 32768, address 0000.0c0c.f573
  Designated bridge has priority 32768, address 0000.0c0c.f68b
  Designated port is 064A, path cost 16, peer 0
  Timers: message age 0, forward delay 0, hold 0
```

A port (spanRSRB) is created with each virtual ring group. The port will be disabled until one or more peers go into open state in the ring group.

## show sse summary

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

**show sse summary**

### Syntax Description

This command has no arguments or keywords.

### Command Mode

EXEC

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.0.

### Sample Display

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

```
Router# show sse summary

SSE utilization statistics

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

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

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

## source-bridge

Use the **source-bridge** interface configuration command to configure an interface for source-route bridging. Use the **no** form of this command to disable source bridging on a particular interface.

**source-bridge** *local-ring bridge-number target-ring*  
**no source-bridge**

### Syntax Description

<i>local-ring</i>	Ring number for this interface's Token Ring or FDDI ring. It must be a decimal number between 1 and 4095 that uniquely identifies a network segment or ring within the bridged Token Ring or FDDI network.
<i>bridge-number</i>	Number that uniquely identifies the bridge connecting the local and target rings. It must be a decimal number between 1 and 15.
<i>target-ring</i>	Decimal ring number of the destination ring on this router. It also must be unique within the bridged Token Ring or FDDI network. The target ring can also be a ring group.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The parser automatically displays the word "active" in the **source-bridge** command in configurations that have source-route bridging enabled. You do not need to enter the **source-bridge command** with an **active** keyword.

### Examples

In the following example, Token Rings 129 and 130 are connected via a router:

```
interface tokenring 0
source-bridge 129 1 130
!
interface tokenring 1
source-bridge active 130 1 129
```

In the following example, an FDDI ring on one router is connected to a Token Ring on a second router across a DLSw+ link.

```
dlsw local-peer peer-id 132.11.11.2
dlsw remote-peer 0 tcp 132.11.11.3
interface fddi 0
no ip address
multiring all
```

```
source-bridge active 26 1 10

dlsw local peer peer-id 132.11.11.3
dlsw remote-peer 0 tcp 132.11.11.2
interface tokenring 0
no ip address
multiring all
source-bridge active 25 1 10
```

### Related Commands

**source-bridge ring-group**  
**source-bridge transparent**

## source-bridge connection-timeout

Use the **source-bridge connection-timeout** global configuration command to establish the interval of time between first attempt to open a connection until a timeout is declared. Use the **no** form of this command to disable this feature.

**source-bridge connection-timeout** *seconds*  
**no source-bridge connection-timeout** *seconds*

### Syntax Description

*seconds* Interval of time, in seconds, before a connection attempt to a remote peer is aborted.

### Default

The default connection-timeout interval is 10 seconds.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

The **source-bridge connection-timeout** command is used for setting timeout intervals in a complex topology such as a large multihop WAN with virtual rings or satellite links. The timeout interval is used when a connection to a remote peer is attempted. If the timeout interval expires before a response is received, the connection attempt is aborted.

### Example

The following example sets the connection timeout interval to 60 seconds:

```
source-bridge connection-timeout 60
```

### Related Commands

**source-bridg ring-group**

## source-bridge enable-80d5

Use the **source-bridge enable-80d5** global configuration command to change the router's Token Ring to Ethernet translation behavior. Use the **no** form of this command to disable this function.

**source-bridge enable-80d5**  
**no source-bridge enable-80d5**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The Cisco IOS software supports two types of Token Ring LLC2 to Ethernet conversion. They are as follows:

- Token Ring LLC2 to Ethernet 802.3 LLC2
- Token Ring LLC2 to Ethernet 0x80d5

Use this global configuration command to change the translation behavior. By default, the Cisco IOS software translates Token Ring LLC2 to Ethernet 802.3 LLC2. This command allows you to configure the software to translate Token Ring LLC2 frames into Ethernet 0x80d5 format frames.

This command is useful when you have a non-IBM device attached to an IBM network with devices that are using the nonstandard Token Ring LLC2 to Ethernet 80d5 translation. If you do not configure your router to enable 80d5 processing, the non-IBM and IBM devices will not be able to communicate.

The parameters specifying the current parameters for the processing of 0x80d5 frames are given at the end of the output of the **show span** command.

---

**Note** The 80d5 frame processing option is available only with SR/TLB. It is not available when source-route transparent bridging (SRT) is used.

---

Use the **show span** to check whether 80d5 processing is enabled. If it is, the following line displays in the output:

```
Translation between LLC2 and Ethernet Type II 80d5 is enabled
```

### Example

The following example enables 0x80d5 processing, removes the translation for SAP 08, and adds the translation for SAP 1c:

```
source-bridge enable-80d5
no source-bridge sap-80d5 08
source-bridge sap-80d5 1c
```

### Related Commands

**show span**

**source-bridge sap-80d5**

## source-bridge explorer-dup-ARE-filter

Use the **source-bridge explorer-dup-ARE-filter** global configuration command to prevent excessive forwarding of explorers in networks with redundant topologies. Use the **no** form of this command to disable this feature.

```
source-bridge explorer-dup-ARE-filter  
no source-bridge explorer-dup-ARE-filter
```

### Syntax Description

This command has no arguments or keywords.

### Default

Duplicate explorer filtering is disabled.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

### Example

The following example enables duplicate explorer filtering:

```
source-bridge explorer-dup-ARE-filter
```

## source-bridge explorer-fastswitch

Use the **source-bridge explorer-fastswitch** global configuration command to enable explorer fast switching. To disable explorer fast switching, use the **no** form of this command.

**source-bridge explorer-fastswitch**  
**no source-bridge explorer-fastswitch**

### Syntax Description

This command has no arguments or keywords.

### Default

Fast switching is enabled.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

Use the **no** form of this command in conjunction with the **source-bridge explorerq-depth** and the **source-bridge explorer-maxrate** command to optimize explorer processing.

### Example

The following example enables explorer fast switching after it has been previously disabled:

```
source-bridge explorer-fastswitch
```

### Related Commands

**source-bridge explorer-maxrate**  
**source-bridge explorerq-depth**

## source-bridge explorer-maxrate

Use the **source-bridge explorer-maxrate** global configuration command to set the maximum byte rate of explorers per ring. To reset the default rate, use the **no** form of this command.

**source-bridge explorer-maxrate** *maxrate*  
**no source-bridge explorer-maxrate** *maxrate*

### Syntax Description

*maxrate*                      Number in the range 100 to 1000000000 (in bytes per second). The default maximum byte rate is 38400 bytes per second.

### Default

The default maximum byte rate is 38400 bytes per second.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

Given the number of different explorer packet types and sizes and the bandwidth limits of the various interfaces, the bus data rate (as opposed to the packet rate) is the common denominator used to decide when to flush incoming explorers. The packets are dropped by the interface before any other processing.

### Example

The following command sets the maximum byte rate of explorers on a ring:

```
source-bridge explorer-maxrate 100000
```

## source-bridge explorerq-depth

Use the **source-bridge explorerq-depth** global configuration command to set the maximum explorer queue depth. To reset the default value, use the **no** form of this command.

```
source-bridge explorerq-depth depth  
no source-bridge explorerq-depth depth
```

### Syntax Description

*depth*                    The maximum number of incoming packets. The valid range is 1 to 500.

### Default

The default maximum depth is 30.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

In this implementation, the limit is on a per-interface basis such that each interface can have up to the maximum (default 30) outstanding packets on the queue before explorers from that particular interface are dropped.

### Example

The following example sets the maximum explorer queue depth:

```
source-bridge explorerq-depth 100
```

## source-bridge input-address-list

Use the **source-bridge input-address-list** interface configuration command to apply an access list to an interface configured for source-route bridging. This command filters source-routed packets received from the router interface based upon the source mac address. Use the **no** form of this command to remove the application of the access list.

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

### Syntax Description

*access-list-number*                      Number of the access list. The value must be in the range 700 to 799.

### Default

No access list is assigned.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

### Example

The following example assigns access list 700 to Token Ring 0:

```
interface tokenring 0
source-bridge input-address-list 700
!
access-list 700 deny 1000.5A00.0000 8000.00FF.FFFF
access-list 700 permit 0000.0000.0000 FFFF.FFFF.FFFF
```

### Related Commands

**access-list**  
**source-bridge output-address-list**

## source-bridge input-lsap-list

Use the **source-bridge input-lsap-list** interface configuration command to filter, on input, FDDI and IEEE 802-encapsulated packets that include the destination service access point (DSAP) and source service access point (SSAP) fields in their frame formats. The access list specifying the type codes to be filtered is given by this variation of the **source-bridge** interface configuration command.

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

### Syntax Description

*access-list-number*

Number of the access list. This access list is applied to all IEEE 802 or FDDI frames received on that interface prior to the source-routing process. Specify zero (0) to disable the filter. The value must be in the range 200 to 299.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example specifies access list 203:

```
interface tokenring 0
source-bridge input-lsap-list 203
```

### Related Commands

**access-list**

**source-bridge output-lsap-list**

## source-bridge input-type-list

Use the **source-bridge input-type-list** interface configuration command to filter SNAP-encapsulated packets on input.

**source-bridge input-type-list** *access-list-number*

### Syntax Description

*access-list-number*

Number of the access list. This access list is applied to all SNAP frames received on that interface prior to the source-routing process. Specify zero (0) to disable the application of the access list on the bridge group. The value must be in the range 200 to 299.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Use the **access list** command to specify type code when using the **source-bridge input-type-list** command.

### Example

The following example specifies access list 202:

```
interface tokenring 0
source-bridge input-type-list 202
!
access-list 202 deny 0x6000 0x0007
access-list 202 permit 0x0000 0xFFFF
```

### Related Commands

**access-list**

**source-bridge output-type-list**

## source-bridge max-hops

Use the **source-bridge max-hops** interface configuration command to control the forwarding or blocking of all-routes explorer frames received on an interface. Use the **no** form of this command to reset the count to the maximum value.

**source-bridge max-hops** *count*  
**no source-bridge max-hops**

### Syntax Description

*count* Determines the number of bridges an explorer packet can traverse. Typically, the maximum number of bridges for interoperability with IBM equipment is seven.

### Default

The maximum number of bridge hops is seven.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

Frames are forwarded only if the number of hops in the routing information field of the input frame plus hops appended by the router is less than or equal to the specified count. If the interface is connected to a destination interface, the router appends one hop. If the interface is tied to a virtual ring, the router appends two hops. This applies only to all-routes explorer frames on input to this interface.

### Example

The following example limits the maximum number of source-route bridge hops to five.

```
source-bridge max-hops 5
```

### Related Commands

**source-bridge**  
**source-bridge max-in-hops**  
**source-bridge max-out-hops**

## source-bridge max-in-hops

Use the **source-bridge max-in-hops** interface configuration command to control the forwarding or blocking of spanning-tree explorer frames received on an interface. Use the **no** form of this command to reset the count to the maximum value.

**source-bridge max-in-hops** *count*  
**no source-bridge max-in-hops**

### Syntax Description

<i>count</i>	Determines the number of bridges an explorer packet can traverse. Typically, the maximum number of bridges for interoperability with IBM equipment is seven.
--------------	--

### Default

The maximum number of bridge hops is seven.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

Frames are forwarded only if the number of hops in the routing information field of the input frame is less than or equal to the specified count. This applies only to spanning-tree explorer frames input to the specified interface.

### Example

The following example limits the maximum number of source-route bridge hops to three.

```
source-bridge max-in-hops 3
```

### Related Commands

**source-bridge**  
**source-bridge max-hops**  
**source-bridge max-out-hops**

## source-bridge max-out-hops

Use the **source-bridge max-out-hops** interface configuration command to control the forwarding or blocking of spanning-tree explorer frames sent from this interface. Use the **no** form of this command to reset the count to the maximum value.

**source-bridge max-out-hops** *count*  
**no source-bridge max-out-hops**

### Syntax Description

*count* Determines the number of bridges an explorer packet can traverse. Typically, the maximum number of bridges for interoperability with IBM equipment is seven.

### Default

The maximum number of bridge hops is seven.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

Frames are forwarded only if the number of hops in the routing information field of the frame (including the hops appended by the router) is less than or equal to the specified count. This applies only to spanning-tree explorer frames output from the specified interface.

### Example

The following example limits the maximum number of source-route bridge hops to five.

```
source-bridge max-out-hops 5
```

### Related Commands

**source-bridge**  
**source-bridge max-hops**  
**source-bridge max-in-hops**

## source-bridge output-address-list

Use the **source-bridge output-address-list** interface configuration command to apply an access list to an interface configured for source-route bridging. This command filters source-routed packets sent to the router interface based upon the destination mac address. Use the **no** form of this command to remove the application of the access list.

**source-bridge output-address-list** *access-list-number*  
**no source-bridge output-address-list** *access-list-number*

### Syntax Description

*access-list-number*                      Number of the access list. The value must be in the range 700 to 799.

### Default

No access list is assigned.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

To disallow the bridging of Token Ring packets of all IBM workstations on Token Ring 1, use this sample configuration. The software assumes that all such hosts have Token Ring addresses with the vendor code 1000.5A00.0000. (The vendor portion of the MAC address is the first three bytes (left to right) of the address. The first line of the access list denies access to all IBM workstations, while the second line permits access to all other devices on the network. Then, the access list can be assigned to the input side of Token Ring 1.

```
access-list 700 deny 1000.5A00.0000 8000.00FF.FFFF
access-list 700 permit 0000.0000.0000 FFFF.FFFF.FFFF
interface tokenring 1
source-bridge output-address-list 700
```

### Related Commands

**access-list**  
**source-bridge input-address-list**

## source-bridge output-lsap-list

Use the **source-bridge output-lsap-list** interface configuration command to filter, on output, FDDI and IEEE 802-encapsulated packets that have destination service access point (DSAP) and source service access point (SSAP) fields in their frame formats.

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

### Syntax Description

*access-list-number*

Number of the access list. This access list is applied just before sending out a frame to an interface. Specify zero (0) to disable the filter. The value must be in the range 200 to 299.

### Default

No filters are applied.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The access list specifying the type codes to be filtered is given by this command.

### Example

The following example specifies access list 251:

```
interface tokenring 0
source-bridge output-lsap-list 251
access-list 251 permit 0xE0E0 0x0101
access-list 251 deny 0x0000 0xFFFF
```

### Related Commands

**access-list**

**source-bridge input-lsap-list**

## source-bridge output-type-list

Use the **source-bridge output-type-list** interface configuration command to filter SNAP-encapsulated frames by type code on output.

**source-bridge output-type-list** *access-list-number*

### Syntax Description

*access-list-number*

Number of the access list. This access list is applied just before sending out a frame to an interface. Specify zero (0) to disable the application of the access list on the bridge group. The value must be in the range 200 to 299.

### Default

No filters are applied.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Input and output type code filtering on the same interface reduces performance and is not recommended.

Access lists for Token Ring- and IEEE 802-encapsulated packets affect only source-route bridging functions. Such access lists do not interfere with protocols that are being routed.

Use the access list specifying the types codes in this command.

### Example

The following example filters SNAP-encapsulated frames on output:

```
! apply interface configuration commands to interface tokenring 0
interface tokenring 0
! filter SNAP-encapsulated frames on output using access list 202
source-bridge output-type-list 202
!
access-list 202 deny 0x6000 0x0007
access-list 202 permit 0x0000 0xFFFF
```

### Related Commands

**access-list**

**source-bridge input-type-list**

## source-bridge proxy-explorer

Use the **source-bridge proxy-explorer** interface configuration command to configure the interface to respond to any explorer packets from a source node that meet the conditions described below. Use the **no** form of this command to cancel responding to explorer packets with proxy explorers.

**source-bridge proxy-explorer**  
**no source-bridge proxy-explorer**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

The *proxy explorer* function allows the source-route bridge interface to respond to a source node on behalf of a particular destination node. The interface responds with proxy explorers. The following conditions must be met in order for the interface to respond to a source node with proxy explorers on behalf of a destination node:

- The destination node must be in the RIF cache.
- The destination node must not be on the same ring as the source node.
- The explorer packet must be an IEEE 802.2 XID or TEST packet.
- The packet cannot be from the IBM Token Ring LAN Network Manager source SAP.

If all of the above conditions are met, the source-route bridge interface will turn the packet around, append the appropriate RIF, and reply to the source node.

Use proxy explorers to limit the amount of explorer traffic propagating through the source-bridge network, especially across low-bandwidth serial lines. The proxy explorer is most useful for multiple connections to a single node.

### Example

The following example configures the router to use proxy explorers on Token Ring 0:

```
interface tokenring 0
source-bridge proxy-explorer
```

## source-bridge proxy-netbios-only

Use the **source-bridge proxy-netbios-only** global configuration command to enable proxy explorers for the NetBIOS name-caching function. Use the **no** form of this command to disable the NetBIOS name-caching function.

**source-bridge proxy-netbios-only**  
**no source-bridge proxy-netbios-only**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example configures the router to use proxy explorers:

```
source-bridge proxy-netbios-only
```

## source-bridge ring-group

Use the **source-bridge ring-group** global configuration command to define or remove a ring group from the configuration. Use the **no** form of this command to cancel previous assignments.

```
source-bridge ring-group ring-group [virtual-mac-address]  
no source-bridge ring-group ring-group [virtual-mac-address]
```

### Syntax Description

<i>ring-group</i>	Ring group number. The valid range is 1 to 4095.
<i>virtual-mac-address</i>	(Optional) 12-digit hexadecimal string written as a dotted triplet (for example, 0010.0a00.20a6).

### Default

No ring group is defined.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

To configure a source-route bridge with more than two network interfaces, the *ring group* concept is used. A ring group is a collection of Token Ring interfaces in one or more routers that are collectively treated as a virtual ring. The ring group is denoted by a ring number that must be unique for the network. The ring group's number is used just like a physical ring number, showing up in any route descriptors contained in packets being bridged.

To configure a specific interface as part of a ring group, its target ring number parameter is set to the ring group number specified in this command. Do not use the number 0; it is reserved to represent the local ring.

To avoid an address conflict on the virtual MAC address, use a locally administered address in the form 4000.xxx.xxx.

### Example

In the following example, multiple Token Rings are source-route bridged to one another through a single router. These Token Rings are all part of ring group 7.

```
! all token rings attached to this bridge/router are part of ring group 7  
source-bridge ring-group 7  
!  
interface tokenring 0  
source-bridge 1000 1 7  
!  
interface tokenring 1  
source-bridge 1001 1 7  
!  
interface tokenring 2  
source-bridge 1002 1 7  
!
```

```
interface tokenring 3
source-bridge 1003 1 7
```

#### Related Command

**source-bridge**

## source-bridge route-cache

Use the **source-bridge route-cache** interface configuration command to enable fast switching. Use the **no** form of this command to disable fast switching.

**source-bridge route-cache**  
**no source-bridge route-cache**

### Syntax Description

This command has no arguments or keywords.

### Default

Enabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

By default, fast-switching software is enabled in the source-route bridging software. Fast switching allows for faster implementations of local source-route bridging between 4/16-Mb Token Ring cards in the same router. This feature also allows for faster implementations of local source-route bridging between two routers using the 4/16-Mb Token Ring cards and the direct interface encapsulation.

### Example

The following example disables use of fast switching between two 4/16-Mb Token Ring interfaces:

```
interface token 0
source-bridge 1 1 2
no source-bridge route-cache
!
interface token 1
source-bridge 2 1 1
no source-bridge route-cache
```

### Related Command

**source-bridge**

## source-bridge route-cache cbus

Use the **source-bridge route-cache cbus** interface configuration command to enable autonomous switching. Use the **no** form of this command to disable autonomous switching.

**source-bridge route-cache cbus**  
**no source-bridge route-cache cbus**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Autonomous switching in source-route bridging software is available for local source-route bridging between ciscoBus Token Ring (CTR) cards in the same router. Autonomous switching provides higher switching rates than does fast-switching between 4/16-MbToken Ring cards. Autonomous switching works for both two-port bridges and multiport bridges that use ciscoBus Token Ring cards.

In a virtual ring that includes both ciscoBus Token Ring and 4/16-MbToken Ring interfaces, frames that flow from one CTR interface to another are autonomously switched, and the remainder of the frames are fast switched. The switching that occurs on the CTR interface takes advantage of the high-speed ciscoBus controller processor.

---

**Note** Using either NetBIOS byte offset access lists or the access-expression capability to logically combine the access filters disables the autonomous or fast switching of SRB frames.

---

### Example

The following example enables use of autonomous switching between two ciscoBus Token Ring interfaces:

```
interface token 0
source-bridge 1 1 2
source-bridge route-cache cbus
!
interface token 1
source-bridge 2 1 1
source-bridge route-cache cbus
```

### Related Command

**source-bridge**

## source-bridge route-cache sse

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

**source-bridge route-cache sse**  
**no source-bridge route-cache sse**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

### Example

The following example enables use of SSE switching between two 4/16-Mb Token Ring interfaces:

```
interface token 0
source-bridge 1 1 2
source-bridge route-cache sse
!
interface token 1
source-bridge 2 1 1
source-bridge route-cache sse
```

### Related Command

**source-bridge**

## source-bridge sap-80d5

Use the **source-bridge sap-80d5** global configuration command to allow non-IBM hosts (attached to a router with 80d5 processing enabled) to use the standard Token Ring to Ethernet LLC2 translation instead of the nonstandard Token Ring to Ethernet 80d5 translation. This command allows you to set the translation on a per-DSAP basis. Use the **no** form of this command to disable this feature.

```
source-bridge sap-80d5 dsap  
no source-bridge sap-80d5 dsap
```

### Syntax Description

*dsap* Destination service access point (DSAP).

### Default

Enabled

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

By default, the following DSAPs are enabled for 0x80d5 translation simply by specifying the **source-bridge enable-80d5** command:

- For SNA—04, 08, 0C, 00
- For NetBIOS—F0

Any of these DSAPs can be disabled with the **no** form of this command.

The parameters specifying the current parameters for the processing of 0x80d5 frames are given at the end of the output of the **show span** command.

---

**Note** The 80d5 frame processing option is available only with SR/TLB. It is not available when source-route transparent bridging (SRT) is used.

---

Use the **show span** to check whether 80d5 processing is enabled for a particular DSAP. The following line displays in the output if 80d5 processing is enabled, listing each DSAP for which it is enabled:

```
Translation is enabled for the following DSAPs:  
04 0C 1C F0
```

### Example

The following example enables 0x80d5 processing, removes the translation for SAP 08, and adds the translation for SAP 1c:

```
source-bridge enable-80d5
no source-bridge sap-80d5 08
source-bridge sap-80d5 1c
```

### Related Commands

**show span**

**source-bridge enable-80d5**

## source-bridge spanning (automatic)

Use the automatic version of the **source-bridge spanning** interface configuration command to enable the automatic spanning-tree function for a specified group of bridged interfaces. Use the **no source-bridge spanning** command to return to the default disabled state. Use the **no source-bridge spanning path-cost** command to return an assigned path cost to the default path cost of 16.

```
source-bridge spanning bridge-group [path-cost path-cost]  
no source-bridge spanning bridge-group [path-cost path-cost]
```

### Syntax Description

<i>bridge-group</i>	Number in the range 1 to 9 that you choose to refer to a particular group of bridged interfaces. This must be the same number as assigned in the <b>bridge protocol ibm</b> command.
<b>path-cost</b>	(Optional) Assign a path cost for a specified interface.
<i>path-cost</i>	(Optional) Path cost for the interface. The valid range is 0 to 65535.

### Defaults

The automatic spanning-tree function is disabled. The default path cost is 16.

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.3.

### Example

The following example adds Token Ring 0 to bridge group 1 and assigns a path cost of 12 to Token Ring 0:

```
interface tokenring 0  
source-bridge spanning 1 path-cost 12
```

### Related Commands

**bridge protocol ibm**  
**show source-bridge**

## source-bridge spanning (manual)

Use the **source-bridge spanning** interface configuration command to enable use of spanning explorers. The **no** form of this command disables their use. Only spanning explorers will be blocked; everything else will be forwarded.

**source-bridge spanning**  
**no source-bridge spanning**

### Syntax Description

This command has no arguments or keywords.

### Default

Disabled

### Command Mode

Interface configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Use of the **source-bridge spanning** command is recommended. This command puts the interface into a forwarding or active state with respect to the spanning tree. There are two types of explorer packets used to collect RIF information:

- All-rings, all-routes explorer packets follow all possible paths to a destination ring. In a worst case scenario, the number of all-rings explorers generated may be exponentially large.
- Spanning or limited-route explorer packets follow a spanning tree when looking for paths, greatly reducing the number of explorer packets required. There is currently no dynamic spanning-tree algorithm to establish that spanning tree; it must be manually configured.

### Example

The following example enables use of spanning explorers:

```
! Global configuration command establishing the ring group for the interface
configuration commands
source-bridge ring-group 48
!
! commands that follow apply to interface token 0
interface tokenring 0
! configure interface tokenring 0 to use spanning explorers
source-bridge spanning
```

### Related Command

**source-bridge**

## source-bridge transparent

Use the **source-bridge transparent** global configuration command to establish bridging between transparent bridging and source-route bridging. Use the **no** form of this command to disable a previously established link between a source-bridge ring group and a transparent bridge group.

**source-bridge transparent** *ring-group pseudo-ring bridge-number tb-group [oui]*  
**no source-bridge transparent** *ring-group pseudo-ring bridge-number tb-group*

### Syntax Description

<i>ring-group</i>	Virtual ring group created by the <b>source-bridge ring-group</b> command. This is the source-bridge virtual ring to associate with the transparent bridge group. This ring group number must match the number you have specified with the <b>source-bridge ring-group</b> command. The valid range is 1 to 4095.
<i>pseudo-ring</i>	Ring number used to represent the transparent bridging domain to the source-route bridged domain. This number must be a unique number, not used by any other ring in your source-route bridged network.
<i>bridge-number</i>	Bridge number of the bridge that leads to the transparent bridging domain.
<i>tb-group</i>	Number of the transparent bridge group that you want to tie into your source-route bridged domain. The <b>no</b> form of this command disables this feature.
<i>oui</i>	(Optional) Organizational unique identifier. Possible values include the following: <ul style="list-style-type: none"> <li>• <b>90-compatible</b></li> <li>• <b>standard</b></li> <li>• <b>cisco</b></li> </ul>

### Default

Not established

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 10.0.

Before using this command, you must have completely configured your router using multiport source-bridging and transparent bridging.

Specify the **90-compatible** OUI when talking to our routers. This OUI provides the most flexibility. Specify the **standard** OUI when talking to IBM 8209 bridges and other vendor equipment. This OUI does not provide for as much flexibility as the other two choices. The **cisco** OUI is provided for compatibility with future equipment.

Do not use the **standard** OUI unless you are forced to interoperate with other vendor equipment, such as the IBM 8209, in providing Ethernet and Token Ring mixed media bridged connectivity. Only use the **standard** keyword when you are transferring data between IBM 8209 Ethernet/Token Ring bridges and routers running the SR/TLB software (to create a Token Ring backbone to connect Ethernets). Use of the **standard** keyword causes the OUI code in Token Ring frames to always be 0x000000. In the context of the **standard** keyword, an OUI of 0x000000 identifies the frame as an Ethernet Type II frame. If the OUI in Token Ring frame is 0x000000 SR/TLB will output an Ethernet Type II frame.

When 8209 compatibility is enabled with the **ethernet transit-oui standard** command, the SR/TLB chooses to translate all Token Ring SNAP frames into Ethernet Type II frames as described earlier in this chapter.

### Example

The following example establishes bridging between a transparent-bridge network and a source-route network:

```
source-bridge ring-group 9
source-bridge transparent 9 6 2 2
!
interface tokenring 0
source-bridge 5 2 9
interface token ring 1
source bridge 4 2 9
!
interface ethernet 0
bridge-group 2
!
interface ethernet 1
bridge-group 2

bridge 2 protocol ieee
```

### Related Commands

**bridge-group**  
**source-bridge**  
**source-bridge ring-group**

## source-bridge transparent fastswitch

Use the **source-bridge transparent fastswitch** global configuration command to enable fast switching of packets between the SRB and transparent domains. Use the **no** form of this command to disable fast switching of packets.

**source-bridge transparent *ring-group* fastswitch**  
**no source-bridge transparent *ring-group* fastswitch**

### Syntax Description

<i>ring-group</i>	Virtual ring group created by the <b>source-bridge ring-group</b> command. This is the source-bridge virtual ring to associate with the transparent bridge group. This ring group number must match the number you have specified with the <b>source-bridge ring-group</b> command. The valid range is 1 to 4095.
<b>fastswitch</b>	Fast-switched SR/TLB enables the Cisco IOS software to process packets at the interrupt level.

### Default

Fast-switched SR/TLB is enabled.

### Command Mode

Global configuration

### Usage Guidelines

This command first appeared in Cisco IOS Release 11.2.

Because fast-switched SR/TLB is enabled by default when the router is configured for SR/TLB, there are no user-specified changes to the operation of the router, and the enabling command does not appear in the configuration.

The **no source-bridge transparent *ring-group* fastswitch** command is provided to disable fast-switched SR/TLB, causing the router to handle packets by process switching. When fast-switched SR/TLB is disabled, the **no** form of the command appears on a separate line of the configuration, immediately below the parent **source-bridge transparent** command.

If fast-switch SR/TLB has been disabled, it can be enabled using the **source-bridge transparent *ring-group* fastswitch** command, but the enabling form of the command will not appear in the configuration.

### Example

The following example disables fast-switched SR/TLB between a transparent-bridge network and a source-route network:

```
source-bridge ring-group 9
source-bridge transparent 9 6 2 2
no source-bridge transparent 9 fastswitch
!
interface tokenring 0
source-bridge 5 2 9
```

```
interface token ring 1
source bridge 4 2 9
!
interface ethernet 0
bridge-group 2
!
interface ethernet 1
bridge-group 2

bridge 2 protocol ieee
```

### Related Commands

A dagger (†) indicates that the command is documented outside this chapter.

**bridge-group** †

**source-bridge**

**source-bridge ring-group**