

Troubleshoot Cisco Router Token Ring Interfaces

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Conventions](#)

[Token Ring Insertion Process](#)

[Lobe Test](#)

[Physical Insertion and Monitor Check](#)

[Duplicate Address Check](#)

[Participation in Ring Poll](#)

[Request Initialization](#)

[Troubleshooting](#)

[Flow Chart](#)

[LAN Network Manager](#)

[Use of Cisco IOS Software Commands](#)

[show interfaces tokenring](#)

[show controllers tokenring](#)

[debug token events](#)

[Ring Mode Definitions](#)

[modified open w/ option](#)

[Setting the Functional and Group Addresses](#)

[Keepalives](#)

[Use of LAN Analyzer](#)

[Related Information](#)

Introduction

This document describes some of the most common issues that cause a Cisco router Token Ring interface to fail to insert into a Token Ring. It provides a flow chart for a quick overview of the steps to take to troubleshoot the Token Ring interface. This document also discusses some of the most commonly used Cisco IOS[®] Software commands and how to use them to gather information about the Token Ring interface, to successfully troubleshoot the problem.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

The information in this document is based on these software and hardware versions:

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Conventions

For more information on document conventions, refer to the [Cisco Technical Tips Conventions](#).

Token Ring Insertion Process

In order to successfully troubleshoot Token Ring interfaces, it is important to understand the sequence of events that take place before a station joins the ring.

There are five phases through which a station proceeds, to join a ring:

1. [Lobe test](#)
2. [Physical insertion and monitor check](#)
3. [Duplicate address check](#)
4. [Participation in ring poll](#)
5. [Request initialization](#)

Lobe Test

The insertion process begins with a lobe test. This phase actually tests the transmitter and receiver of the Token Ring adapter and tests the cable between the adapter and the Multistation Access Unit (MAU). An MAU physically wraps the connection cable's transmit wire back to its receive wire. The effect is that the adapter can transmit media test MAC frames up the cable to the MAU (where it is wrapped) and back to itself. During this phase, the adapter sends lobe media test MAC frames to destination address 00-00-00-00-00-00 (with the source address of the adapter) and a Duplication Address Test (DAT) MAC frame (which contains the address of the adapter as both the source and destination) up the cable. If the lobe test passes, then phase one is complete.

Physical Insertion and Monitor Check

In phase two, a phantom current is sent to open the hub relay, once the hub relay opens the station and attaches itself to the ring. The station then checks to see if an active monitor (AM) is present by checking for any of these frames:

- Active monitor present (AMP) MAC frame
- Standby monitor present (SMP) MAC frame
- Ring purge MAC frames

If none of these frames are detected within 18 seconds, the station assumes that there is no active monitor present and it initiates the monitor contention process. Through the monitor contention process, the station with the highest MAC address becomes the active monitor. If contention is not completed within one second, the adapter fails to open. If the adapter becomes the AM and initiates a purge, and the purge process

does not complete within one second, then the adapter fails to open. If the adapter receives a beacon MAC frame or a remove station MAC frame, then the adapter fails to open.

Duplicate Address Check

As part of the duplicate address check phase, the station transmits a series of duplicate address MAC frames addressed to itself. If the station receives two frames back with the Address Recognized Indicator (ARI) and Frame Copied Indicator (FCI) set to 1, then it knows that this address is a duplicate on this ring, it detaches itself, and it reports a failure to open. This is necessary because Token Ring allows Locally Administered Addresses (LAAs), and you could end up with two adapters with the same MAC address if this check is not done. If this phase does not complete within 18 seconds, the station reports a failure and detaches itself from the ring.

Note: If there is a duplicate MAC address on another ring, which is permissible in source-route bridged Token Ring networks, this will not be detected. The duplicate address check is only locally significant.

Participation in Ring Poll

In the ring poll phase, the station learns the address of its NAUN (Nearest Active Upstream Neighbor) and makes its address known to its nearest downstream neighbor. This process creates the ring map. The station must wait until it receives an AMP or SMP frame with the ARI and FCI bits set to 0. When it does, the station flips both bits (ARI and FCI) to 1, if enough resources are available, and queues an SMP frame for transmission. If no such frames are received within 18 seconds, then the station reports a failure to open and de-inserts from the ring. If the station successfully participates in a ring poll, it proceeds into the final phase of insertion, request initialization.

Request Initialization

In the request initialization phase, the station sends four request initialization MAC frames to the functional address of the Ring Parameter Server (RPS). If there is no RPS present on the ring, the adapter uses its own default values and reports successful completion of the insertion process. If the adapter receives one of its four request initialization MAC frames back with the ARI and FCI bits set to 1, it waits two seconds for a response. If there is no response, it retransmits up to four times. At this time, if there is no response, it reports a request initialization failure and de-inserts from the ring.

This is a list of the functional addresses:

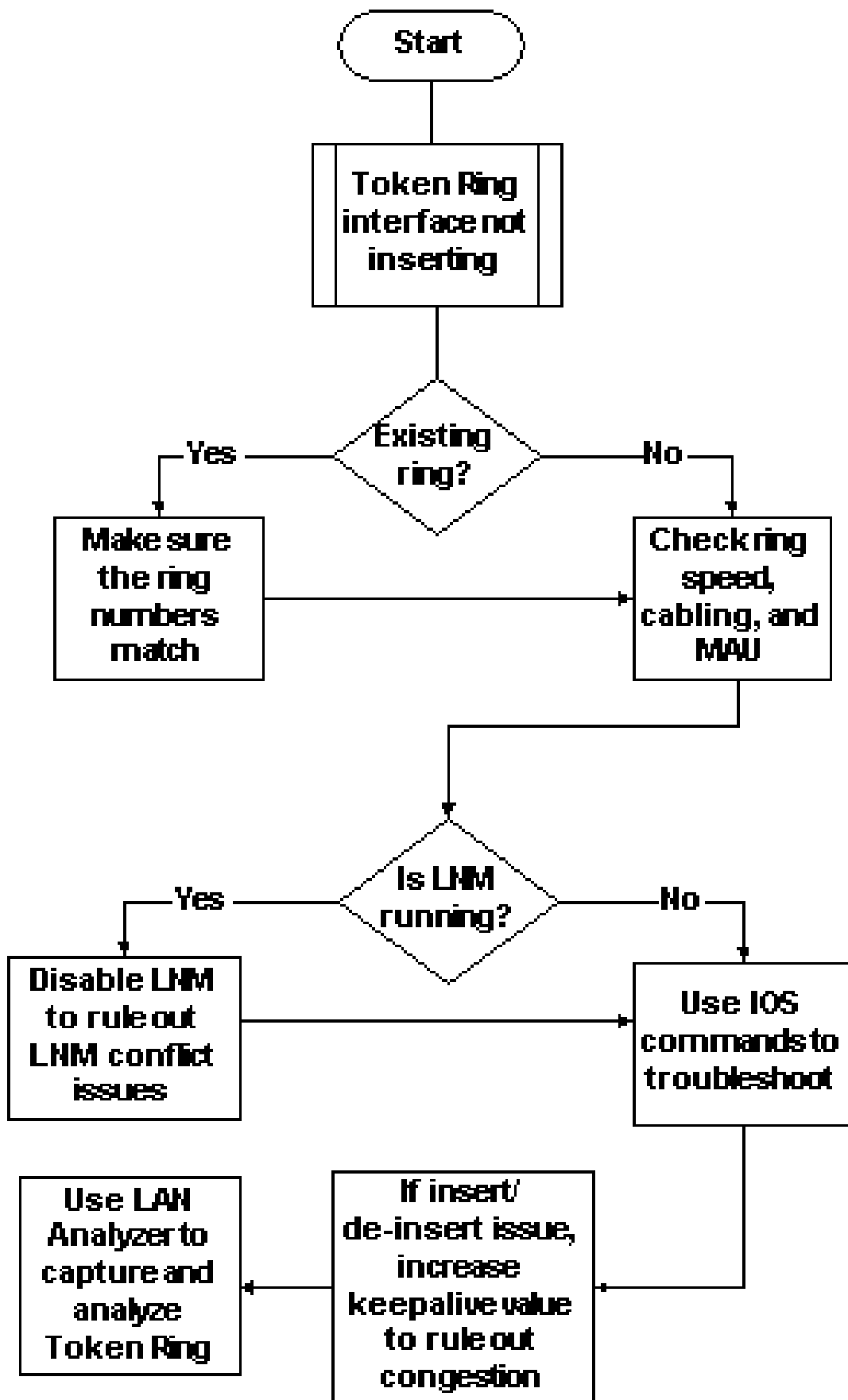
- C000.0000.0001 - Active monitor
- C000.0000.0002 - Ring Parameter Server
- C000.0000.0004 - Network Server Heartbeat
- C000.0000.0008 - Ring Error Monitor
- C000.0000.0010 - Configuration Report Server
- C000.0000.0020 - Synchronous Bandwidth Manager
- C000.0000.0040 - Locate Directory Server
- C000.0000.0080 - NetBIOS
- C000.0000.0100 - Bridge
- C000.0000.0200 - IMPL Server
- C000.0000.0400 - Ring Authorization Server
- C000.0000.0800 - LAN Gateway
- C000.0000.1000 - Ring Wiring Concentrator
- C000.0000.2000 - LAN Manager

For more information on functional addresses, refer to the IEEE802.5 specifications.

Troubleshooting

Flow Chart

Refer to this flow chart for a quick troubleshooting overview:



One of the first things that must be checked, when a Token Ring interface has problems with insertion into the ring, is whether or not you are inserting into a ring that already exists. If yes, you need to match the ring number configured on the Token Ring interface with the existing ring number governed by other Source-

Route Bridges (SRBs).

Note: Cisco routers, by default, accept ring numbers in decimal format, whereas most IBM bridges use hexadecimal notation. Therefore, make sure that you do the conversion from hexadecimal to decimal before you configure this on the Cisco router. For example, if you have an SRB with ring number 0x10, then you need to enter 16 on the Cisco router. Alternatively, you can enter the ring number on the Cisco router's Token Ring interface in hexadecimal, if you precede the ring number with 0x:

```
<#root>

turtle(config)#
interface token

turtle(config)#
interface tokenring 0

turtle(config-if)#
source

turtle(config-if)#
source-bridge 0x10 1 0x100
```

Note: When you display the configuration, the router automatically displays the ring numbers in *decimal* notation. As a result, decimal ring numbers are the most commonly used format on Cisco routers. This is the relevant part from a **show run** command:

```
<#root>

source-bridge ring-group 256
  interface TokenRing0
  no ip address
  ring-speed 16

source-bridge 16 1 256
```

!--- 16 is the physical ring number, 1 is the bridge number or ID, !--- and 256 is the Virtual Ring

```
source-bridge spanning
```

If you do not match the ring numbers, the Cisco Token Ring interface gives a message similar to this and shuts itself down:

```
02:50:25: %TR-3-BADRNGNUM: Unit 0, ring number (6) doesn't match
established number (5).
02:50:25: %LANMGR-4-BADRNGNUM: Ring number mismatch on TokenRing0,
shutting down the interface
```

```
02:50:27: %LINK-5-CHANGED: Interface TokenRing0, changed state
to administratively down
```

You then have to configure the correct ring number on the Token Ring interface???in this case, 5???and then manually issue the **no shutdown** command.

Note: The bridge number (or bridge ID) does not have to match other bridge numbers in the network; you can use a unique value or the same bridge number throughout your network as long as you have a unique Routing Information Field (RIF) path to each device in your SRB network. An example of when you would need different bridge numbers is if you have two rings connected through two parallel bridges. In this case, failure to use different bridge numbers results in two physically different paths, but the same RIF information.

Note: When you add or remove the **source-bridge** command, the Token Ring interface bounces, which causes disruption to and from this router through its Token Ring interface. For more information on how to configure SRB, refer to [Understanding and Troubleshooting Local Source-Route Bridging](#).

As well as matching ring numbers, you also need to ensure that the ring speed is set correctly; that is, 4 or 16 Mbps. Failure to do so causes the generation of a ring beacon and causes a network outage on this ring. If the ring numbers and the ring speed are set up correctly, but the Token Ring interface still fails to insert into the ring, use the process of elimination to rule out issues with cables or with the MAU. Use a wrap plug or ensure that the adapter is connected to a working MAU. Bad cabling causes many adapter problems during the insertion process. Things to look for include:

- Is the adapter configured to use the correct media port, unshielded twisted-pair (UTP) cable, or shielded twisted-pair (STP) cable?
- Is the cable that runs from the adapter to the hub complete and correct?
- What kind of media filter is in use? Keep in mind that what works at 4 Mbps does not always work at 16 Mbps.

It could be that there is a physical layer problem on the ring (for example, wiring, line noise, or jitter) which shows up as more stations insert. This causes purges and beacons, which kick off a newly inserted adapter. This can be eliminated if the Token Ring interface comes up when it is connected to another MAU with no other stations. You can then gradually add more stations to see at what point you get a failure. This test also eliminates possible conflict issues such as Active Monitor, RPS, Configuration Report Server (CRS), and others. See the [LAN Network Manager](#) section for details.

LAN Network Manager

LAN Network Manager (LNM, formerly called LAN Manager) is an IBM product that manages a collection of source-route bridges. LNM uses a version of Common Management Information Protocol (CMIP) to talk to the LNM station manager. LNM allows you to monitor the entire collection of Token Rings that comprise your source-route bridged network. You can use LNM to manage the configuration of source-route bridges, monitor Token Ring errors, and gather information from Token Ring parameter servers.

As of Cisco IOS Software Release 9.0, Cisco routers that use 4 and 16 Mbps Token Ring interfaces that are configured for SRB support the proprietary protocol that LNM uses. These routers provide all of the functions that the IBM Bridge Program currently provides. Thus, LNM can communicate with a router as if it were an IBM source-route bridge - such as the IBM 8209 - and can manage or monitor any Token Ring connected to the router, whether it be a virtual ring or physical ring. LNM is enabled on Cisco routers by default. Also, these hidden interface configuration commands are enabled by default:

- **[no] lnm crs** - The CRS monitors the current logical configuration of a Token Ring and reports any changes to LNM. CRS also reports various other events, such as the change of an active monitor on a Token Ring.
- **[no] lnm rps** - The RPS reports to LNM when any new station joins a Token Ring and ensures that all stations on a ring use a consistent set of reporting parameters.
- **[no] lnm rem** - Ring Error Monitor (REM) monitors errors that are reported by any station on the ring. In addition, REM monitors whether the ring is in a functional or failure state.

Those commands are only visible in the configuration once they have been disabled:

```
<#root>
para#
config terminal

Enter configuration commands, one per line.  End with CNTL/Z.

para(config)#
interface tokenRing 0

para(config-if)#
no lnm crs

para(config-if)#
^z
```

This is part of the Token Ring interface configuration in which the configuration is displayed:

```
<#root>

interface TokenRing0
 ip address 192.168.25.18 255.255.255.240
 no ip directed-broadcast
 ring-speed 16
 source-bridge 200 1 300
 source-bridge spanning

no lnm CRS
```

As you troubleshoot Token Ring interfaces, it might be necessary to disable CRS, RPS, REM, or all three on the Cisco router, to rule out conflict issues with other Token Ring devices. A typical scenario is when a Token Ring station fails to insert into the ring, even though the same station can insert into an isolated ring with no other stations present. You can disable individual servers, such as RPS, CRS, and REM, or disable LNM functionality on the router altogether with this global configuration:

- **lnm disabled** - This command terminates all LNM server input and reporting links. It is a superset of

the functions normally performed on individual interfaces by the **no lnm rem**, **no lnm rps**, and **no lnm rps** commands.

If you disable LNM and that solves the problem, make sure that you are not running into a known bug. If LNM is not required on your network, you may leave it disabled.

You can also make use of LNM functionality on the Cisco router to list stations that are on the local rings attached to the router, to see if there are any isolating error counts, and to see which station is sending them:

```
<#root>
para#
show lnm station

              isolating error counts
station      int   ring  loc.  weight  line inter burst ac  abort
0005.770e.0a8c To0   00C8  0000  00 - N  00000 00000 00000 00000 00000
0006.f425.ce89 To0   00C8  0000  00 - N  00000 00000 00000 00000 00000
```

Note: If you disable LNM, you can not use any of the **show lnm** commands.

From the **show lnm station** command, of particular interest is station address, ring number, and any reported errors. For a full explanation of the fields, refer to the [show lnm station](#) command in the command reference manual.

Another useful LNM command is the **show lnm interface** command:

```
<#root>
para#
show lnm interface tokenring 0

              nonisolating error counts
interface    ring
Active Monitor
  SET      dec lost cong. fc  freq. token
To0        0200
0005.770e.0a8c
  00200  00001 00000 00000 00000 00000 00000

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.
```

station	int	ring	loc.	weight	isolating error counts				
					line	inter	burst	ac	abort
0005.770e.0a8c	To0	00C8	0000	00 - N	00000	00000	00000	00000	00000
0006.f425.ce89	To0	00C8	0000	00 - N	00000	00000	00000	00000	00000

From that command, you can readily see who is the active monitor, the stations that are present on the directly connected ring, and all of the active servers on the ring (such as REM, RPS, and others).

These are the other **show lnm** command options:

```
<#root>
show lnm bridge
show lnm config
show lnm ring
```

Use of Cisco IOS Software Commands

These are the most commonly used Cisco IOS Software troubleshooting commands for Token Ring interfaces:

- [show interfaces tokenring](#)
- [show controllers tokenring](#)
- [debug token events](#)

show interfaces tokenring

These are the highlights of the **show interfaces tokenring** command:

```
<#root>
ankylo#
show interfaces tokenring1/0

TokenRing1/0 is up, line protocol is up

Hardware is IBM2692, address is 0007.78a6.a948 (bia 0007.78a6.a948)
Internet address is 1.1.1.1/24
MTU 4464 bytes, BW 16000 Kbit, DLY 630 usec,
    reliability 255/255, txload 1/255, rxload 1/255
Encapsulation SNAP, loopback not set
Keepalive set (10 sec)
ARP type: SNAP, ARP Timeout 04:00:00
Ring speed:

16 Mbps
```

```
Duplex: half
Mode: Classic token ring station
Source bridging enabled,

srn 5 bn 1 trn 100

(ring group)
spanning explorer enabled
Group Address: 0x00000000, Functional Address: 0x0800001A
Ethernet Transit OUI: 0x000000
```

```
Last Ring Status 18:15:54 <Soft Error> (0x2000)
```

```
Last input 00:00:01, output 00:00:01, output hang never
Last clearing of "show interface" counters never
Queueing strategy: fifo
```

```
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
```

```
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
27537 packets input, 1790878 bytes,
```

```
0 no buffer
```

```
Received 0 broadcasts, 0 runts, 0 giants,
```

```
0 throttles
```

```
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
```

```
7704 packets output, 859128 bytes, 0 underruns
```

```
0 output errors, 0 collisions, 2 interface resets
```

```
0 output buffer failures, 0 output buffers swapped out
```

```
1 transitions
```

[Output drops](#) can be caused when the output media can not accept frames and the output queue reaches the maximum value before it starts to drop packets. Output drops might not necessarily indicate a problem, because an explorer frame that is dropped (because it has already traveled on a particular ring) can increment the output drops counter.

Increasing [input drops](#), on the other hand, can be serious and should be carefully analyzed. Input drops can be caused by insufficient system buffers; see `0 no buffer` in the previous **show interfaces tokenring1/0** output. The incrementing `no buffer` counter of the **show interfaces** output might correlate to the incrementing `misses` counter of the **show buffers** output, and the appropriate buffer pool might need to be tuned. Refer to [Buffer Tuning for all Cisco Routers](#) for more information.

Note: Input and output queues can be increased with the [hold-queue length {in | out}](#) command; however, it is important to understand the reason why those queues are reaching their maximum hold value before you increase them. You might find that, when you increase the hold-queue maximum value, you only increases the time period before they overflow again.

You should also check the `throttles` counter. This counter indicates the number of times that the input buffers of an interface have been cleaned, because they have not been serviced fast enough or because they are overwhelmed. Typically, an explorer storm can cause the `throttles` counter to increment. Refer to the

[source-bridge explorer-maxrate](#) command and to the [Optimized Explorer Processing](#) section of [Configuring Source-Route Bridging](#).

Note: Every time you have a throttle, all of the packets in the input queue get dropped. This causes very slow performance and might also disrupt existing sessions.

A transition occurs when the interface changes its state, such as when it goes from being down to initializing or from initializing to up. A reset occurs when the interface is kick-started. The insertion of other devices into the ring should not cause either of these counters to increase, but it will cause the count of soft errors to increase. Moreover, if the **show interface tokenring** command shows no drops, input errors, or output errors, but you see a significant number of resets and transitions, then the keepalives might be resetting the interface.

Note: When you clear a token ring interface, one reset and two transitions occur: one transition from up to initializing and one from initializing to up.

The Last Ring Status field shows the last ring status for the ring. For example, 0x2000 indicates a software error. This is a list of possible status values:

```
<#root>

RNG_SIGNAL_LOSS  FIXSWAP(0x8000)
RNG_HARD_ERROR   FIXSWAP(0x4000)

RNG_SOFT_ERROR   FIXSWAP(0x2000)

RNG_BEACON       FIXSWAP(0x1000)
RNG_WIRE_FAULT   FIXSWAP(0x0800)
RNG_HW_REMOVAL   FIXSWAP(0x0400)
RNG_RMT_REMOVAL  FIXSWAP(0x0100)
RNG_CNT_OVRFLW   FIXSWAP(0x0080)
RNG_SINGLE       FIXSWAP(0x0040)
RNG_RECOVERY     FIXSWAP(0x0020)
RNG_UNDEFINED    FIXSWAP(0x021F)
RNG_FATAL        FIXSWAP(0x0d00)
RNG_AUTOFIX      FIXSWAP(0x0c00)
RNG_UNUSEABLE    FIXSWAP(0xdd00)
```

Note: Software error 0x2000 is a very common, normal ring status. 0x20 indicates ring initialization and 00 is the length of the subvector; this indicates that a ring station has entered the ring.

show controllers tokenring

The next Cisco IOS Software command to use to troubleshoot is the **show controllers tokenring** command:

```
<#root>

FEP#

show controllers tokenring 0/0

TokenRing0/0:

state up
```

current address:

0000.30ae.8200, burned in address: 0000.30ae.8200

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 100, bnum 1, target 60
max_hops 7, target idb: null
Interface failures: 0

Monitor state: (active)

, chip f/w: '000500.CS1AA5 ', [bridge capable]

ring mode: F00

, internal enables:

SRB REM RPS CRS/NetMgr

internal functional: 0800011A (0800011A)

, group: 00000000 (00000000)

internal addr: SRB: 0288, ARB: 02F6, EXB 0880, MFB: 07F4
Rev: 0170, Adapter: 02C4, Parns 01F6

Microcode counters:

MAC giants 0/0, MAC ignored 0/0
Input runts 0/0, giants 0/0, overrun 0/0
Input ignored 0/0, parity 0/0, RFED 0/0
Input REDI 0/0, null rcp 0/0, recovered rcp 0/0
Input implicit abort 0/0, explicit abort 0/0
Output underrun 0/0, TX parity 0/0, null tcp 0/0
Output SFED 0/0, SEDI 0/0, abort 0/0
Output False Token 0/0, PTT Expired 0/0

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

Internal controller smt state:

Adapter MAC: 0000.30ae.8200

, Physical drop: 00000000

NAUN Address: 0005.770e.0a87

, NAUN drop: 00000000

Last source: 0000.30ae.8200, Last poll: 0000.30ae.8200

```

Last MVID:      0006,      Last attn code:  0006
Txmit priority: 0003,      Auth Class:      7BFF
Monitor Error:  0000,      Interface Errors: 0004
Correlator:     0000,      Soft Error Timer: 00DC
Local Ring:     0000,      Ring Status:     0000
Beacon rcv type: 0000,    Beacon txmit type: 0004
Beacon type:    0000,

Beacon NAUN:    0005.770e.0a87

Beacon drop:    00000000,  Reserved:        0000
Reserved2:      0000

```

Soft errors - This is a combination of all soft errors that are seen by this interface. Soft errors include line errors, multiple monitors, ARI and FCI set errors, burst errors, lost frames, corrupted token, lost token, circulating frame or priority token, lost monitor, and frequency error. Refer to [Soft Errors Information](#) for details.

Hard errors - These are errors that are not recoverable by software routines. The ring has been physically reset. For more information, refer to [Token Ring Abnormal State List](#).

Monitor state: (active) - Indicates the state of the controller. Possible values include active, failure, inactive, and reset.

SRB REM RPS CRS/NetMgr - Indicates that SRB, REM, RPS, and CRS are all enabled on the interface. See the [LAN Network Manager](#) section for details.

Important information that is also provided in the output is the Adapter MAC and NAUN Address, which help to determine the ring topology. You can also find out who is the ring beacon NAUN; that is, the Nearest Active Upstream Neighbor to the beaconing station. This gives you a starting point to determine where the problem might lie: the beaconing station, the beacon NAUN, or the cable that lies between them. For an explanation of the rest of the fields, refer to [show controllers token](#) in the command reference manual.

debug token events

The last Cisco IOS Software command to use to troubleshoot is the **debug token events** command:

```

<#root>

1w6d: TR0 starting.
1w6d: %LINK-5-CHANGED: Interface TokenRing0,
changed state to initializing

1w6d: TR0 receive SRB_FREE, state=2, if_state=6
1w6d: TR0 receive SRB_FREE, state=2, if_state=7 ring mode = F00

1w6d: TR0:
modified open w/ option 1180

1w6d: TR0: Interface is alive, phys. addr 0000.3090.79a0

setting functional address w/ 800011A
setting group address w/ 80000000

```

```
ring mode = F00
```

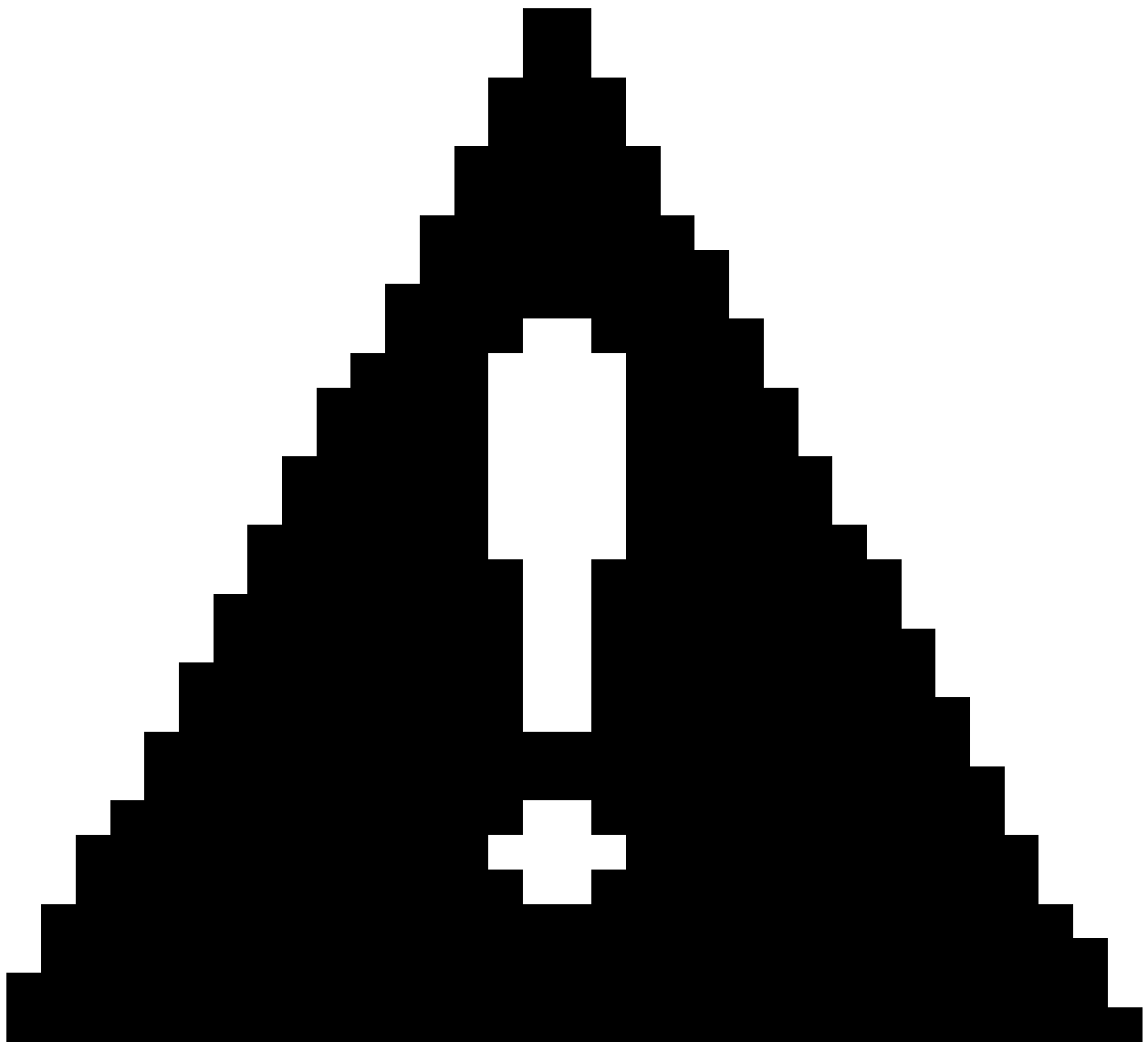
```
1w6d: TR0: modified open w/ option 1180
```

```
1w6d: %LINK-3-UPDOWN: Interface TokenRing0, changed state to up
```

```
1w6d: %LINEPROTO-5-UPDOWN:
```

```
Line protocol on Interface TokenRing0,  
changed state to up
```

```
1w6d: %SYS-5-CONFIG_I: Configured from console by console
```



Caution: **debug token events** should have a minimal impact on the router because it only displays token ring events and not packets. However, if you have a very busy ring with lots of transitions, it is recommended that you issue the **logging buffer** and the **no logging console** commands and that you have physical access to the router.

The previous **debug token events** output is from a Cisco 2500 router. The output can have a wide variety of

messages, but it should give some guidance as to where the problem might lie. In the previous example, it shows a successful initialization of the Token Ring interface. The debug also contains informational messages contained in [ring mode](#) and in [group address and functional address](#).

Ring Mode Definitions

These are values that are passed from the main system to the adapter boards, to indicate which mode the interface should use. They control whether or not certain function bits are turned on and control the command flags that are used when actually inserting into the Token Ring. For the ring mode, this is what those numbers mean:

For the previous sample debug, the ring mode is 0x0F00, which is a 2-byte value that has these meanings:

RINGMODE_LOOPBACK	0x8000	
RINGMODE_NO_RINGSTAT	0x4000	
RINGMODE_ALL_FRAMES	0x2000	
RINGMODE_ALL_LLC	0x1000	
RINGMODE_BRIDGE	0x0800	/* status only */
RINGMODE_REM	0x0400	/* be Ring Error Monitor */
RINGMODE_RPS	0x0200	/* be Ring Parameter Server */
RINGMODE_NETMGR	0x0100	/* be Configuration Report Server */
RINGMODE_TBRIDGE	0x0080	/* be a transparent bridge */
RINGMODE_CONTENDER	0x0040	/* be a contender for AMP */
RINGMODE_RS	0x0020	/* listen to ring maintenance MAC frames */
RINGMODE_ALL_MAC	0x0010	/* listen to all MAC frames */
RINGMODE_ETR	0x0008	/* Early Token Release */
RINGMODE_NEED_MAC	0x0730	/* Needs MAC frames */

The ring mode is therefore a total of those bit settings. 0xF00 indicates Bridge, Ring Error Monitor, Ring Parameter Server, and Configuration Report Server.

modified open w/ option

This is the new setting of the chipset by the Cisco. In the previous sample debug, you can see `modified open w/ option 1180`. This is a 16-bit value read from left to right. The Cisco router can only set options on, but not off.

<#root>

- + Bit 0 - Open in Wrap: the open adapter is executed without inserting phantom drive to allow testing of the lobe.
- + Bit 1 - Disable Hard Error: prevents a change in the Hard Error and Transmit Beacon bits causing a Ring Status Change ARB.
- + Bit 2 - Disable Soft Error: prevents a change in the Soft Error bit from causing a Ring Status Change ARB.
- +
Bit 3 - Pass Adapter MAC frames: Causes adapter class MAC frames not supported by the adapter to be passed back as received Frames. If this bit is off, these frames are discarded.
- + Bit 4 - Pass Attention MAC frames: Causes attention MAC frames that are not the same as the last received attention MAC frame.
- + Bit 5 - reserved: should be 0

+ Bit 6 - reserved: should be 0

+

Bit 7 - Contender: When the contender bit is on, the adapter will participate in claim token upon receiving a claim token frame from another adapter with a lower source address. If this bit is off the adapter will not enter into claim token process if it receives a Claim Token MAC frame. The adapter will enter claim token if a need is detected regardless of the setting of this bit.

+

Bit 8 - Pass Beacon MAC frames: The adapter will pass the first Beacon MAC frame and all subsequent Beacon MAC frames that have a change in the source address of the Beacon type.

+ Bit 9 - reserved: should be 0

+ Bit 10 - reserved: should be 0

+ Bit 11 - Token Release: If this bit is set the adapter will not operate with early token release. If this bit is 0 the adapter will operate with early token release when the selected ring speed is 16 megabits per second.

+ Bit 12 - reserved: should be 0

+ Bit 13 - reserved: should be 0

+ Bit 14 - reserved: should be 0

+ Bit 15 - reserved: should be 0

For option 0x1180, see the previous **bold** bits.

Setting the Functional and Group Addresses

In the previous sample debug, the functional address is set to w/ 800011A and the group address is set to w/ 80000000.

These are reporting attributes for LNM:

```
REPORT_LRM    0x80000000
REPORT_LBS    0x00000100
REPORT_CRS    0x00000010
REPORT_REM    0x00000008
REPORT_RPS    0x00000002
REPORT_AVAIL  0x8000011a
REPORT_ALL    0x8000011a
```

Keepalives

If the problem seems to be the intermittent de-insertion and re-insertion of a random number of Token Ring interfaces, the ring might be extremely congested, which causes the keepalives sent by the Token Ring interface to time out. Issue the **keepalive {0 - 32767}** interface command to increase the keepalive value. (The default value is 10 seconds.)

<#root>

```
tricerca(config)#  
  
interface tokenring 4/0/0  
  
tricerca(config-if)#  
  
keepalive 30
```

Note: When you increase the keepalives, you might keep Token Ring interfaces from bouncing; this does not, however, replace good network design and proper ring segmentation.

Use of LAN Analyzer

Very often, the problems faced in Token Ring networks are of an intermittent nature, with re-occurrences at random intervals. This makes it much more challenging to troubleshoot. This is common in situations where you have a random number of stations that experience slow performance or tend to detach themselves from the ring momentarily. Also, the use of the above techniques to troubleshoot insertion problems sometimes might not provide adequate information.

In order to narrow down the problem, a Token Ring LAN analyzer might be required to capture and analyze frames. The analyzer should be the immediate upstream neighbor to the station that is trying to insert. It is therefore important to know what you should look for in a Token Ring trace and to know what to expect in a healthy Token Ring network. Token Ring frame analysis is beyond the scope of this document, but these frames are what you would expect to see in the Token Ring trace of a successful Token Ring station insertion:

```
MAC: Active Monitor Present
```

```
!--- Normal ring poll.
```

```
MAC: Standby Monitor Present
```

```
!--- Normal ring poll.
```

```
MAC: Duplicate Address Test
```

```
!--- Inserting station sends duplicate address MAC#1 frames.
```

```
MAC: Duplicate Address Test
```

```
!--- Inserting station sends duplicate address MAC#2 frames.
```

```
MAC: Standby Monitor Present
```

```
MAC: Report SUA Change
```

```
!--- Stored Upstream Address reported to Configuration Report Server !--- by inserting station.
```

```
MAC: Standby Monitor Present
```

```
!--- Participate in ring poll by inserting station.
```

```
MAC: Report SUA Change
```

```
!--- SUA reported by station downstream from inserting station.
```

MAC: Standby Monitor Present

!--- Normal ring poll.

MAC: Request Initialization

!--- Request ring initialization MAC#1 from Ring Parameter Server.

MAC: Request Initialization

!--- Request ring initialization MAC#2 from Ring Parameter Server.

MAC: Request Initialization

!--- Request ring initialization MAC#3 from Ring Parameter Server.

MAC: Request Initialization

!--- Request ring initialization MAC#4 from Ring Parameter Server.

MAC: Report Soft Error

MAC: Active Monitor Present

MAC: Standby Monitor Present

!--- Station inserted and participating in ring poll.

MAC: Standby Monitor Present

Note: That trace has been filtered to show only frames of interest (see the comments). On a network analyzer, those frames can be examined more closely to view the detailed information that is contained in those fields.

It is very likely that you will also see soft errors - such as burst errors, line errors, token errors, ring purges, and lost frame errors - caused by the simple act of opening the hub relay. Do not assume that the existence of these errors indicates a problematic ring, as these are normal symptoms that occur during the insertion process.

Other frames for which to look, for example, are AM-issued MAC frames that are called Neighbor Notification Incomplete (NNI) or Ring Poll Failure. This frame should be issued every seven seconds in a failing ring, just prior to an AMP MAC frame. The NNI frame is important because it contains the address of the last station to successfully complete the ring poll process. The downstream neighbor from this station is usually the culprit, and you can remove the downstream neighbor to solve the problem.

Related Information

- [Troubleshooting DLSw](#)
- [DLSw \(Data-Link Switching\) & DLSw+ \(Data-Link Switching Plus\) Support Page](#)
- [Technical Support & Documentation - Cisco Systems](#)