

Basic RSRB Configurations and Concepts

Document ID: 10674

Introduction

Prerequisites

- Requirements

- Conventions

Virtual Rings

Configuration Example – Direct Encapsulation

- ROUTER A

- ROUTER B

- ROUTER C

Configuration Example – FST (Fast Sequenced Transport) Encapsulation

- ROUTER A

- ROUTER B

- ROUTER C

Configuration Example – TCP Encapsulation

- ROUTER A

- ROUTER B

- ROUTER C

Configuration Example – Mixed Encapsulations

- ROUTER A

- ROUTER B

- ROUTER C

Configuration Example – Indirect Bridging

- ROUTER A

- ROUTER B

- ROUTER C

Related Information

Introduction

This document introduces you to RSRB, or Remote Source–Route Bridging. With Cisco routers, RSRB allows you to bridge traffic between two Token Rings from separate routers. The creation of virtual rings, or ring groups, is discussed. Then you are given configuration examples for RSRB, with Direct, FST (Fast Sequenced Transport), and TCP encapsulation, which include some helpful design guidelines. Finally, you get a description of indirect bridging, a technique by which two or more ring groups can be bridged together.

All the examples given here only show the parts of the configuration that pertain to the RSRB configuration. In addition, you need to add the command `source-bridge spanning` to the Token–Ring interfaces in order to specify how single–route broadcast traffic should be handled. For more details on how this is configured, refer to the document on single–route broadcasts and the Automatic Spanning Tree.

Prerequisites

Requirements

There are no specific requirements for this document.

Conventions

Refer to Cisco Technical Tips Conventions for more information on document conventions.

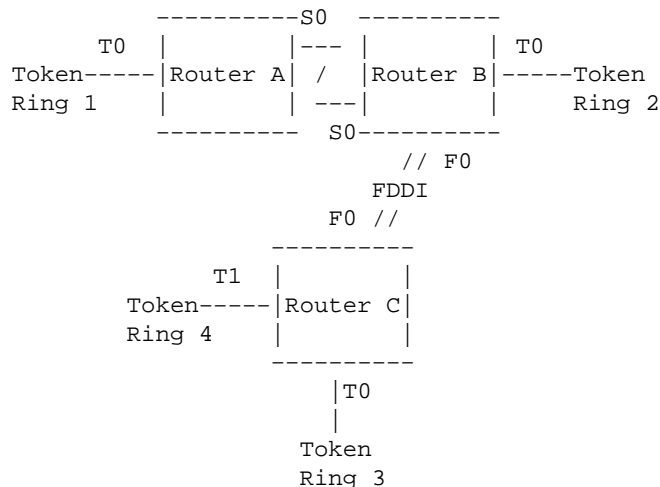
Virtual Rings

Virtual rings or ring groups are mechanisms that bridge together two or more rings. These virtual rings can be extended between multiple routers, so that all of the rings on multiple routers are able to communicate with each other. This extension is done with **source-route remote-peer** statements.

There are some things to keep in mind when you set up virtual rings. In order to pass SRB traffic between them, routers must be part of the same ring-group (virtual ring). Also, routers that are peered together must have source-route remote-peer statements that point to each other. For example, picture three routers (A, B, and C) all in a ring-group. A is peered to B, and B is peered to C, but A is not peered to C. Routers A and C will not be able to pass SRB traffic from one to the other, even though they appear to have a path through router B. The peering is not commutative.

Configuration Example – Direct Encapsulation

Direct encapsulation is the simplest type of remote peering. In order to use direct encapsulation, the routers to be peered together must be directly attached to each other, that is, no intermediate hops, with HDLC-encapsulated serial lines or with LAN media. When serial lines are slow, or not completely clean, FST or TCP are better choices for encapsulation types.



ROUTER A

```
source-bridge ring-group 4095
source-bridge remote-peer 4095 interface serial 0

interface serial 0
encapsulation hdlc {default}

interface tokenring 0
source-bridge 1 1 4095
source-bridge spanning
```

ROUTER B

```
source-bridge ring-group 4095
source-bridge remote-peer 4095 interface serial 0
source-bridge remote-peer 4095 interface fddi 0 4000.0000.0003

interface serial 0
encapsulation hdlc      {default}

interface fddi 0
mac-address 4000.0000.0002

interface tokenring 0
source-bridge 2 1 4095
source-bridge spanning
```

ROUTER C

```
source-bridge ring-group 4095
source-bridge remote-peer 4095 interface fddi 0 4000.0000.0002

interface fddi 0
mac-address 4000.0000.0003

interface tokenring 0
source-bridge 3 1 4095
source-bridge spanning

interface tokenring 1
source-bridge 4 1 4095
source-bridge spanning
```

NOTES:

- You must use HDLC encapsulation when you do direct encapsulation over serial lines. If the line is set up for PPP, X25, or other encapsulation types, direct encapsulation cannot be used. Since HDLC is the default encapsulation, this information is not included in the configuration file. The router accepts the command **encapsulation hdlc** at config time, but, when the commands **show config** or **write term** are used, this information is not displayed. In order to find this information, use the command **show interface**.
- Since LANs are multi-access media, MAC Addresses must be used when you do direct encapsulation over LAN interfaces. In this case, MAC Addresses were defined for the LAN interfaces for clarity, but most of the time the burned-in MAC addresses are preferable. In order to find these addresses for the FDDI interfaces, use command **show interface fddi x**.
- An FDDI interface was used here to demonstrate direct encapsulation over LANs, but it will also work over Ethernet or Token-Ring interfaces. Typically you do not see direct encapsulation over Token Ring since local SRB is usually a better option.
- Even though all three routers are part of ring-group 4095, stations on the Token Ring off of Router A are not able to establish sessions with stations on the Token Ring off of Router C since these two routers are not peered together. Router A and Router C cannot be peered together with direct encapsulation because of hop (Router B) which intervenes. You can either add a direct link between Routers A and C and add the appropriate peer statement, or use one of the other two encapsulation methods (FST, TCP) to peer routers A and C.

Configuration Example – FST (Fast Sequenced Transport) Encapsulation

The major advantage of FST encapsulation over direct encapsulation is that it allows you to extend over multiple hops. It does require a bit more protocol overhead than direct encapsulation (FST encapsulates SRB packets into an IP packet), and it relies on a level 3 protocol (IP), while direct encapsulation only needs level 2 connectivity (HDLC, LAN LLC).

FST introduces a new configuration requirement, as well. Each router that uses FST peering must have an `fst-peername` defined. This name must be an IP address that is in use by an active interface on the router. The `remote-peer` statements of routers that try to access this router must reference the IP address that has been defined as the `fst-peername` for the router – to use any other IP address on the box does not work. Use either a loopback interface or a LAN interface as the `fst-peername` for stability.

ROUTER A

```
source-bridge fst-peername 1.1.1.1 source-bridge ring-group 4095 source-bridge remote-peer 4095 fst 1.2.2.2
```

```
source-bridge remote-peer 4095 fst 1.3.3.3

interface serial 0
ip address 1.255.1.1 255.255.0.0

interface tokenring 0
ip address 1.1.1.1 255.255.0.0
source-bridge 1 1 4095
source-bridge spanning

router rip
network 1.0.0.0
```

ROUTER B

```
source-bridge fst-peername 1.2.2.2
source-bridge ring-group 4095
source-bridge remote-peer 4095 fst 1.1.1.1
source-bridge remote-peer 4095 fst 1.3.3.3

interface serial 0
ip address 1.255.1.2 255.255.0.0

interface fddi 0
ip address 1.254.1.2 255.255.0.0

interface tokenring 0
ip address 1.2.2.2 255.255.0.0
source-bridge 2 1 4095
source-bridge spanning

router rip
network 1.0.0.0
```

ROUTER C

```
source-bridge fst-peername 1.3.3.3
source-bridge ring-group 4095
source-bridge remote-peer 4095 fst 1.1.1.1
source-bridge remote-peer 4095 fst 1.2.2.2

interface fddi 0
ip address 1.254.1.1 255.255.0.0

interface tokenring 0
ip address 1.3.3.3 255.255.0.0
source-bridge 3 1 4095
source-bridge spanning

interface tokenring 1
source-bridge 4 1 4095
source-bridge spanning

router rip
network 1.0.0.0
```

NOTES:

- The router RIP statements and IP addresses on serial and FDDI interfaces show IP connectivity between the routers. If you cannot pass IP traffic between the two IP addresses used by the peers, FST RSRB does not work.
- In this example, the IP addresses of the Token-Ring interfaces were chosen as the fst-peernames. Any active IP addresses, for instance, the IP addresses on the serial or FDDI interfaces, could have been used instead, as long as the FST peername was referenced consistently by the peers. For instance, if the fst-peername statement of Router A was modified to read **source-bridge fst-peername 1.255.1.1**, you need to replace **source-bridge remote-peer 4095 fst 1.1.1.1** with **source-bridge remote-peer 4095 fst 1.255.1.1** on Routers B and C.
- Even though there are multiple rings bridged off of router C, Routers A and B still require only one peer statement to that router.

Configuration Example – TCP Encapsulation

The most commonly used encapsulation type is TCP. This encapsulation has greater overhead than direct or FST, both on the network, where every RSRB packet gets encapsulated within a full IP and TCP header, and in the routers, where there is additional processor overhead to maintain a TCP session for every remote-peer required. The advantage of TCP encapsulation is the reliable delivery of packets, which lessens the recovery responsibility of end stations in the case of a lost or corrupted packet.

TCP configuration is very similar to FST configuration. Instead of a special command to configure the peername, the IP address that represents the local router is treated exactly the same as a remote peer. Here is a TCP peering configuration equivalent to the FST configuration:

ROUTER A

```
source-bridge ring-group 4095
source-bridge remote-peer 4095 tcp 1.1.1.1
source-bridge remote-peer 4095 tcp 1.2.2.2
source-bridge remote-peer 4095 tcp 1.3.3.3

interface serial 0
```

```
ip address 1.255.1.1 255.255.0.0

interface tokenring 0
ip address 1.1.1.1 255.255.0.0
source-bridge 1 1 4095
source-bridge spanning

router rip
network 1.0.0.0
```

ROUTER B

```
source-bridge ring-group 4095
source-bridge remote-peer 4095 tcp 1.2.2.2
source-bridge remote-peer 4095 tcp 1.1.1.1
source-bridge remote-peer 4095 tcp 1.3.3.3

interface serial 0
ip address 1.255.1.2 255.255.0.0

interface fddi 0
ip address 1.254.1.2 255.255.0.0

interface tokenring 0
ip address 1.2.2.2 255.255.0.0
source-bridge 2 1 4095
source-bridge spanning

router rip
network 1.0.0.0
```

ROUTER C

```
source-bridge ring-group 4095
source-bridge remote-peer 4095 tcp 1.3.3.3
source-bridge remote-peer 4095 tcp 1.1.1.1
source-bridge remote-peer 4095 tcp 1.2.2.2

interface fddi 0
ip address 1.254.1.1 255.255.0.0

interface tokenring 0
ip address 1.3.3.3 255.255.0.0
source-bridge 3 1 4095
source-bridge spanning

interface tokenring 1
source-bridge 4 1 4095
source-bridge spanning

router rip
network 1.0.0.0
```

Note: In this configuration, the local remote-peer statement is always the first of the configured remote-peers. This makes it easier to understand for troubleshooting.

Configuration Example – Mixed Encapsulations

It is all right to mix encapsulation types within a ring group, but there are some simple rules to be followed.

- Any router that uses FST peering must have an FST peername. The FST peername is the IP address of one of the interfaces. Use of an address defined for a loopback interface is permissible.
- Any router that uses TCP peering must have a local remote-peer statement. The source bridge remote peername is the IP address of one of the interfaces. Use of an address defined for a loopback interface is permissible.
- When peering two routers, only one encapsulation type can be used. You cannot have a TCP peer statement from A to B, and an FST peer statement from B back to A. Similarly, you cannot peer from A to B with more than one encapsulation type.

This example shows all three encapsulation types within the same ring group. Routers A and B are peered with direct encapsulation; routers A and C use TCP encapsulation; and routers B and C use FST encapsulation.

ROUTER A

```
source-bridge ring-group 4095
source-bridge remote-peer 4095 tcp 1.1.1.1
source-bridge remote-peer 4095 tcp 1.3.3.3
source-bridge remote-peer 4095 interface serial 0

interface serial 0
ip address 1.255.1.1 255.255.0.0

interface tokenring 0
ip address 1.1.1.1 255.255.0.0
source-bridge 1 1 4095
source-bridge spanning

router rip
network 1.0.0.0
```

ROUTER B

```
source-bridge fst-peername 1.2.2.2
source-bridge ring-group 4095
source-bridge remote-peer 4095 interface serial 0
source-bridge remote-peer 4095 fst 1.3.3.3

interface serial 0
ip address 1.255.1.2 255.255.0.0

interface fddi 0
ip address 1.254.1.2 255.255.0.0

interface tokenring 0
ip address 1.2.2.2 255.255.0.0
source-bridge 2 1 4095
source-bridge spanning

router rip
network 1.0.0.0
```

ROUTER C

```
source-bridge fst-peername 1.3.3.3
source-bridge ring-group 4095
source-bridge remote-peer 4095 fst 1.2.2.2
source-bridge remote-peer 4095 tcp 1.3.3.3
source-bridge remote-peer 4095 tcp 1.1.1.1
```

```

interface fddi 0
ip address 1.254.1.1 255.255.0.0

interface tokenring 0
ip address 1.3.3.3 255.255.0.0
source-bridge 3 1 4095
source-bridge spanning

interface tokenring 1
source-bridge 4 1 4095
source-bridge spanning

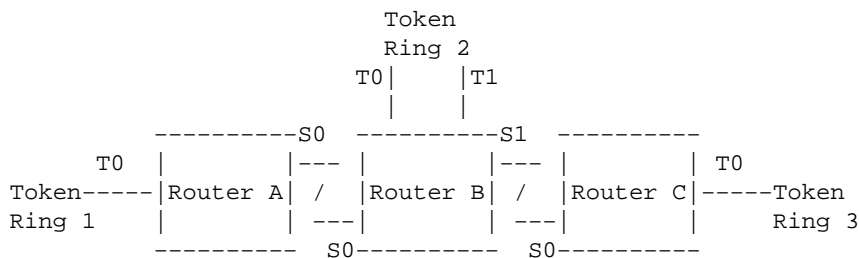
router rip
network 1.0.0.0

```

Configuration Example – Indirect Bridging

So far we have only dealt with situations with one ring-group, but there are times when we want to bridge multiple ring-groups together. There is no command that directly bridges two ring-groups together, but we can accomplish this with indirect bridging. Through the use of multiple Token-Ring interfaces on the same segment, two or more ring groups can be indirectly bridged together.

This is an example:



ROUTER A

```

source-bridge ring-group 4095
source-bridge remote-peer 4095 interface serial 0

interface serial 0
encapsulation hdlc

interface tokenring 0
source-bridge 1 1 4095
source-bridge spanning

```

ROUTER B

```

source-bridge ring-group 4095
source-bridge remote-peer 4095 interface serial 0
source-bridge ring-group 4094
source-bridge remote-peer 4094 interface serial 1

interface serial 0
encapsulation hdlc

interface serial 1
encapsulation hdlc

```

```
interface tokenring 0
source-bridge 2 1 4095
source-bridge spanning

interface tokenring 1
source-bridge 2 1 4094
source-bridge spanning
```

ROUTER C

```
source-bridge ring-group 4094
source-bridge remote-peer 4094 interface serial 0

interface serial 0
encapsulation hdlc

interface tokenring 0
source-bridge 3 1 4094
source-bridge spanning
```

NOTES:

- In the configuration, physical ring 2, connected to the two Token-Ring interfaces of Router B, is used for indirect bridging of ring groups 4095 and 4094. This ring can be used by other devices as well, but be aware that there can be a heavy load on this ring because it handles broadcast traffic from both ring groups.
- Because of the indirect bridging, stations on the Token Ring off of Router A can communicate with stations on the Token Ring off of Router C, even though they are not peered together. If the indirect bridging commands were removed from Router B, this connectivity would be lost.
- Indirect bridging is a mechanism to join two ring groups that exist without the need to reconfigure all of the routers within one of the groups. It is more of a patch than a design feature, and it is not regarded as a good design practice. When you design a network from the start, multiple ring-groups joined in this manner must not be designed into the same RSRB network.

Related Information

- [Technical Support & Documentation – Cisco Systems](#)

All contents are Copyright © 1992–2006 Cisco Systems, Inc. All rights reserved. Important Notices and Privacy Statement.

Updated: Jan 26, 2007

Document ID: 10674
