10

Bridging—Transmitting Non-IP Traffic or Merging Two Networks

Contents

Overview

The ProCurve Secure Router can function as a bridge as well as a router. A bridge, like a switch, is a Layer 2 device that operates at the Data Link Layer of the Open Systems Interconnection (OSI) model. A bridge connects two or more LAN segments together. Bridges and switches also minimize traffic on network segments by breaking up traffic areas, reducing data transmission delays, and increasing the efficiency of the network. A bridged network can provide traffic management by reducing collisions and limiting the amount of bandwidth wasted with unnecessary transmissions when routing is not necessary.

Each device connected by a bridge must be on the same logical network because Layer 2 devices translate and filter only hardware (MAC) addresses. Bridges and switches make forwarding and filtering decisions based on these MAC addresses; upper-Layer protocols—such as IP—are transparent to them.

Bridges can be categorized as either local or remote (see Figure 10-1). Local bridges provide connectivity for multiple LAN segments in one area. A remote bridge, on the other hand, connects LAN segments in different areas. Because remote bridges must connect geographically distant LAN segments, they have special design considerations, including the buffering of the LAN-to-WAN connection speed variation.

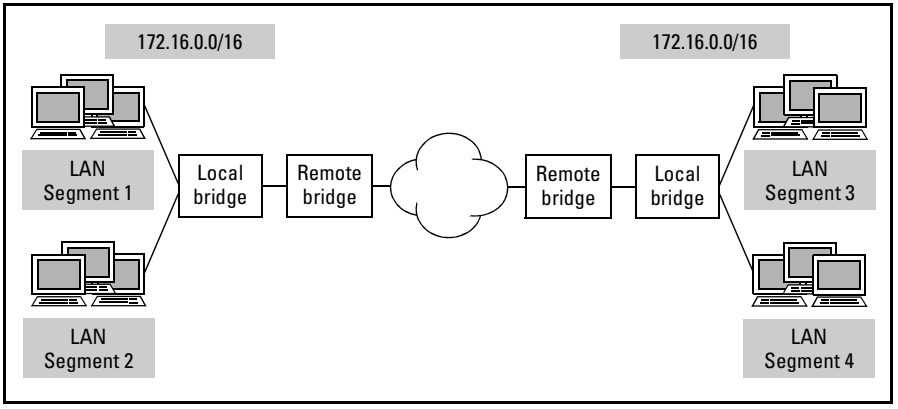

Figure 10-1. Local and Remote Bridges

The ProCurve Secure Router supports bridging using the IEEE 802.2 standards. You would configure a ProCurve Secure Router to act as a remote bridge to allow it to:

- transmit non-IP traffic
- merge two remote networks

Transmitting Non-IP Traffic

The ProCurve Secure Router only routes IP traffic. If one or more of the networks in a WAN use a different Layer 3 protocol, you must configure the router to bridge this traffic. The router will simply pass the traffic through interfaces in the bridge group without examining or modifying the Layer 3 header.

Layer 3 protocols that must be bridged include:

- NetBIOS
- IPX
- AppleTalk
- DecNet

Merging Two Remote Networks

When you configure the ProCurve Secure Router to act as a bridge, you extend a LAN through WAN connections. In essence, the WAN becomes a single LAN. The distance between the bridges does not matter; they connect segments of a single network.

However, practically, LAN connections transmit at much higher speeds than WAN connections. As you design your network, you should take this difference into account. While flooding messages between remote segments is logically equivalent to flooding them between local segments, sending messages to a remote segment costs more in terms of time and relative bandwidth as well as money.

Spanning Tree Protocol

When you configure the ProCurve Secure Router as a bridge, it loses its routing capabilities. Like a switch, it must run a spanning tree protocol to eliminate loops and respond to network topology changes. Bridged interfaces on the ProCurve Secure Router automatically run rapid spanning tree protocol (RSTP), IEEE 802.1W. If necessary, you can alter the default spanning tree settings. See “Configuring Spanning Tree” on page 10-11.

Configuring Bridging

You configure the ProCurve Secure Router to function as a bridge by assigning logical interfaces to be part of a bridge group. For example, you could assign the Ethernet interface and the Point-to-Point Protocol (PPP) interface to a bridge group, or you could assign the Ethernet interface and the Frame Relay subinterface to a bridge group.

When the router receives a packet on a bridged interface, it floods the packet out all interfaces in the bridge group. The router also stores the source MAC address of the packet in a bridge table, together with the interface from which it received the packet. When a packet arrives destined for that address, the router then knows through which interface to forward it. In this way, the router gradually learns how to forward traffic and contain packets.

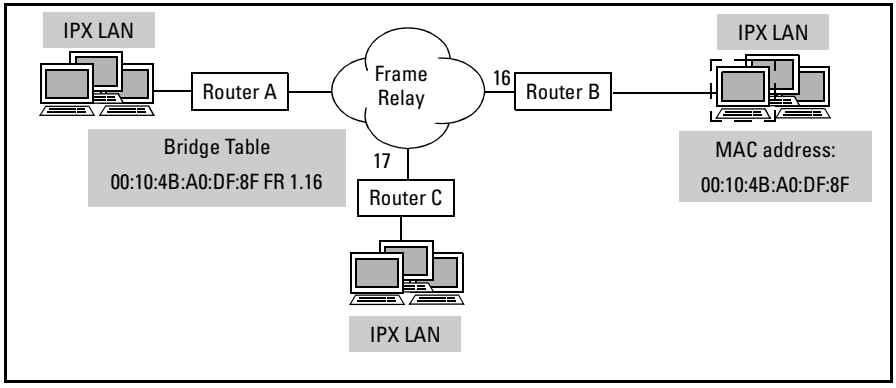

Figure 10-2. Bridging Example

In Figure 10-2, networks at sites A, B, and C use IPX. The sites connect through a Frame Relay network. When configuring bridging for the traffic between these sites, you would assign the Ethernet interface and Frame Relay subinterfaces to the same bridge group. When Router A receives a packet from a local host on its Ethernet interface, it searches its bridge table for the entry corresponding to its destination MAC address. It then transmits it out the correct Frame Relay subinterface, leaving the IPX header unexamined and intact. Router B receives the packet on its Frame Relay subinterface and transmits it out its Ethernet interface. The network at site B can now process the IPX packet.

To configure bridging, you must:

- configure a bridge group
- assign interfaces to the bridge group
- disable IP routing, if you are bridging IP traffic

Note

The ProCurve Secure Router does not both route and bridge IP traffic. If you want to bridge IP traffic, you must disable IP routing.

However, the router can route IP traffic and bridge *non-IP* traffic at the same time. It can even route IP traffic and bridge non-traffic IP traffic on the same Frame Relay or ATM interface. For example, you could configure Frame Relay subinterface 1.101 as part of a bridge group for non-IP traffic, but route IP traffic through Frame Relay subinterface 1.102.

Configuring a Bridge Group

You create bridge groups from the global configuration mode context. When you create the bridge, you must specify that it uses IEEE:

Syntax: bridge <*group number*> protocol ieee

The group number can be between 1 and 255. For example:

```
ProCurve(config)# bridge 1 protocol ieee
```

Assigning an Interface to the Bridge

You configure bridging on Data Link Layer interfaces. Typically, you will assign both LAN and WAN interfaces to the bridge group.

LAN interfaces include:

- Ethernet interfaces

When you enable 802.1Q encapsulation on an Ethernet interface, you can no longer assign it to a bridge group; the interface can now carry traffic for multiple VLANs and you cannot bridge traffic between different VLANs.

WAN interfaces on which you can configure bridging include:

- PPP interfaces
- High-level Data Link Control (HDLC) interfaces
- Frame Relay subinterfaces
- Asynchronous Transfer Mode (ATM) subinterfaces

If you want to configure bridging between more than one switch, remember to assign both Ethernet interfaces to the bridge group. If the router is acting as a remote bridge to more than one remote site (for example, the headquarters router in the Frame Relay network shown in Figure 10-2), you should assign all WAN interfaces to the bridge.

You can also assign only WAN interfaces to a bridge, although you probably would not use this application. In this case, the router would simply act as a corridor between remote sites.

To assign an interface to a bridge group:

1. Move to the logical interface configuration mode context:

   ProCurve(config)# int ppp 1

2. Assign the interface to the bridge group:

   Syntax: bridge-group <*group number*>

   For example:

   ProCurve(config-ppp 1)# bridge-group 1

Note Only one interface in the bridge group should have an IP address. You should remove all IP addresses from other interfaces before configuring the bridge.

Note Remember that every host in a bridged network must be on the same subnet.

If you want to bridge traffic between hosts on multiple subnets, you can change the subnet mask so that all hosts are on the same subnet. You could also enable a different bridge group on interfaces connecting to different subnet. However, in the second case these subnets will not communicate between each other unless a different device supports routing between the subnets.

Disabling IP Routing

The router cannot both route and bridge IP traffic. You must disable IP routing when the router acts as a remote bridge to join two sites using addresses on the same IP network.

Enter the following command to disable IP routing:

ProCurve(config)# no ip routing

Rather than use the router as a bridge in this situation, you could use variable-length subnetting to divide the network into two subnets. This solution works when the sites include contiguous, evenly divided addresses. For example, in Figure 10-3 an organization uses network 192.168.1.0 /24. Site A uses addresses 192.168.1.1 through 192.168.1.127 and Site B uses addresses 192.168.1.128 through 192.168.1.254. You could divide the subnet into subnets 192.168.1.0 / 25 and 192.168.1.128 /25.

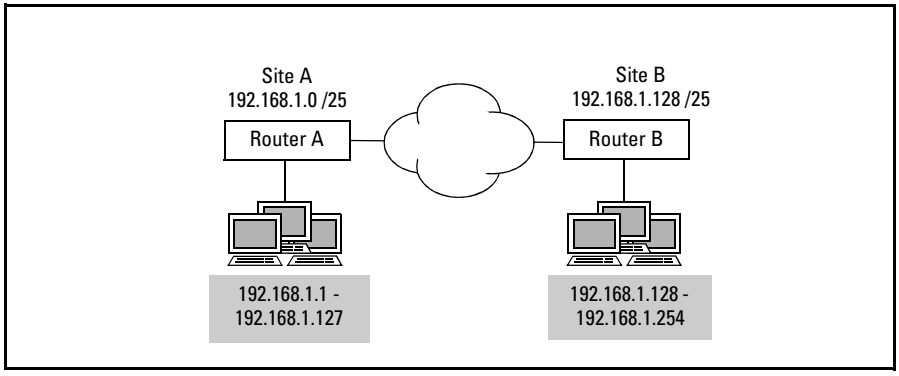

Figure 10-3. Variable-Length Subnetting

Viewing the Bridge Table

The ProCurve Secure Router stores information about how to forward bridged packets in a bridge table. To view the bridge table, move to the enable mode context and enter:

Syntax: show bridge <*group number*>

For example:

```
ProCurve# show bridge 1
```

Note

You must either enter **show** commands from the enable mode context or add **do** to the command. For example, to view the bridge table from the global configuration mode context, you would enter **do show bridge**.

The bridge table contains MAC addresses for hosts in the bridged network and the interface through which the router connects to these hosts. It also displays the age of the entry and the number of frames transmitted to and received from the host. (See Figure 10-4.)

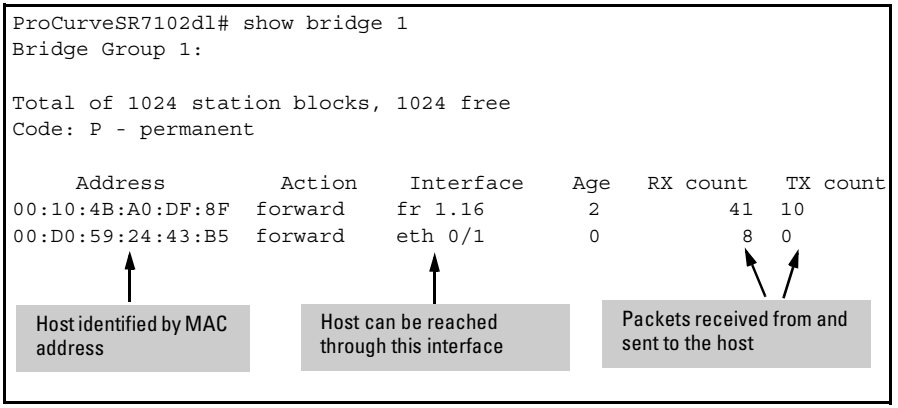

Figure 10-4. Viewing a Bridge Table

You can also view specific portions of the bridge table. Use the commands shown in Table 10-1.

If necessary, you can manually add a host to the bridge table with this global configuration mode context command:

Syntax: mac address-table static <*mac address*> bridge <*group number*> <*interface ID*>

Identify the host by its MAC address and enter the number of the bridge group and the forwarding interface.

Table 10-1. Viewing Portions of the Bridge Table

Display Hosts Connected Through	Command Syntax
a specific bridge group	**show bridge <group number>**
a specific Ethernet interface	**show bridge ethernet <slot>/<port>**
a specific PPP interface	**show bridge ppp <interface number>**
a specific Frame Relay subinterface	**show bridge frame-relay <subinterface number>**
a specific HDLC interface	**show bridge hdlc <interface number>**

Troubleshooting Bridging

When traffic is not able to reach its destination, follow this standard troubleshooting process:

1. Check the Physical Layer:
   a. If the Stat LED for the carrier line's module slot is green, the physical line is up. Move to the second step.
   b. If the Stat LED for the line is red, the physical line is down. Check for bad cables, then for configuration mismatches. (For more detailed instructions, see "Troubleshooting an Ethernet Interface" on page 3-25, "Troubleshooting E1 and T1 WAN Connections" on page 4-31, "Troubleshooting a Serial Connection" on page 5-18, or "Troubleshooting the ADSL Connection" on page 7-47.)
2. Check the Data Link Layer:
   a. View the status of logical interfaces, including Ethernet interfaces. For example:

      ```
      ProCurve# show interface frame-relay 1
      ```

   b. If the interface is up, move to step 3.
   c. If the interface is down, follow the troubleshooting tips in "Troubleshooting an Ethernet Interface" on page 3-25, "Troubleshooting Logical Interfaces" on page 6-59, or "Troubleshooting the ATM Interface" on page 7-49.
3. Check that all interfaces that should be members of a bridge group are members. View the running-config for the interface and look for the bridge group number:

   ```
   ProCurve# show run int eth 0/1
   ```

4. If an interface refuses to join a bridge group, try removing other interfaces from the group (enter **no bridge-group *<group number>*** from the interface configuration mode context). Then configure the Ethernet interface to join the bridge group first.
5. If you are using the bridge to connect remote sites using addresses on the same subnet, you should disable IP routing. Verify that IP routing has been disabled:

   ```
   ProCurve# show running-config
   ```

6. Verify that all hosts participating in a bridge group are on the same subnet. You can also try viewing the bridge table. If the table does not show entries for an interface, this is a good hint that the devices on the other end of that connection are on a different subnet.
7. The bridge runs more smoothly if you remove IP addresses from every interface in the bridge except one. For example, you can assign only the Ethernet interface an IP address. Enter **show ip interfaces** and verify that WAN interfaces in the bridge group do no have IP addresses.

Configuring Spanning Tree

When the router acts as a bridge, it automatically enables Rapid Spanning Tree Protocol (RSTP), or IEEE 802.1W. RSTP eliminates network loops and is fully backwards compatible with Spanning Tree Protocol (STP), or IEEE 802.D.

The router only supports RSTP and STP when it acts as a bridge. The following interfaces join the spanning tree when they join a bridge group:

- Ethernet interfaces
- Frame Relay subinterfaces
- ATM subinterfaces

Often, the router will be able to run RSTP adequately without additional configuration: the default settings match most WAN topologies.

You can configure spanning tree functions on the router in order to:

- raise the router's priority for being elected root device
- change the cost of a connection
- connect the router to an edge device
- connect the router to a hub
- have the router run STP

Configuring spanning tree on a WAN router is usually simpler than configuring it on a switch. A switch might provide many connections—some redundant, some necessary, some faster, some slower, some to end users, some to another switch, some to a hub. The ProCurve Secure Router typically has fewer connections—and these only to other routers and switches—and is part of few or no loops. Therefore, you need not understand STP and RSTP in great technical depth.

The overview provides a brief background in STP and RSTP for those who want to learn more about how the protocols function.

Overview

Network devices in a Data Link Layer network, such as bridges and switches, run STP or RSTP. Bridged interfaces on the ProCurve Secure Router also participate in the spanning tree protocol. The protocol helps devices to generate a loopless topology.

Unlike routers, switches do not time out messages. Loops in a network topology can lead to duplicated messages and broadcast storms, which can bring a network down. However, the redundant links that cause loops can also be desirable: they protect against loss of connectivity when a connection fails.

STP allows network devices to generate a shared loopless topology, blocking all redundant links. However, if active connections fail, redundant links can become active for as long as the original path is down.

RSTP is now the spanning tree standard. It improves convergence time to less than one second and is the recommended implementation.

STP BPDUs

Devices running STP send and listen for configuration bridge protocol data units (BPDUs) to determine the spanning tree topology. Each BPDU includes:

- the identifier (priority plus MAC address) of the source port
- the identifier of the root device
- the cost between the source port and root device

Using these BPDUs, each device can determine:

- Which device is root—The root is the device from which the tree topology originates. All ports on the root must remain active. When STP is originally implemented, each device believes that it is the root. In the initial exchange of BPDUs, the device with the lowest identifier is elected root. You can ensure that a router interface is elected by lowering its priority number.
- Which switch provides the local device the best connection to the root—This switch becomes the device's designated switch.
- Which port provides the best connection to the designated switch—This becomes the root port.

A device then marks the following ports for activation (forwarding frames):

- the root port
- designated ports—which connect to devices that consider the local device as their designated switch (and ports that connect to end users)

All other ports become inactive.

The root device periodically sends BPDUs. If the router is root, these BPDUs will consume some bandwidth. Other devices only send topology change notification BPDUs (TCN BPDUs).

When a device receives a TCN BPDU, it re-evaluates which ports are marked for activation. If necessary, it transmits its own TCN BPDU, informing other devices on the change. The port (or ports) through which the device transmits a BPDU is not necessarily the one that received the BPDU that prompted the change.

Devices determine which ports process BPDUs, learn information about the network topology, forward BPDUs, and forward network traffic according to the ports' STP state.

STP States

STP includes the following port states:

- disabled
- blocking
- listening
- learning
- forwarding

In a stable network, all ports are in either the forwarding or blocking state. Only ports in the forwarding state forward frames. Ports in the blocking state are not considered part of the network topology.

Note

When using STP, it is important to understand the difference between disabled and blocking ports. Neither type forwards frames or learns addresses. Neither processes or transmits BPDUs. However, blocking ports receive BPDUs, while disabled ports do not. If you disable a port, it will not participate in STP at all.

When a change in network topology makes STP determine that a new port must become active, the port first passes through the listening and learning states. (When STP is initially enabled and devices exchange configuration BPDUs, all ports move through the listening and learning states until STP determines whether they should become blocked or forwarding ports.)

In the listening state, the port processes BPDUs to determine whether it is indeed the best connection to the root. If within 15 seconds it does not receive a BPDU advertising a better connection, the port enters the learning state.

In the learning state, the port begins to transmit BPDUs as well as receive them. This notifies other active ports of its presence, and the learning port becomes part of the network topology. The port also listens for frames to build up its address database. After 15 seconds, it enters the forwarding state and begins to forward traffic. (If the port receives a better BPDU than it can transmit during this interval, it returns to blocking.)

As you can see, the process of a port moving from blocked to forwarding can be quite lengthy. A network running STP usually takes a minute to converge after a link failure, and the network outage during this delay is not acceptable for many environments.

RSTP Improvements

RSTP can reduce convergence time to less than 1 second.

RSTP does not always force ports to go through the listening and learning states and removes the distinction between blocked and disabled ports.

RSTP speeds convergence by:

- defining new roles for certain ports:
  - edge ports
  - backup ports
  - alternate ports
  - ports on a point-to-point connection
- using sync to activate point-to-point ports
- immediately purging old information

New Roles. In RSTP, edge ports immediately become forwarding ports; they must forward frames because they are the only connection to the end client. You can configure ports on the ProCurve Secure Router to be edge ports (although this is not a typical application for the router). Important configurations for edge ports are BPDU guards and filters which keep the router from receiving BPDUs from user software or rogue devices.

Blocking ports are divided into backup and alternate ports. Backup ports provide a redundant connection to the root through a different device. Alternate ports provide a redundant connection to the root through the same device. If the root port goes down, alternate ports are allowed to move rapidly into the forwarding state.

Ports on a point-to-point connection can use the rapid sync method to move into the forwarding state. On the ProCurve Secure Router, ports will almost always be on point-to-point connections. You can configure this setting, or you can leave the interface at its default **auto** setting, which defines full-duplex interfaces as point-to-point ports.

Sync. STP assumes that devices best decide which ports to activate by collecting a great deal of information about the network. Therefore, it sets conservative timers for listening for TCN BPDUs. Ports were forced to spend 30 seconds passing through the listening and learning phases before they could begin to forward user traffic.

Many devices now connect through point-to-point connections rather than through shared media. RSTP relies on the fact that the single neighbor at the other end can refuse to activate a link if it has a better connection. Rather than wait 30 seconds collecting information, a port can start forwarding user traffic after a single rapid exchange with its neighbor.

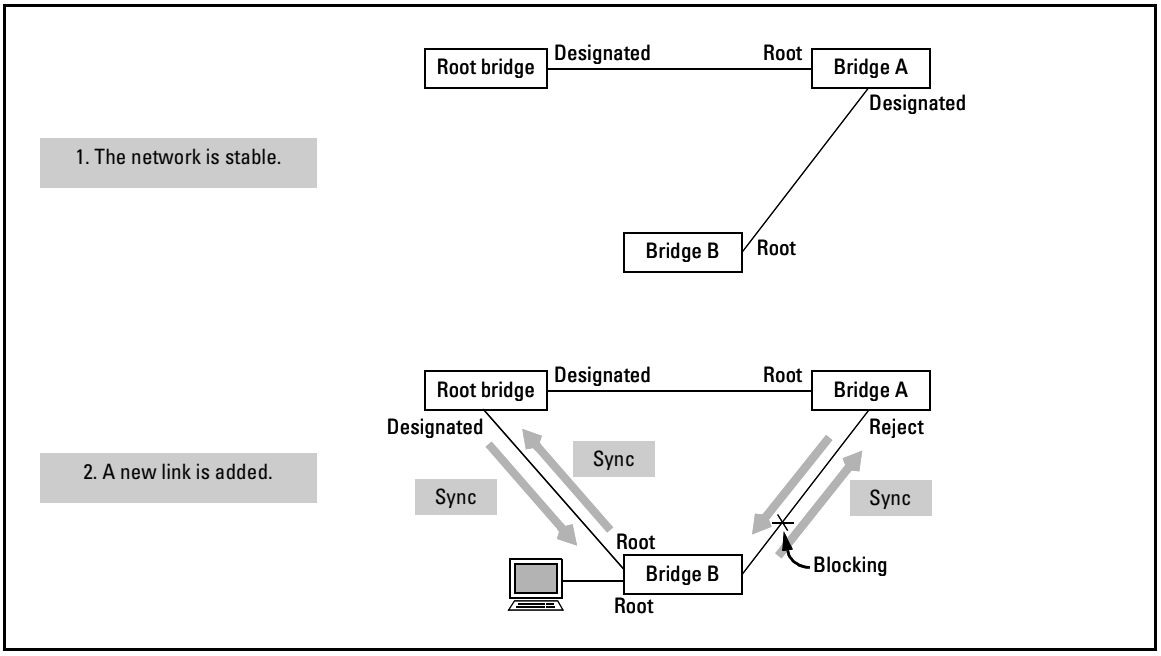

Figure 10-5. Asserting Sync

When network topology changes, devices assert sync to propagate new paths in an ordered flow from devices closer to the root to devices further from the root. A device sends a BDPU to a neighbor on a potential designated port. The BPDU has a proposal flag set, which requests that the two ports immediately transition to the forwarding state. If the neighbor determines that this BPDU is best (the transmitting port is closest to the root), it replies with an agreement BPDU. The neighbor also asserts sync: it makes the port on which it received the BPDU its root port and shuts down all other ports except edge ports.

The neighbor then sends its own proposal BPDUs through the blocked ports. If a neighboring device determines that the connection is best, it brings up its port as root port and continues the process. Otherwise, it sends a non-acknowledgement, and ports on both sides of the link enter the blocking state.

In this way, topology changes propagate rapidly from the root through to edge nodes.

For example, in Figure 10-5, a connection is added between Bridge B and the root. The root bridge first asserts sync with Bridge B. Bridge B blocks its connection to Bridge A. Bridge B attempts to assert sync with Bridge A, but Bridge A rejects the offer because it has a better connection to the root. The link between Bridge A and Bridge B remains blocked.

Immediate Purging. In STP, when devices receive a TCN BPDU withdrawing an entry, they set the timer for the entry in the database to *short*. Only when this timer expires do they flush the entry. In RSTP, devices purge old information as soon as they receive a BPDU indicating a topology change.

RSTP and STP Compatibility

RSTP is designed to be compatible with STP. Even if the LAN is using STP, you should enable RSTP on your router. RSTP automatically detects ports connected to non-RSTP devices and communicates with those devices using 802.1D STP BPDU packets.

Because RSTP is so much more efficient at establishing the network path, it is highly recommended that all your network devices be updated to support RSTP.

Configuring RSTP

RSTP is automatically activated on these interfaces when they act as bridge ports:

- Ethernet interfaces
- Frame Relay subinterfaces
- ATM subinterfaces

You should typically run a spanning tree protocol on these interfaces to prevent the router from handling more traffic than it must. PPP and HDLC interfaces do not participate in the spanning tree.

For most networks, RSTP runs smoothly without any further configuration. However, you can also:

- set the router's priority to influence the election of the root device
- set link cost to influence the selection of a link
- set roles for interfaces
- alter timers

Determining Which Device Becomes Root: Setting the Router's Priority

Spanning tree bridges elect the device with the lowest ID as the root. A bridge's ID consists of its priority value plus its MAC address. By default, all interfaces on the router have a priority of 32,768 (the standard default setting). Unless you alter the priority setting, the switch with the lowest MAC address becomes root.

Default settings, then, leave much to chance. A relatively unimportant device may become root for an entire WAN. Your organization's IT staff should agree on a hub router to become root for the bridged WAN. Lower this router's priority with this global configuration mode command:

Syntax: spanning-tree priority <*value*>

Valid values are from 0 to 63535. Remember that lower values grant higher priority.

Determining Which Links Are Chosen: Setting Link Cost

A BPDU includes the cost of the connection from the source of the BPDU to the root device. Devices calculate this cost from the cost of all intervening links. A device chooses which interface to make its root port according to which interface receives the BPDU with the lowest cost.

A WAN router may have several connections with widely varying link speeds—for example, a 100-Mbps connection to a switch and 3.0-Mbps connection carried on two T1-carrier lines to a Frame Relay network. Assigning a higher cost to the low-speed connection allows the router to take this discrepancy into account when calculating best paths.

The Secure Router OS automatically calculates path cost from bandwidth, and this setting is usually adequate. However, you may also want to consider the monetary cost of link. If you are using a connection as a redundant link, you should raise its cost to keep the router from choosing it as its primary link.

To change the cost of connection, move to the logical interface configuration mode context for that link. Enter this command:

Syntax: spanning-tree path-cost <*value*>

Valid values are from 1 to 63,535. Remember to *raise* the cost for lower-speed or redundant connections.

Another way to force the router to choose one connection over another is to set the port priority. The router only uses this value to choose between two interfaces that have equal cost connections to the root. To set a logical interface's port priority, enter:

Syntax: spanning-tree port-priority <*value*>

Valid values are between 1 and 255. Remember that lower values grant higher priority to the connection. You can only enter values in increments of 16.

Setting Interface Roles

RSTP allows you to define special characteristics for certain ports. These categories speed convergence. Edge ports immediately begin to forward frames. Point-to-point interfaces use sync for rapid activation. (See the "RSTP Improvements" on page 10-14 for more information.)

It is important that interfaces be set to the proper role so that the router can capitalize on RSTP improvements. The ProCurve Secure Router automatically assigns interfaces the roles that they will almost always play.

Interfaces automatically determine whether they are on point-to-point or shared media connections according to the duplex setting. However, if the router connects to a hub, you can manually force the connecting interface to the shared media role.

If the router connects to an end device, you should configure edge port settings.

Configuring an Edge Port. The edge port designation allows interfaces that connect to end devices to immediately enter the forwarding state. This prevents applications on the end device from timing out while they wait for their default gateway to come up. Currently, you will almost always connect your ProCurve Secure Router to a core switch or comparable device, so the edge port option is *disabled* by default.

However, the ProCurve Secure Router does support edge port capabilities. You can enable these capabilities either globally or on an individual interface. Use the commands shown in Table 10-2.

Table 10-2. Defining Edge Ports

Function	Command Syntax	CLI Context
define all spanning tree interfaces on the router as edge ports	**spanning-tree edgeport default**	global configuration mode
define all spanning tree interfaces on the router as *non*-edge ports (default setting)	**no spanning-tree edgeport default**	global configuration mode
enable an Ethernet interface to act as an edge port (overrides global setting)	**spanning-tree edgeport enable**	Ethernet interface configuration mode
prevent an Ethernet interface from acting as an edge port (overrides global setting)	**spanning-tree edgeport disable**	Ethernet interface configuration mode
enable a Frame Relay or ATM subinterface to act as an edge port (overrides global setting)	**spanning-tree edgeport**	Frame Relay or ATM subinterface configuration mode
prevent a Frame Relay or ATM subinterface from acting as an edge port (overrides global setting)	**no spanning-tree edgeport**	Frame Relay or ATM subinterface configuration mode

This global configuration mode command defines all interfaces assigned to a bridge group as edge ports:

Syntax: spanning-tree edgeport default

The default setting is **no spanning-tree edgeport default**. In the default setting, interfaces *do not* act as edge ports. Generally, you should leave this global setting and simply override it for the interface that connects to the end device.

Note

The command to enable an Ethernet interface to act as an edge port is slightly different from the command to enable Frame Relay or ATM subinterfaces to act as edge ports.

To override the global setting for Ethernet interfaces, move to the Ethernet configuration mode context and enter:

Syntax: spanning-tree edgeport [enable | disable]

Enter the command with the **enable** option to allow the interface to act as an edge port. If you have configured a global setting that defines all interfaces as edge ports, the **disable** option overrides this setting.

To enable Frame Relay and ATM subinterfaces to act as edge ports, move to the logical interface configuration mode context and enter:

Syntax: spanning-tree edgeport

When the global setting defines all interfaces as edge ports by default, use the **no** form of the command to disable the **edgeport** setting on the individual subinterface.

You should consider implementing the BPDU guard on edge ports. End devices should not participate in the spanning tree. However, a user running software that implements STP or RSTP can join spanning tree and disrupt the network. If the default priority setting on the user software is low, the end device can even become the root. The BPDU guard prevents the router interface from receiving BPDU messages from the end device. It also prevents the interface from being connected to an unauthorized switch.

You configure the BPDU guard on an individual logical interface with this command:

Syntax: spanning-tree bpduguard [enable | disable]

Use the **enable** option to activate the guard.

You can also configure the BPDU guard on all interfaces from the global configuration mode context:

Syntax: spanning-tree edgeport bpduguard default

You can then override this setting for an individual interface by entering this form of the command from the interface configuration mode context:

ProCurve(config-fr 1.1)# spanning-tree bpduguard disable

Configuring an Interface for a Point-to-Point Versus a Shared Connection. RSTP must know whether an interface uses a point-to-point or shared connection to implement sync.

Point-to-point interfaces use sync to rapidly transition from discarding to forwarding frames. One interface sends a BPDU proposing that it become the neighbor's designated switch. If the neighbor agrees, both interfaces become immediately active.

Interfaces on shared media, which reach more than one neighbor on the same connection, cannot exchange sync BPDUs to activate a connection.

By default, the ProCurve Secure Router uses the **auto** option to determine the connection type. RSTP assumes that full-duplex interfaces are point-to-point and half-duplex interfaces are shared.

If, for whatever reason, you must override this setting, move to the logical interface's configuration mode context and enter this command:

Syntax: spanning-tree link-type [auto | point-to-point | shared]

For example, the Ethernet interface 0/1 connects to a hub. Enter:

```
ProCurve(config-eth 0/1)# spanning-tree link-type shared
```

Altering Timers

Caution You should not alter spanning tree timers unless you have a great deal of experience working with spanning tree.

You configure the timers from the global configuration mode context. Use the commands shown in Table 10-3.

Table 10-3. Spanning Tree Timers

Timer	Function	Default	Range	Command Syntax
forward timer	minimum time between forwarding BPDUs	15 seconds	4 to 30	**spanning-tree forward-time <seconds>**
hello timer	time between hellos	2 seconds	0 to 10	**spanning-tree hello-time <seconds>**
maximum age timer	how long a BPDU remains valid	20 seconds	6 to 40	**spanning-tree max-age <seconds>**

Forward Timer. The forwarding interval determines how long a device waits to forward BPDUs. With STP, this setting determines how long the device stays first in the listening and then in the learning stage.

Hello Timer. Interfaces periodically transmit hellos. If an interface misses three hellos, neighbors assume the connection is down and send out TCN BPDUs to this effect. Take care when altering this timer because incompatible settings can cause devices to believe a connection is down when it is not.

Maximum Age Timer. BPDUs include a maximum age timer. Devices discard information received from a BPDU when this timer expires. With STP, the timer determines how long a device will wait to receive information about a connection from the root before assuming the connection is down.

Configuring STP

It is highly recommended that you implement RSTP, which can reduce network convergence time from more than a minute to less than a second. RSTP is fully compatible with STP, so the router can use it even when some devices on the local network only run STP. When an interface detects STP BPDUs, the router implements STP on that interface. (RSTP improvements will not be enabled for that segment of the network.)

However, the ProCurve Secure Router does support STP, if, for whatever reason, you decide to implement it.

To configure STP, you must:

- change the spanning tree version to STP

Move to the global configuration mode context and enter this command:

```
ProCurve(config)# spanning-tree mode stp
```

Syntax: spanning-tree mode [stp | rstp]

You can also:

- set the router's priority to influence the election of the root device
- set link cost to influence the selection of a link
- alter STP timers

You configure these options exactly as you would for RSTP. See "Determining Which Device Becomes Root: Setting the Router's Priority" on page 10-18, "Determining Which Links Are Chosen: Setting Link Cost" on page 10-18, and "Altering Timers" on page 10-22. When deciding on the root device, remember that it will be the only device to periodically flood BDPU.

Using the BPDU Filter to Disable STP or RSTP

The BPDU filter prevents interfaces from receiving and transmitting BPDUs. With it, you can remove the entire router from the spanning tree or you can remove a single interface.

In a test environment, the filter keeps all connections up so that you can test them.

Caution You should *not* use the global BPDU filter on a live network.

When you enable the filter from the global configuration mode context, the filter applies to all interfaces on the router. Enter this command:

Syntax: [no] spanning-tree edgeport bpdufilter default

To configure a interface to override the global BPDU filter, move to its interface configuration mode context and enter this command:

Syntax: spanning-tree bpdufilter [enable | disable]

The **enable** option removes the interface from the spanning tree. The **disable** option enables an interface to run a spanning tree protocol on a router that blocks it globally. Because the router should always run RSTP or STP, you will very rarely use this option.

Troubleshooting Spanning Tree

This section describes how to test and troubleshoot the router's spanning tree functions.

Note You must enter **show** and **debug** commands from the enable mode context or preface the command with **do**.

Testing Spanning Tree

You can run spanning tree debug commands to test a router's spanning tree functions. (Generally, you will not use these debug commands in a live network.) You can view debug messages to verify that:

- the router chooses the correct primary connection
- appropriate interfaces move quickly into the forwarding state
- when a connection goes down, the network converges within one or two seconds

The syntax for the **debug** commands is shown in Table 10-4.

Table 10-4. Spanning Tree debug Commands

View	Command Syntax
general messages	**debug spanning-tree general**
messages when configuration changes occur	**debug spanning-tree config**
periodic hellos and messages when a change in topology occurs	**debug spanning-tree events**
all BPDUs received	**debug spanning-tree bpdu receive**
all BPDUs transmitted	**debug spanning-tree bpdu transmit**
all BPDUs transmitted and received	**debug spanning-tree bpdu all**

The **debug spanning-tree events** command displays messages dealing with reconvergence when the network topology changes. When you enter the **debug spanning-tree** command with one of the **bpdu** options, the terminal displays a message every time an interface sends or receives a BPDU, or both.

Caution

The **debug spanning-tree events** and **debug spanning-tree bpdu** commands are particularly draining on the processor.

You can also use the BPDU debug commands to determine whether interfaces are actually participating in the spanning tree. If interfaces are not receiving BPDUs at all, you should check the running-config for an inadvertently applied BPDU guard or filter.

Addressing Common Spanning Tree Problems

Problems with spanning tree include slow convergence and routers selecting the wrong primary connection.

Some problems may be caused by other switches on the local network.

You can view information that will help you troubleshoot with this enable mode command:

Syntax: show spanning-tree [<*bridge group number*>] [realtime]

You enter the command without any options to view the following spanning tree information for all bridge groups:

- root ID
- timers
- bridge ID
- interfaces:
  - role
  - status

For example, Figure 10-6 displays the spanning tree instance for bridge group 1.

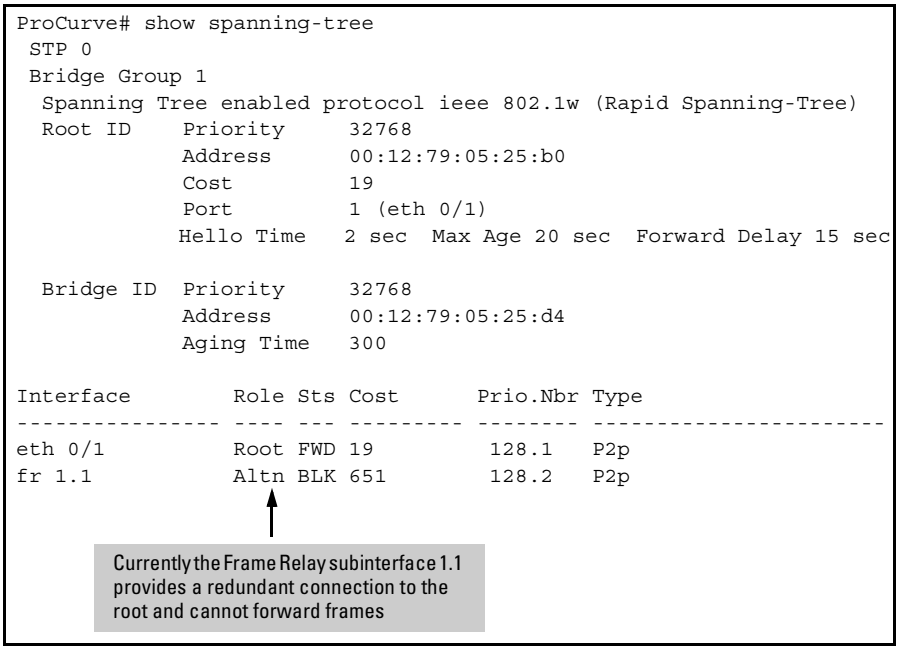

Figure 10-6. Viewing Spanning Tree Information

When the router supports more than one bridge, you may want to view only the information for the bridge group in question. Enter the command with the bridge group number.

You can enter the command with the **realtime** option to view periodic updates of the spanning tree information without re-entering the command. The CLI displays the information in a new screen. You can exit the screen by pressing **Ctrl+C**. You can also pause and restart the display of the updates. (See Figure 10-7).

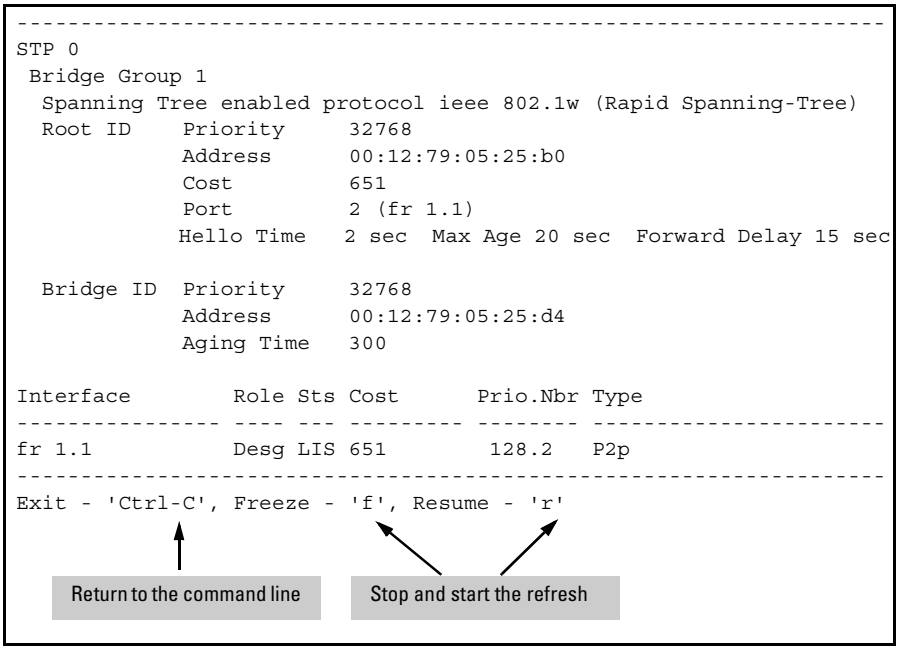

Figure 10-7. Viewing Real-Time Spanning Tree

Slow Convergence

The best way to solve slow convergence is to update all network devices from STP to RSTP.

When a router running RSTP connects to an STP device, it automatically runs STP on that interface. If you have recently updated network devices to RSTP, you may need to force connecting router interfaces to stop running STP. Use this enable mode command:

Syntax: clear spanning-tree detected-protocol [interface ethernet <*slot*>/<*port*>]

You can force the entire router to return to RSTP by simply entering **clear spanning-tree detected-protocol**. Or you can force the single interface that connects to the updated device. For example:

ProCurve# clear spanning-tree detected-protocol interface eth 0/1

Relatively slow convergence with RSTP may be caused by incorrectly configured point-to-point interfaces. View the status for each bridged interface and make sure that it is using full duplex. The router should automatically assign it the point-to-point role. If necessary, force this role by entering this command in the logical interface configuration mode context:

```
ProCurve(config-fr 1.1)# spanning-tree link-type point-to-point
```

Incorrect Path Selection

Devices may choose paths that seem illogical for several reasons:

- an end device or rogue device has been elected root
- connections are configured with an inappropriate cost
- a guard or filter has been applied to an interface

When an interface connects to an end device, enable the BPDU guard so that the router refuses BPDUs from it. Otherwise, software running on the device may cause it to be elected root. (You can view what device has actually been elected root with the **show spanning-tree** command.)

The router selects the primary connection according to which connection provides the lowest-cost link to the root. The **show spanning-tree** command displays which interfaces are active (status = FWD). You can force the router to select a specific connection by lowering its cost.

You can also assign two equivalent connections the same cost, but still have the router choose one as primary and one as redundant. Simply lower the port priority for the primary connection. (See “Determining Which Links Are Chosen: Setting Link Cost” on page 10-18.) Again, the **show spanning-tree** command displays the cost and priority for each interface in the bridge.

If an interface is not participating in the spanning tree, check the running-config for guards or filters that may have been inadvertently assigned to it. Also view the global spanning tree configuration and make sure that the global BPDU guard and/or filter has not been applied.

Quick Start

This section provides the commands you must enter to quickly configure the router to bridge traffic. Only a minimal explanation is provided.

If you need additional information about any of these options, see "Contents" on page 10-1 to locate the section that contains the explanation you need.

1. If you are using the bridge to extend a subnet to a remote site, move to the global configuration mode context and disable routing.

   ProCurve(config)# no ip routing

2. Create a bridge group.

   Syntax: bridge <*group number*> protocol ieee

3. Assign the Ethernet interface(s) to the bridge group from its interface configuration mode context.

   Syntax: bridge-group <*group number*>

4. Assign the WAN interface(s) to the bridge group. You can assign PPP and HDLC interfaces and Frame Relay and ATM subinterfaces to a bridge. Enter the following command from the logical interface configuration mode context:

   Syntax: bridge-group <*group number*>

   For example:

   ProCurve(config)# interface frame-relay 1.1

   ProCurve(config-fr 1.1)# bridge-group 1

5. If necessary, remove IP addresses from the WAN interfaces. For example:

   ProCurve(config-ppp 1)# no ip address 10.1.1.1 /30

The ProCurve Secure Router automatically implements RSTP on bridged Ethernet interfaces and Frame Relay and ATM subinterfaces. Usually, you will not need to make any further configurations. However, you can complete any of the following steps:

1. If so desired, change the spanning tree version from RSTP to STP. (RSTP is fully compatible with STP.) Move to the global configuration mode context and enter:

   Syntax: spanning-tree mode [rstp | stp]

2. If so desired, change the router's priority for becoming the root of the spanning tree.

   Syntax: spanning-tree priority <*value*>

   The value can be from 0 to 63535.

3. If so desired, configure the cost of the connections on the router from the logical interface for the connection.

   Syntax: spanning-tree path-cost <*value*>

   The cost can be from 1 to 63535. A higher cost lowers the chance that the connection will be chosen. For example:

   ```
   ProCurve(config-fr 1.1)# spanning-tree path-cost 60000
   ```

4. If a router interface connects to an edge device, configure the interface as an edge port and enable the BPDU guard. Move to the logical interface and enter:

   ```
   ProCurve(config-eth 0/1)# spanning-tree edgeport enable
   ProCurve(config-eth 0/1)# spanning-tree bpduguard enable
   ```

   For Frame Relay and ATM subinterfaces enter:

   ```
   ProCurve(config-fr 1.1)# spanning-tree edgeport
   ProCurve(config-fr 1.1)# spanning-tree bpduguard enable
   ```