Configuring the Data Link Layer Protocol for E1, T1, and Serial Interfaces

Contents

Configuring the Logical Interface 6-3
PPP Overview 6-4
Establishing a PPP Connection 6-4
Creating a PPP Interface on the ProCurve Secure Router 6-6
Configuring an IP Address for the WAN Connection 6-8
Activating the PPP Interface 6-10
Binding the Physical Interface to the Logical Interface 6-10
PPP Authentication 6-11
Additional Settings 6-16
Settings Explained in Other Chapters 6-18
Frame Relay Overview 6-19
Packet-Switching Network 6-20
Components of a Frame Relay Network 6-21
DLCI 6-22
Create the Frame Relay Interface 6-23
Activate the Frame Relay Interface 6-25
Define the Signaling Role 6-25
Define the Frame Relay Signaling Type 6-26
Configure Frame-Relay Counters 6-26
Create the Frame Relay Subinterface 6-28
Assign a DLCI to the Frame Relay Subinterface 6-28
Configure the IP Address for the WAN Connection 6-29
Set the CIR 6-35
Set the EIR 6-35
Bind the Physical Interface to the Logical Interface 6-36
Additional Settings 6-37
Settings Explained in Other Chapters 6-39

Configuring HDLC as the Data Link Layer Protocol 6-40
Create the HDLC Interface 6-40
Activate the HDLC Interface 6-42
Configure an IP Address for the WAN Connection 6-42
Bind the Physical Interface to the Logical Interface 6-44
Additional Settings 6-45
Settings Explained in Other Chapters 6-47
Example Networks 6-48

Configuring the Logical Interface

As outlined in *Chapter 4: Configuring E1 and T1 Interfaces*, all WAN connections—including E1- and T1-carrier lines—require both a Physical Layer and a Data Link Layer. (See Figure 6-1.) The Physical Layer encompasses:

- the transmission media and other infrastructure required to create and maintain the WAN connection
- the electrical signaling specifications for generating, transmitting, and receiving signals through the various transmission media

The Data Link Layer provides logical flow control for transmitting data between the peers of a WAN connection.

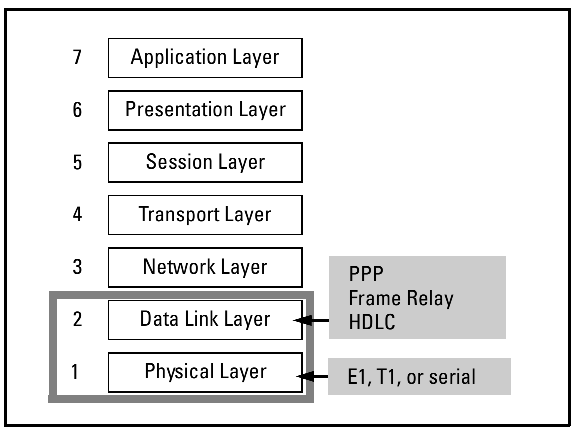

Figure 6-1. Data Link Layer Is Layer 2 in the OSI module.

The ProCurve Secure Router supports the following Data Link Layer protocols for E1, T1, and serial interfaces:

- Point-to-Point Protocol (PPP), including Multilink PPP (MLPPP)
- Frame Relay, including Multilink Frame Relay (MLFR)
- High-Level Data Link Control (HDLC)

For more information about MLPPP and MLFR, see the *Advanced Management and Configuration Guide*, *Chapter 2: Increasing Bandwidth.*

PPP Overview

PPP is a suite of protocols, rather than just a single protocol. (See Figure 6-2.) The PPP suite includes several types of protocols:

- link control protocol (LCP)
- authentication protocols
- network control protocols (NCPs)
- PPP

Each type of protocol has a specific role in establishing and maintaining a PPP connection

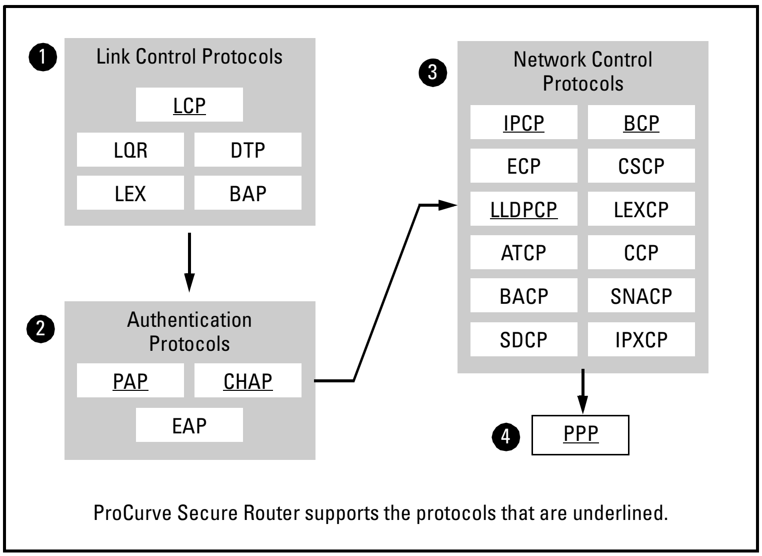

Figure 6-2. Protocols in the PPP Suite

Establishing a PPP Connection

When two peers try to establish a PPP connection, they must exchange protocols in the following order:

1. LCP
2. Authentication protocol
3. NCP
4. PPP

Exchanging an authentication protocol is optional.

Understanding how a PPP session is established can help you troubleshoot problems if they occur. (See Figure 6-3.)

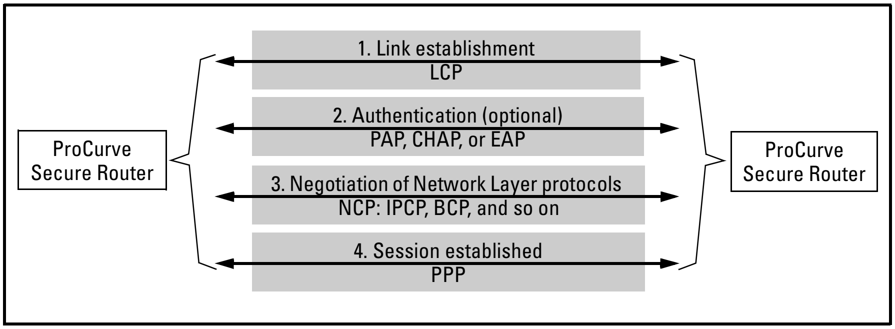

Figure 6-3. Establishing a PPP Link

Link Establishment. Two PPP peers exchange LCP frames to establish, configure, and test the WAN link. These frames allow the peers to determine if the link can accommodate the data they want to transfer. The LCP frames also contain a field called the *configuration option*. Configuration options inform the peer desired settings for the link such as the size of the PPP datagrams that will be sent and their degree of compression.

The two peers negotiate these settings. If the LCP frames do not contain a particular configuration option field, the peers use the default configuration for that option.

Authentication Protocol. If authentication is configured, the two peers authenticate the link. Although authentication is optional, the peers pass through this phase whether or not authentication is chosen.

PPP supports several authentication protocols:

- Password Authentication Protocol (PAP)
- Challenge Handshake Authentication Protocol (CHAP)
- Extensible Authentication Protocol (EAP)

The ProCurve Secure Router supports PAP and CHAP.

NCP. PPP uses an NCP to enable the exchange of Network Layer protocols—such as IP—across a WAN link. As Figure 6-2 shows, there is a specific NCP for each support Network Layer protocol. For example, the NCP for IP is IP Control Protocol (IPCP), and the NCP for IPX (which is a legacy Novell NetWare protocol) is IPX Control Protocol (IPXCP).

The ProCurve Secure Router supports the following NCPs:

- IP Control Protocol (IPCP)
- Bridging Control Protocol (BCP)
- Link-Layer Discovery Protocol (LLDP) Control Protocol (LLDPCP)

In order to exchange Network Layer protocols, the NCP must be in an “opened” state.

PPP. PPP frames carry the actual information being transferred over the WAN link. In PPP terminology, this information is called a datagram.

After the two peers successfully exchange LCP frames, authenticate the link (if authentication is configured), and negotiate the Network Layer protocol, a PPP session is established. The peers can then exchange PPP datagrams.

Creating a PPP Interface on the ProCurve Secure Router

To begin configuring PPP for an E1, T1, or serial interface, you must create a logical interface. From the global configuration mode context, enter:

Syntax: interface <*interface*> <*number*>

Replace ***<interface>*** with **ppp** and replace ***<number>*** with any number between 1 and 1024. Each PPP interface you configure on the router must have a unique number.

For example, if you are configuring the first PPP interface on the router, enter:

```
ProCurve(config)# interface ppp 1
```

The router prompt indicates that you have entered the PPP 1 interface configuration mode context:

```
ProCurve(config-ppp 1)#
```

You can enter the **?** help command to display the commands available from this configuration mode context.

```
ProCurve(config-ppp 1)# ?
```

Table 6-1 shows the main settings that you must configure for an E1, T1, or serial interface connection that uses PPP.

Table 6-1. Options for Configuring an E1, T1, or Serial Interface with PPP

Interface Configuration Mode Context	Command	Explanation	Page
e1	• **tdm-group <number> timeslots <range of numbers>** • **coding [ami | hdb3]** • **frame format [e1 | crc4]** • **clock source [internal | line | through]** • **no shutdown**	• defines the number of channels used for the E1 connection • defines the line coding • defines the frame format • defines the clock source, or timing, for the connection • activates the interface	4-10
t1	• **tdm-group <number> timeslots <range of numbers>** • **coding [ami | b8zs]** • **frame format [esf | d4]** • **clock source [internal | line | through]** • **lbo long <value> | lbo short <value>** • **no shutdown**	• defines the number of channels used for the T1 connection • defines the line coding • defines the frame format • defines the clock source, or timing, for the connection • sets the level of the transmit signal • activates the interface	4-10
serial	• **serial-mode [EIA530 | V35 | X21]** • **et-clock-source [txclock | rxclock]** • **no shutdown**	• configures the serial interface to support the appropriate cable • configures the serial interface to take the clock from the transmit signal (**txclock**) or from the receive signal, (**rxclock**) • activates the interface	5-12
ppp	• **ip address <A.B.C.D> [<subnet mask> | </prefix length>]** or • **ip address negotiated** or • **ip unnumbered <interface>** • **no shutdown**	• assigns a static IP address to the PPP interface • configures the PPP interface to negotiate an IP address from its peer • configures the PPP interface to use the IP address assigned to another interface • activates the interface	6-8
global configuration or interface configuration	• **bind <number> <physical interface> <slot>/<port> [<tdm-group number>] ppp <interface number>**	• binds the physical interface to the PPP interface • requires a tdm-group number for T1 and E1 interfaces (but not for serial interfaces)	6-10

The PPP settings are described in the sections that follow. (For information about E1 and T1 interface settings, see *Chapter 4: Configuring E1 and T1 Interfaces*. For information about serial interface settings, see *Chapter 5: Configuring Serial Interfaces for E1- and T1-Carrier Lines*.)

Configuring an IP Address for the WAN Connection

You configure the IP address for the E1 or T1 WAN connection on the PPP interface rather than on the physical interface. There are several ways to assign an IP address to the PPP interface:

- assign a static IP address
- configure the PPP interface to negotiate an IP address with its PPP peer
- configure the PPP interface as an unnumbered interface

Note

If the PPP interface is part of a bridge and IP routing is disabled, you can configure the PPP interface as a Dynamic Host Configuration Protocol (DHCP) client.

Configure a Static IP Address. To assign the PPP interface a static IP address, enter the following command from the PPP interface configuration mode context:

Syntax: ip address <*A.B.C.D*> <*subnet mask* | */prefix length*>

For example, you might enter:

```
ProCurve(config-ppp 1)# ip address 10.1.1.1 255.255.255.252
```

For subnet mask, you can enter the complete subnet mask or the classless inter-domain routing (CIDR) notation. For example, you might enter:

```
ProCurve(config-ppp 1)# ip address 10.1.1.1 /30
```

Configure a Negotiated IP Address. If you are using your WAN connection for Internet access, your Internet Service Provider (ISP) may want you to configure the PPP interface so that it negotiates the IP address with the ISP's router. From the PPP interface configuration mode context, enter:

Syntax: ip address negotiated [no-default]

Include the **no-default** option if you do not want the router to accept a default route from the PPP peer that is providing the IP address.

Configure the PPP Interface as an Unnumbered Interface. To conserve IP addresses on your network, you may want to create the PPP interface as an unnumbered interface. When you assign a logical interface on the router an IP address, that IP address cannot overlap with the IP addresses that are assigned to other logical interfaces. As a result, each interface that has an IP address represents an entire subnet. Depending on the subnetting scheme you use, this could use more IP addresses than you can spare.

You can configure the PPP interface (and other interfaces on the ProCurve Secure Router) as an unnumbered interface. The PPP interface will then use the IP address of another interface—the interface you specify. The Secure Router OS uses the IP address of the specified interface when sending route updates over the unnumbered interface.

Before configuring the PPP interface as an unnumbered interface, you should be aware of a potential disadvantage: If the interface to which the IP address is actually assigned goes down, the PPP interface will be unavailable as well. For example, suppose you configure the PPP 1 interface as an unnumbered interface that takes its IP address from the Ethernet 0/1 interface. If the Ethernet 0/1 interface goes down, the PPP 1 interface will also be unavailable.

To minimize the chances that the interface with the IP address will go down, you can assign the IP address to a loopback interface, which typically does not go down.

To configure the PPP interface as an unnumbered interface, enter the following command from the PPP interface configuration mode context:

Syntax: ip unnumbered <*interface*>

Valid interfaces include:

- Ethernet interfaces and subinterfaces
- Frame Relay subinterfaces
- other PPP interfaces
- HDLC interfaces
- loopback interfaces
- Asynchronous Transfer Mode (ATM) subinterfaces
- demand interfaces

For example, you would enter the following commands to configure a loopback interface and then configure the PPP 1 interface to use the IP address assigned to that loopback interface:

```
ProCurve(config)# interface loopback 1
ProCurve(config-loop 1)# ip address 10.1.2.2 /30
ProCurve(config-loop 1)# interface ppp 1
ProCurve(config-ppp 1)# ip unnumbered loopback 1
```

Note

You do not have to enter **no shutdown** to activate a loopback interface. The status of a loopback interface changes to up after you enter the **interface loopback *<interface number>*** command.

Activating the PPP Interface

To activate the PPP interface, enter the following command from the PPP interface configuration mode context:

```
ProCurve(config-ppp 1)# no shutdown
```

Although the PPP interface is activated, its status will not change to up until it is bound to a physical interface. It can then begin to negotiate a PPP session with its peer, and if that negotiation is successful, the status of the PPP interface will change to up.

Binding the Physical Interface to the Logical Interface

On the ProCurve Secure Router, you must bind the physical interface to the logical interface so that the router knows which Data Link Layer protocol to use for that WAN connection. When you bind a physical interface to a logical interface, the two are considered a single interface bind group.

From the global configuration mode context, enter:

Syntax: bind <*bind number*> <*physical interface*> <*slot*>/<*port*> [<*tdm-group number*>] <*logical interface*> <*logical interface number*>

You can also enter this command from the PPP interface configuration mode context.

Replace <*bind number*> with a number that is globally significant. That is, each **bind** command you enter on the router must have a unique bind number.

Replace <*physical interface*> with the type of WAN connection, such as E1, T1, or serial. Replace <*slot*> and <*port*> with the correct numbers to identify this interface's location on the ProCurve Secure Router.

If you are binding an E1 or T1 interface to the PPP interface, replace <*TDM-group number*> with the TDM group number you created on the E1 or T1 interface. If you are binding a serial interface to the PPP interface, omit this option.

Note You do not include a TDM group number when binding a serial interface to a logical interface because the serial interface does not use TDM groups.

Replace <*logical interface*> with **ppp** and replace < *logical interface number*> with the number you assigned to this interface. For example, if you want to bind the E1 1/1 interface or the T1 1/1 interface to the PPP 1 interface, enter:

```
ProCurve(config)# bind 1 e1 1/1 1 ppp 1
```

or

```
ProCurve(config)# bind 1 t1 1/1 1 ppp 1
```

If you want to bind the serial 1/1 interface to the PPP 1 interface, enter:

```
ProCurve(config)# bind 1 ser 1/1 ppp 1
```

To see an example configuration that uses PPP, see "Example Networks" on page 6-48.

PPP Authentication

You can increase the security of your WAN by requiring the PPP peer at the other end of the link to vouch that it is, indeed, the authorized router at the remote site. You can also configure the router to provide its own authentication information. Many Internet service providers (ISPs) require authentication so that they grant service only to subscribers who have paid for it.

The ProCurve Secure Router supports two authentication protocols for PPP:

- PAP
- CHAP

PAP. PAP is the simplest possible authentication scheme. It requires a two-way message exchange. One peer sends the password previously agreed upon to the other peer, which is called the authenticator. The authenticator looks up the password in its database. If the password matches, the authenticator returns an authentication acknowledge. The two peers can then send NCPs to negotiate the Network Layer protocols. If this negotiation is successful, the PPP session is established.

With PAP, the two peers authenticate only once, and the username and password are sent in clear text across the connecting private circuit. Because PAP sends the password directly over the wire, anyone capable of tapping into the wire can intercept it.

CHAP. CHAP solves the security problem of PAP by hashing the password and sending the hash value instead of the password over the wire. CHAP follows the process shown in Figure 6-4:

1. The authenticator challenges the peer.
2. The peer combines its password with a string of text and calculates a hash value using the Message Digest 5 (MD5) algorithm. (The password is irreversibly encrypted.) The peer sends the hash value to the authenticator.
3. The authenticator knows both the agreed-upon string of text and the peer's password. The authenticator performs the same hashing calculation and compares its hash value to the hash value sent by the peer.
4. If the hash values match, the authenticator acknowledges the peer, and the peers proceed to exchange NCPs. If the hash values do not match, the authenticator continues to issue challenges until the peer returns a matching hash value or runs out of retry attempts.

Because the encryption prevents hackers from hijacking a password, CHAP provides increased security. In addition, CHAP requires peers to reauthenticate themselves from time to time.

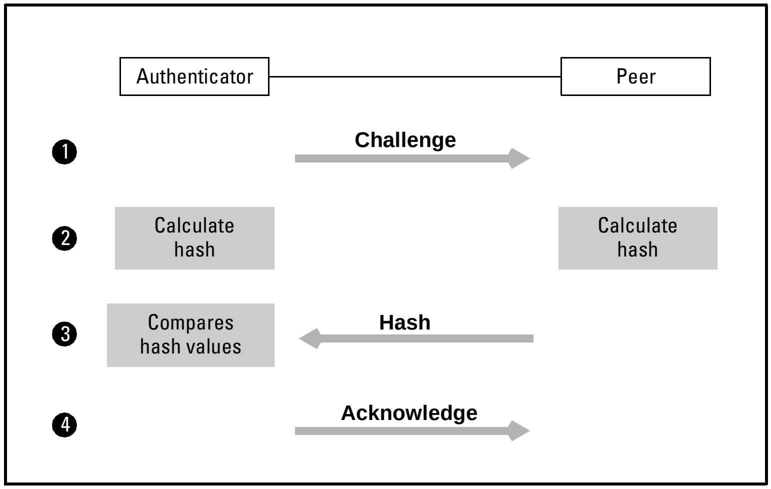

Figure 6-4. CHAP Process

When you configure CHAP on the ProCurve Secure Router, you only need to set the password. The router automatically sends the hostname for the username and computes the hash value.

Requiring a Peer to Authenticate Itself. When you configure PPP authentication on the ProCurve Secure Router, you must first choose whether you want to use PAP or CHAP. To require authentication, you must:

- enable PAP or CHAP on the connection
- set the peer's username and password

You configure authentication for an individual PPP connection. Move to the logical interface for the connection and specify the type of authentication:

Syntax: ppp authentication [chap | pap]

For example, if you want to use CHAP for the PPP 1 interface, enter:

```
ProCurve(config-ppp 1)# ppp authentication chap
```

Note

Both sides of the connection do *not* have to require authentication. However, if both sides require authentication, they must use the same protocol. If your peer requires authentication, you must set the username and password the router will *send*. (These are distinct from the username and password that the router accepts.) See "Setting a PAP Username and Password" on page 6-14 and "Setting a CHAP Username and Password" on page 6-15.

You must add the password you have agreed upon for the peer to the PPP database. The PPP database for each connection is separate and distinct from the global username and password database and the databases of other PPP connections. Because the database is for a point-to-point connection, it stores only one username and password. You manage the database for a PPP connection from its logical interface configuration mode context.

To set the username and password that the ProCurve Secure Router *accepts* from a peer, enter the following command from the global configuration mode context:

Syntax: username <*username*> password <*password*>

For example, you might enter:

```
ProCurve(config-ppp 1)# username SiteB password procurve
```

For CHAP, the username should be the hostname of the peer.

Authenticating to a Peer. The device at the other end of a PPP connection may require the ProCurve Secure Router to authenticate itself. To configure the local router, you must:

- configure which authentication protocol to use
- set the username and password

The authentication protocol must match that requested by the peer. If you do not know which protocol the peer is using, you can view the debug messages and look for PAP or CHAP. From the enable mode context, enter:

```
ProCurve# debug ppp authentication
```

You specify the authentication protocol with the same command that you enter to configure the username and password that the ProCurve Secure Router sends the PPP peer. The company or ISP that is requiring PPP authentication should provide you with the username and password, which are case sensitive.

Setting a PAP Username and Password. To configure PAP authentication information for a WAN connection, you must move to the configuration mode context for the logical interface that provides the Data Link Layer for the connection. To set the username and password that the router will send in clear text over the wire, enter:

Syntax: ppp pap sent-username <*username*> password <*password*>

For example, you might enter:

```
ProCurve(config-ppp 1)# ppp pap sent-username SiteA password procurve
```

Note

PAP will be used only to authenticate this WAN connection. You do not have to actually enable the PAP protocol. It is perfectly acceptable for the local router to authenticate itself to a peer without requiring that peer to authenticate itself in turn.

Setting a CHAP Username and Password. You configure the router to authenticate itself from the PPP interface configuration mode context for the connection. For CHAP, you only have to set the password that the router will hash and send encrypted to the peer. Enter:

Syntax: ppp chap password <*password*>

The peer or ISP should provide this password. For example:

```
ProCurve(config-ppp 1)# ppp chap password procurve
```

The router automatically sends its hostname for its username. Make sure that this hostname actually matches that by which the peer identifies your router. (This can be particularly important when authenticating to an ISP.) If necessary, you can override the hostname with a different username by entering:

Syntax: ppp chap hostname <*username*>

For example, you might enter:

```
ProCurve(config-ppp 1)# ppp chap hostname ProcurveA
```

Recording PPP Authentication Information. If you are configuring PPP authentication, you may want to print Table 6-2 and enter the information for your router.

Table 6-2. PPP Authentication Worksheet

Option	Your Setting
PPP interface number	
authentication protocol	
Are you requiring the peer to authenticate itself?	Yes/No
peer username	

Option	Your Setting
peer password	
Are you authenticating to the peer?	Yes/No
local router's username	
local router's password	

This worksheet will help you enter the PPP authentication command for your router.

Additional Settings

Depending on your company's WAN environment, you may want to configure other settings on the PPP interface.

Configure a Secondary IP Address for the Interface. You can configure a secondary IP address on an interface if the interface supports more than one subnet. For example, the LAN you connect to an Ethernet interface may require more IP addresses than the primary subnet can provide.

Note

When using secondary IP addresses, avoid routing loops by verifying that all devices on the network segment are configured with secondary IP addresses on the secondary subnet.

From the PPP interface configuration mode context, enter:

Syntax: ip address <*A.B.C.D*> <*subnet mask* | */prefix length*> secondary

Replace <*A.B.C.D*> with the secondary IP address and replace <*subnet mask*> with the corresponding subnet mask. Instead of specifying a subnet mask, you can replace <*/prefix length*> with the CIDR notation. Finally, include the **secondary** option.

For example, you might enter:

```
ProCurve(config-ppp 1)# ip address 192.168.115.1 255.255.255.0 secondary
```

You can include an unlimited number of secondary IP addresses.

To remove a secondary IP address, enter:

Syntax: no ip address <*A.B.C.D*> <*subnet mask* | */prefix length*> secondary

Set the MTU. The maximum transmission unit (MTU) defines the largest size that a PPP frame can be. If a frame exceeds this size, it must be fragmented. By default, the MTU for PPP interfaces is 1500 bytes. To change this setting, enter:

Syntax: mtu <*size*>

Replace <*size*> with a number between 64 and 1520.

For most environments, you should leave the MTU at 1500. In some cases, however, you may need to adjust the MTU size. For example, you need to evaluate MTU size if:

- The interface is connected to another router that uses a different MTU size.
- The interface is used in a PPP over Ethernet (PPPoE) environment. (For more information about PPPoE, see *Chapter 7: ADSL WAN Connections*.)

If two PPP peers use different MTU sizes, this mismatch can affect transmissions and routing. For example, if the PPP peer has a smaller MTU and your router sends a frame that exceeds that size, the PPP peer will have to fragment the frame. If the frame is tagged with the "do not fragment" field, then the router cannot forward the frame.

If you have enabled Open Shortest Path First (OSPF) routing on the ProCurve Secure Router, you should be especially careful when setting the MTU. OSPF routers cannot become adjacent if their MTU sizes do not match. You should ensure that the MTU on the router at the far-end of the PPP connection is using the same MTU as the router you are configuring.

You may also need to configure the MTU for PPPoE. When two devices initiate a PPPoE session, they negotiate an MTU of 1492 bytes because the payload of an Ethernet frame cannot exceed 1500 bytes. With the overhead created by PPP, the PPPoE frame is 1500 bytes.

Typically, the two PPPoE devices will negotiate the MTU size of 1492. If there are problems, however, you may need to manually configure the MTU.

Adding a Description. You can add a description to the PPP interface if you want to document information about it. For example, if you have configured multiple PPP interfaces, you may want to document how each PPP interface is being used. To create a description, enter:

Syntax: description <*line*>

Replace <*line*> with a phrase up to 80 characters. For example, you might enter:

```
ProCurve(config-ppp 1)# description WAN link to Denver office
```

This description is displayed only when you enter the s**how running-config** command. From the enable mode context, enter:

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

You must then scroll through the running-config to find the **interface ppp 1** heading. To view only the running-config for the PPP 1 interface, enter:

```
ProCurve# show running-config interface ppp 1
```

Configuration information such as the following is displayed:

```
interface ppp 1
  description WAN link to Denver office
  ip address 192.168.1.1 255.255.255.0
  bind 1 ser 1/1 ppp 1
  no shutdown
```

Settings Explained in Other Chapters

In addition to configuring these settings for the PPP interface, you can:

- assign access control policies (ACPs) or access control lists (ACLs) to the PPP interface
- assign crypto maps to enable virtual private networks (VPNs)
- configure settings for routing protocols
- enable bridging

Table 6-3 lists additional configurations that you can enter from the PPP interface and the page number where you find information about those configurations.

Table 6-3. Additional Configuration Settings for the PPP Interface

Settings	Configuration Guide	Page Number
access controls to filter incoming and outgoing traffic	Advanced	5-19, 5-38
bridging	Basic	10-6
VPNs	Advanced	10-46
routing commands for OSPF, RIP, or BGP	Advanced	15-1
quality of service settings	Advanced	8-28

Frame Relay Overview

For companies that can accept lower transmission speeds during peak usage times, Frame Relay provides a more affordable WAN solution than a dedicated E1- or T1-carrier line. Frame Relay can run over a variety of physical WAN connections, including E1- and T1-carrier lines. Whatever the physical WAN connection is, Frame Relay allocates bandwidth on that connection dynamically. As a result, public carriers provide a subscriber with bandwidth only when that subscriber requires it.

Frame Relay cuts costs both for public carriers and subscribers because it minimizes idle bandwidth: Public carriers can allocate the same bandwidth to multiple subscribers, and subscribers do not pay for bandwidth that they do not use.

When companies purchase Frame Relay service, they negotiate a Service Level Agreement (SLA) that specifies a Committed Information Rate (CIR), the amount of bandwidth they can use. The CIR is contractually guaranteed bandwidth, rather than physically guaranteed as with dedicated E1- or T1-carrier lines. If Frame Relay carriers do not provide the CIR, however, they can be fined. Consequently, carriers usually ensure that the bandwidth stipulated in the CIR is available to the customer. (See Figure 6-5.)

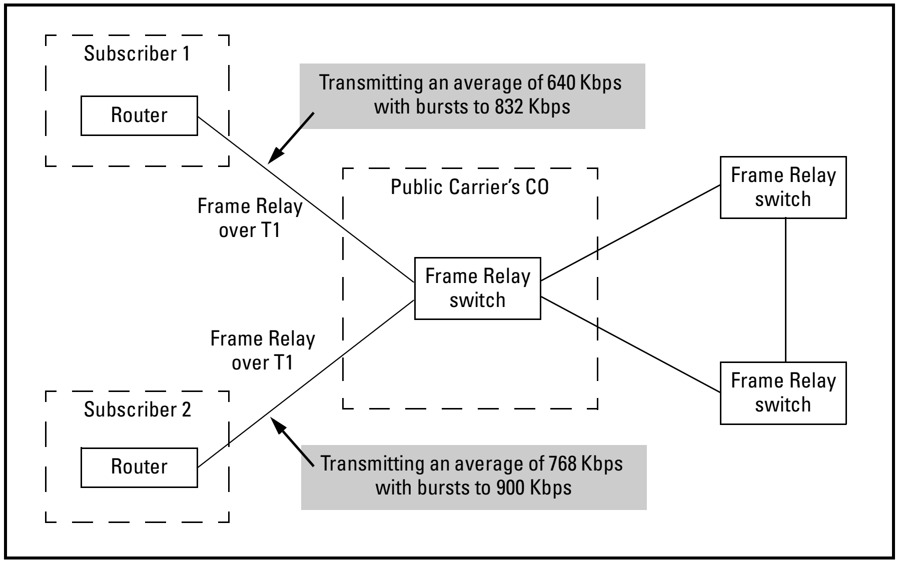

Figure 6-5. A Frame Relay Network Dynamically Allocates Bandwidth.

Packet-Switching Network

Frame Relay transfers data through multiple nodes in a shared network using packet switching. Frame Relay divides data into frames, and each frame travels through the network individually, passing from one Frame Relay switch to another in a non-fixed path, until the frames are reassembled at their destination.

Although frames can take multiple and variable paths through a shared network, two routers, which are identified by administratively assigned circuit IDs, define the fixed endpoints to a permanent virtual circuit (PVC). In a Frame Relay network, a PVC is a logical connection between two sites. (See Figure 6-6.)

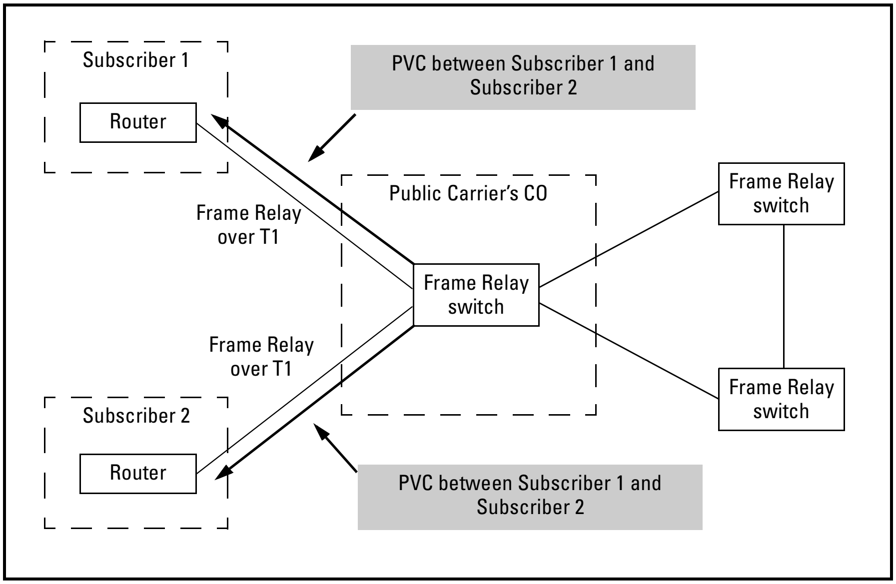

Figure 6-6. A PVC Connects Two Endpoints in the Frame Relay Network.

Components of a Frame Relay Network

The Frame Relay network consists of several components, each of which has a specific role.

- user, or data terminal equipment (DTE)
- network, or data communications equipment (DCE)
- network-to-network interfaces (NNI)
- user-to-network interfaces (UNI)

When you configure Frame Relay on the ProCurve Secure Router, you must define the role that the router will perform in the Frame Relay network. (See Figure 6-7.)

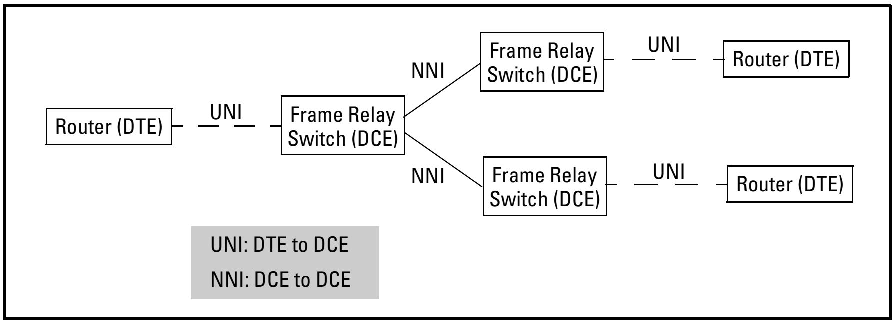

Figure 6-7. Components in a Frame Relay Network

DTE. The DTE receives data from the LAN in the form of multiple protocol packets and encapsulates each packet into a Frame Relay frame. The header of such a frame is called the Data Link Connection Identifier (DLCI) and contains the frame's ultimate destination.

You can configure the DTE to manage congestion and maintain quality of service. For example, the DTE can manipulate the actual size of each frame sent through the network. It also can buffer and fragment packets to reserve bandwidth for particular circuits and ensure quality of service for time-sensitive packets such as voice applications.

DCE. in a Frame Relay network, the DCE is the Frame Relay switch, which establishes and maintains the Frame Relay connection. After receiving frames from the DTE, the DCE converts these frames into signals supported by the physical media of the network. The DCE also reads the DLCI on incoming packets, checks its switch lookup table, and then forwards data to the appropriate outgoing port—which leads to the correct virtual endpoint.

UNI. UNIs connect the DTE to the DCE and provide access to the Frame Relay network.

NNI. NNIs connect a DCE to a DCE, using bidirectional signaling. That is, NNIs connect one Frame Relay switch to another.

DLCI

As mentioned earlier, the DTE marks each outgoing frame with a DLCI, a 10-bit field in the Address Field of the Frame Relay header. The switch reads the DLCI to determine the appropriate PVC endpoint to which to send the frame. DLCIs are locally, not globally, significant. (See Figure 6-8.)

The 10-bit field enables 1024 possible DLCI numbers, but some are reserved for special purposes:

- 0 signals Annex A and D
- 1-15 and 1008-1022 are reserved
- 1023 signals the Link Management Interface (LMI)

The remaining 976 DLCI numbers between 16 and 1007 are available to users. Your Frame Relay service provider will assign you a DLCI.

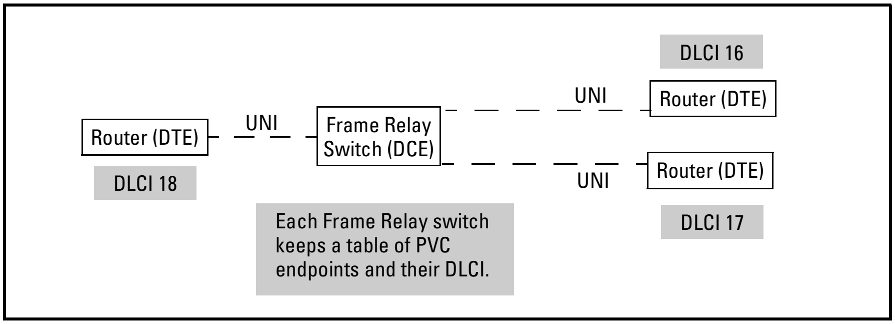

Figure 6-8. The DLCI Identifies the PVC Endpoint.

Create the Frame Relay Interface

To begin configuring Frame Relay as the Data Link Layer protocol for an E1, T1, or serial interface, you must create a logical interface. From the global configuration mode context, enter:

Syntax: interface <*interface*> <*number*>

Replace <*interface*> with **frame-relay**; you can also use the shortcut **fr**. Replace <*number*> with any number between 1 and 1024. Each Frame Relay interface that you create on the router must have a unique number.

For example, if you are configuring the first Frame Relay interface on the router, you might enter:

```
ProCurve(config)# interface frame-relay 1
```

The router prompt indicates that you have entered the proper interface configuration mode context:

```
ProCurve(config-fr 1)#
```

From this configuration mode context, you can enter the **?** help command to display the commands available from this configuration mode context.

ProCurve(config-fr 1)# ?

Table 6-4 shows the main settings that you must configure for an E1, T1, or serial interface that uses Frame Relay.

Table 6-4. Frame Relay Configuration Options

Interface Configuration Mode Context	Command	Description	Page
e1	• **tdm-group <number> timeslots <range of numbers>** • **coding [ami | hdb3]** • **frame format [e1 | crc4]** • **clock source [internal | line | through]** • **no shutdown**	• defines the number of channels used for the E1 connection • defines the line coding • defines the frame format • defines the clock source, or timing, for the connection • activates the interface	4-10
t1	• **tdm-group <number> timeslots <range of numbers]** • **coding [ami | b8zs]** • **frame format [esf | d4]** • **clock source [internal | line | through]** • **no shutdown**	• defines the number of channels used for the T1 connection • defines the line coding • specifies frame format • defines the clock source • activates the interface	4-10
serial	• **serial-mode [EIA530 | V35 | X21]** • **et-clock-source [txclock | rxclock]** • **no shutdown**	• configures the serial interface to support the appropriate cable • configures the serial interface to take the clock from the transmit signal, txclock, or from the receive signal, rxclock • activates the interface	5-12
frame-relay interface	• **no shutdown** • **frame-relay intf-type [dte | dce | nni]** • **frame-relay lmi-type [ansi | auto | cisco | none | q933a]**	• activates the interface • defines the signaling role as user, network, or both • defines Frame Relay signaling type	6-25

Interface Configuration Mode Context	Command	Description	Page
frame-relay subinterface	• **frame-relay interface-dlci <dlci>** • **ip address <A.B.C.D> <subnet mask | /prefix length>** or • **ip address dhcp {client-id [ethernet 0/ <port> | HH:HH:HH:HH:HH:HH:HH] | hostname <word>}** • **ip address dhcp [hostname <word> | no-default-route | no-domain-name | no-nameservers]** or • **ip unnumbered <interface>**	• defines the DLCI for the PVC • defines a static IP address for the interface • configures the Frame Relay subinterface as a DHCP client • configures the Frame Relay as an unnumbered interface, which takes its IP address from another interface	6-28
global configuration or interface configuration	• **bind <number> <physical interface> <slot>/ <port> [<tdm-group number>] Frame Relay <interface number>**	• binds the physical interface to the logical interface • requires tdm-group number for E1 and T1 interfaces (but not for serial interfaces)	6-36

The Frame Relay settings are described in the sections that follow.

Activate the Frame Relay Interface

You must activate the Frame Relay interface. From the Frame Relay interface configuration mode context, enter:

```
ProCurve(config-fr 1)# no shutdown
```

Define the Signaling Role

You must configure the signaling role that the ProCurve Secure Router will fulfill in the Frame Relay network. With few exceptions, the ProCurve Secure Router will function as the user, or DTE, and consequently, this is the default setting.

However, the other options are available if you should ever need to change the signaling role. For example, if you are setting up a test WAN to determine if your applications will run over a Frame Relay connection, you may need to configure the router as a DCE.

To configure the signaling role, enter the following command from the Frame Relay interface configuration mode context:

Syntax: frame-relay intf-type [dte | dce | nni]

Define the Frame Relay Signaling Type

You must configure the Frame Relay interface to use the same signaling type that your Frame Relay service provider uses. From the Frame Relay interface configuration mode context, enter:

Syntax: frame-relay lmi-type [ansi | auto | cisco | none | q933a]

Table 6-5 maps the Frame Relay signaling type to the setting that you must enter for the **frame-relay lmi-type** command.

Table 6-5. Frame Relay Signaling

Signaling type	Option	Complete Command
Annex D	**ansi**	**frame-relay lmi-type ansi**
detect signaling type from incoming message	**auto**	**frame-relay lmi-type auto**
Cisco LMI	**cisco**	**frame-relay lmi-type cisco**
no signaling (disables signaling role as well)	**none**	**frame-relay lmi-type none**
Annex A	**q933a**	**frame-relay lmi-type q933a**

For example, to set the signaling type to **auto**, enter the following command from the Frame Relay interface configuration mode context:

```
ProCurve(config-fr 1)# frame-relay lmi-type auto
```

The default setting is **ansi**.

Configure Frame-Relay Counters

The Frame Relay counters monitor status polls sent and received, track errors, and change the endpoint's signaling status from up to down, depending on the number of errors counted within a set frame of events. Although you can tailor the counter settings to your system, most applications do not require special settings, so you should keep the default settings.

Table 6-6 lists the Frame Relay counters, the possible settings, and the polls that each one controls.

Table 6-6. Frame Relay Counters

Frame Relay Counter	Possible Settings	Default Setting	Description
frame-relay lmi-n391dce <polls>	1-255	6	Configure how many link integrity polls occur in between the full-status polls. Configure this setting for the DCE endpoint.
frame-relay lmi-n391dte <polls>	1-255	6	Configure how many link integrity polls occur between the full status polls. Configure this setting for the DTE endpoint.
frame-relay lmi-n392dce <threshold>	1-10	3	Configure an error threshold number for the DCE. If the error threshold is met, the signaling status is changed to down, which indicates a service-affecting condition. This condition is cleared after this number of consecutive error-free N393 events are received.
frame-relay lmi-n392dte <threshold>	1-10	3	Configure an error threshold number for the DTE. If the error threshold is met, the signaling status is changed to down, which indicates a service-affecting condition. This condition is cleared after this number of consecutive error-free N393 events are received.
frame-relay lmi-n393dce <counter>	1-10	4	Configure the LMI- monitored event counter for the DCE endpoint.
frame-relay lmi-n393dte <counter>	1-10	4	Configure the LMI- monitored event counter for the DTE endpoint.
frame-relay lmi-t391dte <seconds>	5-30 seconds	10 seconds	Set the T391 signal-polling timer for the DTE endpoint.
frame-relay lmi-t392dce <seconds>	5-30 seconds	10 seconds	Set the T392 polling-verification timer for the DCE endpoint.

You can use the **no** command to return counters to their default settings.

Create the Frame Relay Subinterface

You must create a Frame Relay subinterface for each PVC that you want to establish through this Frame Relay interface. To create a Frame Relay sub-interface, enter the following command from the global configuration context or from the Frame Relay interface configuration mode context:

Syntax: interface frame-relay <*number.subinterface number*>

Replace the first *number* in <*number.subinterface number*> with the number of the Frame Relay interface that you have already configured. Then replace *subinterface number* with any number between 16 to 1007. Using the same number as the subinterface's DLCI can help you keep track of the subinterface and troubleshoot any errors.

For example, if your public carrier has assigned your company a DLCI of 16, enter:

```
ProCurve(config-fr 1)# interface frame-relay 1.16
```

You are then moved to the Frame Relay subinterface configuration mode context, which is reflected in the router prompt:

```
ProCurve(config-fr 1.16)#
```

From the Frame Relay subinterface configuration mode context, you can configure a variety of settings for the connection, including the MTU size and excess burst rate. However, to initially establish the sublink, you only need to assign it a DLCI.

Assign a DLCI to the Frame Relay Subinterface

The Frame Relay service provider assigns each PVC endpoint a DLCI on the Frame Relay switch, and the switch maintains a table of each DLCI so that it can pass traffic through an outbound port uniquely associated with a specific peer. Your Frame Relay service provider should tell you the DLCI for the PVC.

To assign the DLCI to the Frame Relay interface, enter the following command from the Frame Relay subinterface configuration mode context:

Syntax: frame-relay interface-dlci <*DLCI*>

Replace <*DLCI*> with a valid DLCI number, ranging from 16 to 1007. You must assign a different DLCI to each PVC established on the same router.

For example, if the Frame Relay service provider assigned your company a DLCI of 16, enter:

```
ProCurve(config-fr 1.16)# frame-relay interface-dlci 16
```

Configure the IP Address for the WAN Connection

You configure the IP address for the WAN connection on the Frame Relay subinterface, rather than on the physical interface or the Frame Relay interface. There are several ways to assign an IP address to the Frame Relay subinterface:

- assign a static IP address
- configure the Frame Relay subinterface as a DHCP client
- configure the Frame Relay subinterface as an unnumbered interface

Configuring a Static IP Address. From the Frame Relay subinterface configuration mode context, enter:

Syntax: ip address <*A.B.C.D*> <*subnet mask* | /*prefix length*>

For <*subnet mask*>, you can enter the complete subnet mask or replace </*prefix length*> with the CIDR notation. For example, you might enter:

```
ProCurve(config-fr 1.16)# ip address 10.10.2.1 /30
```

Configure the Frame Relay Subinterface as a DHCP Client. Your Frame Relay service provider may want to dynamically assign your router an IP address for each Frame Relay PVC. To enable a Frame Relay subinterface to use DHCP to obtain an IP address, use one of the following commands:

Syntax: ip address dhcp {client-id [ethernet 0/<*port*> | HH:HH:HH:HH:HH:HH:HH] | hostname <*word*>} [track <*name*>] [<*administrative distance*>]

Syntax: ip address dhcp [hostname <*word*> | no-default-route | no-domain-name | no-nameservers] [track <*name*>] [<*administrative distance*>]

In addition to enabling the DHCP client for the Frame Relay subinterface, this command allows you to configure the settings shown in Table 6-7.

Table 6-7. Default Settings for the DHCP Client

Option	Use	Default Setting
client-id	configures the client identifier displayed in the DHCP server's table	media type and interface's MAC address
hostname	configures the hostname displayed in the DHCP server's table	router hostname
no-default-route	specifies that the DHCP client should not accept the default route obtained through DHCP	accept default route from the DHCP server
no-domain-name	specifies that the DHCP client should not accept the domain name included with the other lease settings that the DHCP server sends	accept the domain name setting from the DHCP server
no-nameservers	specifies that the DHCP client should not accept the DNS setting included with the other lease settings that the DHCP server sends	accept DNS settings from the DHCP server
track	attaches a network monitoring track to the DHCP client	(none)
<administrative distance>	specifies the administrative distance to use when adding the DHCP gateway into the route table	1

Before you enable the DHCP client, you must decide whether or not you want to configure the settings listed in Table 6-7, and you must then include the settings in the same command that you enter to enable the DHCP client. After you enable the DHCP client, it immediately begins to search for a DHCP server and negotiate a lease. You cannot impose settings on that lease after it is established.

Accepting the Default Settings. If you want to use default DHCP settings for the Frame Relay subinterface, you can simply enter:

```
ProCurve(config-fr 1.16)# ip address dhcp
```

The DHCP client on the Frame Relay subinterface will immediately begin to send DHCP discovery message to find a DHCP server. When a DHCP server responds, the client will negotiate an IP address.

The DHCP client will send DHCP discovery messages whether or not the Frame Relay subinterface is activated or a valid connection has been established. It will continue to send DHCP discovery messages until a DHCP server responds.

You should ensure that the DHCP client receives an IP address so that these discovery messages do not consume router resources or bandwidth on your Frame Relay link. To determine whether the Frame Relay subinterface has been assigned an IP address, enter the following command from the enable mode context:

ProCurve# show interface frame-relay <*number.subinterface number*>

Configuring a Client Identifier. By default, the Secure Router OS populates the client identifier with the media type and the interface's media access control (MAC) address. You can specify that the DHCP client uses the MAC address of an Ethernet port, or you can change the client identifier to a customized MAC address.

To configure a client identifier when you enable the DHCP client, enter:

Syntax: ip address dhcp client-id [ethernet 0/<*port number*> | *HH:HH:HH:HH:HH:HH:HH*]

When you configure the client identifier, you can also configure a hostname, as explained in the next section.

Configuring a Hostname. The Secure Router OS uses the hostname configured for the router as the Frame Relay subinterface's default DHCP client hostname. If you want to override this hostname when you enable the DHCP client, enter the following command:

Syntax: ip address dhcp hostname <*word*>

For example, you might want to specify that the hostname is *RouterB*. In this case, you would enter:

ProCurve(config-fr 1.1)# ip address dhcp hostname RouterB

When you specify the hostname, you can also configure a client identifier at the same time, as shown below.

ProCurve(config-fr 1.1)# ip address dhcp client-id ethernet 0/1 hostname RouterB

If you enter this command, the DHCP client will use the MAC address of the Ethernet 0/1 interface as its client identifier, and it will use the hostname RouterB.

Alternatively, you can specify the hostname and configure the client to ignore the settings received from the DHCP server. These commands are described in the following sections.

Overriding Settings Received from the DHCP Server. If the DHCP server is configured to provide a default-route, a domain name, or a domain name system (DNS) server, the DHCP client for the Frame-Relay subinterface will accept and use these settings. If you do not want to use any or one of these settings, enter the appropriate option when you enable the DHCP client:

Syntax: ip address dhcp [hostname <*word*> | no-default-route | no-domain-name | no-nameservers]

For example, if you do not want the DHCP client to use the default route and name server settings that it receives from the DHCP server, enter:

```
ProCurve(config-fr 1.1)# ip address dhcp no-default-route no-nameservers
```

Attaching a Network Monitoring Track to the DHCP Interface. As a part of the network monitoring feature, you can attach a network monitoring track to the DHCP client, in order to monitor the default route received from a DHCP server. A track uses probes to test routes and servers, with the goal of either removing failed routes or logging poor performance.

You can use the **track** option with the **ip address dhcp** command to configure the ProCurve Secure Router to add the default route as a monitored route. You can combine the **track** option with any of the other options for the **ip address dhcp** command (except **no-default-route**—the router cannot monitor a route that the interface does not accept).

For example, to attach the track named DHCPDefault, enter:

```
ProCurve(config-fr 1.1)# ip address dhcp track DHCPDefault
```

Before entering the command, you should create the track named DHCPDefault. Also, you should create a probe to test the route. For example, the probe could test connectivity to the default gateway listed in the DHCP default route. If the probe fails to reach the gateway, the track determines that the default route has failed and removes it.

For more information about network monitoring and configuring tracks and probes, see *Chapter 9: Network Monitoring* in the *Advanced Management and Configuration Guide.*

Setting the Administrative Distance. You can specify the administrative distance to use when adding the DHCP gateway to the route table. The router uses the administrative distance to determine the best route when multiple routes to the same destination exist. The router assumes that the smaller the administrative distance, the more reliable the route is.

For example, to set an administrative distance of 5, enter:

```
ProCurve(config-fr 1.1)# ip address dhcp 5
```

Changing a Setting for the DHCP Client. If you want to change a setting for the DHCP client, you must first disable the client. Then you can enter the command to enable the client with the setting that you want to change.

Before you disable the client, you should release the IP address obtained through DHCP. This will prevent the DHCP server from holding the IP address and allow it to assign the IP address to another client.

Releasing or Renewing an IP address. If you want to manually force the Frame Relay subinterface to release or renew an IP address, enter these commands from the Frame Relay subinterface configuration mode context:

```
ProCurve(config-fr 1.1)# ip dhcp release
```

```
ProCurve(config-fr 1.1)# ip dhcp renew
```

Remove the DHCP Client Setting. If you decide that you no longer want the Frame-Relay subinterface to be a DHCP client, enter:

```
ProCurve(config-fr 1.1)# no ip address dhcp
```

Configure the Frame Relay Subinterface as an Unnumbered Interface. To conserve IP addresses on your network, you may want to create the Frame Relay subinterface as an unnumbered interface. When you assign a logical interface on the router an IP address, that IP address cannot overlap with the IP addresses assigned to other logical interfaces. As a result, each interface that has an IP address represents a subnet. Depending on the subnetting scheme you use, this could use more IP addresses than you can spare.

You can configure the Frame Relay subinterface as an unnumbered interface that uses the IP address assigned to another interface. The Secure Router OS uses the IP address of the specified interface when sending route updates over the unnumbered interface.

Before configuring the Frame Relay subinterface as an unnumbered interface, you should be aware of a potential disadvantage: If the interface to which the IP address is actually assigned goes down, the Frame Relay subinterface will be unavailable. For example, suppose you configure Frame Relay 1.16 as an unnumbered interface that takes its IP address from the Ethernet 0/1 interface. If the Ethernet 0/1 interface goes down, the Frame Relay 1.16 subinterface will be unavailable as well.

To minimize the chances of the interface with the IP address going down, you can assign the IP address to a loopback interface, which typically does not go down.

To configure a Frame Relay subinterface as an unnumbered interface, enter the following command from the Frame Relay subinterface configuration mode context:

Syntax: ip unnumbered <*interface*>

Valid interfaces include:

- Ethernet interfaces and subinterfaces
- other Frame Relay subinterfaces
- PPP interfaces
- HDLC interfaces
- loopback interfaces
- Asynchronous Transfer Mode (ATM) subinterfaces
- demand interfaces

For example, you would enter the following commands to configure a loopback interface and then configure the Frame Relay 1.16 subinterface to use the IP address assigned to that loopback interface:

```
ProCurve(config)# interface loopback 1
ProCurve(config-loop 1)# ip address 10.1.1.1 /30
ProCurve(config-loop 1)# interface fr 1.16
ProCurve(config-fr 1.16)# ip unnumbered loopback 1
```

Note

You do not have to enter **no shutdown** to activate a loopback interface. The status of a loopback interface changes to up after you enter the **interface loopback <*interface number*>** command.

Set the CIR

You can configure the CIR for the Frame Relay link using the **frame-relay bc** command. As explained earlier, the CIR is the bandwidth that your Frame Relay service provider guarantees your company.

The CIR is calculated from the B_c, which is the maximum number of bits that the Frame Relay carrier guarantees to forward during a certain interval of time (T). The CIR is equal to B_c/T.

You should set a B_c for each Frame Relay subinterface to ensure that the PVC does not exceed its CIR. Some Frame Relay service providers may charge your company extra if your company consistently transmits over its CIR.

The industry standard is to calculate the time interval as 1 second. As a result, the B_c is essentially the CIR. To set the CIR, enter the following command from the Frame Relay subinterface configuration mode context:

Syntax: frame-relay bc <*committed burst value*>

Replace <*committed burst value*> with your CIR expressed in bits. You can set a B_c between 0 and 4,294,967,294 bps.

For example, you might enter:

```
ProCurve(config-fr 1.1)# frame-relay bc 256000
```

Set the EIR

When your company negotiated a SLA, the terms of that agreement probably allowed for a burst rate on the Frame Relay connection. This burst rate is called the Excess Information Rate (EIR), which defines the maximum amount of traffic your company is allowed to send *over* its CIR.

The B_e sets the maximum number of bits that the router can transmit during T. Just as B_c is equal to the CIR, B_e is equal to the EIR. B_e determines the rate at which the ProCurve Secure Router can burst data above the CIR when there is no congestion on the Frame Relay network.

Note

If you enter a value for the **frame-relay bc** command, you should also configure a burst rate for the Frame Relay link. Otherwise, the link will be limited to the bandwidth you specified in the **frame-relay bc** command.

Together, the **frame-relay bc** and the **frame-relay be** commands define the amount of bandwidth you can use on the Frame Relay link. The sum of the values you specify for these two commands should be greater than 8000.

To configure the EIR for the Frame-Relay link, enter:

Syntax: frame-relay be <*excessive burst value*>

Replace <*excessive burst value*> with a burst rate, expressed in bits. You can set a B_e between 0 and 4,294,967,294 bps.

For example, you might enter:

```
ProCurve(config-fr 1.1)# frame-relay be 64000
```

Discard Eligible (DE) Bit. After a PVC reaches its CIR, the Frame Relay switch marks each packet with a Discard Eligible (DE) bit. For example, if a PVC's B_c is 1.0 Mb, its B_e is 1.5 Mb, and it is transmitting traffic at full capacity, then Frame Relay switch will set the DE bit on the last 500 kilobytes of packets. If the Frame Relay network becomes congested, the Frame Relay switch first drops the packets that are marked with the DE bit.

Bind the Physical Interface to the Logical Interface

On the ProCurve Secure Router, you must bind the physical interface to the logical interface so that the router knows which Data Link Layer protocol to use for that WAN connection. When you bind a physical interface to a logical interface, the two are considered a single interface bind group.

You bind the physical interface to the Frame Relay interface, not to individual subinterfaces. In this way, various PVCs can use any available bandwidth on the physical connection to burst data past their CIR. You can enter the **bind** command from the global configuration mode context or from the Frame Relay interface configuration mode context:

Syntax: bind <*bind number*> <*physical interface*> <*slot*>/<*port*> [<*tdm-group number*>] <*logical interface*> <*logical interface number*>

The <*bind number*> is globally significant. That is, each **bind** command you enter on the router must have a unique bind number.

Replace *<physical interface>* with E1, T1, or serial. The *<slot>* and *<port>* pinpoint this interface's location on the ProCurve Secure Router and distinguish multiple lines of the same type from each other.

If you are binding the Frame Relay interface to an E1 or T1 interface, replace *<tdm-group number>* with the TDM group you created when you configured that interface. If you are binding the serial interface to the Frame Relay interface, you do not include this option.

In this instance, the *<logical interface>* is Frame Relay, and the *<logical interface number>* refers to the number you assigned to this interface.

For example, if you want to bind the E1 1/1 interface to the Frame Relay 1 interface, enter:

```
ProCurve(config)# bind 1 e1 1/1 1 fr 1
```

Note

You bind the physical interface to the Frame Relay interface (not to the subinterface).

If you want to bind the serial 1/1 interface to the Frame Relay 1 interface, enter:

```
ProCurve(config)# bind 1 ser 1/1 fr 1
```

Note

When you bind a serial interface to the Frame Relay interface, you do not include a TDM group number because the serial interface does not use TDM groups.

To see an example configuration that uses Frame Relay, see "Example Networks" on page 6-48.

Additional Settings

Depending on your company's WAN environment, you may want to configure other options on the Frame Relay interface or subinterface.

Configure a Secondary IP Address for the Subinterface. You can configure a secondary IP address on the Frame Relay subinterface. Enter:

Syntax: ip address *<A.B.C.D>* *<subnet mask | /prefix length>* secondary

Replace *<A.B.C.D>* with secondary IP address and specify a subnet mask using the *<subnet mask>* option or the *</prefix length>* option. Finally, include the **secondary** option.

For example, you might enter:

ProCurve(config-fr 1.1)# ip address 192.168.115.1 255.255.255.252 secondary

To remove the secondary IP address, enter:

Syntax: no ip address <*A.B.C.D*> <*subnet mask* | /*prefix length*> secondary

You can include an unlimited number of secondary IP addresses.

Set the MTU. The MTU defines the largest size that a frame can be before it must be fragmented. The MTU size on the Frame Relay subinterface should match the MTU used by the remote router and the intervening network devices. Although you can match the MTU on your Frame Relay interface with that used by your public carrier's equipment, you cannot ensure that all the intervening network devices will use the same MTU. To avoid any problems that may occur if an intervening network device is using a small MTU size, you may want to enable Frame Relay fragmentation. For more information about Frame Relay fragmentation, see the *Advanced Management and Configuration Guide*, *Chapter 8: Setting Up Quality of Service*.

Note

If you have enabled Open Shortest Path First (OSPF) routing on the ProCurve Secure Router, you should take special care when setting the MTU. OSPF routers cannot become adjacent if their MTU sizes do not match.

By default, the MTU for Frame Relay subinterfaces is 1500 bytes. To change this setting, enter the following command from the Frame Relay subinterface configuration mode context:

Syntax: mtu <*size*>

Replace <*size*> with a number between 64 and 1520.

Adding a Description. You can add a description to a Frame Relay interface or subinterface if you want to document information about it. For example, if you have multiple PVCs configured on a Frame Relay interface, you may want to document the other end point of each PVC. In this case, you would enter the following command at the Frame Relay subinterface configuration mode context:

Syntax: description <*line*>

Replace <*line*> with a phrase up to 80 characters. For example, you might enter:

ProCurve(config-fr 1.16)# description WAN link to London office

This description is displayed when you enter the **show running-config** command. From the enable mode context, enter:

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

You can also view the description by entering:

```
ProCurve# show running-config interface fr 1.16
```

This command displays the running-config settings for only the Frame Relay 1.16 subinterface, as shown below:

```
interface fr 1.16
  frame-relay interface-dlci 16
  description WAN link to London office
  ip address 192.168.1.1 255.255.255.0
  no shutdown
```

Settings Explained in Other Chapters

In addition to configuring these settings for Frame Relay, you can:

- assign ACPs or ACLs to control access to the Frame Relay subinterface
- enable bridging
- assign crypto maps to enable VPNs
- configure settings for routing protocols
- configure Quality of Service (QoS) settings

Table 6-8 lists additional configurations that you can enter from the Frame Relay interface and subinterface and the page number where you can find information about those configurations.

Table 6-8. Additional Configurations for the Frame Relay

Settings	Apply to Frame Relay Interface or Subinterface	Configuration Guide	Page
access controls to filter incoming and outgoing traffic	Frame Relay subinterface	Advanced	5-19, 5-38
bridging	Frame Relay subinterface	Basic	10-6
VPNs	Frame Relay subinterface	Advanced	10-46
routing commands for OSPF, RIP, or BGP	Frame Relay subinterface	Advanced	15-1
QoS settings	Frame Relay interface	Advanced	8-28

Configuring HDLC as the Data Link Layer Protocol

One of the oldest Data Link Layer protocols for a WAN, HDLC actually predates the PC. Although it was developed for a mainframe environment, which includes primary and secondary devices, HDLC has been updated for use in the PC environment. However, some functionality and terminology have survived from its early use, as evidenced by its modes of operation.

HDLC has three modes of operation:

Normal Response Mode (NRM). A secondary device can transmit only when the primary device specifically instructs it to do so.

Asynchronous Response Mode (ARM). A secondary device can initiate a transmission; however, the primary device controls the establishment and termination of the link.

Asynchronous Balanced Mode (ABM). Devices at both ends of a connection are configured to be both primary and secondary devices and can establish a link, transmit data without permission, and terminate a link.

When you configure the ProCurve Secure Router to use HDLC for an E1 or T1 connection, it operate in ABM.

HDLC uses three different types of frames:

- Unnumbered frames establish a link.
- Supervisory frames carry error and flow control information.
- Information frames carry the Network Layer packets across the WAN link.

Create the HDLC Interface

To begin configuring HDLC as the Data Link Layer protocol for an E1, T1, or serial interface, you must create a logical interface. From the global configuration mode context, enter:

Syntax: interface <*interface*> <*number*>

Replace <*interface*> with **HDLC** and replace <*number*> with any number between 1 and 1024. Each HDLC interface you configure on the router must have a unique number.

For example, if you are configuring the first HDLC interface on the router, you could enter:

```
ProCurve(config)# interface hdlc 1
```

The router prompt indicates that you have entered the appropriate interface configuration mode context:

ProCurve(config-hdlc 1)#

From this configuration mode context, you can enter the **?** help command to display the commands available from this configuration mode context.

ProCurve(config-hdlc 1)# ?

Table 6-9 shows the main settings that you must configure for an E1, T1, or serial interface that uses HDLC.

Table 6-9. Options for Configuring an E1, T1, or Serial Interface with HDLC

Interface Configuration Mode Context	Command	Explanation	Page
e1	• **tdm-group <number> timeslots <range of numbers>** • **coding [ami | hdb3]** • **frame format [e1 | crc4]** • **clock source [internal | line | through]** • **no shutdown**	• defines the number of channels used for the E1 connection • defines the line coding • defines the frame format • defines the clock source, or timing, for the connection • activates the interface	4-10
t1	• **tdm-group <number> timeslots <range of numbers>** • **coding [ami | b8zs]** • **frame format [esf | d4]** • **clock source [internal | line | through]** • **lbo long <value> | lbo short <value>** • **no shutdown**	• defines the number of channels used for the T1 connection • defines the line coding • defines the frame format • defines the clock source, or timing, for the connection • sets the level of the transmit signal • activates the interface	4-10
serial	• **serial-mode [EIA530 | V35 | X21]** • **et-clock-source [txclock | rxclock]** • **no shutdown**	• configures the serial interface to support the appropriate cable • configures the serial interface to take the clock from the transmit signal, txclock, or from the receive signal, rxclock • activates the interface	5-12

Interface Configuration Mode Context	Command	Explanation	Page
hdlc	• **no shutdown** • **ip address <A.B.C.D> <subnet mask | / prefix length>** **or** • **ip unnumbered <interface>**	• activates the interface • assigns a static IP address to the HDLC interface • configures the HDLC interface to use the IP address assigned to another interface	6-42
global configuration or interface configuration	• **bind <number> <physical interface> <slot>/<port> [<tdm-group number>] hdlc <interface number>**	• binds the physical interface to the logical interface • requires the tdm-group number for E1 and T1 interfaces, but not for serial interfaces	6-44

The HDLC settings are described in the sections that follow.

Activate the HDLC Interface

You must activate the HDLC interface. From the HDLC interface configuration mode context, enter:

```
ProCurve(config-hdlc 1)# no shutdown
```

Although the HDLC interface is activated, its status will not change to up until it is bound to a physical interface. It can then begin to negotiate an HDLC session, and if that negotiation is successful, the status of the HDLC interface will change to up.

Configure an IP Address for the WAN Connection

You configure the IP address for the WAN connection on the HDLC interface, rather than on the physical interface. There are two ways to assign an IP address to the HDLC interface:

- assign a static IP address
- configure the HDLC interface as an unnumbered interface

Assign a Static IP Address. To assign the HDLC interface an IP address, enter the following command from the HDLC interface configuration mode context:

Syntax: ip address <*A.B.C.D*> <*subnet mask* | /*prefix length*>

You can replace <*subnet mask*> with the complete subnet mask, or you can replace </*prefix length*> with the CIDR notation. For example, you might enter:

```
ProCurve(config-hdlc 1)# ip address 10.1.1.1 /24
```

Configure the HDLC Interface as an Unnumbered Interface. To conserve IP addresses on your network, you may want to create the HDLC interface as an unnumbered interface. When you assign a logical interface an IP address, that IP address cannot overlap with the IP addresses assigned to other logical interfaces on your network. As a result, each interface that has an IP address represents an entire subnet. Depending on the subnetting scheme you use, this could use more IP addresses than you can spare.

You can configure the HDLC interface (and other interfaces on the ProCurve Secure Router) as an unnumbered interface. The HDLC interface will then use the IP address of another interface—the interface you specify. The Secure Router OS uses the IP address of the specified interface when sending route updates over the unnumbered interface.

Configuring the HDLC interface as an unnumbered interface has a potential disadvantage: If the interface to which the IP address is actually assigned goes down, the HDLC interface will be unavailable as well. For example, suppose you configure the HDLC 1 interface as an unnumbered interface that takes its IP address from the Ethernet 0/1 interface. If the Ethernet 0/1 interface goes down, the HDLC 1 interface will also be unavailable.

To minimize the chances of the interface with the IP address going down, you can assign the IP address to a loopback interface, which typically does not go down.

To configure the HDLC interface as an unnumbered interface, enter the following command from the HDLC interface configuration mode context:

Syntax: ip unnumbered <*interface*>

Valid interfaces include:

- ATM subinterfaces
- Ethernet interfaces and subinterfaces
- Frame Relay subinterfaces
- loopback interfaces
- PPP interfaces
- demand interfaces

For example, you would enter the following commands to configure a loopback interface and then configure the HDLC 1 interface to use the IP address assigned to that loopback interface:

```
ProCurve(config)# interface loopback 1
ProCurve(config-loop 1)# ip address 192.168.5.1 /24
ProCurve(config-loop 1)# interface hdlc 1
ProCurve(config-hdlc 1)# ip unnumbered loopback 1
```

Note You do not have to enter **no shutdown** to activate a loopback interface. The status of a loopback interface changes to up after you enter the **interface loopback** command.

Bind the Physical Interface to the Logical Interface

On the ProCurve Secure Router, you must bind the physical interface to the logical interface so that the router knows which Data Link Layer protocol to use for that WAN connection. When you bind a physical interface to a logical interface, the two are considered a single interface bind group.

You can enter the **bind** command from the global configuration mode context or the HDLC interface configuration mode context:

Syntax: bind *<bind number>* *<physical interface>* *<slot>*/*<port>* [*<tdm-group number>*] *<logical interface>* *<logical interface number>*

You can also enter the **bind** command from the HDLC interface configuration mode context.

Replace *<bind number>* with a number that is globally significant. That is, each **bind** command you enter on the router must have a unique bind number.

Replace *<physical interface>* with E1, T1, or serial. Replace *<slot>* and *<port>* with the numbers that identify the physical interface's location on the ProCurve Secure Router.

If you are binding the HDLC interface to an E1 or T1 interface, include the *<tdm-group number>* that you created when you configured the E1 or T1 interface. If you are binding the HDLC interface to a serial interface, you do not include this option.

Replace *<logical interface>* with **hdlc** and the *<logical interface number>* with the number you assigned to this interface.

For example, if you want to bind the T1 2/1 interface to the HDLC 1 interface, enter:

```
ProCurve(config)# bind 1 t1 2/1 hdlc 1
```

If you want to bind the serial interface to the HDLC interface, enter:

```
ProCurve(config)# bind 1 serial 1/1 hdlc 1
```

Note If you are binding a serial interface to the HDLC interface, you do not include the TDM group number because you do not use TDM groups on a serial interface.

Additional Settings

Depending on your company's WAN environment, you may want to configure other options on the HDLC interface.

Configure a Secondary IP Address for the Interface. You can configure a secondary IP address on the HDLC interface. From the HDLC interface configuration mode context, enter:

Syntax: ip address <*A.B.C.D*> <*subnet mask* | */prefix length*> secondary

Replace <*A.B.C.D*> with secondary IP address and specify a subnet mask using the <*subnet mask*> option or the <*/prefix length*> option. Finally, include the **secondary** option.

For example, you might enter:

```
ProCurve(config-hdlc 1)# ip address 192.168.5.1 255.255.255.252 secondary
```

You can include an unlimited number of secondary IP addresses.

To remove the secondary IP address, enter:

Syntax: no ip address <*A.B.C.D*> <*subnet mask* | */prefix length*> secondary

Set the MTU. The MTU defines the largest size that a frame can be. If a frame exceeds the size limit, it must be fragmented. For best results, the MTU size on the HDLC interface should match the MTU used by the remote router.

Note

If you have enabled Open Shortest Path First (OSPF) routing on the ProCurve Secure Router, you should take special care when setting the MTU. OSPF routers cannot become adjacent if their MTU sizes do not match.

By default, the MTU for HDLC interfaces is 1500 bytes. To change this setting, enter the following command from the HDLC interface configuration mode context:

Syntax: mtu <*size*>

Replace <*size*> with a number between 64 and 1520.

Add a Description. You can add a description to the HDLC interface if you want to document information that will be displayed in the running-config. From the HDLC interface configuration mode context, enter:

Syntax: description <*line*>

Replace <*line*> with a phrase up to 80 characters. For example, you might enter:

```
ProCurve(config-hdlc 1)# description WAN link to Saratoga Street office
```

This description is displayed only when you enter the **show running-config** command. From the enable mode context, enter:

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

You must then scroll through the entire running-config to find the **interface hdlc** heading. To view only the running-config for the HDLC interface, enter:

```
ProCurve# show running-config interface hdlc 1
```

This command displays the running-config settings for only the HDLC interface, as shown below:

```
interface hdlc 1
  description WAN link to Saratoga Street office
  ip address 192.168.1.1 255.255.255.0
  bind 1 e1 1/1 1 hdlc 1
  no shutdown
```

Settings Explained in Other Chapters

In addition to configuring these settings for an HDLC interface, you can:

- assign ACPs or ACLs to control access to the HDLC interface
- enable bridging
- assign crypto maps to enable VPNs
- configure settings for routing protocols
- configure QoS settings

Table 6-10 lists additional configurations that you can enter from the HDLC interface and the page number where you find information about those configurations.

Table 6-10. Additional Configurations for the HDLC Interface

Settings	Configuration Guide	Page
access controls to filter incoming and outgoing traffic	Advanced	5-19, 5-38
bridging	Basic	10-6
VPNs	Advanced	10-46
routing commands for OSPF, RIP, or BGP	Advanced	15-1
QoS settings	Advanced	8-28

Example Networks

This section outlines examples of E1- and T1-carrier lines that use PPP, Frame Relay, and HDLC as the Data Link Layer protocol. It also provides examples of WANs that are using PPP authentication.

Example 1. Figure 6-9 shows a company's WAN that includes a connection between two offices in London. Because this company needed a constant, reliable connection between these two offices, they leased an E1-carrier line for both the Seething Lane and Chelsea Harbor offices. The Data Link Layer protocol is PPP.

The company also required a connection to its Paris office. For this connection, the company negotiated an SLA with a Frame Relay service provider.

Finally, the company set up an Asymmetric Digital Subscriber (ADSL) line to a local Internet Service Provider (ISP). Through this connection, the company's employees can access the Internet. (For information about ADSL, see *Chapter 7: ADSL WAN Connections*.)

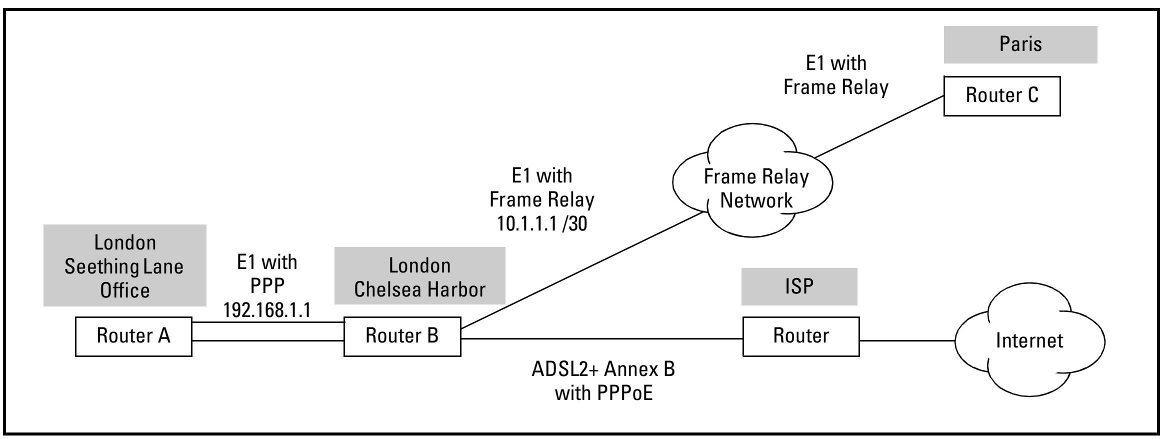

Figure 6-9. Example WAN Using E1-Carrier Lines with PPP and Frame Relay

Figure 6-10 shows the configuration for the E1, PPP, and Frame Relay interfaces, as they appear in the running-config for Router B, the router for the London Chelsea Harbor office.

```
interface e1 1/1
  tdm-group 1 timeslots 1-31 speed 64
  no shutdown
!
interface e1 1/2
  clock source through
  tdm-group 1 timeslots 1-31 speed 64
  no shutdown
!
interface fr 1 point-to-point
  frame-relay intf-type dte
  frame-relay lmi-type q933a
  no shutdown
  bind 2 e1 1/2 1 frame-relay 1
!
interface fr 1.16 point-to-point
  frame-relay interface-dlci 16
  frame-relay bc 1600000
  frame-relay be 128000
  ip address 10.1.1.1 255.255.255.252
!
interface ppp 1
  ip address  192.168.1.1  255.255.255.0
  no shutdown
  bind 1 e1 1/1 1 ppp 1
```

Figure 6-10. Running-Config for Router B in Example 1

Because the company is using default settings for line coding (HDB3) and frame format (E1) on the E1-carrier lines, the network administrator did not enter these settings. Consequently, they are not listed when you enter the following command from the enable mode context:

ProCurve# show running-config

To view all of the configuration settings—including the default settings—you must enter:

ProCurve# show running-config verbose

Example 2. The WAN shown in Figure 6-11 is for a U.S.-based company that has three offices: The main office is in Atlanta, and the two branch offices are in San Francisco and London. To connect the San Francisco office to the Atlanta office, the company leased a T1-carrier line for each office and are using HDLC as the Data Link Layer protocol. The two offices are exchanging confidential information and wanted a dedicated connection with the full bandwidth of a T1-carrier line.

To connect the Atlanta office to the London office, the company chose Frame Relay, which allows them to cross country borders at a more affordable cost than dedicated T1-and E1-carrier lines.

The company uses ADSL for its Internet connection at the Atlanta office. (For information about ADSL, see *Chapter 7: ADSL WAN Connections*.)

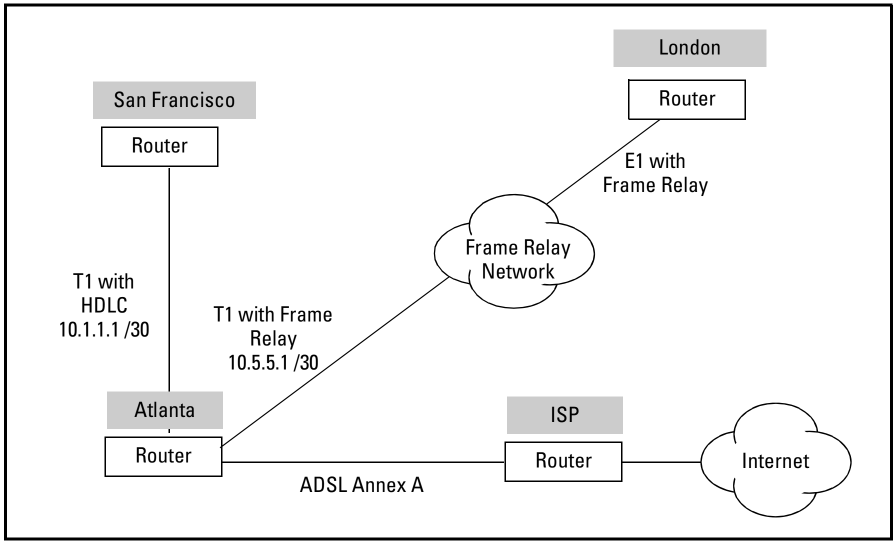

Figure 6-11. Example WAN Using Carrier Lines with HDLC and Frame Relay

Figure 6-12 shows the configurations for the T1, HDLC, and Frame Relay interfaces, as they appear in the running-config for the Atlanta router.

```
interface t1 1/1
  lbo short 550
  tdm-group 1 timeslots 1-24 speed 64
  no shutdown
!
interface t1 1/2
  clock source through
  lbo short 550
  tdm-group 1 timeslots 1-24 speed 64
  no shutdown
!
interface fr 1 point-to-point
  frame-relay intf-type dte
  frame-relay lmi-type ansi
  no shutdown
  bind 2 t1 1/2 1 frame-relay 1
!
interface fr 1.104 point-to-point
  frame-relay interface-dlci 104
  ip address  10.5.5.1  255.255.255.252
!
interface hdlc 1
  ip address  10.1.1.1  255.255.255.252
  no shutdown
  bind 1 t1 1/1 1 hdlc 1
```

Figure 6-12. Running-Config for the Atlanta Router in Example 2

Because the company is using default settings for line coding (B8ZS) and frame format (ESF) on the T1-carrier lines, the network administrator did not enter these settings. Consequently, they are not listed when you enter the following command from the enable mode context:

ProCurve# show running-config

To view all of the configuration settings—including the default settings—you must enter:

ProCurve# show running-config verbose

Example 3: Two Routers Authenticating Each Other with PAP. In this example, the router at Site A (hostname Local) and the router at Site B (hostname Remote) authenticate each other with PAP. Local's password is XXX, and Remote's password is YYY. (See Figure 6-13.)

You would configure Local as follows:

1. Access the PPP interface configuration mode context:

   ```
   Local(config)# interface ppp 1
   ```

2. Configure the router to authenticate Remote with PAP:

   ```
   Local(config-ppp 1)# ppp authentication pap
   ```

3. Set Remote's username and password:

   ```
   Local(config-ppp 1)# username Remote password YYY
   ```

4. Set the router's own PAP username and password:

   ```
   Local(config-ppp 1)# ppp pap sent-username Local password XXX
   ```

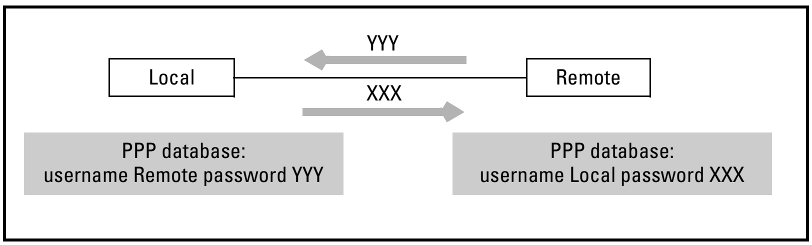

Figure 6-13. Routers Authenticating Each Other

Remote would then be configured in the same way:

1. Access the PPP interface configuration mode and configure the router to authenticate Local with PAP:

   ```
   Remote(config)# interface ppp 1
   Remote(config-ppp 1)# ppp authentication pap
   ```

2. Set Local's username and password:

   ```
   Remote(config-ppp 1)# username Local password XXX
   ```

3. Set the router's own PAP username and password:

   ```
   Remote(config-ppp 1)# ppp pap sent-username Remote password YYY
   ```

Example 4: One Peer Requesting CHAP. Both routers do not have to require authentication. For example, only Local could request Remote to authenticate itself using CHAP. The commands would be as follows for Local:

```
Local(config)# interface ppp 1
Local(config-ppp 1)# ppp authentication chap
Local(config-ppp 1)# username Remote password YYY
```

Remote would be configured as follows:

```
Remote(config)# interface ppp 1
Remote(config-ppp 1)# ppp chap password YYY
```

Example 5: CHAP Authentication to an ISP. In this example, the ISP has provided an ID (ID-GIVEN-BY-ISP) and password (PWD-GIVEN-BY-ISP) to be used when authenticating through CHAP. (See Figure 6-14.)

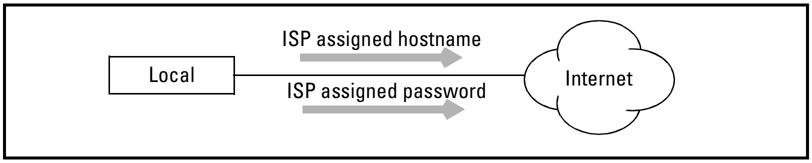

Figure 6-14. Authentication to an ISP

You would configure the router being authenticated as follows:

1. Access the PPP interface:

   ```
   ProCurve(config)# interface ppp 1
   ```

2. Configure the ID given by the ISP to override the local hostname.

   ```
   ProCurve(config-ppp 1)# ppp chap hostname ID-GIVEN-BY-ISP
   ```

3. Set the password given by the ISP:

   ```
   Remote(config-ppp 1)# ppp chap password PWD-GIVEN-BY-ISP
   ```

Checking the Status of Logical Interfaces

After you configure the physical and logical interfaces and bind them together, the ProCurve Secure Router should be able to exchange data with the device at the other end of the WAN connection.

View the Status of Interfaces

To view the status of the logical interface you have bound to the E1, T1, or serial interface, you can use **show** commands. Table 6-11 lists the **show** commands that you can use to view information about interfaces.

Table 6-11. show Commands for Logical Interfaces

Command	Explanation
show interfaces	displays information about all the interfaces—active or inactive—on the ProCurve Secure Router
show interface <interface> <number> [realtime]	displays information about a specific logical interface
show running-config	displays all of the settings that you have configured for the ProCurve Secure Router
show running-config verbose	displays the entire running-config, including the default settings
show running-config interface <interface> <number>	displays the settings that you have configured for a particular interface
show running-config interface <interface> <number> verbose	displays the entire running-config for a particular interface, including the default settings

Viewing the Status of PPP Interfaces

For example, if you want to view the status of the PPP 1 interface, enter the following command from the enable mode context:

Syntax: show interface ppp 1

Figure 6-15 shows the results of this command for a sample network.

```
ppp 1 is UP
  Configuration:
    Keep-alive is set (10 sec.)
    No multilink
      MTU = 1492
    No authentication
    IP is configured
      192.168.1.20  255.255.255.0
  Link thru atm 1.1 is UP; LCP state is OPENED, negotiated MTU is 1492
    Receive: bytes=20296, pkts=2727, errors=0
    Transmit: bytes=27728, pkts=2214, errors=0
    5 minute input rate 208 bits/sec, 0 packets/sec
    5 minute output rate 112 bits/sec, 0 packets/sec
  Bundle information
    Queueing method: fifo
    HDLC tx ring limit: 0
    Output queue: 0/1/200/0 (size/highest/max total/drops)
    IP is UP, IPCP state is OPENED
      Address=192.168.1.20 Mask=255.255.255.0
      Peer address=192.168.1.1
      IP MTU=1492, Bandwidth=896 Kbps
    LLDPCP State is STOPPED
```

Status of interface
No authentication is configured
IP address
Negotiated MTU
Status of NCP
IP address of PPP peer

Figure 6-15. show interface ppp <number>

This command displays a report about the logical interface's status, including information such as:

- whether the interface is up or down
- whether the physical link bound to the logical interface is up or down
- whether the LCP is opened
- endpoint settings
- errors
- queuing method
- available bandwidth
- the negotiated NCP and whether it is opened
- IP address
- peer IP address

Viewing the Status of Frame Relay Interfaces and Subinterfaces

For Frame Relay, you can view the status of both the interface and the subinterface. To view information about the Frame Relay interface, enter the following command from the enable mode context:

Syntax: show interface frame-relay <*number*>

Figure 6-16 shows the results of this command for a sample network.

```
fr 1 is UP
  Configuration:
    Signaling type is AUTO, signaling role is USER
    Multilink disabled
    Polling interval is 10 seconds,
    full inquiry interval is 6 polling intervals
  Link information:
    5 minute input rate 0 bits/sec, 0 packets/sec
    5 minute output rate 8 bits/sec, 0 packets/sec
    BW 1984 Kbit
    Queueing method: weighted fair
    HDLC tx ring limit: 2
   Output queue:0/1/100/64/0 (size/highest/max total/threshold/drops)
      Conversations  0/1/256 (active/max active/max total)
      Available Bandwidth 1488 kilobits/sec
      0 packets input, 0 bytes
      1 pkts discarded, 0 error pkts, 0 unknown protocol pkts
      25 packets output, 334 bytes
      1 tx pkts discarded, 0 tx error pkts
```

Status of interface

signaling type and role

Figure 6-16. show interface frame-relay <number>

With this command, you can view the Frame Relay signaling role and signaling type, and you can view the information about packet discards or errors.

You can view this information in real-time by adding this option when you enter the **show** command:

Syntax: show interface frame-relay <*number*> realtime

Figure 6-17 shows the results of this command for a sample network.

```
-------------------------------------------------------------------
fr 1 is UP
  Configuration:
    Signaling type is ANSI, signaling role is USER
    Multilink disabled
    Polling interval is 10 seconds,
    full inquiry interval is 6 polling intervals
  Link information:
    5 minute input rate 24 bits/sec, 0 packets/sec
    5 minute output rate 8 bits/sec, 0 packets/sec
    BW 1984 Kbit
    Queueing method: weighted fair
    HDLC tx ring limit: 2
   Output queue:0/1/428/64/0 (size/highest/max total/threshold/drops)
      Conversations  0/1/256 (active/max active/max total)
      Available Bandwidth 1488 kilobits/sec
      44 packets input, 915 bytes
      1 pkts discarded, 0 error pkts, 0 unknown protocol pkts
      23 packets output, 322 bytes
      1 tx pkts discarded, 0 tx error pkts
Exit - 'Ctrl-C', Freeze - 'f', Resume - 'r'
```

Figure 6-17. show interface frame-relay <number> realtime

The Secure Router OS will continue to refresh this display with current information until you enter **Ctrl+C** to end the display.

To view information about the Frame Relay subinterface, enter the following command from the enable mode context:

Syntax: show interface frame-relay <*number.subinterface number*>

Figure 6-18 shows the results of this command for a sample network.

```
fr 1.1 is Active                              <---- Status of interface
  Ip address is 10.10.10.1, mask is 255.255.255.252
  Interface-dlci is 104                          <---- DLCI
  MTU is 1500 bytes, BW is 128000 Kbit (limited)
  Average utilization is 92%    <---- Utilization
```

Figure 6-18. show interface frame-relay <number.subinterface number>

As Figure 6-18 shows, you can view the status of the Frame Relay subinterface, the IP address, the DLCI, the MTU size, and the average utilization.

Viewing the Status of HDLC Interfaces

To view information about the HDLC interface, enter the following command from the enable mode context:

Syntax: show interface hdlc <*number*>

Figure 6-19 shows the results of this command for a sample network.

```
hdlc 1 is UP
  Configuration:
    Keep-alive is set (10 sec.)
    IP is configured
      10.1.1.1  255.255.255.252
  Link information:
    Receive: bytes=6896, pkts=65, errors=0, broadcast=22
    Transmit: bytes=8158, pkts=79, errors=0, broadcast=29
    5 minute input rate 184 bits/sec, 0 packets/sec
    5 minute output rate 216 bits/sec, 0 packets/sec
    IP is UP      Address=10.1.1.1 Mask=255.255.255.252
      IP MTU=1500, Bandwidth=1984 Kbps
```

Status of interface

IP address

MTU and bandwidth

Figure 6-19. show interface hdlc <number>

Viewing Configuration Information

You can view the running-config for a logical interface by entering the following command from the enable mode context:

Syntax: show running-config interface <*interface*> <*number*>

Replace <*interface*> with the logical interface and replace <*number*> with the number you used to create that interface. For example, to view the running-config for an HDLC interface, enter:

ProCurve# show running-config interface hdlc 1

Figure 6-20 shows the results of this command for a sample network.

```
interface hdlc 1
  ip address  10.1.1.1  255.255.255.252
  no shutdown
  bind 1 e1 1/1 1 hdlc 1
```

Figure 6-20. show running-config interface hdlc <number>

Troubleshooting Logical Interfaces

If the physical interface is up but the logical interface is not, the steps you take to troubleshoot the problem vary, depending on the Data Link Layer protocol you are using. This section is organized into three sections:

- troubleshooting the PPP interface
- troubleshooting the Frame Relay interface and subinterface
- troubleshooting the HDLC interface

Note

Enter the **show** and **debug** commands described in this troubleshooting section from the enable mode context. You can also access these commands from any configuration mode context by adding **do** to the beginning of the command.

Troubleshooting the PPP Interface

The first tool in troubleshooting a logical interface is the **show interfaces** command. From the enable mode context, enter the following command to check the status of a PPP interface that is bound to the E1, T1, or serial interface:

Syntax: show interfaces ppp <*number*>

```
ppp 2 is DOWN
  Configuration:
    Keep-alive is set (10 sec.)
    No multilink
      MTU = 1500
    No authentication
    IP is configured
      15.1.1.1  255.0.0.0
  Link thru ser 2/1 is DOWN; LCP state is INITIAL
    Receive: bytes=0, pkts=0, errors=0
    Transmit: bytes=0, pkts=0, errors=0
    5 minute input rate 0 bits/sec, 0 packets/sec
    5 minute output rate 0 bits/sec, 0 packets/sec
  Bundle information
    Queueing method: weighted fair
    HDLC driver does not support quality-of-service, or is not cross-connected
    Output queue: 0/0/-1512133286/64/0 (size/highest/max total/threshold/drops)
      Conversations  0/0/0 (active/max active/max total)
      Available Bandwidth 0 kilobits/sec
```

Figure 6-21. show interface ppp <number>

If the PPP interface is down, you should recheck the configuration to see if there are any errors. (See Figure 6-21.) You should also ensure that you have bound the physical interface to the PPP interface. If you have entered a **bind** command, it should be displayed when you enter **show running-config interface ppp <number>** from the enable mode context.

You should then determine if all steps for establishing a PPP session were completed successfully. The output for the **show interface ppp** command provides basic information about different PPP protocols, and you can use this information to determine if these PPP protocols were exchanged. If you want more detailed information to troubleshoot the PPP session, you can use **debug** commands, which are explained later in this section.

LCP State. When you enter s**how interfaces ppp** command, the status report will indicate whether the LCP state is opened, initial, or starting.

- If the LCP is opened, the ProCurve Secure Router was able to exchange LCP packets with its peer.
- If the LCP is in the initial state, the ProCurve Secure Router has not yet succeeded in establishing a link with the peer.
- If the LCP state is starting, the PPP interface is attempting to reopen a link that has been lost.

If the LCP status is not opened, you may need to double-check your configuration settings with your public carrier. For example, the carrier may have allocated a different number of DS0 channels to the physical line. You will need to reconfigure the physical interface to the correct number of DS0 channels. The public carrier may also be using a different Data Link Layer protocol.

NCP State. If the router has been able to exchange LCPs and has successfully passed through the authentication phase, the **show interfaces ppp** command displays:

- the type of NCP the router is using
- the status of the NCP

```
ppp 1 is UP
  Configuration:
    Keep-alive is set (10 sec.)
    No multilink
      MTU = 1500
    No authentication
    IP is configured
      10.1.1.1  255.255.255.252
  Link thru t1 1/1 is UP; LCP state is OPENED, negotiated MTU is 1500
    Receive: bytes=870, pkts=68, errors=0
    Transmit: bytes=1070, pkts=48, errors=0
    5 minute input rate 24 bits/sec, 0 packets/sec
    5 minute output rate 24 bits/sec, 0 packets/sec
  Bundle information
    Queueing method: weighted fair
    HDLC tx ring limit: 2
   Output queue: 0/1/400/64/0 (size/highest/max total/threshold/drops)
      Conversations  0/1/256 (active/max active/max total)
      Available Bandwidth 1536 kilobits/sec
    IP is DOWN, IPCP state is CLOSED
    LLDPCP State is OPENED
```

Check the status of NCP

Figure 6-22. Using the show interface ppp Command to Check the NCP

In Figure 6-22, PPP is using IPCP as the NCP. If the NCP is not open, it cannot encapsulate one or both of the two peers' network protocols. Verify that both ends of the connection are using viable upper-layer protocols.

Debug Commands. You can also isolate problems by examining frames coming through the PPP interface in real time. You can use this information to track the establishment of the PPP session and determine when and why the connection is not established.

Note

Debug commands are processor intensive.

Table 6-12 lists the **debug** commands you can use to monitor PPP interfaces.

Table 6-12. Debug commands for PPP Interfaces

Command	Explanation
debug ppp verbose	displays detailed information about all PPP frames as they arrive on or are sent from the PPP interface
debug ppp errors	displays error messages relating to PPP
debug ppp negotiation	displays events relating to link negotiation; shows if link protocols are able to open; reveals when negotiations between two PPP peers fail
debug ppp authentication	displays real-time messages relating to PAP and CHAP
undebug all	turns off debug messages

For example, if the status of the NCP is stopped, you may want to enter the **debug ppp negotiations** command. You should be able to see each stage in the process of establishing a PPP connection. Figure 6-23 shows the debug messages when a PPP connection is established successfully.

```
2005.08.12 17:51:01 PPP.NEGOTIATION PPPrx[e1 1/1] LCP: Conf-Ack  ID=33 Len=16 ACCM(00000000)
MAGIC(d418e92e)
2005.08.12 17:51:02 PPP.NEGOTIATION PPPrx[e1 1/1] LCP: Conf-Req  ID=188 Len=16
ACCM(00000000) MAGIC(2656e0ba)
2005.08.12 17:51:02 PPP.NEGOTIATION PPPtx[e1 1/1] LCP: Conf-Ack  ID=188 Len=16
ACCM(00000000) MAGIC(2656e0ba)
2005.08.12 17:51:02 PPP.NEGOTIATION PPPFSM: layer up, Protocol=c021
2005.08.12 17:51:02 PPP.NEGOTIATION e1 1/1: LCP up                  <-- LCP successful
2005.08.12 17:51:02 PPP.NEGOTIATION PPPtx[e1 1/1] LLDPCP: Conf-Req  ID=1 Len=4
2005.08.12 17:51:02 PPP.NEGOTIATION PPPtx[e1 1/1] IPCP: Conf-Req  ID=1 Len=10 IP (10.1.1.1)
2005.08.12 17:51:02 PPP.NEGOTIATION PPPrx[e1 1/1] LCP: Identification MAGIC(2656e0ba)
Msg(A04)
2005.08.12 17:51:02 PPP.NEGOTIATION PPPrx[e1 1/1] IPCP: Conf-Req  ID=1 Len=22 IP(10.3.3.2)
PriDNS(0.0.0.0) SecDNS(0.0.0.0)
2005.08.12 17:51:02 PPP.NEGOTIATION PPPtx[e1 1/1] IPCP: Conf-Rej  ID=1 Len=16
PriDNS(0.0.0.0) SecDNS(0.0.0.0)2005.08.12 17:51:02 PPP.NEGOTIATION PPPrx[e1 1/1] LCP:
ProtoRej (82cc)
2005.08.12 17:51:02 PPP.NEGOTIATION PPPrx[e1 1/1] IPCP: Conf-Ack  ID=1 Len=10 IP(10.1.1.1)
2005.08.12 17:51:02 INTERFACE_STATUS.ppp 1 changed state to up
2005.08.12 17:51:02 PPP.NEGOTIATION PPPrx[e1 1/1] IPCP: Conf-Req  ID=2 Len=10 IP(10.3.3.2)
2005.08.12 17:51:02 PPP.NEGOTIATION PPPtx[e1 1/1] IPCP: Conf-Ack  ID=2 Len=10 IP(10.3.3.2)
2005.08.12 17:51:02 PPP.NEGOTIATION PPPFSM: layer up, Protocol=8021
2005.08.12 17:51:02 PPP.NEGOTIATION ppp 1: IPCP up                  <-- NCP successful
```

Figure 6-23. debug ppp negotiation

Troubleshooting PPP Authentication

If you are troubleshooting a PPP connection and you notice that the LCP state is continually going up and down, it is possible that one or both of the peers cannot authenticate themselves. You can view debug authentication messages to determine whether the local or the remote router has failed to authenticate itself. When troubleshooting PAP, you can also view the usernames and passwords the routers are sending.

To monitor the PPP authentication process, enter the following command from the enable mode context:

```
ProCurve# debug ppp authentication
```

Troubleshooting PAP. If you are using PAP authentication, look for messages such as those shown in Figure 6-24.

The local router is attempting to authenticate itself.

```
ProCurve# debug ppp authentication
2005.07.08 09:03:44 PPP.AUTHENTICATION PPPtx[t1 1/1] PAP: Authen-Req
ID=1 Len=10 PeerID(Local) Password()
2005.07.08 09:03:44 PPP.AUTHENTICATION PPPrx[t1 1/1] PAP: Authen-Nak
ID=1 Len=5 Message()
```

The remote router rejects the password.

Figure 6-24. PAP Authentication Messages

The Authen-Req message is the message the authenticating peer sends with its username and password. If you see such a message marked with PPPtx, you know that your router is attempting to authenticate itself to the remote endpoint. The PeerID and Password fields are the values that this router sends as its username and password. When the interface receives an Authen-Nak, as shown above, the peer has rejected these values.

In this example, the interface has not been configured to send a password. You would need to obtain the correct username and password from your peer and configure them in the PPP interface configuration mode context.

When the local router is the authenticator, you can check the debug messages for the username and password the remote router is sending. Because PAP does not use encryption, the password will be readable in plain text. (See Figure 6-25.)

```
ProCurve# debug ppp authentication

2005.07.08 09:03:44 PPP.AUTHENTICATION PPPrx[t1 1/1] PAP: Authen-Req
ID=1 Len=10 PeerID(Remote) Password(procurve)
```

Peer's username

Peer's password

Figure 6-25. Finding the Peer's PAP Password

If you recognize the PeerID as that of a legitimate endpoint and the password seems correct, make sure that the username and password in the PPP database have been entered correctly. Enter **show run interface ppp *<interface number>*** and look for **username** and **password**. Otherwise, contact the remote site and inform it that it is sending the wrong password.

When a peer successfully authenticates itself, the authenticator returns an Authen-Ack:

2005.07.08 09:05:08 PPP.AUTHENTICATION PPPtx[t1 1/1] PAP: Authen-Ack ID=1 Len=10 Message(Hello)

Note Usernames and passwords are case-sensitive.

Troubleshooting CHAP. If you are using CHAP authentication, look for messages such as those shown in Figure 6-26.

```
ProCurve# debug ppp authentication

2005.07.08 08:59:02 PPP.AUTHENTICATION PPPtx[t1 1/1] CHAP: Challenge
ID=1 Len=28 ValLen=16 Name(Local)

2005.07.08 08:59:02 PPP.AUTHENTICATION PPPrx[t1 1/1] CHAP: Response
ID=1 Len=25 ValLen=16 Name(Remote)     <---- Peer's hostname

2005.07.08 08:59:02 PPP.AUTHENTICATION PPPtx[t1 1/1] CHAP: Failure
ID=1 Len=4 Message()
```

Figure 6-26. CHAP Authentication Messages

The Challenge message indicates which router requires the other to authenticate itself. In this example, the router with the hostname Local authenticates Remote. (The PPPtx also indicates that the local router transmits the challenge.) The Failure message indicates that Remote could not correctly identify itself.

View the running config for the interface (**show run int ppp *<interface number>***) and look for miskeyed passwords.

If the local router cannot authenticate itself, check the **ppp chap hostname** and **ppp chap password**. If they seem correct, contact the remote site or ISP and explain your problem.

If the remote router cannot authenticate itself, check the **ppp username** and **password** in the running config, which may have been miskeyed. If they are correct, contact the remote site and inform the network administrator that the router is sending the wrong authentication information.

Note Usernames and passwords are case sensitive.

Incompatible Authentication Protocols. If you do not receive any PPP authentication debug messages at all, the local and remote routers may be requesting different authentication protocols. In this case, the LCP state will not come up because the peers cannot negotiate the authentication option.

You could test this theory by debugging PPP negotiation events and looking for a Conf-Nak message. This message indicates that one of the peer's must refuse an option proposed by the other.

Caution

PPP debug messages are processor intensive: peers exchange LCP frames again and again in an attempt to negotiate the session. If the router is currently supporting network traffic, debugs can compromise its functions. When you suspect that authentication is keeping a connection from going up, you can simply try changing the type of authentication you require or send. If the PPP connection then goes up (or if PPP authentication debug messages appear), you know that incompatible authentication protocols were at least partially at fault.

In Figure 6-27, the local router requires PAP, but the remote router is configured for CHAP.

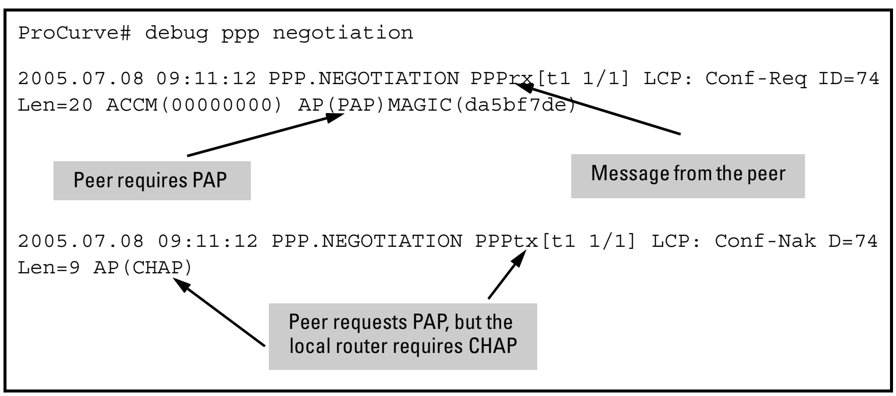

Figure 6-27. Debugs for Incompatible Authentication Protocols

Troubleshooting the Frame Relay Interface

When you troubleshoot the Frame Relay connection, you should first check the Frame Relay interface and then check the Frame Relay subinterface. From the enable mode context, enter the following command to check the status of a Frame Relay interface that is bound to the E1, T1, or serial interface:

Syntax: show interface frame-relay <*number*>

If the interface is administratively down, you need to activate it. From the Frame Relay interface configuration mode context, enter **no shutdown**.

If the interface is down, check your configuration and ensure that you are using the same Frame Relay signaling type as your Frame Relay carrier. Ensure that you have entered the correct **bind** command to bind this interface to the physical interface that is providing the connection.

If the Frame Relay interface is up, check the status of the Frame Relay subinterface. From the enable mode context, enter:

Syntax: show interface frame-relay <*number.subinterface number*>

If the status of the Frame Relay subinterface is "deleted," the DLCI that you entered does not match the DLCI that the provider is using. Recheck the DLCI with your Frame Relay service provider. If the status of the Frame Relay subinterface is "inactive," check the IP address and other configuration settings.

Table 6-13 shows the commands that you can use to troubleshoot a Frame Relay interface.

Table 6-13. show and debug Commands for Troubleshooting Frame Relay

Command	Explanation
show frame-relay lmi	displays LMI (signaling) type and information about LMI messages and updates
show frame-relay pvc	displays TX and RX status messages and the DLCI state
debug frame-relay lmi	displays LMI messages in real-time
undebug all	turns off debug messages

View LMI Statistics. From the enable mode context, enter:

ProCurve# show frame-relay lmi

Examine the polling information.

- "Num Status Enq. Sent" indicates the number of polls that the interface has sent. By default, the interface sends out one poll every 10 seconds.
- "Num Status Msgs Rcvd" indicates the number of polls that the interface has received from the other end of the connection. If the other endpoint is using typical Frame Relay settings, the interface should receive one poll every 10 seconds.

- "Num Update Status Rcvd" indicates the number of full status reports the interface has received. By default, the interface receives one full status report every six polls, or one every 60 seconds.
- "Num Status Timeouts" indicate the number of times the signal has been lost. When the router misses three out of four polls, it takes down the connection. When the interface continually sends out polls for which it does not receive a reply, the link has a problem, such as:
  - Signaling-type mismatch—Steadily incrementing status timeouts signal mismatched signaling-types. Check the signaling type listed in the LMI statistics as "LMI Type" and verify that it matches that of the service provider.
  - DS0 channel mismatch—If you double-check your Data Link Layer configurations but cannot discover what is causing the problem, you may want to recheck the physical interface, even if its status is up. Mismatched channels might not cause a problem until you attempt to transmit data across a link. Use the **show interface** command for the physical interface and check that you have dedicated the same number of channels to the carrier line as your service provider. Use the **tdm-group** command to establish the correct number of channels for the interface.
  - DLCI error—If you have configured the wrong DLCI number for the Frame Relay interface, the Frame Relay connection cannot be established. Double-check the DLCI to ensure that you are using the correct setting.

Displaying the PVC Status. You can view PVC statistics to monitor the connection end-to-end and check for problems with traffic congestion and dropped packets. From the enable mode context, enter:

```
ProCurve# show frame-relay pvc
```

The CLI displays information about each Frame Relay port, including how many active, inactive, and deleted connections it has established. Table 6-14 shows possible PVC status terms and explains what each one means.

Table 6-14. Status of the PVC

Status of the PVC	Explanation
active	The PVC is functional, end-to-end, from the local router to the switch and then to the far-end router
inactive	The PVC is functional from the router to the Frame Relay switch. The other side of the connection is not configured or is down.
deleted	The PVC was announced to the Frame Relay switch but was then deleted. This status appears if the DLCI on the router does not match the DLCI configured for this PVC at the Frame Relay switch.

Information about each PVC is listed under the sublink's DLCI and subinterface numbers. Check the settings listed in Table 6-15.

Table 6-15. Checking the Frame Relay Settings

Status of the PVC	Explanation
DLCI	Misconfiguring the DLCI can prevent traffic from reaching its destination. Verify that the sublink's DLCI is valid. You should configure a unique DLCI in a separate subinterface for each site to which you want to make a Frame Relay connection.
dropped packets	The interface may drop more packets when the Frame Relay network is congested or when the two endpoints of a PVC use different amounts of bandwidth.
FECN/BECN packets	The endpoint that is transmitting data sends forward explicit congestion notification (FECN) packets when the receiver is sending too many requests for data. When its queues fill, the endpoint that is receiving data sends backward explicit congestion notification (BECN) packets to request the source to stop sending so many packets. Endpoints use these messages to minimize the number of dropped packets. A large number of incoming FECN and BECN packets indicate that the other end of the circuit cannot transmit and receive data as quickly as this interface. This discrepancy can lead to dropped packets.
DE packets	When the interface bursts data across the PVC at rates beyond its CIR, the excess packets are marked with the DE bit. If the network becomes congested, these packets will be the first to be dropped.

View LMI Messages. To receive real-time messages, enter the following command from the enable mode context:

```
ProCurve# debug frame-relay lmi
```

The CLI displays events dealing with the establishment and negotiation of connection as they occur. You can then determine when and why problems occur.

LMI statistics report on the LMI messages that are exchanged between the Frame Relay DTE and the DCE. The DCE uses LMI messages to advertise its DLCI. In addition, the LMI messages serve as a local keepalive, indicating that the interface is receiving polls from the other end of the connection.

Clear Counters. If you view the LMI statistics, you will see a running count of polls sent and received, including those incremented before the interface began having a problem. Because you are not interested in how many polls the interface was receiving when it was functioning properly, you should reset the counters to isolate the problem. To reset all counters associated with a Frame Relay interface, enter the following command from the enable mode context:

Syntax: clear counters frame-relay <*number*>

After you clear the counters, you can reproduce the problem and then view the LMI statistics to check whether the interface is receiving polls.

Troubleshooting HDLC

You should begin troubleshooting the HDLC interface by entering the **show interface hdlc** command. From the enable mode context, enter:

Syntax: show interface hdlc <*number*>

Replace <*number*> with the number you assigned the HDLC interface.

If the HDLC interface is administratively down, enter **no shutdown** from the HDLC interface configuration mode context. If the HDLC interface is down, check the running-config to ensure that the HDLC interface is bound to the correct physical interface. From the enable mode context, enter:

Syntax: show running-config interface hdlc <*number*>

Debug HDLC. You can view real-time events about the HDLC interface by entering:

Syntax: debug hdlc [errors | verbose]

Use the **errors** option to view statistics and messages about protocol errors. Use the **verbose** option to increase the level of detail provided in the debug messages.

To disable the hdlc debug messages, enter one of the following commands from the enable mode context:

ProCurve# no debug hdlc [errors | verbose]

or

ProCurve# undebug all

Quick Start

After you configure the physical connection—the E1, T1, or serial interface—you must configure the Data Link Layer protocol that controls the data being transmitted across the WAN link. The ProCurve Secure Router supports the following Data Link Layer protocols for E1, T1, and serial interfaces:

- Point-to-Point Protocol (PPP)
- Frame Relay
- High-Level Data Link Control (HDLC)

This section provides the commands you must enter to quickly configure the Data Link Layer protocol for an E1, T1, or serial interface. Only a minimal explanation is provided.

If you need additional information about any of these options, see "Contents" on page 6-1 to locate the section and page number that contains the explanation you need. (For information about E1 or T1 interface, see *Chapter 4: Configuring E1 and T1 Interfaces*. For information about serial interfaces, see *Chapter 5: Configuring Serial Interfaces for E1- and T1-Carrier Lines*.)

PPP

To configure PPP for an E1, T1, or serial interface, complete these steps:

1. From the global configuration mode context, create a PPP interface.

   Syntax: interface <*interface*> <*number*>

   For example, you might enter:

   ProCurve(config)# interface ppp 1

2. Set a static IP address.

   Syntax: ip address <*A.B.C.D*> <*subnet mask* | /*prefix length*>

   For example, you might enter:

   ProCurve(config-ppp 1)# ip address 10.1.1.1 /24

3. Activate the PPP interface

   ProCurve(config-ppp 1)# no shutdown

4. Bind the physical interface to the logical interface.

   Syntax: bind <*number*> <*physical interface*> <*slot*>/<*port*> [<*tdm-group number*>] <*logical interface*> <*logical interface number*>

   For example, to bind the T1 interface to the PPP interface, enter:

   ProCurve(config-ppp 1)# bind 1 t1 1/1 1 ppp 1

   To bind the serial interface to the PPP interface, enter:

   ProCurve(config-ppp 1)# bind 1 ser 1/1 ppp 1

Note If you are binding a serial interface to the PPP interface, you do not include the TDM group number because you do not use TDM groups on a serial interface.

5. View the status of the PPP interface.

   ProCurve(config-ppp 1)# do show interface ppp 1

PPP Authentication

If you are configuring PPP authentication, you may want to print Table 6-16 and enter the information for your router.

Table 6-16. Quick Start Worksheet

Parameter	Your Setting
PPP interface number	
authentication protocol	
Are you requiring the peer to authenticate itself?	Yes/No
peer username	
peer password	

Parameter	Your Setting
Are you authenticating to the peer?	Yes/No
local router's username	
local router's password	

Requiring the Peer to Authenticate Itself

1. Move to the PPP interface for the connection whose endpoint you want to authenticate. From the global configuration mode context, enter:

   Syntax: interface ppp <*interface number*>

2. Choose the authentication type:

   Syntax: ppp authentication [chap | pap]

3. Enter the *peer's* username and password. If you are using CHAP, the username should be the peer's hostname:

   Syntax: username <*username*> password <*password*>

   For example, if the peer's hostname is Remote and the password is procurve, enter:

   ProCurve(config-ppp 1)# username Remote password procurve

Authenticating to a Peer

1. Move to the PPP interface for the connection whose endpoint requires the router to authenticate itself. From the global configuration mode context, enter:

   Syntax: interface ppp <*interface number*>

2. Determine whether the peer uses PAP or CHAP authentication.

3. For PAP, enter the username and password you have agreed upon for the *local* router:

   Syntax: ppp pap sent-username <*username*> password <*password*>

   For example, you might enter:

   ProCurve(config-ppp 1)# ppp pap sent-username Local password procurve

4. For CHAP, enter the password you have agreed upon for the local router:

   Syntax: ppp chap password <*password*>

5. For CHAP, enter a username only if it is different from the router's hostname:

   Syntax: ppp chap hostname <*username*>

   For example, you might enter:

   ProCurve(config-ppp 1)# ppp chap hostname ProCurveA

Frame Relay

Before you begin to configure the Frame Relay interface, you should know the settings that you must enter for the following:

- Frame Relay signaling role:
  - user, or data terminal equipment (DTE)
  - network, or data communications equipment (DCE)
  - both, or network-to-network interfaces (NNI)
- Frame Relay signaling type, which is referred to as the link management interface (LMI)
- data link connection identifier (DLCI)
- your negotiated committed information rate (CIR), which is configured as your B_c
- your negotiated excess information rate (EIR), which is configured as your B_e.

Your public carrier should provide you with this information.

Note With few exceptions, you will configure the signaling role as DTE. However, the other options are available if you need to change the setting for any reason. For example, you may want the router to act as a DCE in a test WAN environment.

To configure Frame Relay for an E1, T1, or serial interface, complete these steps:

1. From the global configuration mode context, create a Frame Relay interface.

   Syntax: interface <*interface*> <*number*>

   ProCurve(config)# interface frame-relay 1

2. Define the signaling role for the Frame Relay interface. The default setting is **dte**, or user.

   Syntax: frame-relay intf-type [dce | dte | nni]

   ProCurve(config-fr 1)# frame-relay intf-type dte

3. Define the signaling type (the LMI). The default setting is **ansi**, or Annex D.

   Syntax: frame-relay lmi-type [ansi | auto | cisco | none | q933a]

   For example, to set the signaling type to **auto**, enter:

   ProCurve(config-fr 1)# frame-relay lmi-type auto

4. Activate the Frame Relay interface.

   ProCurve(config-fr 1)# no shutdown

5. Create a Frame Relay subinterface for each permanent virtual circuit (PVC). Enter any number from 16 to 1007 for the sublink number. Using the same number as the subinterface's DLCI will help you keep track of and troubleshoot the sublink.

   Syntax: interface frame-relay <*number.subinterface number*>

   ProCurve(config-fr 1)# interface frame-relay 1.103

6. Assign the subinterface a DLCI.

   Syntax: frame-relay interface-dlci <*DLCI*>

   ProCurve(config-fr 1.103)# frame-relay interface-dlci 103

7. Assign the interface a static IP address.

   Syntax: ip address <*A.B.C.D*> <*subnet mask* | /*prefix length*>

   ProCurve(config-fr 1.103)# ip address 10.1.1.1 /24

8. Configure a CIR.

   Syntax: frame-relay bc <*committed burst value*>

   Replace <*committed burst value*> with your CIR expressed in bytes. For example, you might enter:

   ProCurve(config-fr 1.1)# frame-relay bc 128000

9. Set the excessive burst rate.

   Syntax: frame-relay be <*excessive burst value*>

   Replace <*excessive burst value*> with a burst rate, expressed in bytes. For example, you might enter:

   ProCurve(config-fr 1.1)# frame-relay be 64000

Note

Together, the **frame-relay bc** command and the **frame-relay be** command define the amount of bandwidth you can use on the Frame Relay link. The sum of the values you specify for these two settings should be greater than 8000.

10. Bind the physical interface—the E1, T1, or serial interface—to the Frame Relay interface. From the global configuration mode context, enter:

    Syntax: bind <*number*> <*physical interface*> <*slot*> /<*port*> [<*tdm-group number*>] <*logical interface*> <*logical interface number*>

    For example, to bind the E1 1/1 interface to the Frame Relay 1 interface, enter:

    ```
    ProCurve(config)# bind 1 e1 1/1 1 fr 1
    ```

    To bind the serial 1/1 interface to the Frame Relay 1 interface, enter:

    ```
    ProCurve(config)# bind 1 ser 1/1 fr 1
    ```

Note

If you are binding a serial interface to the Frame Relay interface, you do not include the TDM group number because you do not use TDM groups on a serial interface.

11. View the status of the Frame Relay interface and subinterface. From the enable mode context, enter:

    ```
    ProCurve# show interface fr 1
    ProCurve# show interface fr 1.103
    ```

HDLC

To configure HDLC for an E1, T1, or serial interface, complete these steps:

1. From the global configuration mode context, create an HDLC interface.

   Syntax: interface <*interface*> <*number*>

   ```
   ProCurve(config)# interface hdlc 1
   ```

2. Enter the IP address.

   Syntax: ip address <*A.B.C.D*> <*subnet mask* | /*prefix length*>

   ```
   ProCurve(config-hdlc 1)# ip address 10.1.1.1 /24
   ```

3. Activate the HDLC 1 interface

   ```
   ProCurve(config-hdlc 1)# no shutdown
   ```

4. Bind the physical interface—the E1, T1, or serial interface—to the logical interface.

   Syntax: bind <*number*> <*physical interface*> <*slot*>/<*port*> [<*tdm-group number*>] <*logical interface*> <*logical interface number*>

   For example, to bind the E1 1/1 interface to the HDLC 1 interface, enter:

   ```
   ProCurve(config-hdlc 1)# bind 1 e1 1/1 1 hdlc 1
   ```

   To bind the serial 1/1 interface to the HDLC 1 interface, enter:

   ```
   ProCurve(config-hdlc 1)# bind 1 ser 1/1 hdlc 1
   ```

Note

If you are binding a serial interface to the HDLC interface, you do not include the TDM group number because you do not use TDM groups on a serial interface.

5. View the status of the HDLC interface. From the enable mode context, enter:

   ```
   ProCurve# show interface hdlc 1
   ```