CHAPTER 3

Point-to-Point Connections

Objectives

Upon completion of this chapter

- What are the fundamentals of point-to-point serial communications across a WAN?
- How do you configure HDLC encapsulation on a point-to-point serial link?
- What are the benefits of using PPP over HDLC in a WAN?
- What is the PPP layered architecture and the functions of LCP and NCP?
- How is a PPP session established?
- How do you configure PPP encapsulation on a point-to-point serial link?
- How do you configure PPP authentication protocols?
- How are the **show** and **debug** commands used to troubleshoot PPP?

Key Terms

This chapter uses the following key terms. You can find the definitions in the glossary.

point-to-point connections *page 80*

clock skew *page 82*

time-division multiplexing (TDM) *page 85*

statistical time-division multiplexing (STDM) *page 85*

data stream *page 85*

transmission link *page 85*

demarcation point *page 88*

null modem *page 91*

DS (digital signal level) *page 94*

E1 *page 95*

E3 *page 95*

bit-oriented *page 97*

Synchronous Data Link Control (SDLC) *page 97*

primary station *page 99*

Cisco 7000 *page 103*

trunk lines *page 105*

Link Control Protocol (LCP) *page 105*

Network Control Protocols (NCPs) *page 105*

Novell IPX *page 105*

SNA Control Protocol *page 105*

Password Authentication Protocol (PAP) *page 119*

Challenge Handshake Authentication Protocol (CHAP) *page 119*

fragmentation *page 119*

reassembly *page 119*

message digest 5 (MD5) *page 130*

TACACS/TACACS+ *page 135*

Introduction (3.0.1.1)

One of the most common types of WAN connections, especially in long-distance communications, is a *point-to-point connection*, also called a serial or leased line connection. Because these connections are typically provided by a carrier, such as a telephone company, boundaries between what is managed by the carrier and what is managed by the customer must be clearly established.

This chapter covers the terms, technology, and protocols used in serial connections. The High-Level Data Link Control (HDLC) and Point-to-Point Protocol (PPP) are introduced. PPP is a protocol that is able to handle authentication, compression, error detection, monitor link quality, and logically bundles multiple serial connections together to share the load.

Class Activity 3.0.1.2: PPP Persuasion

Your network engineering supervisor recently attended a networking conference where Layer 2 protocols were discussed. He knows that you have Cisco equipment on the premises, but he would also like to offer security and advanced TCP/IP options and controls on that same equipment by using the Point-to-Point Protocol (PPP).

After researching the PPP protocol, you find it offers some advantages over the HDLC protocol, currently used on your network.

Create a matrix listing the advantages and disadvantages of using the HDLC vs. PPP protocols. When comparing the two protocols, include

- Ease of configuration
- Adaptability to non-proprietary network equipment
- Security options
- Bandwidth usage and compression
- Bandwidth consolidation

Share your chart with another student or class. Justify whether or not you would suggest sharing the matrix with the network engineering supervisor to justify a change being made from HDLC to PPP for Layer 2 network connectivity.

Serial Point-to-Point Overview (3.1)

This section gives an overview of point-to-point serial communications. A basic understanding of point-to-point serial communications is essential to understanding protocols that are used over these types of serial links. HDLC encapsulation and configuration is discussed later in this section.

Serial Communications (3.1.1)

The earliest form of computer communications involved serial links between mainframe computers. Serial communications is still a widely used method of connecting two networks usually over long distances.

Serial and Parallel Ports (3.1.1.1)

One of the most common types of WAN connections is the point-to-point connection. As shown in Figure 3-1, point-to-point connections are used to connect LANs to service provider WANs, and to connect LAN segments within an enterprise network.

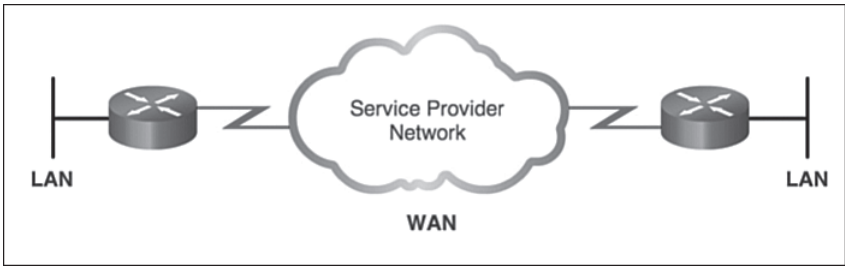


Figure 3-1 Serial Point-to-Point Communications

A LAN-to-WAN point-to-point connection is also referred to as a serial connection or leased line connection. This is because the lines are leased from a carrier (usually a telephone company) and are dedicated for use by the company leasing the lines. Companies pay for a continuous connection between two remote sites, and the line is continuously active and available. Leased lines are a frequently used type of WAN access, and they are generally priced based on the bandwidth required and the distance between the two connected points.

Understanding how point-to-point serial communication across a leased line works is important to an overall understanding of how WANs function.

Communications across a serial connection is a method of data transmission in which the bits are transmitted sequentially over a single channel. This is equivalent to a pipe only wide enough to fit one ball at a time. Multiple balls can go into the pipe, but only one at a time, and they only have one exit point, the other end of the pipe. A serial port is bidirectional, and often referred to as a bidirectional port or a communications port.

This is in contrast to parallel communications in which bits can be transmitted simultaneously over multiple wires. As shown in Figure 3-2, a parallel connection theoretically transfers data eight times faster than a serial connection. Based on this theory, a parallel connection sends a byte (eight bits) in the time that a serial connection sends a single bit. However, parallel communications do have issues with crosstalk across

wires, especially as the wire length increases. *Clock skew* is also an issue with parallel communications. Clock skew occurs when data across the various wires does not arrive at the same time, creating synchronization issues. Finally, most parallel communications support only one-direction, outbound-only communication from the hard drive.

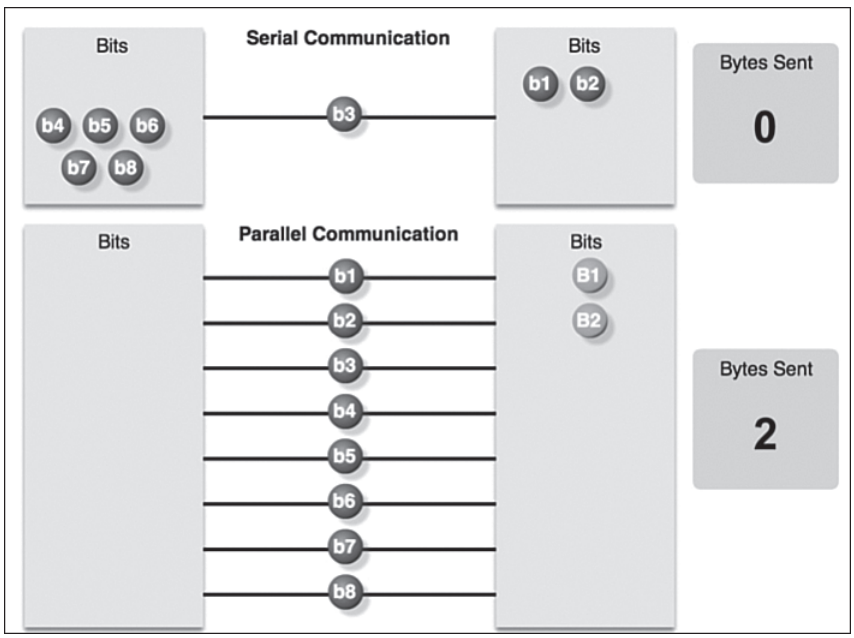


Figure 3-2 Serial and Parallel Communications

At one time, most PCs included both serial and parallel ports. Parallel ports were used to connect printers, computers, and other devices that required relatively high bandwidth. Parallel ports were also used between interior components. For external communications, a serial bus was primarily used for signal conversion. Because of their bidirectional ability, serial communications are considerably less expensive to implement. Serial communications use fewer wires, cheaper cables, and fewer connector pins.

On most PCs, parallel ports and RS-232 serial ports have been replaced by the higher speed serial Universal Serial Bus (USB) interfaces. However, for long-distance communication, many WANs use still serial transmission.

Serial Communication (3.1.1.2)

Figure 3-3 shows a simple representation of a serial communication across a WAN. Data is encapsulated by the communications protocol used by the sending router. The encapsulated frame is sent on a physical medium to the WAN. There are various ways to traverse the WAN, but the receiving router uses the same communications protocol to de-encapsulate the frame when it arrives.

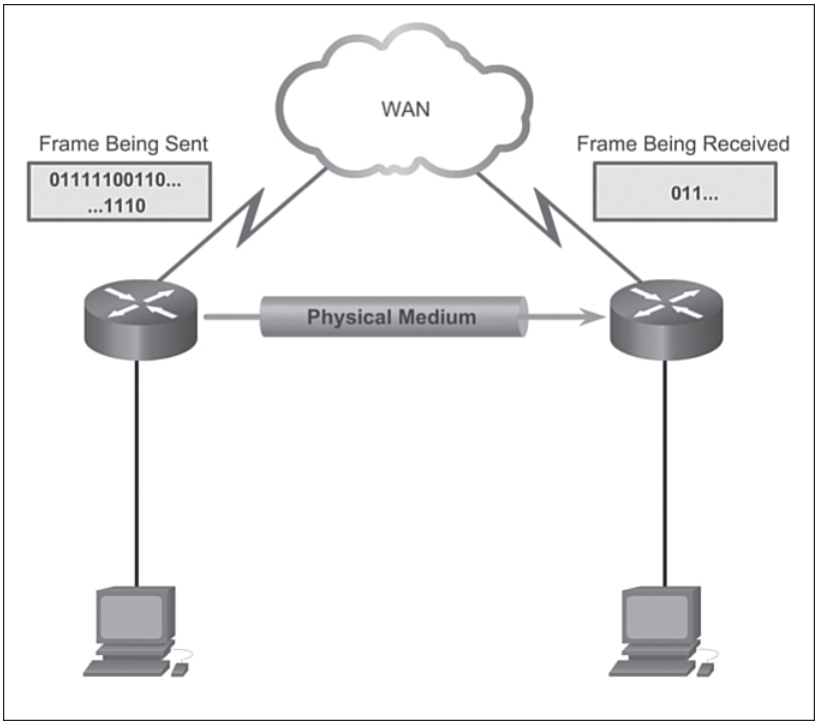


Figure 3-3 Serial Communication Process

There are many different serial communication standards, each one using a different signaling method. There are three important serial communication standards affecting LAN-to-WAN connections:

- **RS-232**: Most serial ports on personal computers conform to the RS-232C or newer RS-422 and RS-423 standards. Both 9-pin and 25-pin connectors are used. A serial port is a general-purpose interface that can be used for almost any type of device, including modems, mice, and printers. These types of peripheral devices for computers have been replaced by new and faster standards such as USB but many network devices use RJ-45 connectors that conform to the original RS-232 standard.
- **V.35**: Typically used for modem-to-multiplexer communication, this ITU standard for high-speed, synchronous data exchange combines the bandwidth of several telephone circuits. In the U.S., V.35 is the interface standard used by most routers and DSUs that connect to T1 carriers. V.35 cables are high-speed serial assemblies designed to support higher data rates and connectivity between DTEs and DCEs over digital lines. There is more on DTEs and DCEs later in this section.

- **HSSI:** A High-Speed Serial Interface (HSSI) supports transmission rates up to 52 Mbps. Engineers use HSSI to connect routers on LANs with WANs over high-speed lines, such as T3 lines. Engineers also use HSSI to provide high-speed connectivity between LANs, using Token Ring or Ethernet. HSSI is a DTE/DCE interface developed by Cisco Systems and T3 Plus Networking to address the need for high-speed communication over WAN links.

Point-to-Point Communication Links (3.1.1.3)

When permanent dedicated connections are required, a point-to-point link is used to provide a single, pre-established WAN communications path from the customer premises, through the provider network, to a remote destination, as shown in Figure 3-4.

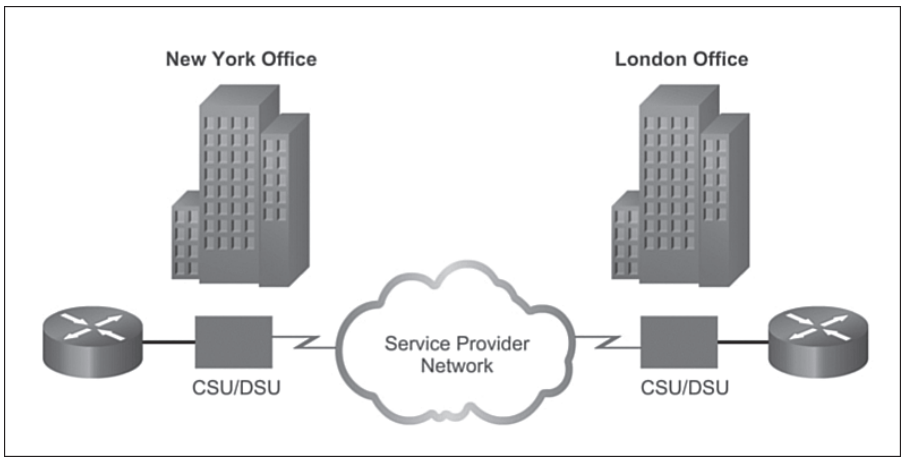


Figure 3-4 Point-to-Point Communication Links

A point-to-point link can connect two geographically distant sites, such as a corporate office in New York and a regional office in London. For a point-to-point line, the carrier dedicates specific resources for a line that is leased by the customer (leased line).

Note

Point-to-point connections are not limited to connections that cross land. There are hundreds of thousands of miles of undersea fiber-optic cables that connect countries and continents worldwide. An Internet search of "undersea Internet cable map" produces several cable maps of these undersea connections.

Point-to-point links are usually more expensive than shared services. The cost of leased line solutions can become significant when used to connect many sites over increasing distances. However, there are times when the benefits outweigh the cost of the leased line. The dedicated capacity removes latency or jitter between the endpoints. Constant availability is essential for some applications such as VoIP or video over IP.

Time-Division Multiplexing (3.1.1.4)

With a leased line, despite the fact that customers are paying for dedicated services, and dedicated bandwidth is provided to the customer, the carrier still uses multiplexing technologies within the network. Multiplexing refers to a scheme that allows multiple logical signals to share a single physical channel. Two common types of multiplexing are ***time-division multiplexing (TDM)*** and ***statistical time-division multiplexing (STDM)***.

TDM

Bell Laboratories originally invented TDM to maximize the amount of voice traffic carried over a medium. Before multiplexing, each telephone call required its own physical link. This was an expensive and unscalable solution. TDM divides the bandwidth of a single link into separate time slots. TDM transmits two or more channels (***data stream***) over the same link by allocating a different time slot for the transmission of each channel. In effect, the channels take turns using the link.

TDM is a physical layer concept. It has no regard for the nature of the information that is multiplexed on to the output channel. TDM is independent of the Layer 2 protocol that has been used by the input channels.

TDM can be explained by an analogy to highway traffic. To transport traffic from four roads to another city, all traffic can be sent on one lane if the roads are equally serviced and the traffic is synchronized. If each of the four roads puts a car on to the main highway every four seconds, the highway gets a car at the rate of one each second. As long as the speed of all the cars is synchronized, there is no collision. At the destination, the reverse happens and the cars are taken off the highway and fed to the local roads by the same synchronous mechanism.

This is the principle used in synchronous TDM when sending data over a link. TDM increases the capacity of the ***transmission link*** by dividing transmission time into smaller, equal intervals so that the link carries the bits from multiple input sources.

In Figure 3-5, a multiplexer (MUX) at the transmitter accepts three separate signals. The MUX breaks each signal into segments. The MUX puts each segment into a single channel by inserting each segment into a time slot.

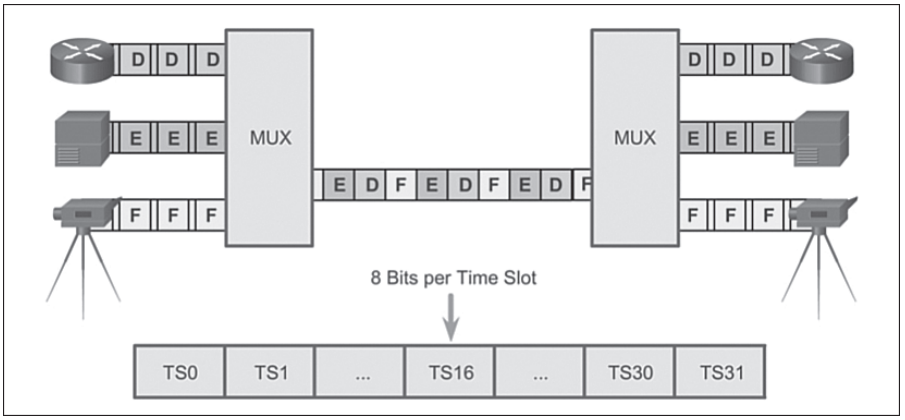


Figure 3-5 Time-Division Multiplexing

A MUX at the receiving end reassembles the TDM stream into the three separate data streams based only on the timing of the arrival of each bit. A technique called bit interleaving keeps track of the number and sequence of the bits from each specific transmission so that they can be quickly and efficiently reassembled into their original form upon receipt. Byte interleaving performs the same functions, but because there are eight bits in each byte, the process needs a bigger or longer time slot.

The operations of TDM are summarized as follows:

- TDM shares available transmission time on a medium by assigning a time slot to users.
- The MUX accepts input from attached devices in an alternating sequence (round-robin) and transmits the data in a recurrent pattern.
- T1/E1 and ISDN telephone lines are common examples of synchronous TDM.

Statistical Time-Division Multiplexing (3.1.1.5)

In another analogy, compare TDM to a train with 32 railroad cars. Each car is owned by a different freight company, and every day the train leaves with the 32 cars attached. If one of the companies has cargo to send, the car is loaded. If the company has nothing to send, the car remains empty, but stays on the train. Shipping empty containers is not very efficient. TDM shares this inefficiency when traffic is intermittent, because the time slot is still allocated even when the channel has no data to transmit.

STDM

STDM was developed to overcome this inefficiency. As shown in Figure 3-6, STDM uses a variable time slot length allowing channels to compete for any free slot space. It employs a buffer memory that temporarily stores the data during periods of peak traffic. STDM does not waste high-speed line time with inactive channels using this scheme. STDM requires each transmission to carry identification information or a channel identifier.

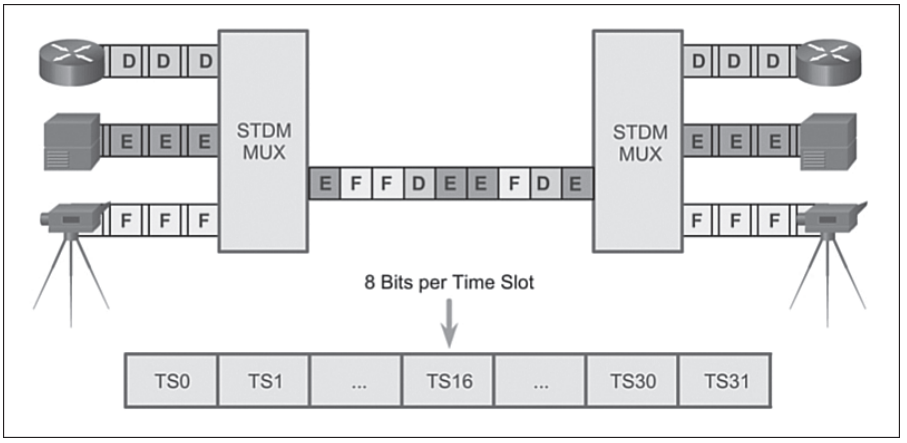


Figure 3-6 Statistical Time-Division Multiplexing

TDM Examples – Sonet and SDM (3.1.1.6)

On a larger scale, the telecommunications industry uses the Synchronous Optical Networking (SONET) or Synchronous Digital Hierarchy (SDH) standard for optical transport of TDM data. SONET, used in North America, and SDH, used elsewhere, are two closely related standards that specify interface parameters, rates, framing formats, multiplexing methods, and management for synchronous TDM over fiber.

Figure 3-7 displays SONET, which is an example of STDM. SONET/SDH takes *n* bit streams, multiplexes them, and optically modulates the signals. It then sends the signals out using a light emitting device over fiber with a bit rate equal to (incoming bit rate) ´ *n*. Thus, traffic arriving at the SONET multiplexer from four places at 2.5 Gbps goes out as a single stream at 4 ´ 2.5 Gbps, or 10 Gbps. This principle is illustrated in the figure, which shows an increase in the bit rate by a factor of four in time slot T.

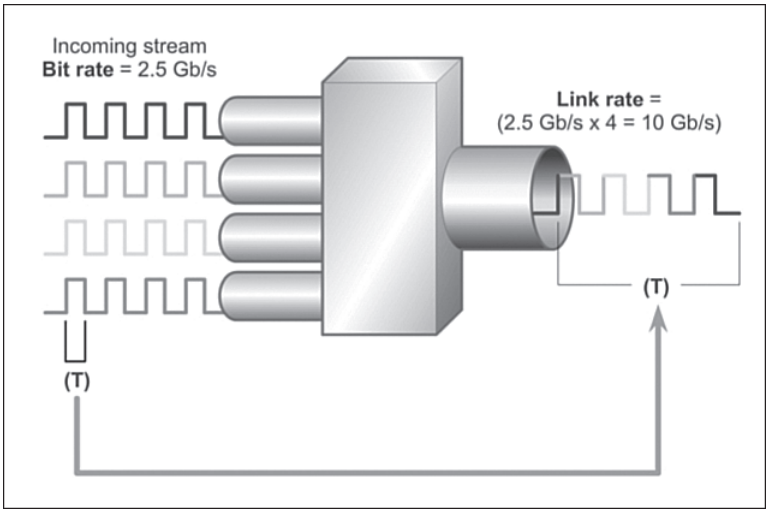


Figure 3-7 TDM Example: SONET

Demarcation Point (3.1.1.7)

Prior to deregulation in North America and other countries, telephone companies owned the local loop, including the wiring and equipment on the premises of the customers. The local loop refers to the line from the premises of a telephone subscriber to the telephone company central office. Deregulation forced telephone companies to unbundle their local loop infrastructure to allow other suppliers to provide equipment and services. This led to a need to delineate which part of the network the telephone company owned and which part the customer owned. This point of delineation is the *demarcation point*, or demarc. The demarcation point marks the point where your network interfaces with a network that is owned by another organization. In telephone terminology, this is the interface between customer premises equipment (CPE) and network service provider equipment. The demarcation point is the point in the network where the responsibility of the service provider ends, as shown in Figure 3-8.

The differences in demarcation points can best be shown using ISDN. In the United States, a service provider provides the local loop into the customer premises, and the customer provides the active equipment such as the channel service unit / data service unit (CSU/DSU) on which the local loop is terminated. This termination often occurs in a telecommunications closet, and the customer is responsible for maintaining, replacing, or repairing the equipment. In other countries, the network terminating unit (NTU)

is provided and managed by the service provider. This allows the service provider to actively manage and troubleshoot the local loop with the demarcation point occurring after the NTU. The customer connects a CPE device, such as a router or Frame Relay access device, to the NTU using a V.35 or RS-232 serial interface.

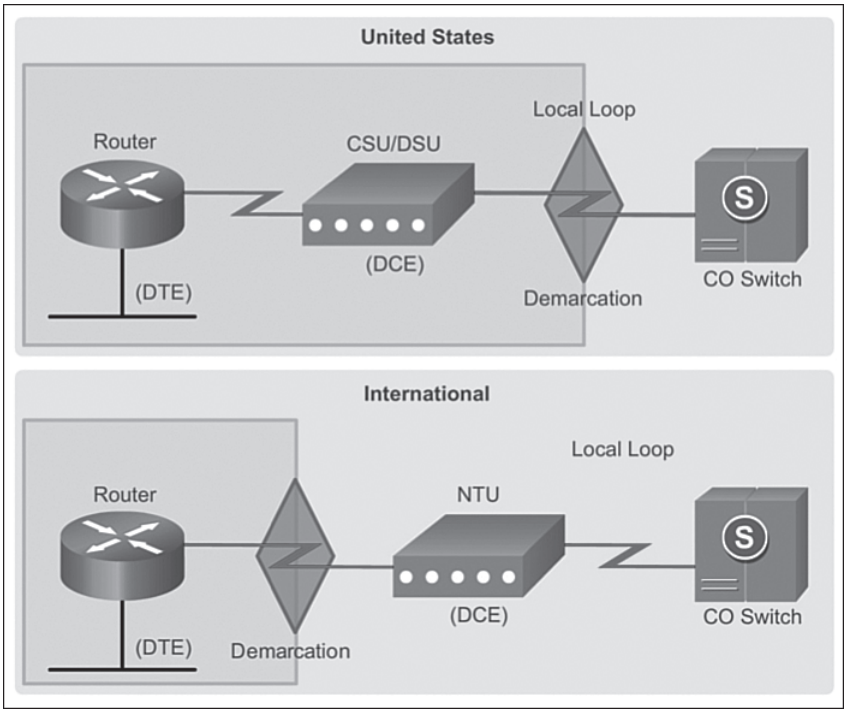


Figure 3-8 Demarcation Point

A router serial port is required for each leased line connection. If the underlying network is based on the T-carrier or E-carrier technologies, the leased line connects to the network of the carrier through a CSU/DSU. The purpose of the CSU/DSU is to provide a clocking signal to the customer equipment interface from the DSU and terminate the channelized transport media of the carrier on the CSU. The CSU also provides diagnostic functions such as a loopback test.

As shown in Figure 3-9, most T1 or E1 TDM interfaces on current routers include CSU/DSU capabilities. A separate CSU/DSU is not required because this functionality is embedded in the interface. IOS commands are used to configure the CSU/DSU operations.

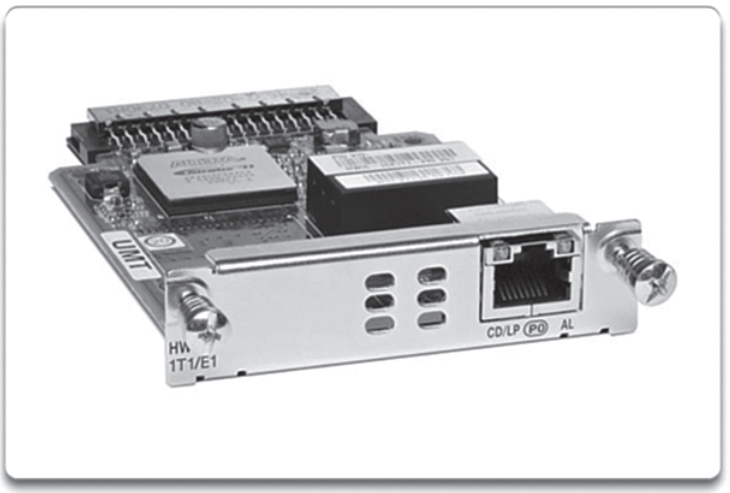


Figure 3-9 T1/E1 with Embedded CSU/DSU

DTE-DCE (3.1.1.8)

From the point of view of connecting to the WAN, a serial connection has a data terminal equipment (DTE) device at one end of the connection and a data circuit-terminating equipment or data communications equipment (DCE) device at the other end. The connection between the two DCE devices is the WAN service provider transmission network, as shown in Figure 3-10. In this example

- The CPE, which is generally a router, is the DTE. The DTE could also be a terminal, computer, printer, or fax machine if they connect directly to the service provider network.
- The DCE, commonly a modem or CSU/DSU, is the device used to convert the user data from the DTE into a form acceptable to the WAN service provider transmission link. This signal is received at the remote DCE, which decodes the signal back into a sequence of bits. The remote DCE then signals this sequence to the remote DTE.

The Electronics Industry Association (EIA) and the International Telecommunication Union Telecommunications Standardization Sector (ITU-T) have been most active in the development of standards that allow DTEs to communicate with DCEs.

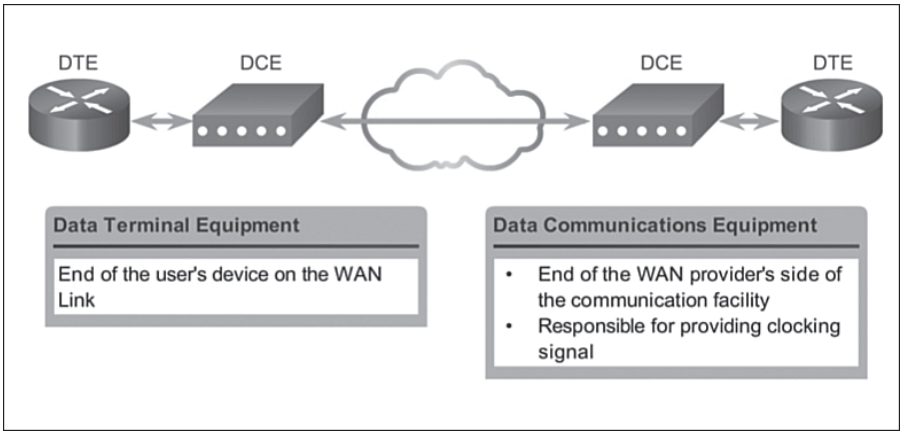


Figure 3-10 Serial DCE and DTE WAN Connections

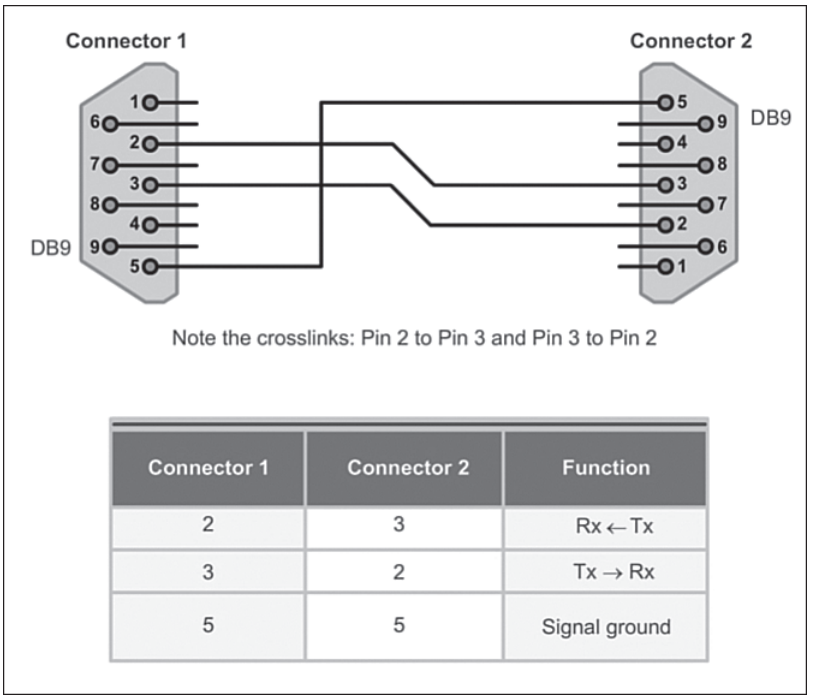
Serial Cables (3.1.1.9)

Originally, the concept of DCEs and DTEs was based on two types of equipment: terminal equipment that generated or received data, and the communication equipment that only relayed data. In the development of the RS-232 standard, there were reasons why 25-pin RS-232 connectors on these two types of equipment must be wired differently. These reasons are no longer significant, but there are two different types of cables remaining: one for connecting a DTE to a DCE, and another for connecting two DTEs directly to each other.

The DTE/DCE interface for a particular standard defines the following specifications:

- **Mechanical/physical:** Number of pins and connector type
- **Electrical:** Defines voltage levels for 0 and 1
- **Functional:** Specifies the functions that are performed by assigning meanings to each of the signaling lines in the interface
- **Procedural:** Specifies the sequence of events for transmitting data

The original RS-232 standard only defined the connection of DTEs with DCEs, which were modems. However, to connect two DTEs, such as two computers or two routers in a lab, a special cable called a null modem eliminates the need for a DCE. In other words, the two devices can be connected without a modem. A ***null modem*** is a communication method to directly connect two DTEs using an RS-232 serial cable. With a null modem connection, the transmit (Tx) and receive (Rx) lines are cross-linked, as shown in Figure 3-11.



Connector 1	Connector 2	Function
2	3	Rx ← Tx
3	2	Tx → Rx
5	5	Signal ground

Figure 3-11 Null Modem to Connect Two DTEs

The cable for the DTE to DCE connection is a shielded serial transition cable. The router end of the shielded serial transition cable may be a DB-60 connector, which connects to the DB-60 port on a serial WAN interface card, as shown in Figure 3-12. The other end of the serial transition cable is available with the connector appropriate for the standard that is to be used. The WAN provider or the CSU/DSU usually dictates this cable type. Cisco devices support the EIA/TIA-232, EIA/TIA-449, V.35, X.21, and EIA/TIA-530 serial standards, as shown in Figure 3-13.

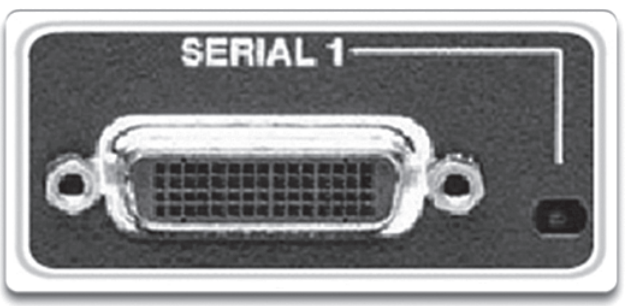


Figure 3-12 DB-60 Router Connection

To support higher port densities in a smaller form factor, Cisco has introduced a Smart Serial cable, as shown in Figure 3-14. The router interface end of the Smart Serial cable is a 26-pin connector that is significantly more compact than the DB-60 connector.

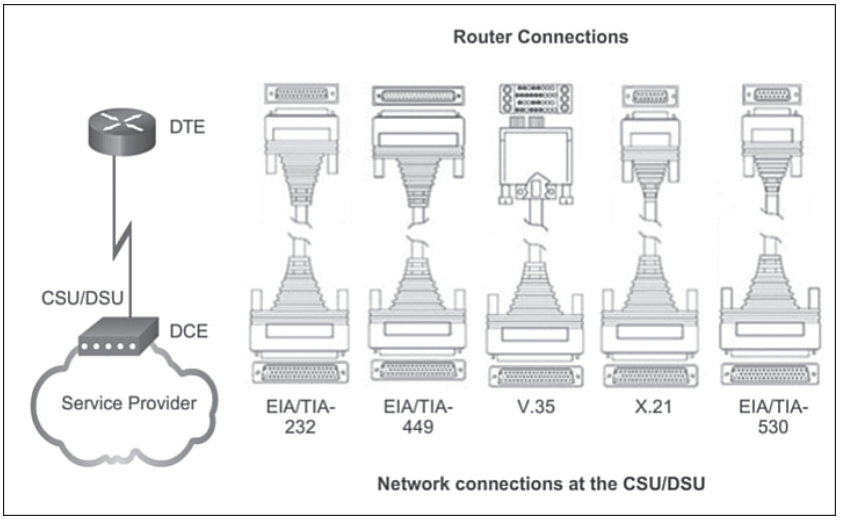


Figure 3-13 WAN Serial Connection Options

Figure 3-14 Smart Serial Connector

When using a null modem, synchronous connections require a clock signal. An external device can generate the signal, or one of the DTEs can generate the clock signal. When a DTE and DCE are connected, the serial port on a router is the DTE end of the connection, by default, and the clock signal is typically provided by a CSU/DSU, or similar DCE device. However, when using a null modem cable in a router-to-router connection, one of the serial interfaces must be configured as the DCE end to provide the clock signal for the connection, as shown in Figure 3-15.

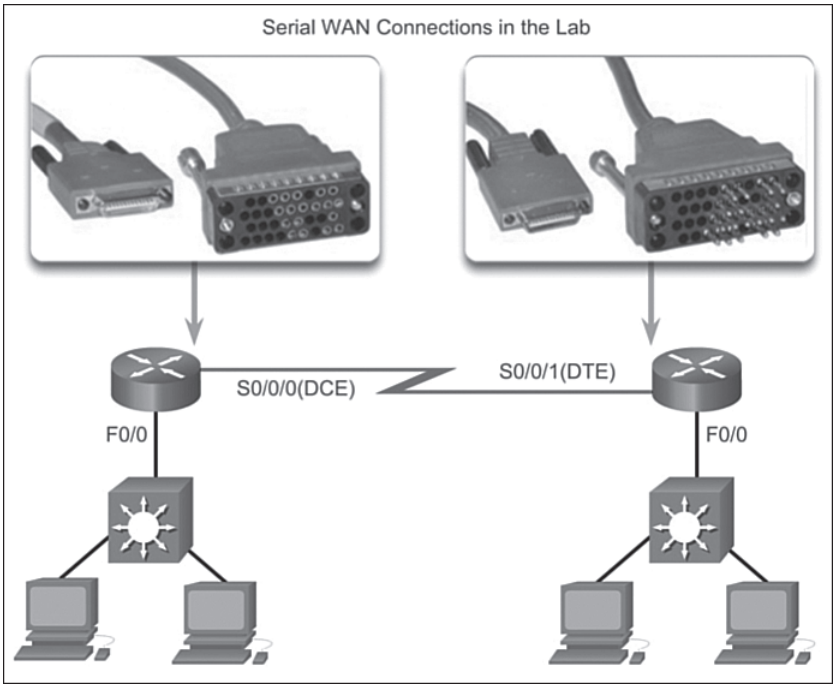


Figure 3-15 Smart Serial Connections in the Lab

Serial Bandwidth (3.1.1.10)

Bandwidth refers to the rate at which data is transferred over the communication link. The underlying carrier technology depends on the bandwidth available. There is a difference in bandwidth points between the North American (T-carrier) specification and the European (E-carrier) system. Optical networks also use a different bandwidth hierarchy, which again differs between North America and Europe. In the United States, Optical Carrier (OC) defines the bandwidth points.

In North America, the bandwidth is usually expressed as a ***DS (digital signal level)*** number (DS0, DS1, etc.), which refers to the rate and format of the signal. The most fundamental line speed is 64 Kbps, or DS-0, which is the bandwidth required for an uncompressed, digitized phone call. Serial connection bandwidths can be incrementally increased to accommodate the need for faster transmission. For example, 24 DS0s can be bundled to get a DS1 line (also called a T1 line) with a speed of 1.544 Mbps. Also, 28 DS1s can be bundled to get a DS3 line (also called a T3 line) with a speed of 44.736 Mbps. Leased lines are available in different capacities and are generally priced based on the bandwidth required and the distance between the two connected points.

OC transmission rates are a set of standardized specifications for the transmission of digital signals carried on SONET fiber-optic networks. The designation uses OC, followed by an integer value representing the base transmission rate of 51.84 Mbps. For example, OC-1 has a transmission capacity of 51.84 Mbps, whereas an OC-3 transmission medium would be three times 51.84 Mbps, or 155.52 Mbps.

Table 3-1 lists the most common line types and the associated bit rate capacity of each.

Table 3-1 Carrier Transmission Rates

Line Type	Bit Rate Capacity
56	56 Kbps
64	64 Kbps
T1	1.544 Mbps
E1	2.048 Mbps
J1	2.048 Mbps
E3	34.064 Mbps
T3	44.736 Mbps
OC-1	51.84 Mbps
OC-3	155.54 Mbps
OC-9	466.56 Mbps
OC-12	622.08 Mbps
OC-18	933.12 Mbps
OC-24	1.244 Gbps
OC-36	1.866 Gbps
OC-48	2.488 Gbps
OC-96	4.976 Gbps
OC-192	9.954 Gbps
OC-768	39.813 Gbps

Note

E1 (2.048 Mbps) and *E3* (34.368 Mbps) are European standards like T1 and T3, but with different bandwidths and frame structures.

Interactive Graphic

Activity 3.1.1.11: Identify the Serial Communications Terminology

Go to the course online to perform this practice activity.

HDLC Encapsulation (3.1.2)

HDLC is a synchronous data link layer protocol developed by the International Organization for Standardization (ISO). Although HDLC can be used for point-to-multipoint connections, the most common usage of HDLC is for point-to-point serial communications.

WAN Encapsulation Protocols (3.1.2.1)

On each WAN connection, data is encapsulated into frames before crossing the WAN link. To ensure that the correct protocol is used, the appropriate Layer 2 encapsulation type must be configured. The choice of protocol depends on the WAN technology and the communicating equipment. Figure 3-16 displays the more common WAN protocols and where they are used. The following are short descriptions of each type of WAN protocol:

- **HDLC:** The default encapsulation type on point-to-point connections, dedicated links, and circuit-switched connections when the link uses two Cisco devices. HDLC is now the basis for synchronous PPP used by many servers to connect to a WAN, most commonly the Internet.
- **PPP:** Provides router-to-router and host-to-network connections over synchronous and asynchronous circuits. PPP works with several network layer protocols, such as IPv4 and IPv6. PPP uses the HDLC encapsulation protocol, but also has built-in security mechanisms such as PAP and CHAP.
- **Serial Line Internet Protocol (SLIP):** A standard protocol for point-to-point serial connections using TCP/IP. SLIP has been largely displaced by PPP.
- **X.25/Link Access Procedure, Balanced (LAPB):** An ITU-T standard that defines how connections between a DTE and DCE are maintained for remote terminal access and computer communications in public data networks. X.25 specifies LAPB, a data link layer protocol. X.25 is a predecessor to Frame Relay.
- **Frame Relay:** An industry standard, switched, data link layer protocol that handles multiple virtual circuits. Frame Relay is a next-generation protocol after X.25. Frame Relay eliminates some of the time-consuming processes (such as error correction and flow control) employed in X.25.
- **ATM:** The international standard for cell relay in which devices send multiple service types, such as voice, video, or data, in fixed-length (53-byte) cells. Fixed-length cells allow processing to occur in hardware; thereby, reducing transit delays. ATM takes advantage of high-speed transmission media such as E3, SONET, and T3.

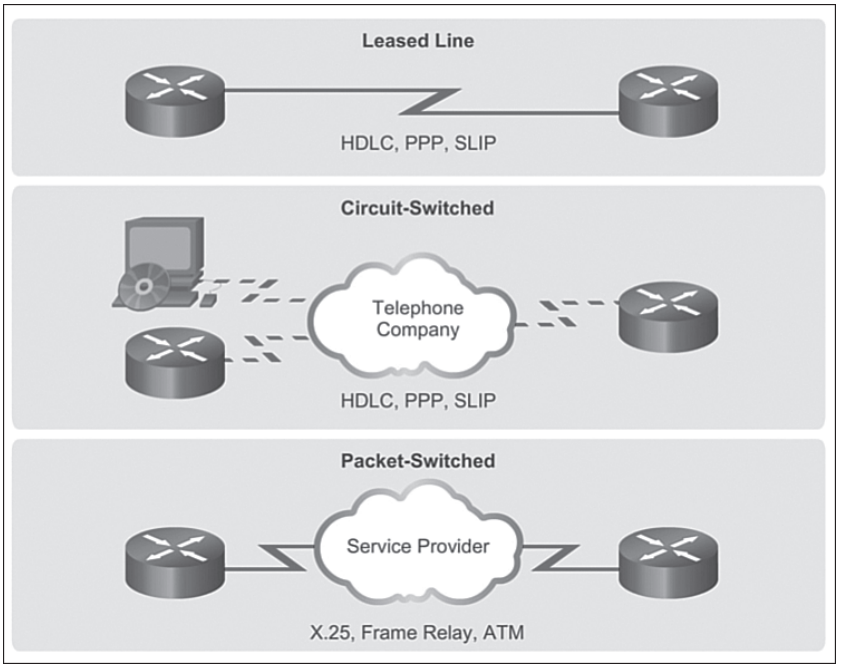


Figure 3-16 WAN Encapsulation Protocols

HDLC Encapsulation (3.1.2.2)

HDLC is a ***bit-oriented*** synchronous data link layer protocol developed by the International Organization for Standardization (ISO). The current standard for HDLC is ISO 13239. HDLC was developed from the ***Synchronous Data Link Control (SDLC)*** standard proposed in the 1970s. HDLC provides both connection-oriented and connectionless service.

HDLC uses synchronous serial transmission to provide error-free communication between two points. HDLC defines a Layer 2 framing structure that allows for flow control and error control through the use of acknowledgments. Each frame has the same format, whether it is a data frame or a control frame.

When frames are transmitted over synchronous or asynchronous links, those links have no mechanism to mark the beginning or end of frames. For this reason, HDLC uses a frame delimiter, or flag, to mark the beginning and the end of each frame.

Cisco has developed an extension to the HLDC protocol to solve the inability to provide multiprotocol support. Although Cisco HLDC (also referred to as cHDLC) is proprietary, Cisco has allowed many other network equipment vendors to implement it. Cisco HDLC frames contain a field for identifying the network protocol being encapsulated. Figure 3-17 compares standard HLDC to Cisco HLDC.

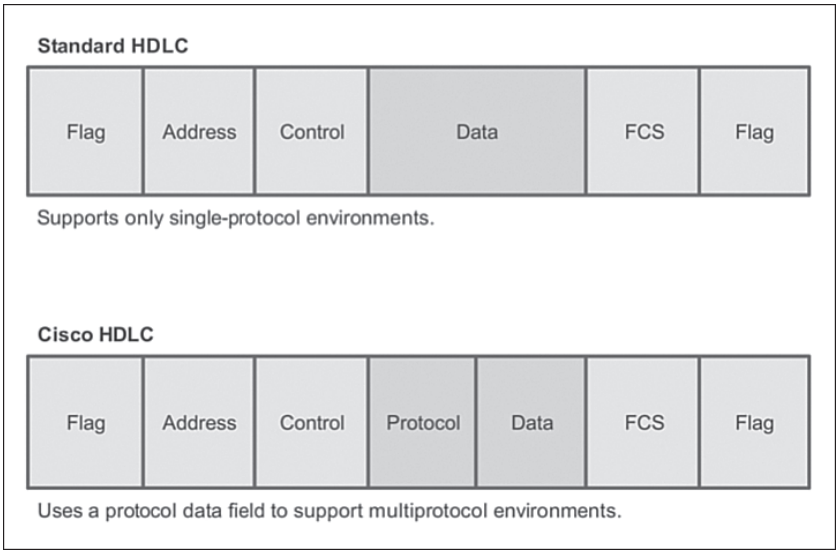


Figure 3-17 Standard and Cisco HLDC Frame Format

HDLC Frame Types (3.1.2.3)

HDLC defines three types of frames, each with a different control field format.

Flag

The Flag field initiates and terminates error checking. The frame always starts and ends with an 8-bit Flag field. The bit pattern is 01111110. Because there is a likelihood that this pattern occurs in the actual data, the sending HDLC system always inserts a 0 bit after every five consecutive 1s in the data field, so in practice the flag sequence can only occur at the frame ends. The receiving system strips out the inserted bits. When frames are transmitted consecutively, the end flag of the first frame is used as the start flag of the next frame.

Address

The Address field contains the HDLC address of the secondary station. This address can contain a specific address, a group address, or a broadcast address. A primary address is either a communication source or a destination, which eliminates the need to include the address of the primary.

Control

The Control field, shown in Figure 3-18, uses three different formats, depending on the type of HDLC frame used:

- **Information (I) frame:** I-frames carry upper layer information and some control information. This frame sends and receives sequence numbers, and the poll final (P/F) bit performs flow and error control. The send sequence number refers to the number of the frame to be sent next. The receive sequence number provides the number of the frame to be received next. Both sender and receiver maintain send and receive sequence numbers. A *primary station* uses the P/F bit to tell the secondary whether it requires an immediate response. A secondary station uses the P/F bit to tell the primary whether the current frame is the last in its current response.
- **Supervisory (S) frame:** S-frames provide control information. An S-frame can request and suspend transmission, report on status, and acknowledge receipt of I-frames. S-frames do not have an information field.
- **Unnumbered (U) frame:** U-frames support control purposes and are not sequenced. Depending on the function of the U-frame, its Control field is 1 or 2 bytes. Some U-frames have an Information field.

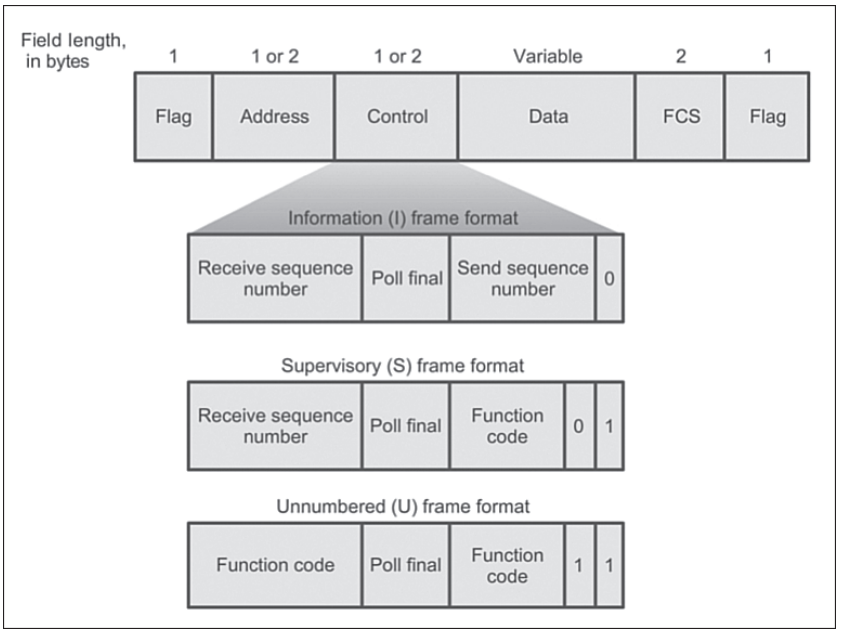


Figure 3-18 HDLC Frame Types

Protocol

Only used in Cisco HDLC. This field specifies the protocol type encapsulated within the frame (e.g., 0x0800 for IP).

Data

The Data field contains a path information unit (PIU) or exchange identification (XID) information.

Frame Check Sequence (FCS)

The FCS precedes the ending flag delimiter and is usually a cyclic redundancy check (CRC) calculation remainder. The CRC calculation is redone in the receiver. If the result differs from the value in the original frame, an error is assumed.

Configuring HDLC Encapsulation (3.1.2.4)

Cisco HDLC is the default encapsulation method used by Cisco devices on synchronous serial lines.

Use Cisco HDLC as a point-to-point protocol on leased lines between two Cisco devices. If connecting non-Cisco devices, use synchronous PPP.

If the default encapsulation method has been changed, use the **encapsulation hdlc** command in privileged EXEC mode to reenable HDLC.

There are two steps to re-enable HDLC encapsulation:

Step 1. Enter the interface configuration mode of the serial interface.

Step 2. Enter the **encapsulation hdlc** command to specify the encapsulation protocol on the interface.

The following shows an example of HDLC reenabled on a serial interface:

```
R2(config)# interface s0/0/0
R2(config-if)# encapsulation hdlc
```

Troubleshooting a Serial Interface (3.1.2.5)

The output of the **show interfaces serial** command displays information specific to serial interfaces. When HDLC is configured, **encapsulation HDLC** should be reflected in the output, as highlighted in Example 3-1. **Serial 0/0/0 is up, line protocol is up** indicates that the line is up and functioning; **encapsulation HDLC** indicates that the default serial encapsulation (HDLC) is enabled.

Example 3-1 Displaying Serial Interface Information

```
R1# show interface serial 0/0/0
Serial0/0/0 is up, line protocol is up
  Hardware is GT96K Serial
  Internet address is 172.16.0.1/30
  MTU 1500 bytes, BW 1544 Kbit/sec, DLY 20000 usec,
```

```
   reliability 255/255, txload 1/255, rxload 1/255
Encapsulation HDLC, loopback not set
Keepalive set (10 sec)
CRC checking enabled
<Output omitted for brevity>
```

The **show interfaces serial** command returns one of six possible states:

- Serial *x* is up, line protocol is up.
- Serial *x* is down, line protocol is down.
- Serial *x* is up, line protocol is down.
- Serial *x* is up, line protocol is up (looped).
- Serial *x* is up, line protocol is down (disabled).
- Serial *x* is administratively down, line protocol is down.

Of the six possible states, there are five problem states. Table 3-2 lists the five problem states, the issues associated with that state, and how to troubleshoot the issue.

Table 3-2 Troubleshooting a Serial Interface

Status Line Condition	Possible Problem	Solution
Serial *x* is up, line protocol is up.	This is proper status line condition.	No action is required.
Serial *x* is down, line protocol is down.	The router is not sensing a carrier detect (CD) signal (that is, the CD is not active). The line is down or is not connected on the far end. Cabling is faulty or incorrect. Hardware failure has occurred (CSU/DSU).	1. Check the CD LEDs on the CSU/DSU to see whether the CD is active, or insert a breakout box on the line to check for the CD signal. 2. Verify that the proper cable and interface are being used by looking at the hardware installation documentation. 3. Insert a breakout box and check all control leads. 4. Contact the leased line or other carrier service to see whether there is a problem. 5. Swap faulty parts. 6. If you suspect faulty router hardware, change the serial line to another port. If the connection comes up, the previously connected interface has a problem.

Status Line Condition	Possible Problem	Solution
Serial x is up, line protocol is down (DCE mode).	The **clock rate** interface configuration command is missing. The DTE device does not support or is not set up for SCTE mode (terminal timing). The remote CSU or DSU has failed.	1. Add the **clockrate** interface configuration command on the serial interface. Syntax: **clock rate** *bps* Syntax Description: ■ *bps*: Desired clock rate in bits per second: 1200, 2400, 4800, 9600, 19200, 38400, 56000, 64000, 72000, 125000, 148000, 250000, 500000, 800000, 1000000, 1300000, 2000000, 4000000, or 8000000. 2. If the problem appears to be on the remote end, repeat Step 1 on the remote modem, CSU or DSU. 3. Verify that the correct cable is being used. 4. If the line protocol is still down, there is a possible hardware failure or cabling problem. Insert a breakout box and observe leads. 5. Replace faulty parts as necessary.
Serial *x* is up, line protocol is up (looped).	A loop exists in the circuit. The sequence number in the keepalive packet changes to a random number when a loop is initially detected. If the same random number is returned over the link, a loop exists.	1. Use the **show running-config** privileged exec command to look for any **loopback** interface configuration command entries. 2. If you find a **loopback** interface configuration command entry, use the **no loopback** interface configuration command to remove the loop. 3. If you do not find the **loopback** interface configuration command, examine the CSU/DSU to determine whether they are configured in manual loopback mode. If they are, disable manual loopback. 4. Reset the CSU/DSU, and inspect the line status. If the line protocol comes up, no other action is needed. 5. If the CSU/DSU is not configured in manual loopback mode, contact the leased line or other carrier service for line troubleshooting assistance.

Status Line Condition	Possible Problem	Solution
Serial *x* is up, line protocol is down (disabled) .	A high error rate has occurred due to a remote device problem. A CSU or DSU hardware problem has occurred. Router hardware (interface) is bad.	1. Troubleshoot the line with a serial analyzer and breakout box. Look for toggling CTS and DSR signals. 2. Loop CSU/DSU (DTE loop). If the problem continues, it is likely that there is a hardware problem. If the problem does not continue, it is likely that there is a telephone company problem. 3. Swap out bad hardware, as required (CSU, DSU, switch, local or remote router).
Serial *x* is administratively down, line protocol is down.	The router configuration includes the `shutdown` interface configuration command. A duplicate IP address exists.	1. Check the router configuration for the **shutdown** command. 2. Use the **no shutdown** interface configuration command to remove the **shutdown** command. 3. Verify that there are no identical IP addresses using the **show running-config** privileged exec command or the **show interfaces** exec command. 4. If there are duplicate addresses, resolve the conflict by changing one of the IP addresses.

The **show controllers** command is another important diagnostic tool when troubleshooting serial lines, as shown in Example 3-2. The output indicates the state of the interface channels and whether a cable is attached to the interface. In example 3-2, interface serial 0/0/0 has a V.35 DCE cable attached. The command syntax varies depending on the platform. *Cisco 7000* series routers use a cBus controller card for connecting serial links. With these routers, use the **show controllers cbus** command.

If the electrical interface output displays as `UNKNOWN` instead of `V.35`, `EIA/TIA-449`, or some other electrical interface type, the likely problem is an improperly connected cable. A problem with the internal wiring of the card is also possible. If the electrical interface is unknown, the corresponding display for the **show interfaces serial** command shows that the interface and line protocol are down.

Example 3-2 Displaying Controller Hardware Information on a Serial Interface

```
R1# show controllers serial 0/0/0
Interface Serial0/0/0
Hardware is GT96K
DCE V.35, clock rate 64000
idb at 0x66855120, driver data structure at 0x6685C93C
wic_info 0x6685CF68
```

```
Physical Port 0, SCC Num 0
MPSC Registers:
<Output omitted for brevity>
```

Interactive Graphic

Activity 3.1.2.6: Troubleshooting a Serial Interface

Go to the course online to use the Syntax Checker to perform troubleshooting on a serial interface.

Packet Tracer Activity

Packet Tracer Activity 3.1.2.7: Troubleshooting Serial Interfaces

Background/Scenario

You have been asked to troubleshoot WAN connections for a local telephone company (Telco). The Telco router is supposed to communicate with four remote sites, but none of them are working. Use your knowledge of the OSI model and a few general rules to identify and repair the errors in the network.

PPP Operation (3.2)

This section discusses the PPP operations, including the benefits of PPP, LCP, and NCP protocols, and establishing a PPP session.

Benefits of PPP (3.2.1)

PPP has several advantages over its predecessor HDLC. In this section, PPP is introduced along with examining the benefits of PPP.

Introducing PPP (3.2.1.1)

Recall that HDLC is the default serial encapsulation method when connecting two Cisco routers. With an added protocol type field, the Cisco version of HDLC is proprietary. Thus, Cisco HDLC can only work with other Cisco devices. However, when there is a need to connect to a non-Cisco router, PPP encapsulation should be used, as shown in the Figure 3-19.

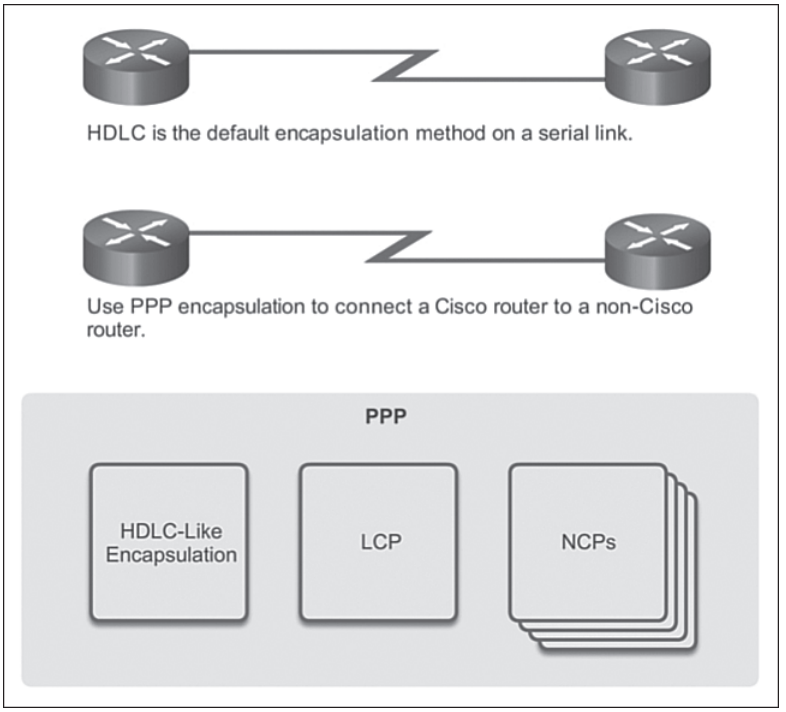


Figure 3-19 What is PPP?

PPP encapsulation has been carefully designed to retain compatibility with most commonly used supporting hardware. PPP encapsulates data frames for transmission over Layer 2 physical links. PPP establishes a direct connection using serial cables, phone lines, ***trunk lines***, cellular telephones, specialized radio links, or fiber-optic links.

PPP contains three main components:

- HDLC-like framing for transporting multiprotocol packets over point-to-point links.
- Extensible ***Link Control Protocol (LCP)*** for establishing, configuring, and testing the data-link connection.
- Family of ***Network Control Protocols (NCPs)*** for establishing and configuring different network layer protocols. PPP allows the simultaneous use of multiple network layer protocols. Some of the more common NCPs are Internet Protocol (IPv4) Control Protocol, IPv6 Control Protocol, AppleTalk Control Protocol, ***Novell IPX*** Control Protocol, Cisco Systems Control Protocol, ***SNA Control Protocol***, and Compression Control Protocol.

Advantages of PPP (3.2.1.2)

PPP originally emerged as an encapsulation protocol for transporting IPv4 traffic over point-to-point links. PPP provides a standard method for transporting multiprotocol packets over point-to-point links.

There are many advantages to using PPP including the fact that it is not proprietary. PPP includes many features not available in HDLC:

- The link quality management feature, as shown in Figure 3-20, monitors the quality of the link. If too many errors are detected, PPP takes the link down.
- PPP supports PAP and CHAP authentication. This feature is explained and practiced in a later section.

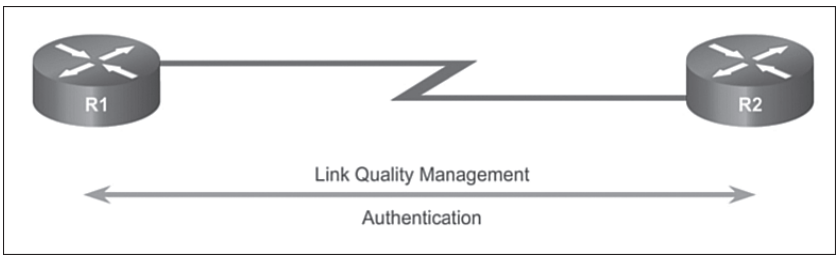


Figure 3-20 Advantages of PPP

LCP and NCP (3.2.2)

LCP and NCP are two key components to PPP. An understanding of these two protocols will help you understand and troubleshoot PPP operations.

PPP Layered Architecture (3.2.2.1)

A layered architecture is a logical model, design, or blueprint that aids in communication between interconnecting layers. Figure 3-21 maps the layered architecture of PPP against the Open System Interconnection (OSI) model. PPP and OSI share the same physical layer, but PPP distributes the functions of LCP and NCP differently.

At the physical layer, you can configure PPP on a range of interfaces, including

- Asynchronous serial, such as leased line services
- Synchronous serial, such as those that use basic telephone service for modem dialup connections
- HSSI
- ISDN

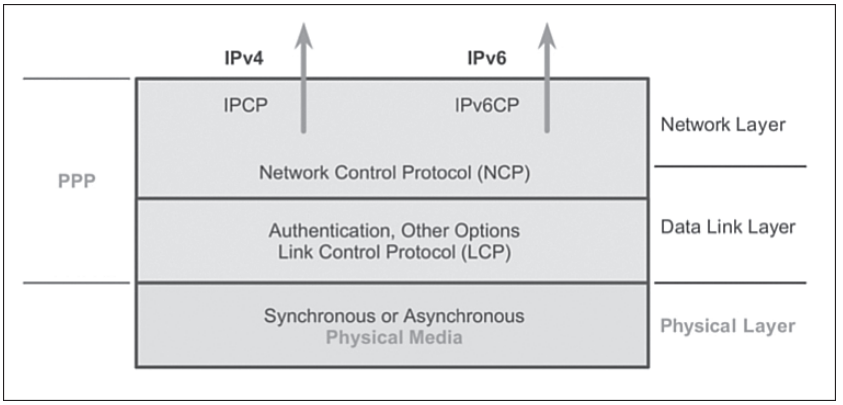


Figure 3-21 PPP Layered Architecture

PPP operates across any DTE/DCE interface (RS-232-C, RS-422, RS-423, or V.35). The only absolute requirement imposed by PPP is a full-duplex circuit, either dedicated or switched, that can operate in either an asynchronous or synchronous bit-serial mode, transparent to PPP link layer frames. PPP does not impose any restrictions regarding transmission rate other than those imposed by the particular DTE/DCE interface in use.

Most of the work done by PPP is at the data link and network layers by the LCP and NCPs. The LCP sets up the PPP connection and its parameters, the NCPs handle higher layer protocol configurations, and the LCP terminates the PPP connection.

PPP – Link Control Protocol (LCP) (3.2.2.2)

The LCP functions within the data link layer and has a role in establishing, configuring, and testing the data-link connection. The LCP establishes the point-to-point link. The LCP also negotiates and sets up control options on the WAN data link, which are handled by the NCPs.

The LCP provides automatic configuration of the interfaces at each end, including

- Handling varying limits on packet size
- Detecting common misconfiguration errors
- Terminating the link
- Determining when a link is functioning properly or when it is failing

After the link is established, PPP also uses the LCP to agree automatically on encapsulation formats such as authentication, compression, and error detection. Figure 3-21 shows the relationship of LCP to the physical layer and NCP.

PPP – Network Control Protocol (NCP) (3.2.2.3)

PPP permits multiple network layer protocols to operate on the same communications link. For every network layer protocol used, PPP uses a separate NCP, as shown in Figure 3-21. For example, IPv4 uses the IP Control Protocol (IPCP) and IPv6 uses IPv6 Control Protocol (IPv6CP).

NCPs include functional fields containing standardized codes to indicate the network layer protocol that PPP encapsulates. Table 3-3 lists the PPP protocol field numbers. Each NCP manages the specific needs required by its respective network layer protocols. The various NCP components encapsulate and negotiate options for multiple network layer protocols.

Table 3-3 Protocol Fields

Value (in hex)	Protocol Name
8021	Internet Protocol (IPv4) Control Protocol
8057	Internet Protocol Version 6 (IPv6) Control Protocol
8023	OSI Network Layer Control Protocol
8029	Appletalk Control Protocol
802b	Novell IPX Control Protocol
c021	Link Control Protocol
c023	Password Authentication Protocol
c223	Challenge Handshake Authentication Protocol

PPP Frame Structure (3.2.2.4)

A PPP frame consists of six fields. The following descriptions summarize the PPP frame fields illustrated in Figure 3-22:

- **Flag**: A single byte that indicates the beginning or end of a frame. The Flag field consists of the binary sequence 01111110. In successive PPP frames, only a single Flag character is used.
- **Address**: A single byte that contains the binary sequence 11111111, the standard broadcast address. PPP does not assign individual station addresses.
- **Control**: A single byte that contains the binary sequence 00000011, which calls for transmission of user data in an unsequenced frame. This provides a connectionless link service that does require the establishment of data links or links stations. On a point-to-point link, the destination node does not need to

be addressed. Therefore, for PPP, the Address field is set to 0xFF, the broadcast address. If both PPP peers agree to perform Address and Control field compression during the LCP negotiation, the Address field is not included.

- **Protocol:** Two bytes that identify the protocol encapsulated in the information field of the frame. The 2-byte Protocol field identifies the protocol of the PPP payload. If both PPP peers agree to perform Protocol field compression during LCP negotiation, the Protocol field is 1 byte for the protocol identification in the range 0x00-00 to 0x00-FF. The most up-to-date values of the Protocol field are specified in the most recent Assigned Numbers Request For Comments (RFC).
- **Data:** Zero or more bytes that contain the datagram for the protocol specified in the Protocol field. The end of the Information field is found by locating the closing flag sequence and allowing 2 bytes for the FCS field. The default maximum length of the Information field is 1500 bytes. By prior agreement, consenting PPP implementations can use other values for the maximum Information field length.
- **Frame Check Sequence (FCS):** Normally 16 bits (2 bytes). By prior agreement, consenting PPP implementations can use a 32-bit (4-byte) FCS for improved error detection. If the receiver's calculation of the FCS does not match the FCS in the PPP frame, the PPP frame is silently discarded.

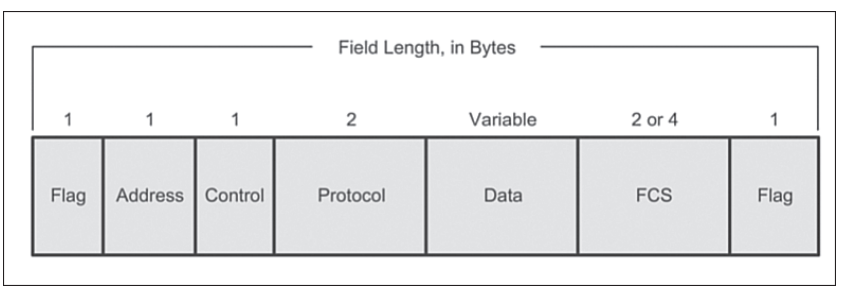


Figure 3-22 PPP Frame Fields

LCPs can negotiate modifications to the standard PPP frame structure. Modified frames, however, are always distinguishable from standard frames.

Interactive Graphic

Activity 3.2.2.5: Troubleshooting a Serial Interface

Go to the course online to perform this practice activity.

PPP Sessions (3.2.3)

Understanding PPP session establishment, LCP and NCP are important parts of implementing and troubleshooting PPP. These topics are discussed next.

Establishing a PPP Session (3.2.3.1)

There are three phases of establishing a PPP session, as shown in Figure 3-23:

- **Phase 1: Link establishment and configuration negotiation:** Before PPP exchanges any network layer datagrams, such as IP, the LCP must first open the connection and negotiate configuration options. This phase is complete when the receiving router sends a configuration-acknowledgment frame back to the router initiating the connection.
- **Phase 2: Link quality determination (optional):** The LCP tests the link to determine whether the link quality is sufficient to bring up network layer protocols. The LCP can delay transmission of network layer protocol information until this phase is complete.
- **Phase 3: Network layer protocol configuration negotiation:** After the LCP has finished the link quality determination phase, the appropriate NCP can separately configure the network layer protocols, and bring them up and take them down at any time. If the LCP closes the link, it informs the network layer protocols so that they can take appropriate action.

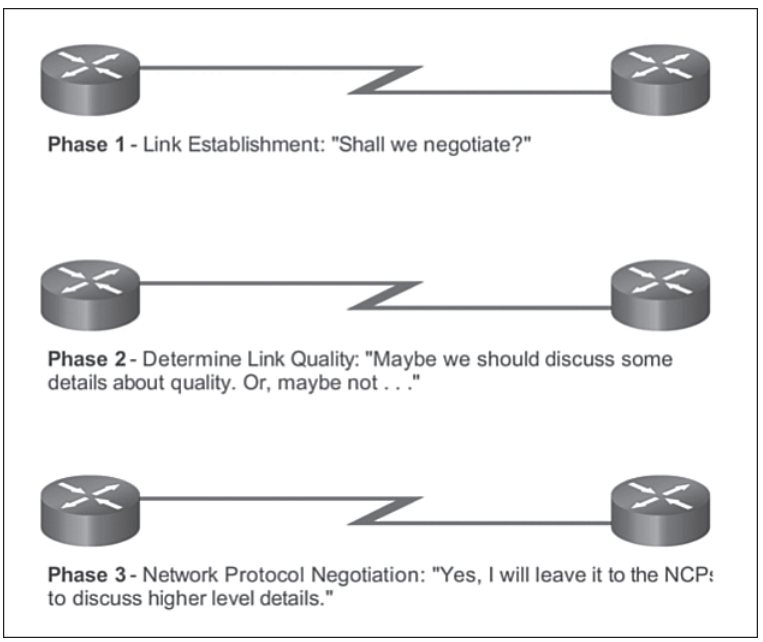


Figure 3-23 Establishing a PPP Session

The link remains configured for communications until explicit LCP or NCP frames close the link, or until some external event occurs such as an inactivity timer expiring, or an administrator intervening.

The LCP can terminate the link at any time. This is usually done when one of the routers requests termination, but can happen because of a physical event, such as the loss of a carrier or the expiration of an idle-period timer.

LCP Operation (3.2.3.2)

LCP operation includes provisions for link establishment, link maintenance, and link termination. LCP operation uses three classes of LCP frames to accomplish the work of each of the LCP phases:

- Link-establishment frames establish and configure a link (Configure-Request, Configure-Ack, Configure-Nak, and Configure-Reject).
- Link-maintenance frames manage and debug a link (Code-Reject, Protocol-Reject, Echo-Request, Echo-Reply, and Discard-Request).
- Link-termination frames terminate a link (Terminate-Request and Terminate-Ack).

Link Establishment

Link establishment is the first phase of LCP operation, as seen in Figure 3-24. This phase must complete successfully, before any network layer packets can be exchanged. During link establishment, the LCP opens the connection and negotiates the configuration parameters. The link establishment process starts with the initiating device sending a Configure-Request frame to the responder. The Configure-Request frame includes a variable number of configuration options needed to set up on the link.

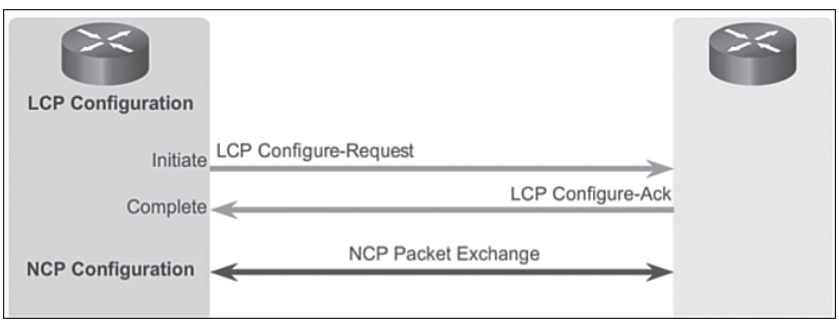


Figure 3-24 PPP Link Establishment

The initiator includes the options for how it wants the link created, including protocol or authentication parameters. The responder processes the request:

- If the options are not acceptable or not recognized, the responder sends a Configure-Nak or Configure-Reject message. If this occurs and the negotiation fails, the initiator must restart the process with new options.

- If the options are acceptable, the responder responds with a Configure-Ack message and the process moves on to the authentication stage. The operation of the link is handed over to the NCP.

When NCP has completed all necessary configurations, including validating authentication if configured, the line is available for data transfer. During the exchange of data, LCP transitions into link maintenance.

Link Maintenance

During link maintenance, LCP can use messages to provide feedback and test the link, as shown in Figure 3-25.

- **Echo-Request, Echo-Reply, and Discard-Request:** These frames can be used for testing the link.
- **Code-Reject and Protocol-Reject:** These frame types provide feedback when one device receives an invalid frame due to either an unrecognized LCP code (LCP frame type) or a bad protocol identifier. For example, if an uninterpretable packet is received from the peer, a Code-Reject packet is sent in response. The sending device will resend the packet.

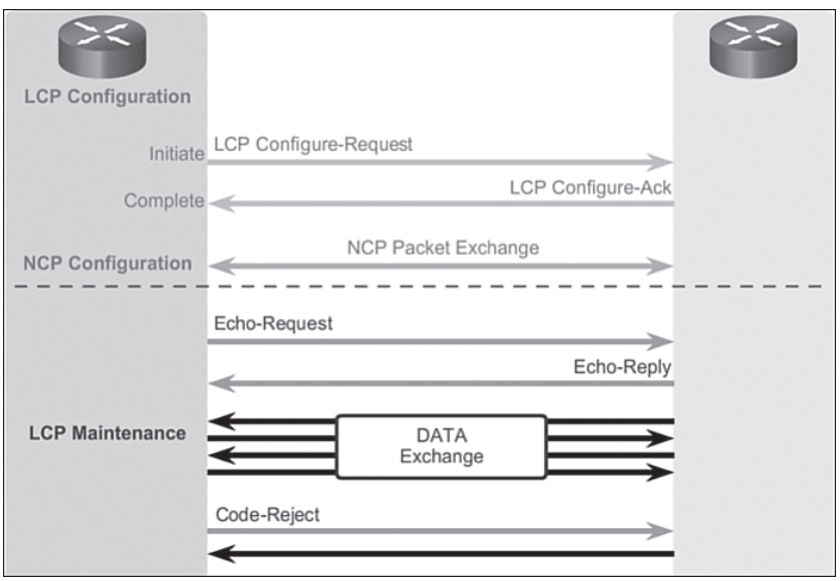


Figure 3-25 PPP Link Maintenance

Link Termination

After the transfer of data at the network layer completes, the LCP terminates the link, as shown in Figure 3-26. NCP only terminates the network layer and NCP link. The link remains open until the LCP terminates it. If the LCP terminates the link before NCP, the NCP session is also terminated.

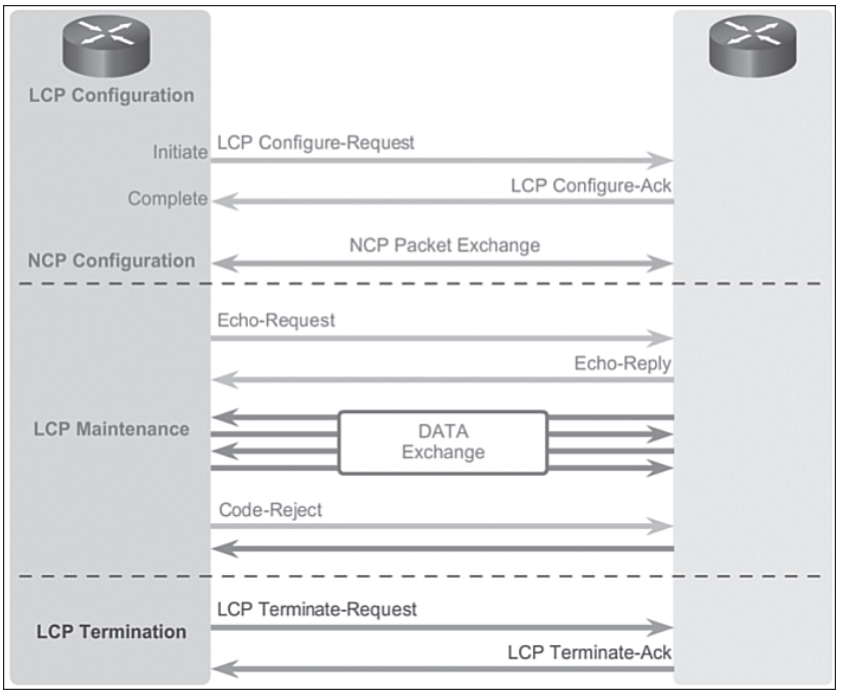


Figure 3-26 PPP Link Termination

PPP can terminate the link at any time. This might happen because of the loss of the carrier, authentication failure, link quality failure, the expiration of an idle-period timer, or the administrative closing of the link. The LCP closes the link by exchanging Terminate packets. The device initiating the shutdown sends a Terminate-Request message. The other device replies with a Terminate-Ack. A termination request indicates that the device sending it needs to close the link. When the link is closing, PPP informs the network layer protocols so that they may take appropriate action.

LCP Packet (3.2.3.3)

Figure 3-27 shows the fields in an LCP packet:

- **Code:** The Code field is 1 byte in length and identifies the type of LCP packet.
- **Identifier:** The Identifier field is 1 byte in length and is used to match packet requests and replies.

- **Length:** The Length field is 2 bytes in length and indicates the total length (including all fields) of the LCP packet.
- **Data:** The Data field is 0 or more bytes as indicated by the length field. The format of this field is determined by the code.

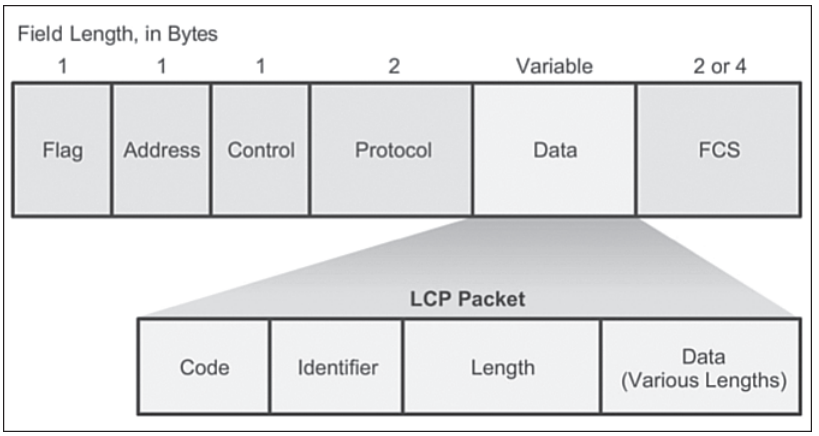


Figure 3-27 LCP Packet Codes

Each LCP packet is a single LCP message consisting of an LCP Code field identifying the type of LCP packet, an identifier field so that requests and replies can be matched, and a Length field indicating the size of the LCP packet and LCP packet type-specific data.

Each LCP packet has a specific function in the exchange of configuration information depending on its packet type. The Code field of the LCP packet identifies the packet type according to Table 3-4.

Table 3-4 LCP Packet Fields

LCP Code	LCP Packet Type	Description
1	Configure-Request	Sent to open or reset a PPP connection. Configure-Request contains a list of LCP options with changes to default option values.
2	Configure-Ack	Sent when all of the values of all of the LCP options in the last Configure-Request received are recognized and acceptable. When both PPP peers send and receive Configure-Acks, the LCP negotiation is complete.
3	Configure-Nak	Sent when all the LCP options are recognized, but the values of some options are not acceptable. Configure-Nak includes the mismatching options and their acceptable values.

LCP Code	LCP Packet Type	Description
4	Configure-Reject	Set when LCP options are not recognized or not acceptable for negotiation. Configure-Reject includes the unrecognized or non-negotiable options.
5	Terminate-Request	Optionally sent to close the PPP connection.
6	Terminate-Ack	Sent in response to a Terminate-Request.
7	Code-Reject	Sent when the LCP code is unknown. The Code-Request message includes the rejected LCP packet.
8	Protocol-Reject	Sent when the PPP frame contains an unknown Protocol ID. The Protocol-Reject message includes the rejected LCP packet. Protocol-Reject is typically sent by a PPP peer in response to PPP NCP for a LAN protocol not enabled on the PPP peer.
9	Echo-Request	Optionally sent to test PPP connection.
10	Echo-Reply	Sent in response to an Echo-Request. The PPP Echo-Request and Echo-Reply are not related to the ICMP Echo Request and Echo Reply messages.
11	Discard-Request	Optionally sent to exercise the link in the outbound direction.

PPP Configuration Options (3.2.3.4)

PPP can be configured to support various optional functions, as shown in Figure 3-28. These optional functions include

- Authentication using either PAP or CHAP
- Compression using either Stacker or Predictor
- Multilink that combines two or more channels to increase the WAN bandwidth

To negotiate the use of these PPP options, the LCP link-establishment frames contain option information in the data field of the LCP frame, as shown in Figure 3-29. If a configuration option is not included in an LCP frame, the default value for that configuration option is assumed.

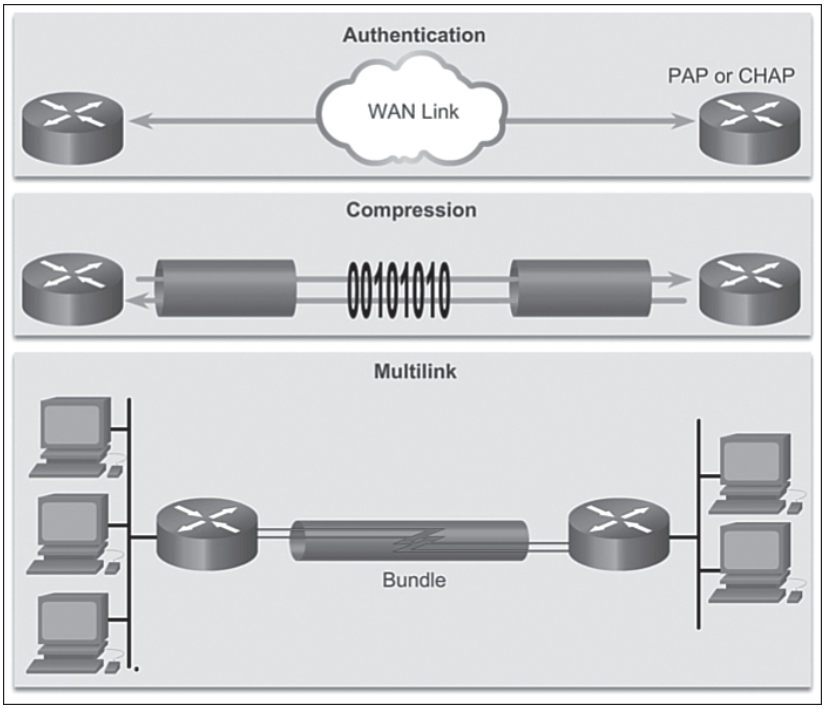


Figure 3-28 PPP Configuration Options

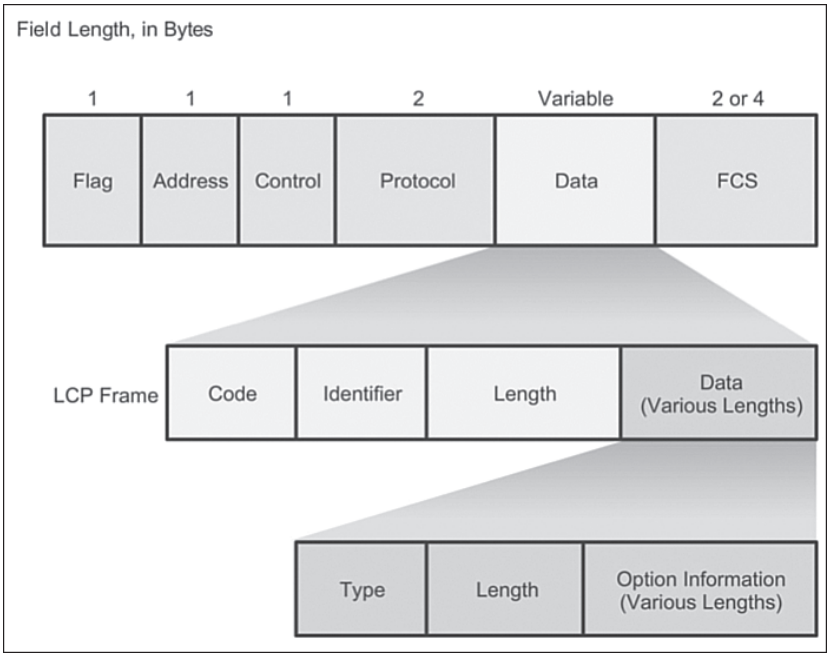


Figure 3-29 LCP Option Fields

This phase is complete when a configuration acknowledgment frame has been sent and received.

NCP Explained (3.2.3.5)

After the link has been initiated, the LCP passes control to the appropriate NCP.

NCP Process

Although initially designed for IP packets, PPP can carry data from multiple network layer protocols by using a modular approach in its implementation. PPP's modular model allows LCP to set up the link and then transfer the details of a network protocol to a specific NCP. Each network protocol has a corresponding NCP and each NCP has a corresponding RFC.

There are NCPs for IPv4, IPv6, IPX, AppleTalk, and many others. NCPs use the same packet format as the LCPs.

After the LCP has configured and authenticated the basic link, the appropriate NCP is invoked to complete the specific configuration of the network layer protocol being used. When the NCP has successfully configured the network layer protocol, the network protocol is in the open state on the established LCP link. At this point, PPP can carry the corresponding network layer protocol packets.

IPCP Example

As an example of how the NCP layer works, the NCP configuration of IPv4, which is the most common Layer 3 protocol, is shown in Figure 3-30. After LCP has established the link, the routers exchange IPCP messages, negotiating options specific to the IPv4 protocol. IPCP is responsible for configuring, enabling, and disabling the IPv4 modules on both ends of the link. IPv6CP is an NCP with the same responsibilities for IPv6.

IPCP negotiates two options:

- **Compression**: Allows devices to negotiate an algorithm to compress TCP and IP headers and save bandwidth. The Van Jacobson TCP/IP header compression reduces the size of the TCP/IP headers to as few as 3 bytes. This can be a significant improvement on slow serial lines, particularly for interactive traffic.
- **IPv4-Address**: Allows the initiating device to specify an IPv4 address to use for routing IP over the PPP link, or to request an IPv4 address for the responder. Prior to the advent of broadband technologies such as DSL and cable modem services, dialup network links commonly used the IPv4 address option.

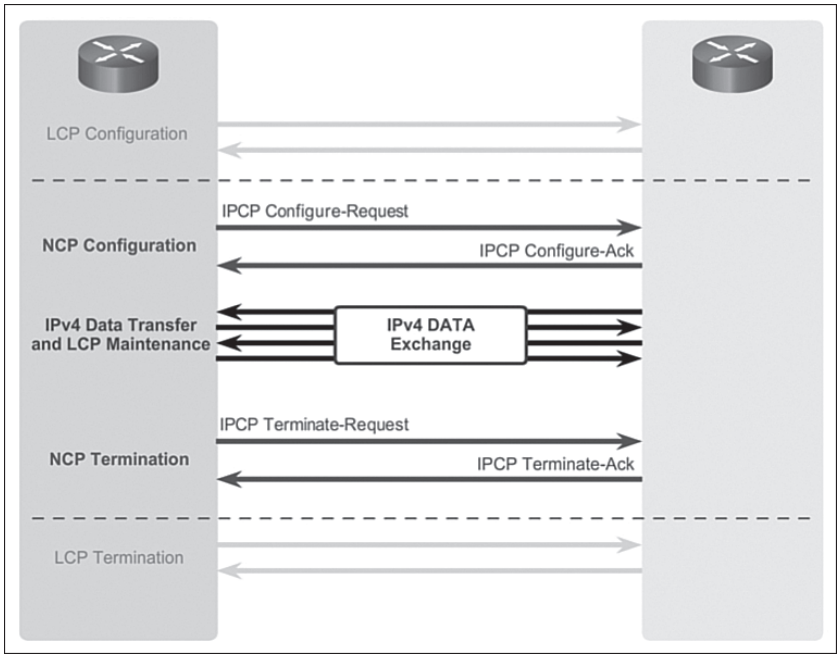


Figure 3-30 PPP NCP Operation

After the NCP process is complete, the link goes into the open state, and LCP takes over again in a link maintenance phase. Link traffic consists of any possible combination of LCP, NCP, and network layer protocol packets. When data transfer is complete, NCP terminates the protocol link; LCP terminates the PPP connection.

Interactive Graphic

Activity 3.2.3.6: Identify the Steps in the LCP Link Negotiation Process

Go to the course online to perform this practice activity.

Configure PPP (3.3)

This section describes the configuration of PPP. Basic PPP configuration is discussed along with optional PPP features and PPP authentication.

Configure PPP (3.3.1)

Basic PPP configuration commands are discussed next along with PPP compression, PPP quality link monitoring, and PPP Multilink.

PPP Configuration Options (3.3.1.1)

In the previous section, configurable LCP options were introduced to meet specific WAN connection requirements. PPP may include the following LCP options:

- **Authentication:** Peer routers exchange authentication messages. Two authentication choices are *Password Authentication Protocol (PAP)* and *Challenge Handshake Authentication Protocol (CHAP)*.
- **Compression:** Increases the effective throughput on PPP connections by reducing the amount of data in the frame that must travel across the link. The protocol decompresses the frame at its destination. Two compression protocols available in Cisco routers are Stacker and Predictor.
- **Error detection:** Identifies fault conditions. The Quality and Magic Number options help ensure a reliable, loop-free data link. The Magic Number field helps in detecting links that are in a looped-back condition. Until the Magic-Number Configuration Option has been successfully negotiated, the Magic-Number must be transmitted as zero. Magic numbers are generated randomly at each end of the connection.
- **PPP Callback:** PPP callback is used to enhance security. With this LCP option, a Cisco router can act as a callback client or a callback server. The client makes the initial call, requests that the server call it back, and terminates its initial call. The callback router answers the initial call and makes the return call to the client based on its configuration statements. The command is **ppp callback [accept | request]**.
- **Multilink:** This alternative provides load balancing over the router interfaces that PPP uses. Multilink PPP, also referred to as MP, MPPP, MLP, or Multilink, provides a method for spreading traffic across multiple physical WAN links while providing packet *fragmentation* and *reassembly*, proper sequencing, multivendor interoperability, and load balancing on inbound and outbound traffic.

When options are configured, a corresponding field value is inserted into the LCP option field, shown in Table 3-5.

Table 3-5 Configurable Options Field Codes

Option Name	Option Type	Option Length	Description
Authentication Protocol	3	5 or 6	This field indicates the authentication protocol, either PAP or CHAP.
Protocol Compression	7	2	A flag indicating that the PPP protocol ID be compressed to a single octet when the 2-byte protocol field is in the range of 0x00-00 to 0x00-FF.

Option Name	Option Type	Option Length	Description
Address and Control Field Compression	8	2	A flag indicating that the PPP Address field (always set to 0xFF) and the PPP Control field (always set to 0x03) be removed from the PPP header.
Magic Number (Error Detection)	5	6	This is a random number chosen to distinguish a peer and detect looped back lines.
Callback	13 or 0x0D	3	A 1-octet indicator of how callback is to be determined.

PPP Basic Configuration Command (3.3.1.2)

Basic PPP configuration is very straightforward. After PPP is configured on an interface the network administrator can then apply one or more PPP options.

Enabling PPP on an Interface

To set PPP as the encapsulation method used by a serial interface, use the **encapsulation ppp** interface configuration command.

The following example enables PPP encapsulation on interface serial 0/0/0:

```
R3# configure terminal
R3(config)# interface serial 0/0/0
R3(config-if)# encapsulation ppp
```

The **encapsulation ppp** interface command has no arguments. Remember that if PPP is not configured on a Cisco router, the default encapsulation for serial interfaces is HDLC.

Figure 3-31 and the listing that follows, shows that routers R1 and R2 have been configured with both an IPv4 and an IPv6 address on the serial interfaces. PPP is a Layer 2 encapsulation that supports various Layer 3 protocols including IPv4 and IPv6.

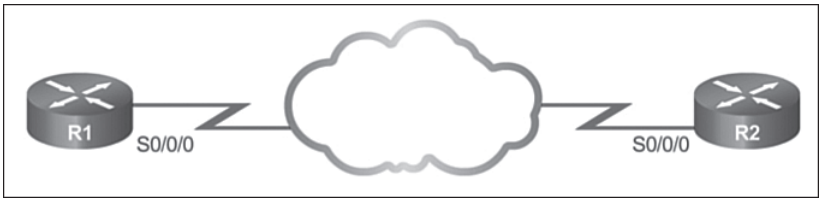


Figure 3-31 PPP Basic Configuration

Partial running-config for R1

```
hostname R1
!
interface Serial 0/0/0
 ip address 10.0.1.1 255.255.255.252
 ipv6 address 2001:db8:cafe:1::1/64
 encapsulation ppp
```

Parital running-config for R2

```
hostname R2
!
interface Serial 0/0/0
 ip address 10.0.1.2 255.255.255.252
 ipv6 address 2001:db8:cafe:1::2/64
 encapsulation ppp
```

PPP Compression Commands (3.3.1.3)

Point-to-point software compression on serial interfaces can be configured after PPP encapsulation is enabled. Because this option invokes a software compression process, it can affect system performance. If the traffic already consists of compressed files, such as .zip, .tar, or .mpeg, do not use this option. The command syntax for the **compress** command is

```
Router(config-if)# compress [ predictor | stac ]
```

- **predictor** (optional): Specifies that a predictor compression algorithm will be used
- **stac** (optional): Specifies that a Stacker (LZS) compression algorithm will be used

To configure compression over PPP, enter the following commands:

```
R2(config)# interface serial 0/0/0
R2(config-if)# encapsulation ppp
R2(config-if)# compress [ predictor | stac ]
```

The following example shows predictor compression used between R1 and R2:

Partial running-config for R1

```
hostname R1
!
interface Serial 0/0/0
 ip address 10.0.1.1 255.255.255.252
 ipv6 address 2001:db8:cafe:1::1/64
```

```
 encapsulation ppp
 compress predictor
```

Partial running-config for R2

```
hostname R2
!
interface Serial 0/0/0
 ip address 10.0.1.2 255.255.255.252
 ipv6 address 2001:db8:cafe:1::2/64
 encapsulation ppp
 compress predictor
```

PPP Link Quality Monitoring Command (3.3.1.4)

Recall that LCP provides an optional link quality determination phase. In this phase, LCP tests the link to determine whether the link quality is sufficient to use Layer 3 protocols. The **ppp quality** *percentage* command ensures that the link meets the quality requirement set; otherwise, the link closes down. The command syntax for the **ppp quality** command is

```
Router(config-if)# ppp quality percentage
```

- *percentage*: Specifies the link quality threshold. Range is 1 to 100.

The percentages are calculated for both incoming and outgoing directions. The outgoing quality is calculated by comparing the total number of packets and bytes sent to the total number of packets and bytes received by the destination node. The incoming quality is calculated by comparing the total number of packets and bytes received to the total number of packets and bytes sent by the destination node.

If the link quality percentage is not maintained, the link is deemed to be of poor quality and is taken down. Link Quality Monitoring (LQM) implements a time lag so that the link does not bounce up and down.

The following configuration example monitors the data dropped on the link and avoids frame looping:

```
R2(config)# interface serial 0/0/0
R2(config-if)# encapsulation ppp
R2(config-if)# ppp quality 80
```

Use the **no ppp quality** command to disable LQM. The following example shows link quality being used between R1 and R2:

Partial running-config for R1

```
hostname R1
!
interface Serial 0/0/0
 ip address 10.0.1.1 255.255.255.252
 ipv6 address 2001:db8:cafe:1::1/64
 encapsulation ppp
 ppp quality 80
```

Partial running-config for R2

```
hostname R2
!
interface Serial 0/0/0
 ip address 10.0.1.2 255.255.255.252
 ipv6 address 2001:db8:cafe:1::2/64
 encapsulation ppp
 ppp quality 80
```

Interactive Graphic

Activity 3.3.1.4: PPP Link Quality Monitoring Command

Go to the course online to use the checker to LQM on R1's serial 0/0/1 interface.

PPP Multilink Commands (3.3.1.5)

Multilink PPP (also referred to as MP, MPPP, MLP, or Multilink) provides a method for spreading traffic across multiple physical WAN links, as shown in Figure 3-32. Multilink PPP also provides packet fragmentation and reassembly, proper sequencing, multivendor interoperability, and load balancing on inbound and outbound traffic.

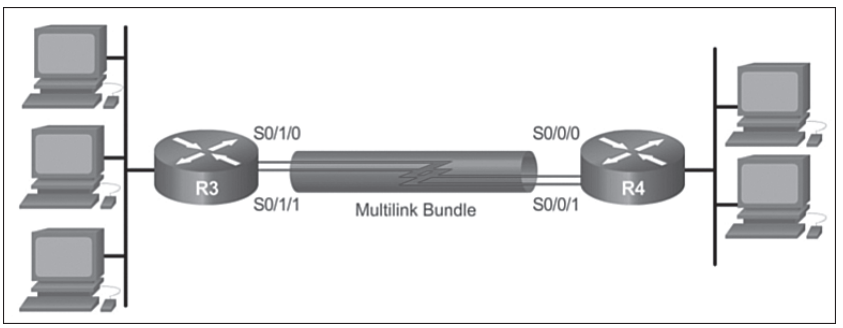


Figure 3-32 PPP Multilink

MPPP allows packets to be fragmented and sends these fragments simultaneously over multiple point-to-point links to the same remote address. The multiple physical links come up in response to a user-defined load threshold. MPPP can measure the

load on just inbound traffic, or on just outbound traffic, but not on the combined load of both inbound and outbound traffic.

Configuring MPPP requires two steps.

Step 1. Create a multilink bundle.

The **interface multilink** *number* command creates the multilink interface.

In interface configuration mode, an IP address is assigned to the multilink interface. In this example, both IPv4 and IPv6 addresses are configured on routers R3 and R4.

The interface is enabled for multilink PPP.

The interface is assigned a multilink group number.

Step 2. Assign interfaces to the multilink bundle. Each interface that is part of the multilink group:

Is enabled for PPP encapsulation.

Is enabled for multilink PPP.

Is bound to the multilink bundle using the multilink group number configured in Step 1.

The following example shows multilink PPP configured between R3 and R4:

Partial running-config for R3

```
hostname R3
!
interface Multilink 1
 ip address 10.0.1.1 255.255.255.252
 ipv6 address 2001:db8:cafe:1::1/64
 ppp multilink
 ppp multilink group 1
!
interface Serial 0/1/0
 no ip address
 encapsulation ppp
 ppp multilink
 ppp multilink group 1
!
interface Serial 0/1/1
 no ip address
 encapsulation ppp
 ppp multilink
 ppp multilink group 1
```

Partial running-config for R4

```
hostname R4
!
interface Multilink 1
 ip address 10.0.1.2 255.255.255.252
 ipv6 address 2001:db8:cafe:1::2/64
 ppp multilink
 ppp multilink group 1
!
interface Serial 0/0/0
 no ip address
 encapsulation ppp
 ppp multilink
 ppp multilink group 1
!
interface Serial 0/0/1
 no ip address
 encapsulation ppp
 ppp multilink
 ppp multilink group 1
```

To disable PPP multilink, use the **no ppp multilink** command.

Verifying PPP Configuration (3.3.1.6)

Use the **show interfaces serial** command to verify proper configuration of HDLC or PPP encapsulation. Example 3-3 shows a PPP configuration.

Example 3-3 Using **show interfaces serial** to Verify a PPP Encapsulation

```
R2#show interfaces serial 0/0/0
Serial0/0/0 is up, line protocol is up
  Hardware is GT96K Serial
  Internet address is 10.0.1.2/30
  MTU 1500 bytes, BW 1544 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, LCP Open
  Open: IPCP, IPV6CP, CCP, CDPCP,, loopback not set
  Keepalive set (10 sec)
  CRC checking enabled
<Output omitted for brevity>
```

When you configure HDLC, the output of the **show interfaces serial** command should display **encapsulation HDLC**. When PPP is configured, the LCP and NCP states also display. Notice that NCPs IPCP and IPv6CP are open for IPv4 and IPv6 because R1 and R2 were configured with both IPv4 and IPv6 addresses.

Table 3-6 summarizes commands used when verifying PPP.

Table 3-6 Verifying PPP Commands

Command	Description
show interfaces	Displays statistics for all interfaces configured on the router.
show interfaces serial	Displays information about a serial interface.
show ppp multilink	Displays information about a PPP multilink interface.

The **show ppp multilink** command verifies that PPP multilink is enabled on R3, as shown in Example 3-4. The output indicates the interface Multilink 1, the hostnames of both the local and remote endpoints, and the serial interfaces assigned to the multilink bundle.

Example 3-4 Verifying PPP Multilink

```
R3# show ppp multilink
Multilink1
   Bundle name: R4
   Remote Endpoint Discriminator: [1] R4
   Local Endpoint Discriminator: [1] R3
   Bundle up for 00:01:20, total bandwidth 3088, load 1/255
   Receive buffer limit 24000 bytes, frag timeout 1000 ms
     0/0 fragments/bytes in reassembly list
     0 lost fragments, 0 reordered
     0/0 discarded fragments/bytes, 0 lost received
     0x2 received sequence, 0x2 sent sequence
   Member links: 2 active, 0 inactive (max 255, min not set)
     Se0/1/1, since 00:01:20
     Se0/1/0, since 00:01:06
No inactive multilink interfaces
R3#
```

PPP Authentication (3.3.2)

PPP authentication protocols and configuration of PPP authentication is discussed in this section.

PPP Authentication Protocols (3.3.2.1)

PPP defines an extensible LCP that allows negotiation of an authentication protocol for authenticating its peer before allowing network layer protocols to transmit over the link. RFC 1334 defines two protocols for authentication, PAP and CHAP, as shown in Figure 3-33.

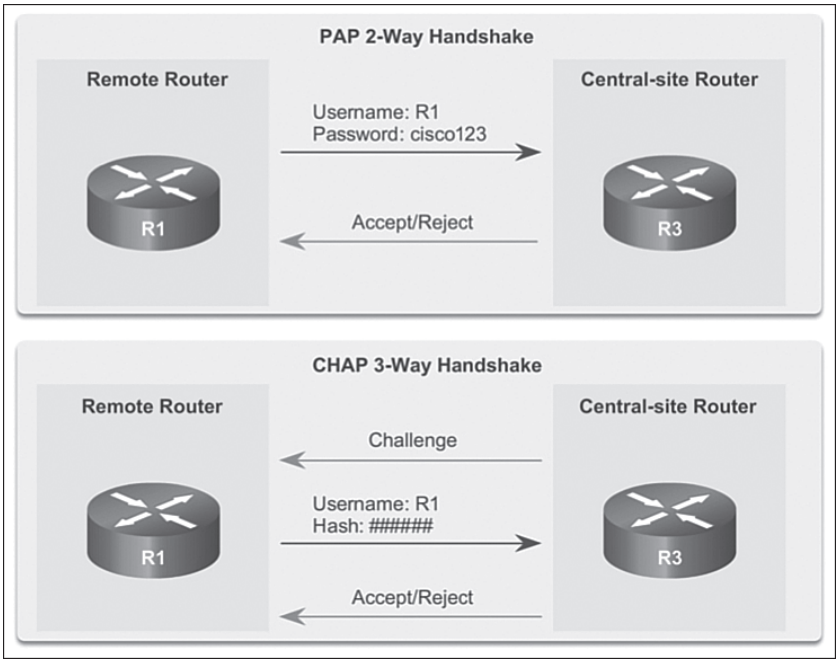


Figure 3-33 PPP Authentication Protocols

PAP is a very basic two-way process. There is no encryption. The username and password are sent in plaintext. If it is accepted, the connection is allowed. CHAP is more secure than PAP. It involves a three-way exchange of a shared secret.

The authentication phase of a PPP session is optional. If used, the peer is authenticated after LCP establishes the link and chooses the authentication protocol. If it is used, authentication takes place before the network layer protocol configuration phase begins.

The authentication options require that the calling side of the link enter authentication information. This helps to ensure that the user has the permission of the network administrator to make the call. Peer routers exchange authentication messages.

Password Authentication Protocol (PAP) (3.3.2.2)

One of the many features of PPP is that it performs Layer 2 authentication in addition to other layers of authentication, encryption, access control, and general security procedures.

Initiating PAP

PAP provides a simple method for a remote node to establish its identity using a two-way handshake. PAP is not interactive. When the **ppp authentication pap** command is used, the username and password are sent as one LCP data package, rather than the server sending a login prompt and waiting for a response, as shown in Figure 3-34. After PPP completes the link establishment phase, the remote node repeatedly sends a username-password pair across the link until the receiving node acknowledges it or terminates the connection.

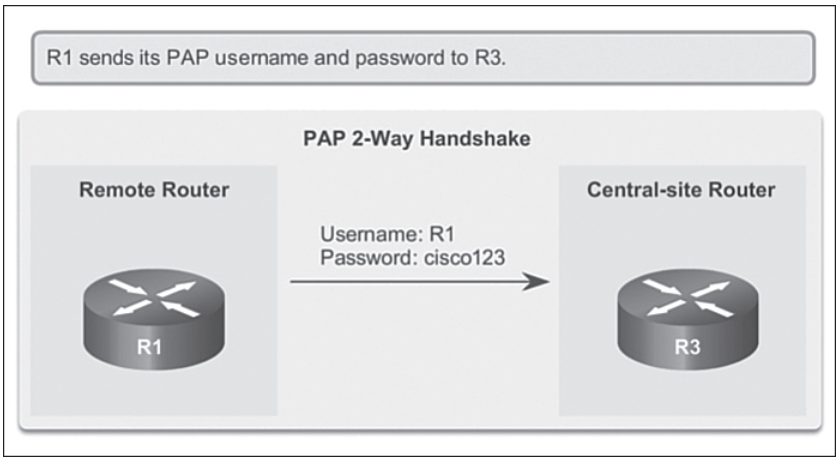


Figure 3-34 Initiating PAP

Completing PAP

At the receiving node, the username-password is checked by an authentication server that either allows or denies the connection. An accept or reject message is returned to the requester, as shown in Figure 3-35.

PAP is not a strong authentication protocol. Using PAP, passwords are sent across the link in plaintext, and there is no protection from playback or repeated trial-and-error attacks. The remote node is in control of the frequency and timing of the login attempts.

Nonetheless, there are times when using PAP can be justified. For example, despite its shortcomings, PAP may be used in the following environments:

- A large installed base of client applications that do not support CHAP
- Incompatibilities between different vendor implementations of CHAP
- Situations where a plaintext password must be available to simulate a login at the remote host

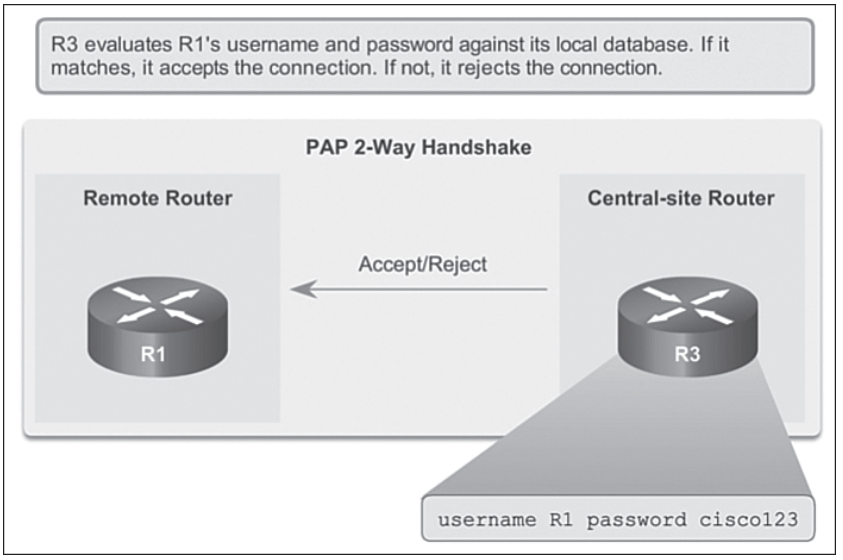


Figure 3-35 Completing PAP

Challenge Handshake Authentication Protocol (CHAP) (3.3.2.3)

After authentication is established with PAP, it does not reauthenticate. This leaves the network vulnerable to attack. Unlike PAP, which only authenticates once, CHAP conducts periodic challenges to make sure that the remote node still has a valid password value. The password value is variable and changes unpredictably while the link exists.

After the PPP link establishment phase is complete, the local router sends a challenge message to the remote node, as shown in Figure 3-36.

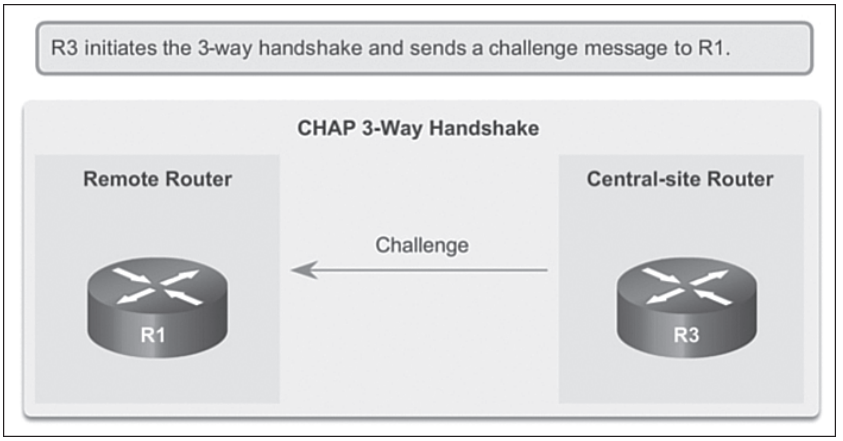


Figure 3-36 Initiating CHAP

The remote node responds with a value calculated using a one-way hash function, which is typically ***message digest 5 (MD5)*** based on the password and challenge message, as shown in Figure 3-37.

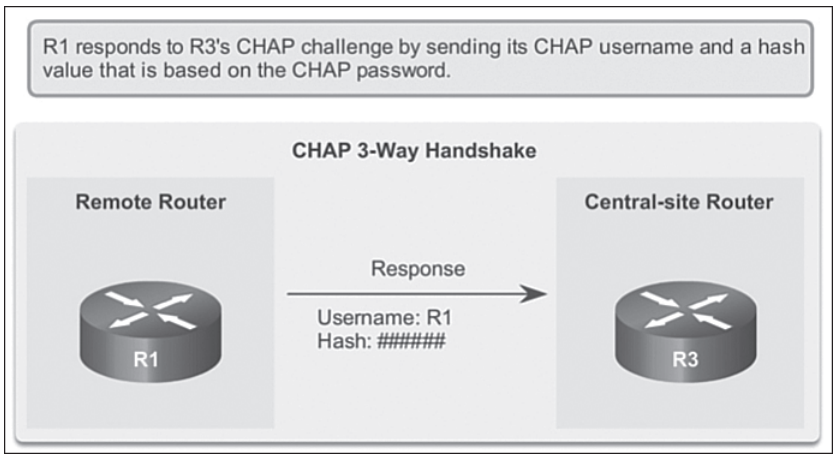


Figure 3-37 Responding CHAP

The local router checks the response against its own calculation of the expected hash value. If the values match, the initiating node acknowledges the authentication, as shown in Figure 3-38. If the value does not match, the initiating node immediately terminates the connection.

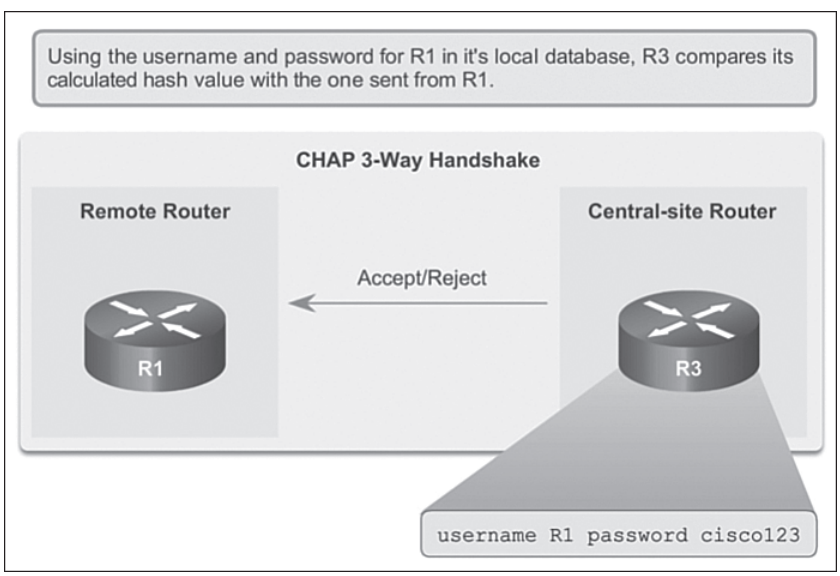


Figure 3-38 Completing CHAP

CHAP provides protection against playback attack by using a variable challenge value that is unique and unpredictable. Because the challenge is unique and random, the resulting hash value is also unique and random. The use of repeated challenges limits the time of exposure to any single attack. The local router or a third-party authentication server is in control of the frequency and timing of the challenges.

PPP Encapsulation and Authentication Process (3.3.2.4)

The flowchart in Figure 3-39 can be used to help understand the PPP authentication process when configuring PPP. The flowchart provides a visual example of the logic decisions made by PPP.

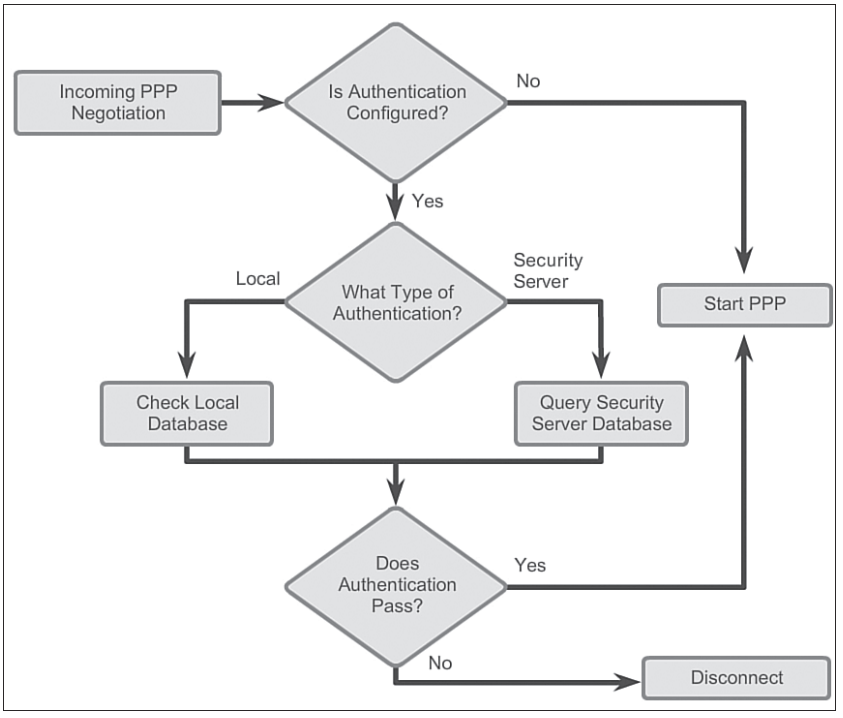


Figure 3-39 PPP Encapsulation and Authentication Process

For example, if an incoming PPP request requires no authentication, then PPP progresses to the next level. If an incoming PPP request requires authentication, then it can be authenticated using either the local database or a security server. As illustrated in the flowchart, successful authentication progresses to the next level, while an authentication failure disconnects and drops the incoming PPP request.

Follow the steps to view R1 establishing an authenticated PPP CHAP connection with R2.

Step 1. As shown in Figure 3-40, R1 initially negotiates the link connection using LCP with router R2 and the two systems agree to use CHAP authentication during the PPP LCP negotiation.

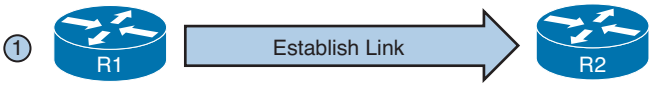


Figure 3-40 Establishing a Link

Step 2. As shown in Figure 3-41, R2 generates an ID and a random number, and sends that and its username as a CHAP challenge packet to R1.

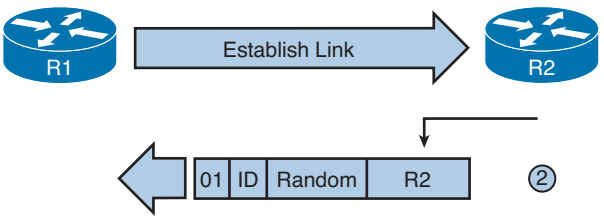


Figure 3-41 Sending a CHAP Challenge to R1

Step 3. As shown in Figure 3-42, R1 uses the username of the challenger (R2) and cross references it with its local database to find its associated password. R1 then generates a unique MD5 hash number using the R2's username, ID, random number and the shared secret password. In this example, the shared secret password is boardwalk.

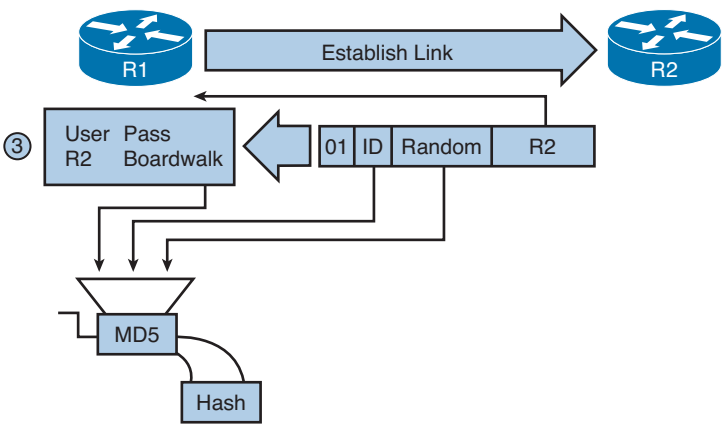


Figure 3-42 R1 Validates R2

Step 4. As shown in Figure 3-43, Router R1 then sends the challenge ID, the hashed value, and its username (R1) to R2.

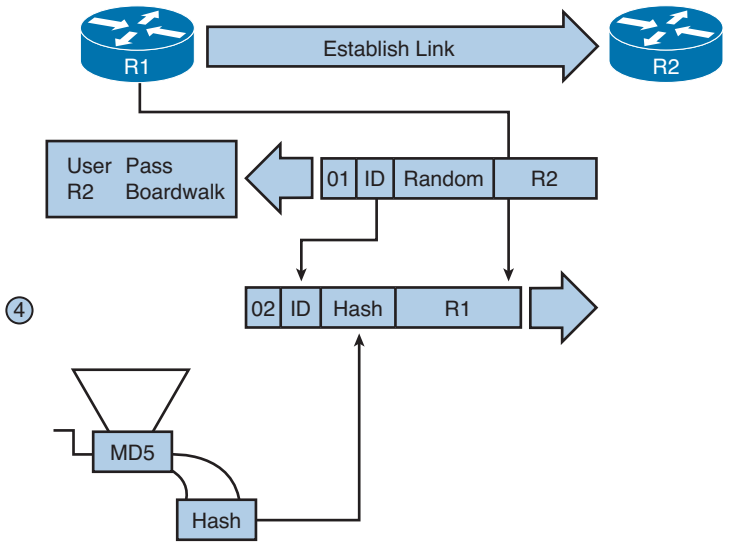


Figure 3-43 R1 Sends the Challenge to R2

Step 5. As shown in Figure 3-44, R2 generates its own hash value using the ID, the shared secret password, and the random number it originally sent to R1.

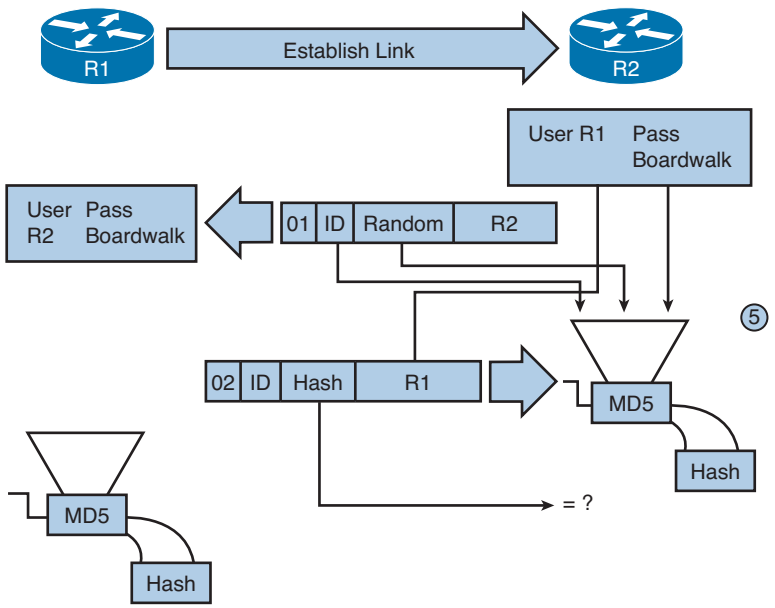


Figure 3-44 R2 Validates R1

Step 6. As shown in Figure 3-45, R2 compares its hash value with the hash value sent by R1. If the values are the same, R2 sends a link established response to R1.

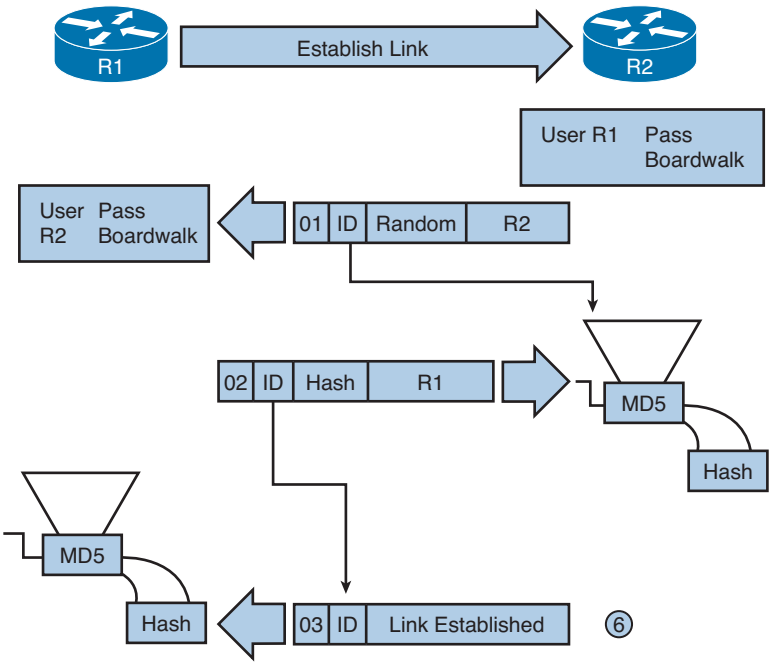


Figure 3-45 R2 Establishes the Link

If the authentication failed, a CHAP failure packet is built from the following components:

- 04 = CHAP failure message type
- id = copied from the response packet
- "Authentication failure" or some similar text message, which is meant to be a user-readable explanation

The shared secret password must be identical on R1 and R2.

Configuring PPP Authentication (3.3.2.5)

To specify the order in which the CHAP or PAP protocols are requested on the interface, use the **ppp authentication** interface configuration command:

```
Router(config-if)# ppp authentication {chap | chap pap | pap chap | pap} [if needed]
   [list-name | default] [callin]
```

- Use the **no** form of the command to disable this authentication.

Table 3-7 explains the syntax for the **ppp authentication** interface configuration command.

Table 3-7 PPP Command Syntax

`Chap`	Enables CHAP on serial interface.
`Pap`	Enables PAP on serial interface.
`chap pap`	Enables both CHAP and PAP on serial interface, and performs CHAP authentication before PAP.
`pap chap`	Enables both CHAP and PAP on serial interface, and performs PAP authentication before CHAP.
if-needed (Optional)	Used with TACACS and XTACACS. Do not perform CHAP or PAP authentication if the user has already provided authentication. This option is available only on asynchronous interfaces.
list-name (Optional)	Used with AAA/TACACS+. Specifies the name of a list of TACACS+ methods of authentication to use. If no list name is specified, the system uses the default. Lists are created with the **aaa authentication ppp** command.
default (Optional)	Used with AAA/TACACS+. Created with the **aaa authentication ppp** command.
Callin	Specifies authentication on incoming (received) calls only.

After you have enabled CHAP or PAP authentication, or both, the local router requires the remote device to prove its identity before allowing data traffic to flow. This is done as follows:

- PAP authentication requires the remote device to send a name and password to be checked against a matching entry in the local username database or in the remote *TACACS/TACACS+* database.
- CHAP authentication sends a challenge to the remote device. The remote device must encrypt the challenge value with a shared secret and return the encrypted value and its name to the local router in a response message. The local router uses the name of the remote device to look up the appropriate secret in the local username or remote TACACS/TACACS+ database. It uses the looked-up secret to encrypt the original challenge and verify that the encrypted values match.

Note

Authentication, authorization, and accounting (AAA)/TACACS is a dedicated server used to authenticate users. TACACS clients send a query to a TACACS authentication server. The server can authenticate the user, authorize what the user can do, and track what the user has done.

Either PAP or CHAP or both can be enabled. If both methods are enabled, the first method specified is requested during link negotiation. If the peer suggests using the

second method or simply refuses the first method, the second method should be tried. Some remote devices support CHAP only and some PAP only. The order in which you specify the methods is based on your concerns about the ability of the remote device to correctly negotiate the appropriate method as well as your concern about data line security. PAP usernames and passwords are sent as plaintext strings and can be intercepted and reused. CHAP has eliminated most of the known security holes.

Configuring PPP with Authentication (3.3.2.6)

The procedure outlined in the table describes how to configure PPP encapsulation and PAP/CHAP authentication protocols. Correct configuration is essential, because PAP and CHAP use these parameters to authenticate.

Configuring PAP Authentication

Figure 3-46 shows the topology used in an example of a two-way PAP authentication configuration, with the configuration in the following listing. Both routers authenticate and are authenticated, so the PAP authentication commands mirror each other. The PAP username and password that each router sends must match those specified with the **username** *name* **password** *password* command of the other router.

Figure 3-46 Topology for PPP

Partial running-config for R1

```
hostname R1
username R2 password sameone
!
interface Serial0/0/0
 ip address 10.0.1.1 255.255.255.252
 ipv6 address 2001:DB8:CAFE:1::1/64
 encapsulation ppp
 ppp authentication pap
 ppp pap sent-username R2 password sameone
```

Partial running-config for R2

```
hostname R2
username R1 password 0 sameone
!
interface Serial 0/0/0
```

```
 ip address 10.0.1.2 255.255.255.252
 ipv6 address 2001:db8:cafe:1::2/64
 encapsulation ppp
 ppp authentication pap
 ppp pap sent-username R2 password sameone
```

PAP provides a simple method for a remote node to establish its identity using a two-way handshake. This is done only on initial link establishment. The hostname on one router must match the username the other router has configured for PPP. The passwords must also match. Specify the username and password parameters, use the following command: **ppp pap sent-username** *name* **password** *password*.

Interactive Graphic

Activity 3.3.2.6: PPP PAP Authentication

Go to the course online to use the Syntax Checker to configure PAP authentication on router R1's serial 0/0/1 interface.

Configuring CHAP Authentication

CHAP periodically verifies the identity of the remote node using a three-way handshake. The hostname on one router must match the username the other router has configured. The passwords must also match. This occurs on initial link establishment and can be repeated any time after the link has been established. The following is an example of a CHAP configuration.

Partial running-config for R1

```
hostname R1
username R2 password sameone
!
interface Serial0/0/0
 ip address 10.0.1.1 255.255.255.252
 ipv6 address 2001:DB8:CAFE:1::1/64
 encapsulation ppp
 ppp authentication chap
```

Partial running-config for R2

```
hostname R2
username R1 password 0 sameone
!
interface Serial 0/0/0
 ip address 10.0.1.2 255.255.255.252
 ipv6 address 2001:db8:cafe:1::2/64
 encapsulation ppp
 ppp authentication chap
```

Interactive Graphic

Activity 3.3.2.6: PPP CHAP Authentication

Go to the course online to use the Syntax Checker to configure CHAP authentication on router R1's serial 0/0/1 interface.

Packet Tracer Activity

Packet Tracer Activity 3.3.2.7: Configuring PAP and CHAP Authentication

Background/Scenario

In this activity, you will practice configuring PPP encapsulation on serial links. You will also configure PPP PAP authentication and PPP CHAP authentication.

Lab 3.3.2.8: Configuring Basic PPP with Authentication

In this lab, you will complete the following objectives:

- Part 1: Configure Basic Device Settings
- Part 2: Configure PPP Encapsulation
- Part 3: Configure PPP CHAP Authentication

Troubleshoot WAN Connectivity (3.4)

Troubleshooting is an important component to understanding and implementing any technology. This section discusses troubleshooting WAN connectivity, specifically point-to-point serial communications using PPP.

Troubleshoot PPP (3.4.1)

Similar to other protocols implemented on a router, troubleshooting PPP involves a combination of **debug** and **show** commands. This section discusses how to use these commands to troubleshoot PPP negotiation and authentication.

Troubleshooting PPP Serial Encapsulation (3.4.1.1)

Recall that the **debug** command is used for troubleshooting and is accessed from privileged EXEC mode of the command-line interface. A **debug** output displays information about various router operations, related traffic generated or received by the router, and any error messages. It can consume a significant amount of resources, and the router is forced to process-switch the packets being debugged. The **debug** command must not be used as a monitoring tool; rather, it is meant to be used for a short period of time for troubleshooting.

Use the **debug ppp** command to display information about the operation of PPP.

```
Router# debug ppp {packet | negotiation | error | authentication | compression |
   cbcp}
```

Table 3-8 shows the command syntax. Use the `no` form of this command to disable debugging output.

Table 3-8 `debug ppp` Command Parameters

Parameter	Usage
packet	Displays PPP packets being sent and received. (This command displays low-level packet dumps.)
negotiation	Displays PPP packets transmitted during PPP startup, where PPP options are negotiated.
error	Displays protocol errors and error statistics associated with PPP connection negotiation and operation.
authentication	Displays authentication protocol messages, including Challenge Authentication Protocol (CHAP) packet exchanges and Password Authentication Protocol (PAP) exchanges.
compression	Displays information specific to the exchange of PPP connections using MPPC. This command is useful for obtaining incorrect packet sequence number information where MPPC compression is enabled.
cbcp	Displays protocol errors and statistics associated with PPP connection negotiations using MSCB.

Use the **debug ppp** command when trying to search the following:

- NCPs that are supported on either end of a PPP connection
- Any loops that might exist in a PPP internetwork
- Nodes that are (or are not) properly negotiating PPP connections
- Errors that have occurred over the PPP connection
- Causes for CHAP session failures
- Causes for PAP session failures
- Information specific to the exchange of PPP connections using the Callback Control Protocol (CBCP), used by Microsoft clients
- Incorrect packet sequence number information where MPPC compression is enabled

Debug PPP (3.4.1.2)

In addition to the **debug ppp** command, there are other commands that are available for troubleshooting a PPP connection.

A good command to use when troubleshooting serial interface encapsulation is the **debug ppp packet** command, as shown in Example 3-5. The example depicts packet exchanges under normal PPP operation, including LCP state, LQM procedures, and the LCP magic number.

Example 3-5 Output of **debug ppp packet** Command

```
R1# debug ppp packet
PPP packet display debugging is on
R1#
*Apr  1 16:15:17.471: Se0/0/0 LQM: O state Open magic 0x1EFC37C3 len 48
*Apr  1 16:15:17.471: Se0/0/0 LQM:    LastOutLQRs 70 LastOutPackets/Octets 194/9735
*Apr  1 16:15:17.471: Se0/0/0 LQM:    PeerInLQRs 70 PeerInPackets/Discards/Errors/
  Octets 0/0/0/0
*Apr  1 16:15:17.471: Se0/0/0 LQM:    PeerOutLQRs 71 PeerOutPackets/Octets 197/9839
*Apr  1 16:15:17.487: Se0/0/0 PPP: I pkt type 0xC025, datagramsize 52 link[ppp]
*Apr  1 16:15:17.487: Se0/0/0 LQM: I state Open magic 0xFE83D624 len 48
*Apr  1 16:15:17.487: Se0/0/0 LQM:    LastOutLQRs 71 LastOutPackets/Octets 197/9839
*Apr  1 16:15:17.487: Se0/0/0 LQM:    PeerInLQRs 71 PeerInPackets/Discards/Errors/
  Octets 0/0/0/0
*Apr  1 16:15:17.487: Se0/0/0 LQM:    PeerOutLQRs 71 PeerOutPackets/Octets 196/9809
*Apr  1 16:15:17.535: Se0/0/0 LCP: O ECHOREQ [Open] id 36 len 12 magic 0x1EFC37C3
*Apr  1 16:15:17.539: Se0/0/0 LCP-FS: I ECHOREP [Open] id 36 len 12 magic 0xFE83D624
*Apr  1 16:15:17.539: Se0/0/0 LCP-FS: Received id 36, sent id 36, line up
R1# undebug all
```

Example 3-6 displays the **debug ppp negotiation** command in a normal negotiation, where both sides agree on NCP parameters. In this case, protocol types IPv4 and IPv6 are proposed and acknowledged. The **debug ppp negotiation** command enables the network administrator to view the PPP negotiation transactions, identify the problem or stage when the error occurs, and develop a resolution. The output includes the LCP negotiation, authentication, and NCP negotiation.

Example 3-6 Output of **debug ppp negotiation** Command

```
R1# debug ppp negotiation
PPP protocol negotiation debugging is on
R1#
*Apr  1 18:42:29.831: %LINK-3-UPDOWN: Interface Serial0/0/0, changed state to up
*Apr  1 18:42:29.831: Se0/0/0 PPP: Sending cstate UP notification
*Apr  1 18:42:29.831: Se0/0/0 PPP: Processing CstateUp message
```

```
*Apr  1 18:42:29.835: PPP: Alloc Context [66A27824]
*Apr  1 18:42:29.835: ppp2 PPP: Phase is ESTABLISHING
*Apr  1 18:42:29.835: Se0/0/0 PPP: Using default call direction
*Apr  1 18:42:29.835: Se0/0/0 PPP: Treating connection as a dedicated line
*Apr  1 18:42:29.835: Se0/0/0 PPP: Session handle[4000002] Session id[2]
*Apr  1 18:42:29.835: Se0/0/0 LCP: Event[OPEN] State[Initial to Starting]
*Apr  1 18:42:29.835: Se0/0/0 LCP: O CONFREQ [Starting] id 1 len 23
*Apr  1 18:42:29.835: Se0/0/0 LCP:    AuthProto CHAP (0x0305C22305)
*Apr  1 18:42:29.835: Se0/0/0 LCP:    QualityType 0xC025 period 1000
  (0x0408C025000003E8)
*Apr  1 18:42:29.835: Se0/0/0 LCP:    MagicNumber 0x1F887DD3 (0x05061F887DD3)
<Output omitted>
*Apr  1 18:42:29.855: Se0/0/0 PPP: Phase is AUTHENTICATING, by both
*Apr  1 18:42:29.855: Se0/0/0 CHAP: O CHALLENGE id 1 len 23 from "R1"
<Output omitted>
*Apr  1 18:42:29.871: Se0/0/0 IPCP: Authorizing CP
*Apr  1 18:42:29.871: Se0/0/0 IPCP: CP stalled on event[Authorize CP]
*Apr  1 18:42:29.871: Se0/0/0 IPCP: CP unstall
<Output omitted>
*Apr  1 18:42:29.875: Se0/0/0 CHAP: O SUCCESS id 1 len 4
*Apr  1 18:42:29.879: Se0/0/0 CHAP: I SUCCESS id 1 len 4
*Apr  1 18:42:29.879: Se0/0/0 PPP: Phase is UP
*Apr  1 18:42:29.879: Se0/0/0 IPCP: Protocol configured, start CP. state[Initial]
*Apr  1 18:42:29.879: Se0/0/0 IPCP: Event[OPEN] State[Initial to Starting]
*Apr  1 18:42:29.879: Se0/0/0 IPCP: O CONFREQ [Starting] id 1 len 10
*Apr  1 18:42:29.879: Se0/0/0 IPCP:    Address 10.0.1.1 (0x03060A000101)
*Apr  1 18:42:29.879: Se0/0/0 IPCP: Event[UP] State[Starting to REQsent]
*Apr  1 18:42:29.879: Se0/0/0 IPV6CP: Protocol configured, start CP. state[Initial]
*Apr  1 18:42:29.883: Se0/0/0 IPV6CP: Event[OPEN] State[Initial to Starting]
*Apr  1 18:42:29.883: Se0/0/0 IPV6CP: Authorizing CP
*Apr  1 18:42:29.883: Se0/0/0 IPV6CP: CP stalled on event[Authorize CP]
<Output omitted>
*Apr  1 18:42:29.919: Se0/0/0 IPCP: State is Open
*Apr  1 18:42:29.919: Se0/0/0 IPV6CP: State is Open
*Apr  1 18:42:29.919: Se0/0/0 CDPCP: State is Open
*Apr  1 18:42:29.923: Se0/0/0 CCP: State is Open
*Apr  1 18:42:29.927: Se0/0/0 Added to neighbor route AVL tree: topoid 0, address
  10.0.1.2
*Apr  1 18:42:29.927: Se0/0/0 IPCP: Install route to 10.0.1.2
*Apr  1 18:42:39.871: Se0/0/0 LQM: O state Open magic 0x1F887DD3 len 48
*Apr  1 18:42:39.871: Se0/0/0 LQM:    LastOutLQRs 0 LastOutPackets/Octets 0/0
*Apr  1 18:42:39.871: Se0/0/0 LQM:    PeerInLQRs 0 PeerInPackets/Discards/Errors/
  Octets 0/0/0/0
*Apr  1 18:42:39.871: Se0/0/0 LQM:    PeerOutLQRs 1 PeerOutPackets/Octets 3907/155488
*Apr  1 18:42:39.879: Se0/0/0 LQM: I state Open magic 0xFF101A5B len 48
*Apr  1 18:42:39.879: Se0/0/0 LQM:    LastOutLQRs 0 LastOutPackets/Octets 0/0
```

```
*Apr  1 18:42:39.879: Se0/0/0 LQM:     PeerInLQRs 0 PeerInPackets/Discards/Errors/
  Octets 0/0/0/0
*Apr  1 18:42:39.879: Se0/0/0 LQM:     PeerOutLQRs 1 PeerOutPackets/Octets 3909/155225
<Output omitted>
```

The **debug ppp error** command is used to display protocol errors and error statistics associated with PPP connection negotiation and operation, as shown in Example 3-7. These messages might appear when the Quality Protocol option is enabled on an interface that is already running PPP.

Example 3-7 Output of **debug ppp error** Command

```
R1# debug ppp error

PPP Serial3(i): rlqr receive failure. successes = 15
PPP: myrcvdiffp = 159 peerxmitdiffp = 41091
PPP: myrcvdiffo = 2183 peerxmitdiffo = 1714439
PPP: threshold = 25
PPP Serial4(i): rlqr transmit failure. successes = 15
PPP: myxmitdiffp = 41091 peerrcvdiffp = 159
PPP: myxmitdiffo = 1714439 peerrcvdiffo = 2183
PPP: l->OutLQRs = 1 LastOutLQRs = 1
PPP: threshold = 25
PPP Serial3(i): lqr_protrej() Stop sending LQRs.
PPP Serial3(i): The link appears to be looped back.
```

Troubleshooting a PPP Configuration with Authentication (3.4.1.3)

Authentication is a feature that needs to be implemented correctly or the security of your serial connection may be compromised. Always verify your configuration with the **show interfaces serial** command, in the same way as you did without authentication.

Note

Never assume your authentication configuration works without testing it. Debugging allows you to confirm your configuration and correct any deficiencies. For debugging PPP authentication, use the **debug ppp authentication** command.

Example 3-8 shows an example output of the **debug ppp authentication** command.

Example 3-8 Troubleshooting a PPP Configuration with Authentication

```
R2# debug ppp authentication

Serial0/0/0: Unable to authenticate. No name received from peer
Serial0/0/0: Unable to validate CHAP response. USERNAME pioneer not found.
Serial0/0/0: Unable to validate CHAP response. No password defined for USERNAME pio-
  neer
Serial0/0/0: Failed CHAP authentication with remote.
Remote message is Unknown name
Serial0/0/0: remote passed CHAP authentication.
Serial0/0/0: Passed CHAP authentication with remote.
Serial0/0/0: CHAP input code = 4 id = 3 len = 48
```

The following is an interpretation of the output:

Line 1 says that the router is unable to authenticate on interface Serial0/0/0 because the peer did not send a name.

Line 2 says the router was unable to validate the CHAP response because username pioneer was not found.

Line 3 says no password was found for pioneer. Other possible responses at this line might have been no name received to authenticate, unknown name, no secret for given name, short MD5 response received, or MD5 compare failed.

In the last line, the code 4 means that a failure has occurred. Other code values are as follows:

- **1**: Challenge.
- **2**: Response.
- **3**: Success.
- **4**: Failure.
- **id**: 3 is the ID number per LCP packet format.
- **len**: 48 is the packet length without the header.

Packet Tracer Activity 3.4.1.4: Troubleshooting PPP with Authentication

Background/Scenario

The routers at your company were configured by an inexperienced network engineer. Several errors in the configuration have resulted in connectivity issues. Your boss has asked you to troubleshoot and correct the configuration errors and document your work. Using your knowledge of PPP and standard testing methods, find and correct the errors. Make sure that all of the serial links use PPP CHAP authentication, and that all of the networks are reachable. The passwords are cisco and class.

Lab 3.4.1.5: Troubleshooting Basic PPP with Authentication

In this lab, you will complete the following objectives:

Part 1: Build the Network and Load Device Configurations

Part 2: Troubleshoot the Data Link Layer

Part 3: Troubleshoot the Network Layer

Summary (3.5)

Class Activity 3.5.1.1: PPP Validation

Three friends who are enrolled in the Cisco Networking Academy want to check their knowledge of PPP network configuration.

They set up a contest where each person will be tested on configuring PPP with defined PPP scenario requirements and varying options. Each person devises a different configuration scenario.

The next day they get together and test each other's configuration using their PPP scenario requirements.

Packet Tracer Activity

Packet Tracer Activity 3.5.1.2: Skills Integration Challenge

Background/Scenario

This activity allows you to practice a variety of skills including configuring VLANs, PPP with CHAP, static and default routing, using IPv4 and IPv6. Due to the sheer number of graded elements, feel free to click Check Results and Assessment Items to see if you correctly entered a graded command. Use the passwords cisco and class to access EXEC modes of the CLI for routers and switches.

Serial transmissions sequentially send 1 bit at a time over a single channel. A serial port is bidirectional. Synchronous serial communications require a clocking signal.

Point-to-Point links are usually more expensive than shared services; however, the benefits may outweigh the costs. Constant availability is important for some protocols, such as VoIP.

SONET is an optical network standard that uses STDM for efficient use of bandwidth. In the United States, OC transmission rates are standardized specifications for SONET.

The bandwidth hierarchy used by carriers is different in North America (T-carrier) and Europe (E-carrier). In North America, the fundamental line speed is 64 Kbps, or DS0. Multiple DS0s are bundled together to provide higher line speeds.

The demarcation point is the point in the network where the responsibility of the service provider ends and the responsibility of the customer begins. The CPE, usually a router, is the DTE device. The DCE is usually a modem or CSU/DSU.

A null modem cable is used to connect two DTE devices together without the need for a DCE device by crossing the Tx and Rx lines. When using this cable between routers in a lab, one of the routers must provide the clocking signal.

Cisco HDLC is a bit-oriented synchronous data link layer protocol extension of HDLC and is used by many vendors to provide multiprotocol support. This is the default encapsulation method used on Cisco synchronous serial lines.

Synchronous PPP is used to connect to non-Cisco devices, to monitor link quality, provide authentication, or bundle links for shared use. PPP uses HDLC for encapsulating datagrams. LCP is the PPP protocol used to establish, configure, test, and terminate the data link connection. LCP can optionally authenticate a peer using PAP or CHAP. A family of NCPs are used by the PPP protocol to simultaneously support multiple network layer protocols. Multilink PPP spreads traffic across bundled links by fragmenting packets and simultaneously sending these fragments over multiple links to the same remote address, where they are reassembled.

Practice

The following activities provide practice with the topics introduced in this chapter. The Labs and Class Activities are available in the companion Introduction to Connecting Networks Lab Manual (978-1-58713-331-2). The Packet Tracer Activity PKA files are found in the online course.

Class Activities

Class Activity 3.0.1.2: PPP Persuasion

Class Activity 3.5.1.1: PPP Validation

Labs

Lab 3.3.2.8: Configuring Basic PPP with Authentication

Lab 3.4.1.5: Troubleshooting Basic PPP with Authentication

Packet Tracer Activities

Packet Tracer Activity 3.1.2.7: Troubleshooting Serial Interfaces

Packet Tracer Activity 3.3.2.7: Configuring PAP and CHAP Authentication

Packet Tracer Activity 3.4.1.4: Troubleshooting PPP with Authentication

Packet Tracer Activity 3.5.1.2: Skills Integration Challenge

Check Your Understanding Questions

Complete all the review questions listed here to test your understanding of the topics and concepts in this chapter. The appendix, "Answers to the 'Check Your Understanding' Questions," lists the answers.

1. Match each PPP establishment step with its appropriate sequence number:

   Step 1:

   Step 2:

   Step 3:

   Step 4:

   Step 5:

   A. Test link quality (optional).
   B. Negotiate Layer 3 protocol options.
   C. Send link-establishment frames to negotiate options such as MTU size, compression, and authentication.
   D. Send configuration-acknowledgment frames.
   E. NCP reaches Open state.

2. Which output from the **show interfaces s0/0/0** command indicates that the far end of a point-to-point link has a different encapsulation set than the local router?

   A. Serial 0/0/0 is down, line protocol is down.
   B. Serial 0/0/0 is up, line protocol is down.
   C. Serial 0/0/0 is up, line protocol is up (looped).
   D. Serial 0/0/0 is up, line protocol is down (disabled).
   E. Serial 0/0/0 is administratively down, line protocol is down.

3. What is the default encapsulation for serial interfaces on a Cisco router?

   A. HDLC
   B. PPP
   C. Frame Relay
   D. X.25

4. What is the function of the Protocol field in a PPP frame?

   A. It identifies the application layer protocol that will process the frame.
   B. It identifies the transport layer protocol that will process the frame.
   C. It identifies the data link layer protocol encapsulation in the frame's Data field.
   D. It identifies the network layer protocol encapsulated in the frame's Data field.

5. Match each description with its corresponding term:

   Error control:

   Authentication protocols:

   Allows load balancing:

   Compression protocols:

   A. Stacker/predictor
   B. Magic number
   C. Multilink
   D. CHAP/PAP
   E. Call in

6. Which of the following statements describe the function of statistical time-division multiplexing (STDM)? (Choose three.)

   A. Multiple data streams share one common channel.
   B. Bit interleaving controls the timing mechanism that places data on the channel.
   C. Time slots are used on a first-come, first-served basis.
   D. STDM was developed to overcome the inefficiency caused by time slots still being allocated even when the channel has no data to transmit.
   E. Sources of data alternate during transmission and are reconstructed at the receiving end.
   F. Priority can be dedicated to one data source.

7. Which of the following describes the serial connection between two routers using the High-level Data Link Control (HDLC) protocol?

   A. Synchronous or asynchronous bit-oriented transmissions using a universal frame format
   B. Synchronous bit-oriented transmissions using a frame format that allows flow control and error detection
   C. Asynchronous bit-oriented transmissions using a frame format derived from the Synchronous Data Link Control (SDLC) protocol
   D. Asynchronous bit-oriented transmissions using a V.35 DTE/DCE interface

8. If an authentication protocol is configured for PPP operation, when is the client or user workstation authenticated?

   A. Before link establishment
   B. During the link establishment phase
   C. Before the network layer protocol configuration begins
   D. After the network layer protocol configuration has ended

9. Why are Network Control Protocols used in PPP?

   A. To establish and terminate data links
   B. To provide authentication capabilities to PPP
   C. To manage network congestion and to allow quality testing of the link
   D. To allow multiple Layer 3 protocols to operate over the same physical link

10. Which statement describes the PAP authentication protocol?

    A. It sends encrypted passwords by default.
    B. It uses a two-way handshake to establish identity.
    C. It protects against repeated trial-and-error attacks.
    D. It requires the same username to be configured on every router.

11. A technician testing the functionality of a recently installed router is unable to ping the serial interface of a remote router. The technician executes the **show interfaces serial 0/0/0** command on the local router and sees the following line in the router:

    Serial0/0/0 is down, line protocol is down

    What are two possible causes of this command output?

    A. The **clock rate** command is missing.
    B. The carrier detect signal is not sensed.
    C. Keepalives are not being sent.
    D. The interface is disabled due to a high error rate.
    E. The interface is shut down.
    F. The cabling is faulty or incorrect.

12. The network administrator is configuring Router1 to connect to Router2 using three-way handshake authentication. Match each description with the command necessary to configure Router1:

Configure the username and password:

Enter interface configuration mode:

Specify the encapsulation type:

Configure authentication:

A. **username Router2 password cisco**
B. **username Router1 password cisco**
C. **interface serial 0/1/0**
D. **encapsulation ppp**
E. **encapsulation hdlc**
F. **ppp authentication pap**
G. **ppp authentication chap**

13. What is required to successfully establish a connection between two routers using CHAP authentication?

A. The hostnames of both routers must be the same.
B. The usernames of both routers must be the same.
C. The enable secret passwords configured on both routers must be the same.
D. The password configured with the router's username must be the same on both routers.
E. The **ppp chap sent-username** command must be configured the same on both routers.

14. For each characteristic, indicate whether it is associated with PAP or CHAP:

Two-way handshake:

Three-way handshake:

Open to trial-and-error attacks:

Password sent in cleartext:

Periodic verification:

Uses a one-way hash function:

15. For each description, indicate whether it is associated with LCP or NCP:

    Negotiates link establishment parameters:

    Negotiates Layer 3 protocol parameters:

    Maintains/debugs a link:

    Can negotiate multiple Layer 3 protocols:

    Terminates a link:

16. Describe the functions of LCP and NCP.

17. Describe the five configurable LCP encapsulation options.

18. Refer to the following configurations for Router R1 and Router R3:

```
hostname R1
username R1 password cisco123
!
int serial 0/0
ip address 128.0.1.1 255.255.255.0
encapsulation ppp
ppp authentication pap
----------------------------
hostname R3
username R1 password cisco
!
int serial 0/0
ip address 128.0.1.2 255.255.255.0
encapsulation ppp
ppp authentication CHAP
```

    Router R1 is unable to connect with Router R3. On the basis of the information presented, which configuration changes on Router R1 would correct the problem?