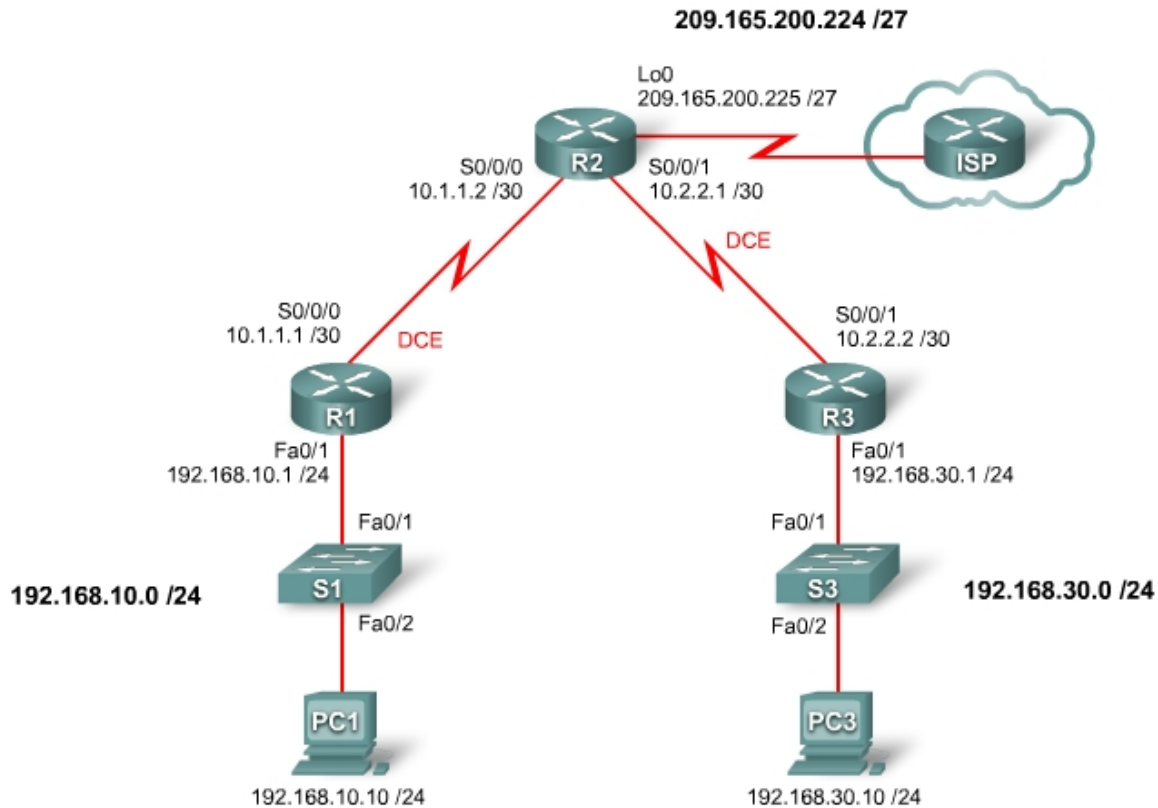


Lab 2.5.1: Basic PPP Configuration Lab

Topology Diagram



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	Fa0/1	192.168.10.1	255.255.255.0	N/A
	S0/0/0	10.1.1.1	255.255.255.252	N/A
R2	Lo0	209.165.200.225	255.255.255.224	N/A
	S0/0/0	10.1.1.2	255.255.255.252	N/A
	S0/0/1	10.2.2.1	255.255.255.252	N/A
R3	Fa0/1	192.168.30.1	255.255.255.0	N/A
	S0/0/1	10.2.2.2	255.255.255.252	N/A
PC1	NIC	192.168.10.10	255.255.255.0	192.168.10.1
PC3	NIC	192.168.30.10	255.255.255.0	192.168.30.1

Learning Objectives

Upon completion of this lab, you will be able to:

- Cable a network according to the topology diagram.
- Erase the startup configuration and reload a router to the default state.
- Perform basic configuration tasks on a router.
- Configure and activate interfaces.
- Configure OSPF routing on all routers.
- Configure PPP encapsulation on all serial interfaces.
- Learn about the **debug ppp negotiation** and **debug ppp packet** commands.
- Learn how to change the encapsulation on the serial interfaces from PPP to HDLC.
- Intentionally break and restore PPP encapsulation.
- Configure PPP PAP and CHAP authentication.
- Intentionally break and restore PPP PAP and CHAP authentication.

Scenario

In this lab, you will learn how to configure PPP encapsulation on serial links using the network shown in the topology diagram. You will also learn how to restore serial links to their default HDLC encapsulation. Pay special attention to what the output of the router looks like when you intentionally break PPP encapsulation. This will assist you in the Troubleshooting lab associated with this chapter. Finally, you will configure PPP PAP authentication and PPP CHAP authentication.

Task 1: Prepare the Network

Step 1: Cable a network that is similar to the one in the topology diagram.

You can use any current router in your lab as long as it has the required interfaces shown in the topology diagram.

Note: If you use 1700, 2500, or 2600 routers, the router outputs and interface descriptions appear differently.

Step 2: Clear any existing configurations on the routers.

Task 2: Perform Basic Router Configuration

Configure the R1, R2, and R3 routers according to the following guidelines:

- Configure the router hostname.
- Disable DNS lookup.
- Configure an EXEC mode password.
- Configure a message-of-the-day banner.
- Configure a password for console connections.

- Configure synchronous logging.
- Configure a password for vty connections.

Task 3: Configure and Activate Serial and Ethernet Addresses

Step 1: Configure interfaces on R1, R2, and R3.

Configure the interfaces on the R1, R2, and R3 routers with the IP addresses from the addressing table at the beginning of the lab. Be sure to include the clock rate on the serial DCE interfaces.

Step 2: Verify IP addressing and interfaces.

Use the **show ip interface brief** command to verify that the IP addressing is correct and that the interfaces are active.

When you have finished, be sure to save the running configuration to the NVRAM of the router.

Step 3: Configure the Ethernet interfaces of PC1 and PC3.

Configure the Ethernet interfaces of PC1 and PC3 with the IP addresses and default gateways from the addressing table.

Step 4: Test the configuration by pinging the default gateway from the PC.

Task 4: Configure OSPF on the Routers

If you need to review the OSPF commands, see Exploration 2, module 11.

Step 1: Enable OSPF routing on R1, R2, and R3.

Use the **router ospf** command with a process ID of 1. Be sure to advertise the networks.

```
R1(config)#router ospf 1
R1(config-router)#network 192.168.10.0 0.0.0.255 area 0
R1(config-router)#network 10.1.1.0 0.0.0.3 area 0
*Aug 17 17:49:14.689: %OSPF-5-ADJCHG: Process 1, Nbr 209.165.200.225 on
Serial0/0/0 from LOADING to FULL, Loading Done
R1(config-router)#
```

```
R2(config)#router ospf 1
R2(config-router)#network 10.1.1.0 0.0.0.3 area 0
*Aug 17 17:48:40.645: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.10.1 on
Serial0/0/0 from LOADING to FULL, Loading Done
R2(config-router)#network 10.2.2.0 0.0.0.3 area 0
R2(config-router)#network 209.165.200.224 0.0.0.31 area 0
R2(config-router)#
*Aug 17 17:57:44.729: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.1 on
Serial0/0/1 from LOADING to FULL, Loading Done
R2(config-router)#
```

```
R3(config)#router ospf 1
R3(config-router)#network 10.2.2.0 0.0.0.3 area 0
*Aug 17 17:58:02.017: %OSPF-5-ADJCHG: Process 1, Nbr 209.165.200.225 on
Serial0/0/1 from LOADING to FULL, Loading Done
R3(config-router)#network 192.168.30.0 0.0.0.255 area 0
R3(config-router)#
```

Step 2: Verify that you have full network connectivity.

Use the **show ip route** and **ping** commands to verify connectivity.

R1#**show ip route**

<output omitted>

```
O   192.168.30.0/24 [110/1563] via 10.1.1.2, 00:33:56, Serial0/0/0
C   192.168.10.0/24 is directly connected, FastEthernet0/1
    209.165.200.0/27 is subnetted, 1 subnets
O     209.165.200.225 [110/782] via 10.1.1.2, 00:33:56, Serial0/0/0
    10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
O     10.2.2.0/30 [110/1562] via 10.1.1.2, 00:33:56, Serial0/0/0
C     10.1.1.0/30 is directly connected, Serial0/0/0
```

R1#**ping 192.168.30.1**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.30.1, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 32/32/32 ms

R1#

R2#**show ip route**

<output omitted>

```
O   192.168.30.0/24 [110/782] via 10.2.2.2, 00:33:04, Serial0/0/1
O   192.168.10.0/24 [110/782] via 10.1.1.1, 00:33:04, Serial0/0/0
    209.165.200.0/27 is subnetted, 1 subnets
C     209.165.200.224 is directly connected, Loopback0
    10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C     10.2.2.0/30 is directly connected, Serial0/0/1
C     10.1.1.0/30 is directly connected, Serial0/0/0
```

R2#**ping 192.168.30.1**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.30.1, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 16/16/16 ms

R2#**ping 192.168.10.1**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.10.1, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 16/16/16 ms

R2#

R3#**show ip route**

<output omitted>

```
C   192.168.30.0/24 is directly connected, FastEthernet0/1
O   192.168.10.0/24 [110/1563] via 10.2.2.1, 00:32:01, Serial0/0/1
    209.165.200.0/27 is subnetted, 1 subnets
```

```
O      209.165.200.225 [110/782] via 10.2.2.1, 00:32:01, Serial0/0/1
      10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
C      10.2.2.0/30 is directly connected, Serial0/0/1
O      10.1.1.0/30 [110/1562] via 10.2.2.1, 00:32:01, Serial0/0/1

R3#ping 209.165.200.225

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.200.225, timeout is 2
seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/16/16 ms
R3#ping 192.168.10.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/32/32 ms
R3#
```

Task 5: Configure PPP Encapsulation on Serial Interfaces

Step 1: Use the show interface command to check whether HDLC is the default serial encapsulation.

```
R1#show interface serial0/0/0
Serial0/0/0 is up, line protocol is up
  Hardware is GT96K Serial
  Internet address is 10.1.1.1/30
  MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
```

<output omitted>

```
R2#show interface serial 0/0/0
Serial0/0/0 is up, line protocol is up
  Hardware is GT96K Serial
  Internet address is 10.1.1.2/30
  MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
```

<output omitted>

```
R2#show interface serial 0/0/1
Serial0/0/1 is up, line protocol is up
  Hardware is GT96K Serial
  Internet address is 10.2.2.1/30
  MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
```

<output omitted>

```
R3#show interface serial 0/0/1
Serial0/0/1 is up, line protocol is up
  Hardware is GT96K Serial
  Internet address is 10.2.2.2/30
  MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set

<output omitted>
```

Step 2: Use debug commands on R1 and R2 to see the effects of configuring PPP.

```
R1#debug ppp negotiation
PPP protocol negotiation debugging is on
R1#debug ppp packet
PPP packet display debugging is on
R1#
```

```
R2#debug ppp negotiation
PPP protocol negotiation debugging is on
R2#debug ppp packet
PPP packet display debugging is on
R2#
```

Step 3: Change the encapsulation of the serial interfaces from HDLC to PPP.

Change the encapsulation type on the link between R1 and R2, and observe the effects. If you start to receive too much debug data, use the **undebg all** command to turn debugging off.

```
R1(config)#interface serial 0/0/0
R1(config-if)#encapsulation ppp
R1(config-if)#
*Aug 17 19:02:53.412: %OSPF-5-ADJCHG: Process 1, Nbr 209.165.200.225 on
Serial0/0/0 from FULL to DOWN, Neighbor Down: Interface down or
detached
R1(config-if)#
*Aug 17 19:02:53.416: Se0/0/0 PPP: Phase is DOWN, Setup
*Aug 17 19:02:53.416: Se0/0/0 PPP: Using default call direction
*Aug 17 19:02:53.416: Se0/0/0 PPP: Treating connection as a dedicated
line
*Aug 17 19:02:53.416: Se0/0/0 PPP: Session handle[E4000001] Session
id[0]
*Aug 17 19:02:53.416: Se0/0/0 PPP: Phase is ESTABLISHING, Active Open
*Aug 17 19:02:53.424: Se0/0/0 LCP: O CONFREQ [Closed] id 1 len 10
*Aug 17 19:02:53.424: Se0/0/0 LCP: MagicNumber 0x63B994DE
(0x050663B994DE)
R1(config-if)#
*Aug 17 19:02:55.412: Se0/0/0 PPP: Outbound cdp packet dropped
*Aug 17 19:02:55.432: Se0/0/0 LCP: TIMEOUT: State REQsent
*Aug 17 19:02:55.432: Se0/0/0 LCP: O CONFREQ [REQsent] id 2 len 10
*Aug 17 19:02:55.432: Se0/0/0 LCP: MagicNumber 0x63B994DE
(0x050663B994DE)
*Aug 17 19:02:56.024: Se0/0/0 PPP: I pkt type 0x008F, datagramsize 24
link[illegal]
*Aug 17 19:02:56.024: Se0/0/0 UNKNOWN(0x008F): Non-NCP packet,
discarding
R1(config-if)#
```

```
*Aug 17 19:02:57.252: Se0/0/0 PPP: I pkt type 0x000F, datagramsize 84
link[illegal]
*Aug 17 19:02:57.252: Se0/0/0 UNKNOWN(0x000F): Non-NCP packet,
discarding
*Aug 17 19:02:57.448: Se0/0/0 LCP: TIMEout: State REQsent
*Aug 17 19:02:57.448: Se0/0/0 LCP: O CONFREQ [REQsent] id 3 len 10
*Aug 17 19:02:57.448: Se0/0/0 LCP:      MagicNumber 0x63B994DE
(0x050663B994DE)
R1(config-if)#
*Aug 17 19:02:58.412: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/0, changed state to down
```

```
R2(config)#interface serial 0/0/0
R2(config-if)#encapsulation ppp
R2(config-if)#
*Aug 17 19:06:48.848: Se0/0/0 PPP: Phase is DOWN, Setup
*Aug 17 19:06:48.848: Se0/0/0 PPP: Using default call direction
*Aug 17 19:06:48.848: Se0/0/0 PPP: Treating connection as a dedicated
line
*Aug 17 19:06:48.848: Se0/0/0 PPP: Session handle[C6000001] Session
id[0]
*Aug 17 19:06:48.848: Se0/0/0 PPP: Phase is ESTABLISHING, Active Open
*Aug 17 19:06:48.856: Se0/0/0 LCP: O CONFREQ [Closed] id 1 len 10
*Aug 17 19:06:48.856: Se0/0/0 LCP:      MagicNumber 0x63BD388C
(0x050663BD388C)
*Aug 17 19:06:48.860: Se0/0/0 PPP: I pkt type 0xC021, datagramsize 14
link[ppp]
*Aug 17 19:06:48.860: Se0/0/0 LCP: I CONFACK [REQsent] id 1 len 10
R2(config-if)#
*Aug 17 19:06:48.860: Se0/0/0 LCP:      MagicNumber 0x63BD388C
(0x050663BD388C)
R2(config-if)#
*Aug 17 19:06:50.864: Se0/0/0 LCP: TIMEout: State ACKrcvd
*Aug 17 19:06:50.864: Se0/0/0 LCP: O CONFREQ [ACKrcvd] id 2 len 10
*Aug 17 19:06:50.864: Se0/0/0 LCP:      MagicNumber 0x63BD388C
(0x050663BD388C)
*Aug 17 19:06:50.868: Se0/0/0 PPP: I pkt type 0xC021, datagramsize 14
link[ppp]
*Aug 17 19:06:50.868: Se0/0/0 LCP: I CONFREQ [REQsent] id 61 len 10
*Aug 17 19:06:50.868: Se0/0/0 LCP:      MagicNumber 0x63BDB9A8
(0x050663BDB9A8)
*Aug 17 19:06:50.868: Se0/0/0 LCP: O CONFACK [REQsent] id 61 len 10
*Aug 17 19:06:50.868: Se0/0/0 LCP:      MagicNumber 0x63BDB9A8
(0x050663BDB9A8)
*Aug 17 19:06:50.868: Se0/0/0 PPP: I pkt type 0xC021, datagramsize 14
link[ppp]
*Aug 17 19:06:50.868: Se0/0/0 LCP: I CONFACK [ACKsent] id 2 len 10
*Aug 17 19:06:50.868: Se0/0/0 LCP:      MagicNumber 0x63BD388C
(0x050663BD388C)
*Aug 17 19:06:50.868: Se0/0/0 LCP: State is Open
*Aug 17 19:06:50.872: Se0/0/0 PPP: Phase is FORWARDING, Attempting
Forward
*Aug 17 19:06:50.872: Se0/0/0 PPP: Phase is ESTABLISHING, Finish LCP
*Aug 17 19:06:50.872: Se0/0/0 PPP: Phase is UP
*Aug 17 19:06:50.872: Se0/0/0 IPCP: O CONFREQ [Closed] id 1 len 10
*Aug 17 19:06:50.872: Se0/0/0 IPCP:      Address 10.1.1.2
```

```
(0x03060A010102)
*Aug 17 19:06:50.872: Se0/0/0 CDPCP: O CONFREQ [Closed] id 1 len 4
*Aug 17 19:06:50.872: Se0/0/0 PPP: Process pending ncp packets
*Aug 17 19:06:50.876: Se0/0/0 PPP: I pkt type 0x8021, datagramsize 14
link[ip]
*Aug 17 19:06:50.876: Se0/0/0 IPCP: I CONFREQ [REQsent] id 1 len 10
*Aug 17 19:06:50.876: Se0/0/0 IPCP:      Address 10.1.1.1
(0x03060A010101)
*Aug 17 19:06:50.876: Se0/0/0 PPP: I pkt type 0x8207, datagramsize 8
link[cdp]
*Aug 17 19:06:50.876: Se0/0/0 IPCP: O CONFACK [REQsent] id 1 len 10
*Aug 17 19:06:50.876: Se0/0/0 IPCP:      Address 10.1.1.1
(0x03060A010101)
*Aug 17 19:06:50.876: Se0/0/0 CDPCP: I CONFREQ [REQsent] id 1 len 4
*Aug 17 19:06:50.876: Se0/0/0 CDPCP: O CONFACK [REQsent] id 1 len 4
*Aug 17 19:06:50.876: Se0/0/0 PPP: I pkt type 0x8021, datagramsize 14
link[ip]
*Aug 17 19:06:50.876: Se0/0/0 IPCP: I CONFACK [ACKse
R2(config-if)#nt] id 1 len 10
*Aug 17 19:06:50.876: Se0/0/0 IPCP:      Address 10.1.1.2
(0x03060A010102)
*Aug 17 19:06:50.876: Se0/0/0 IPCP: State is Open
*Aug 17 19:06:50.876: Se0/0/0 PPP: I pkt type 0x8207, datagramsize 8
link[cdp]
*Aug 17 19:06:50.876: Se0/0/0 IPCP: Install route to 10.1.1.1
*Aug 17 19:06:50.880: Se0/0/0 CDPCP: I CONFACK [ACKsent] id 1 len 4
*Aug 17 19:06:50.880: Se0/0/0 CDPCP: State is Open
*Aug 17 19:06:50.880: Se0/0/0 PPP: O pkt type 0x0021, datagramsize 80
*Aug 17 19:06:50.880: Se0/0/0 IPCP: Add link info for cef entry
10.1.1.1
*Aug 17 19:06:50.884: Se0/0/0 PPP: I pkt type 0x0021, datagramsize 80
link[ip]
*Aug 17 19:06:51.848: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/0, changed state to up
R2(config-if)#
*Aug 17 19:06:51.888: Se0/0/0 LCP-FS: I ECHOREQ [Open] id 1 len 12
magic 0x63BDB9A8
*Aug 17 19:06:51.888: Se0/0/0 LCP-FS: O ECHOREP [Open] id 1 len 12
magic 0x63BD388C

<output omitted>

*Aug 17 19:07:00.936: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.10.1 on
Serial0/0/0 from LOADING to FULL, Loading Done
```

What happens when one end of the serial link is encapsulated with PPP and the other end of the link is encapsulated with HDLC?

What steps does PPP go through when the other end of the serial link on R2 is configured with PPP encapsulation?

What happens when PPP encapsulation is configured on each end of the serial link?

Step 4: Turn off debugging.

Turn off debugging if you have not already used the **undebug all** command.

```
R1#undebug all
```

Port Statistics for unclassified packets is not turned on.

All possible debugging has been turned off

```
R1#
```

```
R2#undebug all
```

Port Statistics for unclassified packets is not turned on.

All possible debugging has been turned off

```
R2#
```

Step 5: Change the encapsulation from HDLC to PPP on both ends of the serial link between R2 and R3.

```
R2(config)#interface serial0/0/1
```

```
R2(config-if)#encapsulation ppp
```

```
R2(config-if)#
```

```
*Aug 17 20:02:08.080: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.1 on  
Serial0/0/1 from FULL to DOWN, Neighbor Down: Interface down or  
detached
```

```
R2(config-if)#
```

```
*Aug 17 20:02:13.080: %LINEPROTO-5-UPDOWN: Line protocol on Interface  
Serial0/0/1, changed state to down
```

```
R2(config-if)#
```

```
*Aug 17 20:02:58.564: %LINEPROTO-5-UPDOWN: Line protocol on Interface  
Serial0/0/1, changed state to up
```

```
R2(config-if)#
```

```
*Aug 17 20:03:03.644: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.1 on  
Serial0/0/1 from LOADING to FULL, Loading Done
```

```
R2(config-if)#
```

```
*Aug 17 20:03:46.988: %LINEPROTO-5-UPDOWN: Line protocol on Interface  
Serial0/0/1, changed state to down
```

```
R3(config)#interface serial 0/0/1
```

```
R3(config-if)#encapsulation ppp
```

```
R3(config-if)#
```

```
*Aug 17 20:04:27.152: %LINEPROTO-5-UPDOWN: Line protocol on Interface  
Serial0/0/1, changed state to up
```

```
*Aug 17 20:04:30.952: %OSPF-5-ADJCHG: Process 1, Nbr 209.165.200.225 on  
Serial0/0/1 from LOADING to FULL, Loading Done
```

When does the line protocol on the serial link come up and the OSPF adjacency is restored?

Step 6: Verify that PPP is now the encapsulation on the serial interfaces.

```
R1#show interface serial0/0/0
```

```
Serial0/0/0 is up, line protocol is up  
  Hardware is GT96K Serial  
  Internet address is 10.1.1.1/30  
  MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,  
    reliability 255/255, txload 1/255, rxload 1/255  
  Encapsulation PPP, LCP Open  
  Open: CDPCP, IPCP, loopback not set
```

```
<output omitted>
```

```
R2#show interface serial 0/0/0
```

```
Serial0/0/0 is up, line protocol is up  
  Hardware is GT96K Serial  
  Internet address is 10.1.1.2/30  
  MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,  
    reliability 255/255, txload 1/255, rxload 1/255  
  Encapsulation PPP, LCP Open  
  Open: CDPCP, IPCP, loopback not set
```

```
<output omitted>
```

```
R2#show interface serial 0/0/1
```

```
Serial0/0/1 is up, line protocol is up  
  Hardware is GT96K Serial  
  Internet address is 10.2.2.1/30  
  MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,  
    reliability 255/255, txload 1/255, rxload 1/255  
  Encapsulation PPP, LCP Open  
  Open: CDPCP, IPCP, loopback not set
```

```
<output omitted>
```

```
R3#show interface serial 0/0/1
```

```
Serial0/0/1 is up, line protocol is up
```

```
Hardware is GT96K Serial
Internet address is 10.2.2.2/30
MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
Encapsulation PPP, LCP Open
Open: CDPCP, IPCP, loopback not set
```

<output omitted>

Task 6: Break and Restore PPP Encapsulation

By intentionally breaking PPP encapsulation, you will learn about the error messages that are generated. This will help you later in the Troubleshooting lab.

Step 1: Return both serial interfaces on R2 to their default HDLC encapsulation.

```
R2(config)#interface serial 0/0/0
R2(config-if)#encapsulation hdlc
R2(config-if)#
*Aug 17 20:36:48.432: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.10.1 on
Serial0/0/0 from FULL to DOWN, Neighbor Down: Interface down or
detached
*Aug 17 20:36:49.432: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/0, changed state to down
R2(config-if)#
*Aug 17 20:36:51.432: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/0, changed state to up
R2(config-if)#interface serial 0/0/1
*Aug 17 20:37:14.080: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/0, changed state to down
R2(config-if)#encapsulation hdlc
R2(config-if)#
*Aug 17 20:37:17.368: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.1 on
Serial0/0/1 from FULL to DOWN, Neighbor Down: Interface down or
detached
*Aug 17 20:37:18.368: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/1, changed state to down
R2(config-if)#
*Aug 17 20:37:20.368: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/1, changed state to up
R2(config-if)#
*Aug 17 20:37:44.080: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/1, changed state to down
R2(config-if)#
```

Why is it useful to intentionally break a configuration?

Why do both serial interfaces go down, come back up, and then go back down?

Can you think of another way to change the encapsulation of a serial interface from PPP to the default HDLC encapsulation other than using the **encapsulation hdlc** command? (Hint: It has to do with the **no** command.)

Step 2: Return both serial interfaces on R2 to PPP encapsulation.

```
R2(config)#interface s0/0/0
R2(config-if)#encapsulation ppp
*Aug 17 20:53:06.612: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/0, changed state to up
R2(config-if)#interface s0/0/1
*Aug 17 20:53:10.856: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.10.1 on
Serial0/0/0 from LOADING to FULL, Loading Done
R2(config-if)#encapsulation ppp
*Aug 17 20:53:23.332: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/1, changed state to up
R2(config-if)#
*Aug 17 20:53:24.916: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.1 on
Serial0/0/1 from LOADING to FULL, Loading Done
R2(config-if)#
```

Task 7: Configure PPP Authentication

Step 1: Configure PPP PAP authentication on the serial link between R1 and R2.

```
R1(config)#username R1 password cisco
R1(config)#int s0/0/0
R1(config-if)#ppp authentication pap
R1(config-if)#
*Aug 22 18:58:57.367: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/0, changed state to down
R1(config-if)#
*Aug 22 18:58:58.423: %OSPF-5-ADJCHG: Process 1, Nbr 209.165.200.225 on
Serial0/0/0 from FULL to DOWN, Neighbor Down: Interface down or
detached
R1(config-if)#ppp pap sent-username R2 password cisco
```

What happens when PPP PAP authentication is only configured on one end of the serial link?

```
R2(config)#username R2 password cisco
R2(config)#interface Serial0/0/0
R2(config-if)#ppp authentication pap
R2(config-if)#ppp pap sent-username R1 password cisco
R2(config-if)#
*Aug 23 16:30:33.771: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/0, changed state to up
R2(config-if)#
*Aug 23 16:30:40.815: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.10.1 on
Serial0/0/0 from LOADING to FULL, Loading Done
R2(config-if)#
```

What happens when PPP PAP authentication is configured on both ends of the serial link?

Step 2: Configure PPP CHAP authentication on the serial link between R2 and R3.

In PAP authentication, the password is not encrypted. While this is certainly better than no authentication at all, it is still highly preferable to encrypt the password that is being sent across the link. CHAP encrypts the password.

```
R2(config)#username R3 password cisco
R2(config)#int s0/0/1
R2(config-if)#ppp authentication chap
R2(config-if)#
*Aug 23 18:06:00.935: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/1, changed state to down
R2(config-if)#
*Aug 23 18:06:01.947: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.1 on
Serial0/0/1 from FULL to DOWN, Neighbor Down: Interface down or
detached
R2(config-if)#

R3(config)#username R2 password cisco
*Aug 23 18:07:13.074: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/1, changed state to up
R3(config)#int s0/0/1
R3(config-if)#
*Aug 23 18:07:22.174: %OSPF-5-ADJCHG: Process 1, Nbr 209.165.200.225 on
Serial0/0/1 from LOADING to FULL, Loading Done
R3(config-if)#ppp authentication chap
R3(config-if)#
```

Notice that the line protocol on interface serial 0/0/1 changes state to UP even before the interface is configured for CHAP authentication. Can you guess why this is the case?

Step 3: Review the debug output.

To better understand the CHAP process, view the output of the **debug ppp authentication** command on R2 and R3. Then shut down interface serial 0/0/1 on R2, and issue the **no shutdown** command on interface serial 0/0/1 on R2.

```
R2#debug ppp authentication
PPP authentication debugging is on
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#int s0/0/1
R2(config-if)#shutdown
R2(config-if)#
*Aug 23 18:19:21.059: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.1 on
Serial0/0/1 from FULL to DOWN, Neighbor Down: Interface down or
detached
R2(config-if)#
*Aug 23 18:19:23.059: %LINK-5-CHANGED: Interface Serial0/0/1, changed
state to administratively down
*Aug 23 18:19:24.059: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/1, changed state to down
R2(config-if)#no shutdown

*Aug 23 18:19:55.059: Se0/0/1 PPP: Using default call direction
*Aug 23 18:19:55.059: Se0/0/1 PPP: Treating connection as a dedicated
line
*Aug 23 18:19:55.059: Se0/0/1 PPP: Session handle[5B000005] Session
id[49]
*Aug 23 18:19:55.059: Se0/0/1 PPP: Authorization required
*Aug 23 18:19:55.063: %LINK-3-UPDOWN: Interface Serial0/0/1, changed
state to up
*Aug 23 18:19:55.063: Se0/0/1 CHAP: O CHALLENGE id 48 len 23 from "R2"
*Aug 23 18:19:55.067: Se0/0/1 CHAP: I CHALLENGE id 2 len 23 from "R3"
*Aug 23 18:19:55.067: Se0/0/1 CHAP: Using hostname from unknown source
*Aug 23 18:19:55.067: Se0/0/1 CHAP: Using password from AAA
*Aug 23 18:19:55.067: Se0/0/1 CHAP: O RESPONSE id 2 len 23 from "R2"
*Aug 23 18:19:55.071: Se0/0/1 CHAP: I RESPONSE id 48 len 23 from "R3"
*Aug 23 18:19:55.071: Se0/0/1 PPP: Sent CHAP LOGIN Request
*Aug 23 18:19:55.071: Se0/0/1 PPP: Received LOGIN Response PASS
*Aug 23 18:19:55.071: Se0/0/1 PPP: Sent LCP AUTHOR Request
*Aug 23 18:19:55.075: Se0/0/1 PPP: Sent IPCP AUTHOR Request
*Aug 23 18:19:55.075: Se0/0/1 LCP: Received AAA AUTHOR Response PASS
*Aug 23 18:19:55.075: Se0/0/1 IPCP: Received AAA AUTHOR Response PASS
```

```
*Aug 23 18:19:55.075: Se0/0/1 CHAP: O SUCCESS id 48 len 4
*Aug 23 18:19:55.075: Se0/0/1 CHAP: I SUCCESS id 2 len 4
*Aug 23 18:19:55.075: Se0/0/1 PPP: Sent CDPCP AUTHOR Request
*Aug 23 18:19:55.075: Se0/0/1 CDPCP: Received AAA AUTHOR Response PASS
*Aug 23 18:19:55.079: Se0/0/1 PPP: Sent IPCP AUTHOR Request
*Aug 23 18:19:56.075: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/1, changed state to up
R2(config-if)#
*Aug 23 18:20:05.135: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.1 on
Serial0/0/1 from LOADING to FULL, Loading Done
```

R3#debug ppp authentication

PPP authentication debugging is on

R3#

```
*Aug 23 18:19:04.494: %LINK-3-UPDOWN: Interface Serial0/0/1, changed
state to down
```

R3#

```
*Aug 23 18:19:04.494: %OSPF-5-ADJCHG: Process 1, Nbr 209.165.200.225 on
Serial0/0/1 from FULL to DOWN, Neighbor Down: Interface down or
detached
```

```
*Aug 23 18:19:05.494: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/1, changed state to down
```

R3#

```
*Aug 23 18:19:36.494: %LINK-3-UPDOWN: Interface Serial0/0/1, changed
state to up
```

```
*Aug 23 18:19:36.494: Se0/0/1 PPP: Using default call direction
```

```
*Aug 23 18:19:36.494: Se0/0/1 PPP: Treating connection as a dedicated
line
```

```
*Aug 23 18:19:36.494: Se0/0/1 PPP: Session handle[3C000034] Session
id[52]
```

```
*Aug 23 18:19:36.494: Se0/0/1 PPP: Authorization required
```

```
*Aug 23 18:19:36.498: Se0/0/1 CHAP: O CHALLENGE id 2 len 23 from "R3"
```

```
*Aug 23 18:19:36.502: Se0/0/1 CHAP: I CHALLENGE id 48 len 23 from "R2"
```

```
*Aug 23 18:19:36.502: Se0/0/1 CHAP: Using hostname from unknown source
```

```
*Aug 23 18:19:36.506: Se0/0/1 CHAP: Using password from AAA
```

```
*Aug 23 18:19:36.506: Se0/0/1 CHAP: O RESPONSE id 48 len 23 from "R3"
```

```
*Aug 23 18:19:36.506: Se0/0/1 CHAP: I RESPONSE id 2 len 23 from "R2"
```

R3#

```
*Aug 23 18:19:36.506: Se0/0/1 PPP: Sent CHAP LOGIN Request
```

```
*Aug 23 18:19:36.506: Se0/0/1 PPP: Received LOGIN Response PASS
```

```
*Aug 23 18:19:36.510: Se0/0/1 PPP: Sent LCP AUTHOR Request
```

```
*Aug 23 18:19:36.510: Se0/0/1 PPP: Sent IPCP AUTHOR Request
```

```
*Aug 23 18:19:36.510: Se0/0/1 LCP: Received AAA AUTHOR Response PASS
```

```
*Aug 23 18:19:36.510: Se0/0/1 IPCP: Received AAA AUTHOR Response PASS
```

```
*Aug 23 18:19:36.510: Se0/0/1 CHAP: O SUCCESS id 2 len 4
```

```
*Aug 23 18:19:36.510: Se0/0/1 CHAP: I SUCCESS id 48 len 4
```

```
*Aug 23 18:19:36.514: Se0/0/1 PPP: Sent CDPCP AUTHOR Request
```

```
*Aug 23 18:19:36.514: Se0/0/1 PPP: Sent IPCP AUTHOR Request
```

```
*Aug 23 18:19:36.514: Se0/0/1 CDPCP: Received AAA AUTHOR Response PASS
```

R3#

```
*Aug 23 18:19:37.510: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/1, changed state to up
```

R3#

```
*Aug 23 18:19:46.570: %OSPF-5-ADJCHG: Process 1, Nbr 209.165.200.225 on
Serial0/0/1 from LOADING to FULL, Loading Done
```

R3#

Task 8: Intentionally Break and Restore PPP CHAP Authentication

Step 1: Break PPP CHAP authentication.

On the serial link between R2 and R3, change the authentication protocol on interface serial 0/0/1 to PAP.

```
R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#int s0/0/1
R2(config-if)#ppp authentication pap
R2(config-if)#^Z
R2#
*Aug 24 15:45:47.039: %SYS-5-CONFIG_I: Configured from console by
console
R2#copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
R2#reload
```

Does changing the authentication protocol to PAP on interface serial 0/0/1 break authentication between R2 and R3?

Step 2: Restore PPP CHAP authentication on the serial link.

Notice that it is not necessary to reload the router for this change to take effect.

```
R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#int s0/0/1
R2(config-if)#ppp authentication chap
R2(config-if)#
*Aug 24 15:50:00.419: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/1, changed state to up
R2(config-if)#
*Aug 24 15:50:07.467: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.30.1 on
Serial0/0/1 from LOADING to FULL, Loading Done
R2(config-if)#
```

Step 3: Intentionally Break PPP CHAP authentication by changing the password on R3.

```
R3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R3(config)#username R2 password ciisco
R3(config)#^Z
R3#
*Aug 24 15:54:17.215: %SYS-5-CONFIG_I: Configured from console by
console
R3#copy run start
Destination filename [startup-config]?
```


Building configuration...
[OK]
R3#**reload**

After reloading, what is the status of the line protocol on serial 0/0/1?

Step 4: Restore PPP CHAP authentication by changing the password on R3.

```
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#username R2 password cisco
R3(config)#
*Aug 24 16:11:10.679: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial0/0/1, changed state to up
R3(config)#
*Aug 24 16:11:19.739: %OSPF-5-ADJCHG: Process 1, Nbr 209.165.200.225 on
Serial0/0/1 from LOADING to FULL, Loading Done
R3(config)#
```

Task 9: Document the Router Configurations

On each router, issue the **show run** command and capture the configurations.

```
R1#show run
!<output omitted>
!
hostname R1
!
!
enable secret class
!
!
!
no ip domain lookup
!
username R1 password 0 cisco
!
!
!
interface FastEthernet0/1
 ip address 192.168.10.1 255.255.255.0
 no shutdown
!
!
interface Serial0/0/0
 ip address 10.1.1.1 255.255.255.252
 encapsulation ppp
 clockrate 64000
 ppp authentication pap
 ppp pap sent-username R2 password 0 cisco
 no shutdown
```

```
!  
!  
!  
router ospf 1  
  network 10.1.1.0 0.0.0.3 area 0  
  network 192.168.10.0 0.0.0.255 area 0  
!  
!  
banner motd ^CCUnauthorized access strictly prohibited and prosecuted  
to the full extent of the law^C  
!  
line con 0  
  exec-timeout 0 0  
  password cisco  
  logging synchronous  
  login  
line aux 0  
line vty 0 4  
  password cisco  
  login  
!  
end
```

R2#show run

!<output omitted>

```
!  
hostname R2  
!  
!  
enable secret class  
!  
!  
no ip domain lookup  
!  
username R3 password 0 cisco  
username R2 password 0 cisco  
!  
!  
!  
interface Loopback0  
  ip address 209.165.200.225 255.255.255.224  
!  
!  
!  
interface Serial0/0/0  
  ip address 10.1.1.2 255.255.255.252  
  encapsulation ppp  
  ppp authentication pap  
  ppp pap sent-username R1 password 0 cisco  
  no shutdown  
!  
interface Serial0/0/1  
  ip address 10.2.2.1 255.255.255.252  
  encapsulation ppp  
  clockrate 64000
```

```
    ppp authentication chap
    no shutdown
!
!
router ospf 1
  network 10.1.1.0 0.0.0.3 area 0
  network 10.2.2.0 0.0.0.3 area 0
  network 209.165.200.224 0.0.0.31 area 0
!
!
banner motd ^CUnauthorized access strictly prohibited and prosecuted to
the full extent of the law^C
!
line con 0
  exec-timeout 0 0
  password cisco
  logging synchronous
  login
line aux 0
line vty 0 4
  password cisco
  login
!
end
```

R3#show run

!<output omitted>

```
!
hostname R3
!
!
enable secret class
!
!
!
no ip domain lookup
!
username R2 password 0 cisco
!
!
!
interface FastEthernet0/1
  ip address 192.168.30.1 255.255.255.0
  no shutdown
!
!
interface Serial0/0/1
  ip address 10.2.2.2 255.255.255.252
  encapsulation ppp
  ppp authentication chap
  no shutdown
!
router ospf 1
  network 10.2.2.0 0.0.0.3 area 0
  network 192.168.30.0 0.0.0.255 area 0
```

```
!  
!  
banner motd ^CUnauthorized access strictly prohibited and prosecuted to  
the full extent of the law^C  
!  
line con 0  
  exec-timeout 0 0  
  password cisco  
  logging synchronous  
  login  
line aux 0  
line vty 0 4  
  password cisco  
  login  
!  
end
```

Task 10: Clean Up

Erase the configurations and reload the routers. Disconnect and store the cabling. For PC hosts that are normally connected to other networks, such as the school LAN or the Internet, reconnect the appropriate cabling and restore the TCP/IP settings.