Al-Najah National University
Network Design Lab

FRAME RELAY

Instructor:Dr. Amjad Hawash

Mohammed Adnan
Dima Bshara
Amall ALdeek

1 Abstract

In this experiment, we configure Frame Relay encapsulation on serial links. Then we configure a router to simulate a Frame Relay switch. In the end, we configure Frame Relay point-to-point subinterfaces.

2 Introduction

Frame Relay is a standardized wide area network (WAN) technology that specifies the physical and data link layers of digital telecommunications channels using a packet switching methodology. Originally designed for transport across Integrated Services Digital Network (ISDN) infrastructure, it may be used today in the context of many other network interfaces. Network providers commonly implement Frame Relay for voice (VoFR) and data as an encapsulation technique used between local area networks (LANs) over a WAN. Each end-user gets a private line (or leased line) to a Frame Relay node. The Frame Relay network handles the transmission over a frequently changing path transparent to all end-user extensively used WAN protocols. It is less expensive than leased lines and that is one reason for its popularity. The extreme simplicity of configuring user equipment in a Frame Relay network offers another reason for Frame Relay's popularity. With the advent of Ethernet over fiber optics, MPLS, VPN and dedicated broadband services such as cable modem and DSL. The virtual circuits in frame relay are called as Data Link Connection Identifier **(DLCI)**.This is actually a number which identifies a virtual circuit in frame relay.
Frame relay can provide two types of virtual circuits: 1. Permanent Virtual Circuits (PVC).2. Switched Virtual Circuits (SVC).**PVC** refers to a Permanent Virtual Connection. Packet switching or frame relay operation takes you data at one location and provides a path through the network to the destination you specified. Frame Relay supports two types of subinterfaces: point-to-point and point-to-multipoint. Point-to-multipoint subinterfaces support non-broadcast multiaccess topologies. For example, a hub and spoke topology would use a point-to-multipoint subinterface. In this experiment, we will create a point-to-point subinterface.

3 Producer

We Build the Network as a topology show then we Configure Basic Device Settings.

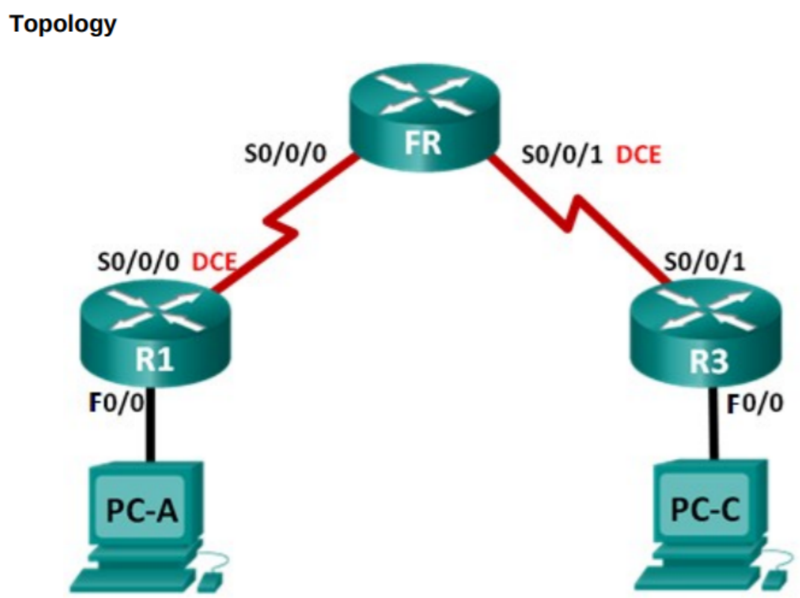


Figure 1: The Topology

3.1 Configure a Frame Relay Switch

1. In first we do a basic configuration for all router such as putting password and rename and SO on.
2. Then, we Configure the IPv4 addresses listed in the Addressing Table for all interfaces, after that we test the connectivity using ping between PC, as we know the the ping will not success. But it will success if we ping the gateway.
3. After that we create permanent virtual circuits (PVCs) and assign Data Link Connection Identifiers (DLCIs). This configuration creates two PVCs: one from R1 to R3(DLCI 103), and one from R3 to R1 (DLCI 301).
4. Then, we Configure the FR router as a Frame Relay switch. The **frame-relay switching** command enables Frame Relay switching globally on a router, allowing it to forward frames based on the incoming DLCI rather than an IP address. After that we Change the interface encapsulation type to Frame Relay. Like HDLC or PPP, Frame Relay is a data-link layer protocol that specifies the framing of Layer 2 traffic.
5. Then we Configure the FR router to forward incoming traffic on interface S0/0/0 with DLCI 103 to S0/0/1 with an output of DLCI of 301. Also we Configure Frame Relay on S0/0/1 interface to forward incoming traffic on interface S0/0/0 with DLCI 301 to S0/0/0 with an output of DLCI of 103.
6. After all those configuration we Verify Frame Relay using show **frame-relay pvc** command. As shown in figures 2,3 and 4.

```
Router#show frame-relay pvc

PVC Statistics for interface Serial0/0/0 (Frame Relay DCE)

              Active     Inactive      Deleted       Static
  Local          0           0            0             0
  Switched       0           1            0             0
  Unused         0           0            0             0

DLCI = 103, DLCI USAGE = SWITCHED, PVC STATUS = INACTIVE, INTERFACE = Serial0/0/
```

Figure 2: Verify Frame Relay configuration

```
PVC Statistics for interface Serial0/0/1 (Frame Relay DCE)
```

Figure 3: Verify Frame Relay configuration

```
DLCI = 301, DLCI USAGE = SWITCHED, PVC STATUS = INACTIVE, INTERFACE = Serial0/0/1

  input pkts 0             output pkts 0            in bytes 0
  out bytes 0              dropped pkts 0           in pkts dropped 0
  out pkts dropped 0                 out bytes dropped 0
  in FECN pkts 0           in BECN pkts 0           out FECN pkts 0
  out BECN pkts 0          in DE pkts 0             out DE pkts 0
  out bcast pkts 0         out bcast bytes 0
  30 second input rate 0 bits/sec, 0 packets/sec
  30 second output rate 0 bits/sec, 0 packets/sec
  switched pkts 0
  Detailed packet drop counters:
  no out intf 0            out intf down 0          no out PVC 0
  in PVC down 0            out PVC down 0           pkt too big 0
  shaping Q full 0         pkt above DE 0           policing drop 0
  connected to interface Serial0/0/0 103
  pvc create time 00:01:31, last time pvc status changed 00:01:31
```

Figure 4: Verify Frame Relay configuration

7. We have issued the show frame-relay route command, which is the layer 2 path that frame relay traffic takes through the network. As shown in figure 5. As we conclude that the status is inactive because those DLC is not mapped yet for IP.

```
Router# show frame-relay route
Input Intf      Input Dlci      Output Intf     Output Dlci     Status
Serial0/0/0     103             Serial0/0/1     301             inactive
Serial0/0/1     301             Serial0/0/0     103             inactive
Router#
```

Figure 5:

3.2 Configure Basic Frame Relay

On the R1 and R3 routers, we configured Frame Relay. After the frame relay is configured, and we enabled the EIGRP routing protocol to provide it end-to-end communication .

- Step 1: : Configure R1 for Frame Relay

1. Inverse ARP allows distant ends of a Frame Relay link to discover each other dynamically, and provides a dynamic method of mapping IP addresses to DLCIs. Although Inverse ARP is useful, it is not always reliable. The best practice is to map IP addresses to DLCIs statically and disable Inverse ARP.
2. So we Change the encapsulation on S0/0/0 to Frame Relay. And we use the no frame-relay inverse-arp command to disable Inverse ARP.
3. Then we Mapped the ip in both routers, we use the frame-relay map command to map an IP address to a DLCI statically. In addition to mapping an IP to a DLCI, Cisco IOS software allows several other Layer 3 protocol addresses to be mapped. In the following command, the broadcast keyword sends any multicast or broadcast traffic destined for this link over the DLCI. Most routing protocols require the broadcast keyword to function properly over Frame Relay. You can use the broadcast keyword on multiple DLCIs on the same interface. The traffic is replicated to all PVCs.
4. **Question:**Why is the no shutdown command used after the no frame-relay inverse-arp command? If we entered the no shutdown command first, Inverse ARP may cause Frame relay to learn Layer

2 to Layer 3 mappings that we may not want it to learn. by turning off Frame relay Inverse ARP before issuing the no shutdown command, we ensured that only the statically mapped connections that we wanted were part of the Frame relay maps.

- Step 3: : Verify that Frame Relay is active.

5. We ping from R1 to R3, as shown in figure 6. Then we ping from R3 to R1, as shown in figure 7.

```
Router#ping 10.1.1.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/4 ms
Router#
```

Figure 6: PING form R1 to R3

```
Router#ping 10.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/4 ms
Router#
```

Figure 7: PING from R3 to R1

6. Then, We have implemented show frame-relay PVC command to display PVC status information on R1 and R3. As shown in figure 8.
7. After that, We issued the show frame-relay route command on FR to check the status of the Frame Relay map formats. As figure 9 show.

```
Router#show frame-relay pvc

PVC Statistics for interface Serial0/0/0 (Frame Relay DTE)

              Active     Inactive      Deleted       Static
  Local          1           0             0            0
  Switched       0           0             0            0
  Unused         0           0             0            0

DLCI = 103, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0/

  input pkts 10            output pkts 10            in bytes 1040
  out bytes 1040           dropped pkts 0            in pkts dropped 0
  out pkts dropped 0                out bytes dropped 0
  in FECN pkts 0           in BECN pkts 0            out FECN pkts 0
  out BECN pkts 0          in DE pkts 0              out DE pkts 0
  out bcast pkts 0         out bcast bytes 0
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
  pvc create time 00:08:21, last time pvc status changed 00:04:36
Router#
```

```
Router# show frame-relay pvc

PVC Statistics for interface Serial0/0/1 (Frame Relay DTE)

              Active     Inactive      Deleted       Static
  Local          1           0             0            0
  Switched       0           0             0            0
  Unused         0           0             0            0

DLCI = 301, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0/0/1

  input pkts 10            output pkts 10            in bytes 1040
  out bytes 1040           dropped pkts 0            in pkts dropped 0
  out pkts dropped 0                out bytes dropped 0
  in FECN pkts 0           in BECN pkts 0            out FECN pkts 0
  out BECN pkts 0          in DE pkts 0              out DE pkts 0
  out bcast pkts 0         out bcast bytes 0
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
  pvc create time 00:07:16, last time pvc status changed 00:06:40
Router#
```

Figure 8: PVC status information

```
Router#show frame-relay route
Input Intf      Input Dlci      Output Intf     Output Dlci     Status
Serial0/0/0     103             Serial0/0/1     301             active
Serial0/0/1     301             Serial0/0/0     103             active
Router#
```

Figure 9: status of the Frame Relay map formats

8. Then, We issued show frame-relay map command on R1 and R3 to display a static line summary and dynamic assignments of Layer 3 addresses to DLCIs. As shown in figures 10 and 11.

```
Router#show frame-relay map
Serial0/0/0 (up): ip 10.1.1.1 dlci 103(0x67,0x1870), static,
              CISCO, status defined, active
Serial0/0/0 (up): ip 10.1.1.2 dlci 103(0x67,0x1870), static,
              broadcast,
              CISCO, status defined, active
Router#
```

Figure 10:

9. Also we configure EIGRP on R1 and R3 to make connectivity for those network. After the configuration we verify the connectivity by ping from pc-c to pc-a as shown in figure 12.

```
Router# show frame-relay map
Serial0/0/1 (up): ip 10.1.1.2 dlci 301(0x12D,0x48D0), static,
              CISCO, status defined, active
Serial0/0/1 (up): ip 10.1.1.1 dlci 301(0x12D,0x48D0), static,
              broadcast,
              CISCO, status defined, active
Router#
```

Figure 11:

```
C:\Users\NetLab>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:
Reply from 192.168.1.1: bytes=32 time=1ms TTL=255
Reply from 192.168.1.1: bytes=32 time=1ms TTL=255
Reply from 192.168.1.1: bytes=32 time=1ms TTL=255
Reply from 192.168.1.1: bytes=32 time=1ms TTL=255

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms

C:\Users\NetLab>
```

Figure 12: PING from pc-c to pc-a

3.3 Configure a Frame Relay Sub interface

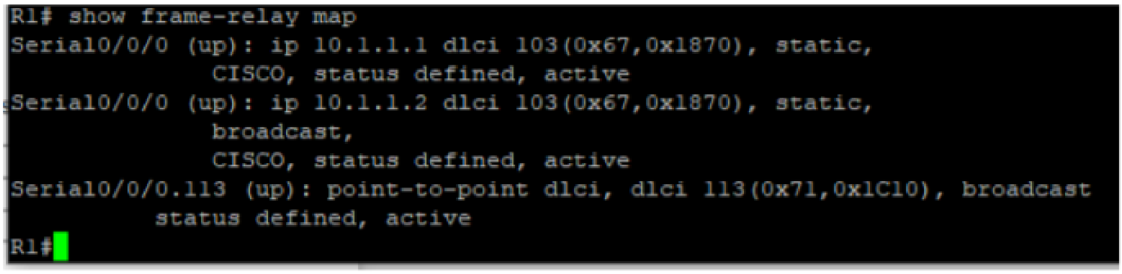
Frame Relay supports two types of subinterfaces: point-to-point and point-to-multipoint. Point-to-multipoint subinterfaces support non-broadcast multiaccess topologies. For example, a hub and spoke topology would use a point-to-multipoint subinterface. In this Part , we will create a point-to-point subinterface.

1. On the FR router, create new PVCs between R1 and R3. Then we create and configure a point-to-point subinterface on R1 and R3.
2. In first we create subinterface 113 as a point-to-point interface on R1. Then we create subinterface 311 as a point-to-point subinterface on R3.
3. So, we verify the connection using ping, and the ping work successfully.
4. Also, we We issued the show frame-relay route command on FR to check the status of the Frame Relay map formats, as shown in figure 13

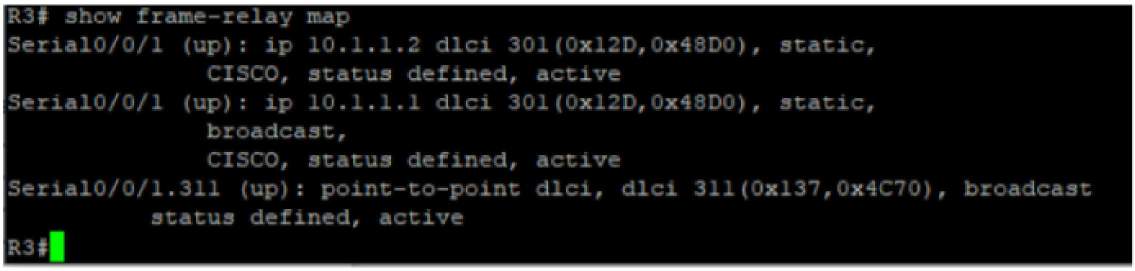
```
FR# show frame-relay route
Input Intf      Input Dlci      Output Intf     Output Dlci     Status
Serial0/0/0     103             Serial0/0/1     301             active
Serial0/0/0     113             Serial0/0/1     311             active
Serial0/0/1     301             Serial0/0/0     103             active
Serial0/0/1     311             Serial0/0/0     113             active
FR#
```

Figure 13:

5. At the end,We issued show frame-relay map command on R1 and R3 to check status of frame relay map formulations. As shown in figure 14



```
R1# show frame-relay map
Serial0/0/0 (up): ip 10.1.1.1 dlci 103(0x67,0x1870), static,
              CISCO, status defined, active
Serial0/0/0 (up): ip 10.1.1.2 dlci 103(0x67,0x1870), static,
              broadcast,
              CISCO, status defined, active
Serial0/0/0.113 (up): point-to-point dlci, dlci 113(0x71,0x1C10), broadcast
          status defined, active
R1#
```



```
R3# show frame-relay map
Serial0/0/1 (up): ip 10.1.1.2 dlci 301(0x12D,0x48D0), static,
              CISCO, status defined, active
Serial0/0/1 (up): ip 10.1.1.1 dlci 301(0x12D,0x48D0), static,
              broadcast,
              CISCO, status defined, active
Serial0/0/1.311 (up): point-to-point dlci, dlci 311(0x137,0x4C70), broadcast
          status defined, active
R3#
```

Figure 14:

4 Conclusion

At the end of the experiment, we achieved the required goals. We trained on how to configure frame relay protocol and learn how it works. Plus, we understand the frame migration diagram rather than using " arp -inverse".

5 References

https://en.wikipedia.org/wiki/Frame$_{R}elay$
https://ecomputernotes.com/computernetworkingnotes/network-technologies/frame-relay