IPv6 Module 9 – IPv6 Access

Objective: Explore three methods of providing IPv6 Access to End-User networks

Prerequisites: Module 1 (IPv4 & IPv6)

The following will be the common topology used for this lab.

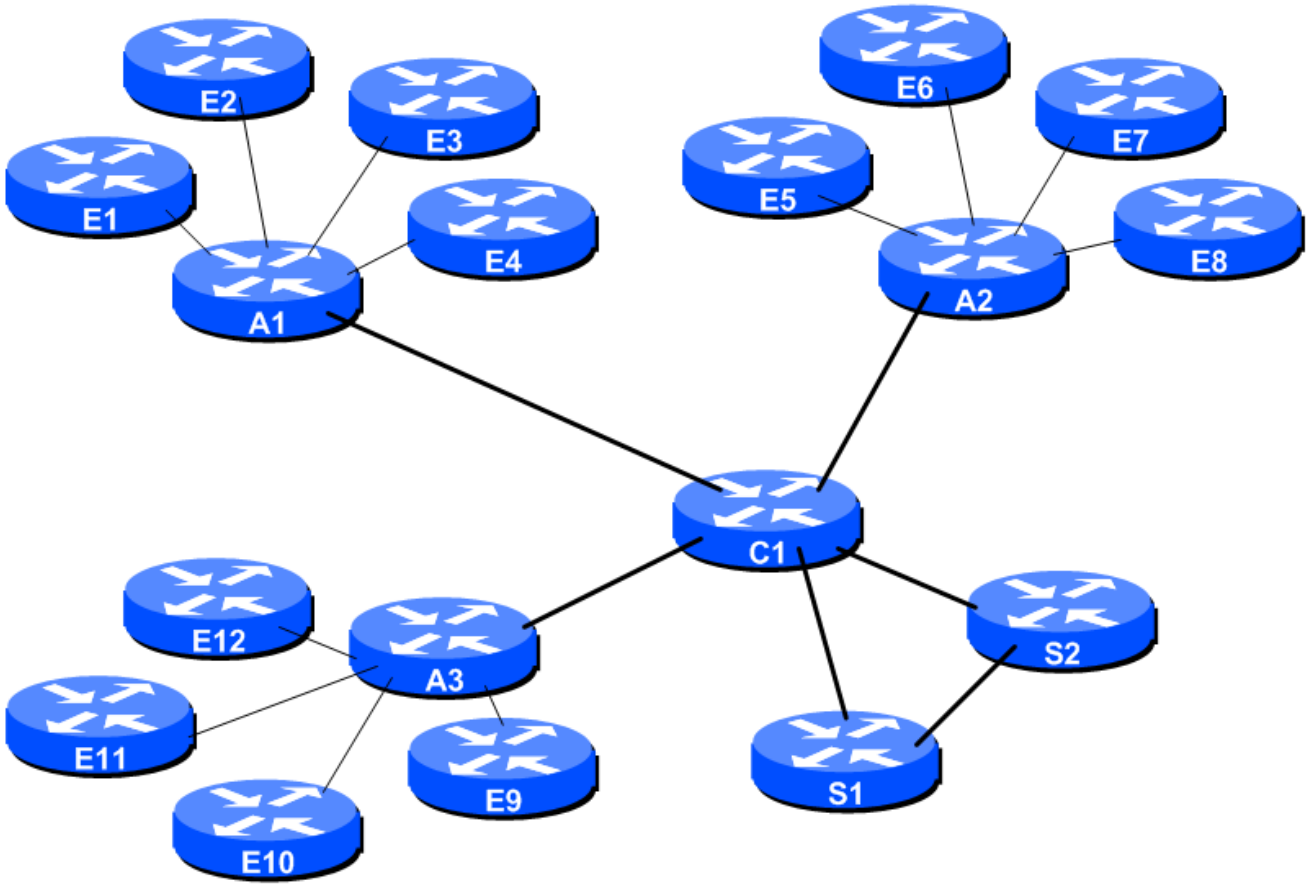

Figure 1 – ISP Lab Basic Configuration

Lab Notes

This lab explores three different technologies for providing IPv6 access to end-user networks. The three technologies are IPv6 in IP tunnels, 6rd, and native IPv6 with DHCPv6PD. Your instructors will have already explained to you during the lectures how each of these technologies work.

The routers used for this portion of the workshop must support IPv6 and they must support 6rd and DHCPv6PD. This is basically any IP Plus image from 15.1T onwards. Best to check the Cisco Feature Navigator www.cisco.com/go/fn to be absolutely sure which images set and platform supports IPv6.

The lab has one Core Router (C1), two Services Routers (S1 and S2) and three Access Routers (A1, A2 and A3). This ISP has 12 customers, donated by routers E1 through to E12. All the links are by Ethernet.

Lab Exercise

1. **Aim.** The purpose of this lab is to set up three types of IPv6 access connections each customer to the ISP backbone. The goal will be to see IPv6 connectivity from each customer to the two Services Routers in the lab. Connectivity will be tested by using ping and by traceroute on the routers in the lab.

2. **Preparation.** The instructors will have set up the lab with a clean and working IPv4 configuration. Right now it is possible for each customer router to use their IPv4 connectivity to all other routers in the lab. When setting up IPv6 configuration, do not forget the basic requirements for setting up IPv6 on a Cisco router – refer to Module 1 and the IPv6 presentations.

3. **Teamwork.** The classroom network has 12 customer routers, three access routers, one core router and two services. The class should divide into 12 teams, with each team operating one customer router. The lab instructors will operate the other routers and the class will not modify configurations on those unless directed otherwise.

4. **IPv4 Addressing.** The IPv4 address plan for the network is as follows:

Loopbacks:	
C1	**10.0.0.1**
S1	**10.0.0.2**
S2	**10.0.0.3**
A1	**10.0.0.4**
A2	**10.0.0.5**
A3	**10.0.0.6**

Backbone Point to Point Links:	
C1-S1	**10.0.1.0/30**
C1-S2	**10.0.1.4/30**
C1-A1	**10.0.1.8/30**
C1-A2	**10.0.1.12/30**
C1-A3	**10.0.1.16/30**
S1-S2	**10.0.2.0/2**

Customer Point to Point Links:	
A1-E1	**10.0.2.0/30**
A1-E2	**10.0.2.4/30**
A1-E3	**10.0.2.8/30**
A1-E4	**10.0.2.12/30**
A2-E5	**10.0.2.16/30**
A2-E6	**10.0.2.20/30**
A2-E7	**10.0.2.24/30**
A2-E8	**10.0.2.28/30**
A3-E9	**10.0.2.32/30**
A3-E10	**10.0.2.36/30**
A3-E11	**10.0.2.40/30**
A3-E12	**10.0.2.44/30**

Address blocks per customer:	
E1	**10.0.11.0/24**
E2	**10.0.12.0/24**
E3	**10.0.13.0/24**
E4	**10.0.14.0/24**
E5	**10.0.15.0/24**
E6	**10.0.16.0/24**
E7	**10.0.17.0/24**
E8	**10.0.18.0/24**
E9	**10.0.19.0/24**
E10	**10.0.20.0/24**
E11	**10.0.21.0/24**
E12	**10.0.22.0/24**

All the routers in the network have been pre-configured with the above addressing. ISIS is running in the core network (between A1-3, C1 and S1/S2). For point to point links, the router listed on the left gets the lower address of the subnet, the router listed on the right gets the upper address of the subnet.

iBGP is also running, with C1 being the route reflector, and the other routers the clients. The ASN the network is using is 131076.

Note that the Customer routers are not part of the backbone – there is just static routing between those routers and their respective Access routers.

5. **IPv6 Addressing.** The network has also been preconfigured with IPv6 operating between C1, S1 and S2 only. The three access routers do not have IPv6 configured on them. The address plan for the three backbone routers with IPv6 is as follows:

Loopbacks:		**Point to point links:**	
C1	**2001:db8::1/128**	**C1-S1**	**2001:db8:0:1::/127**
S1	**2001:db8::2/128**	**C1-S2**	**2001:db8:0:2::/127**
S2	**2001:db8::3/128**	**S1-S2**	**2001:db8:1:0::/64**

Multi-topology ISIS is in use, and the above prefixes are transported between the three routers using ISIS. iBGP has also been configured for the IPv6 address family between these three routers.

SCENARIO 1 – IPv6 in IP Tunnel (6in4)

6. **IPv6-in-IP Tunnels.** The first scenario we want to cover is to implement a simple tunnel between the customer router and the core router so that the customers get IPv6 connectivity. The Access Routers do not have IPv6 enabled, representing legacy infrastructure which cannot run IPv6. Here IPv6 packets are encapsulated in IPv4 packets by the routers.

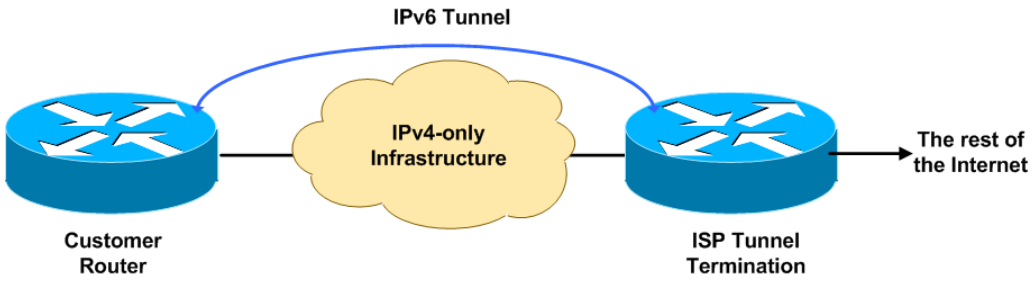

Figure 2 – IPv6 in IPv4 Tunnel

7. **Edge Router IPv6 Addressing.** Each edge router has a local LAN interface (FastEthernet 0/1) which has also been configured with an IPv6 address. This Ethernet has been connected to a switch. The following lists the address blocks assigned to each router. Note that the router has the :1 address.

E1	**2001:db8:11::/64**	**E7**	**2001:db8:17::/64**
E2	**2001:db8:12::/64**	**E8**	**2001:db8:18::/64**
E3	**2001:db8:13::/64**	**E9**	**2001:db8:19::/64**
E4	**2001:db8:14::/64**	**E10**	**2001:db8:1a::/64**
E5	**2001:db8:15::/64**	**E11**	**2001:db8:1b::/64**
E6	**2001:db8:16::/64**	**E12**	**2001:db8:1c::/64**

8. **Creating the Tunnel.** Each group should create a tunnel on their router, pointing to the Core Router in the ISP network. The tunnel should be sourced from the LAN interface of the Customer

Router, and destination of the Loopback of the Core Router. Here is an example for Customer Router E3:

```
interface Tunnel0
 tunnel source FastEthernet0/1
 tunnel mode ipv6ip
 tunnel destination 10.0.0.1
 tunnel path-mtu-discovery
!
```

9. **Tunnel IPv6 addressing.** We now need to set up an IPv6 address on the tunnel (we could just use the link-local address). The point to point link is addressed as a /127, from 2001:db8:3::/48 address block. We will reserve a /64 for each point-to-point link, but use a /127 network mask, as per industry best practice. Here is the addressing for each tunnel:

C1-E1	**2001:db8:3:1::/127**	**C1-E7**	**2001:db8:3:7::/127**
C1-E2	**2001:db8:3:2::/127**	**C1-E8**	**2001:db8:3:8::/127**
C1-E3	**2001:db8:3:3::/127**	**C1-E9**	**2001:db8:3:9::/127**
C1-E4	**2001:db8:3:4::/127**	**C1-E10**	**2001:db8:3:a::/127**
C1-E5	**2001:db8:3:5::/127**	**C1-E11**	**2001:db8:3:b::/127**
C1-E6	**2001:db8:3:6::/127**	**C1-E12**	**2001:db8:3:c::/127**

   The C1 router gets the 0 address, the Customer Router gets the 1 address. Which makes the example for Customer Router E3:

```
interface Tunnel0
 ipv6 address 2001:db8:3:3::1/127
 tunnel source FastEthernet0/1
 tunnel mode ipv6ip
 tunnel destination 10.0.0.1
 tunnel path-mtu-discovery
!
```

   Notice that the tunnel does not have any IPv4 addresses assigned to it – it is an IPv6 only tunnel, running over IPv4.

10. **Ping Test #1.** Once the tunnel has been created, try and ping the IPv6 address at the other end of the tunnel.

```
E3# ping 2001:db8:3:3::
```

    This should result in echo responses from the other end of the tunnel. If not, troubleshoot the problem.

11. **Default IPv6 Route.** To reach either S1 or S2 Services Routers, the Customer Router needs a default IPv6 route to be set up to point to the Tunnel. This will ensure that all non-local IPv6 traffic will be sent through to the tunnel for the Core Router to send onwards:

```
ipv6 route ::/0 Tunnel0
```

12. **Ping Test #2.** Now try and ping the IPv6 addresses of the S1 and S2 routers – they are 2001:db8:1::1 and 2001:db8:1::2 respectively. Run the ping so that it is sourced from the FastEthernet 0/1 interface of the Customer Router, for example:

```
E3# ping 2001:db8:3:3:: source FastEthernet0/1
```

If there are problems, use the following commands to help determine the problem:

```
show ipv6 route             : see if there is a route for the intended destination
show ipv6 interface brief   : see IPv6 status of the tunnel interface
```

13. **Aside: Core Router Configuration.** Here is the configuration of the Core Router for completeness – it shows how the tunnels are typically terminated on a service provider network infrastructure:

```
!
interface Tunnel1
 description IPv6 Tunnel to EDGE1
 ipv6 address 2001:DB8:3:1::/127
 tunnel source Loopback0
 tunnel mode ipv6ip
 tunnel destination 10.0.11.1
!
interface Tunnel2
 description IPv6 Tunnel to EDGE2
 ipv6 address 2001:DB8:3:2::/127
 tunnel source Loopback0
 tunnel mode ipv6ip
 tunnel destination 10.0.12.1
!
interface Tunnel3
 description IPv6 Tunnel to EDGE3
 ipv6 address 2001:DB8:3:3::/127
 tunnel source Loopback0
 tunnel mode ipv6ip
 tunnel destination 10.0.13.1
!
interface Tunnel4
 description IPv6 Tunnel to EDGE4
 ipv6 address 2001:DB8:3:4::/127
 tunnel source Loopback0
 tunnel mode ipv6ip
 tunnel destination 10.0.14.1
!
interface Tunnel5
 description IPv6 Tunnel to EDGE5
 ipv6 address 2001:DB8:3:5::/127
 tunnel source Loopback0
 tunnel mode ipv6ip
 tunnel destination 10.0.15.1
!
interface Tunnel6
 description IPv6 Tunnel to EDGE6
 ipv6 address 2001:DB8:3:6::/127
 tunnel source Loopback0
 tunnel mode ipv6ip
 tunnel destination 10.0.16.1
!
interface Tunnel7
 description IPv6 Tunnel to EDGE7
 ipv6 address 2001:DB8:3:7::/127
 tunnel source Loopback0
 tunnel mode ipv6ip
 tunnel destination 10.0.17.1
!
interface Tunnel8
 description IPv6 Tunnel to EDGE8
 ipv6 address 2001:DB8:3:8::/127
 tunnel source Loopback0
 tunnel mode ipv6ip
 tunnel destination 10.0.18.1
!
interface Tunnel9
 description IPv6 Tunnel to EDGE9
 ipv6 address 2001:DB8:3:9::/127
 tunnel source Loopback0
 tunnel mode ipv6ip
 tunnel destination 10.0.19.1
!
interface Tunnel10
 description IPv6 Tunnel to EDGE10
 ipv6 address 2001:DB8:3:A::/127
 tunnel source Loopback0
 tunnel mode ipv6ip
 tunnel destination 10.0.20.1
!
interface Tunnel11
 description IPv6 Tunnel to EDGE11
 ipv6 address 2001:DB8:3:B::/127
 tunnel source Loopback0
 tunnel mode ipv6ip
 tunnel destination 10.0.21.1
!
interface Tunnel12
 description IPv6 Tunnel to EDGE12
 ipv6 address 2001:DB8:3:C::/127
 tunnel source Loopback0
 tunnel mode ipv6ip
 tunnel destination 10.0.22.1
!
ipv6 route 2001:DB8:11::/64 Tunnel1
ipv6 route 2001:DB8:12::/64 Tunnel2
ipv6 route 2001:DB8:13::/64 Tunnel3
ipv6 route 2001:DB8:14::/64 Tunnel4
ipv6 route 2001:DB8:15::/64 Tunnel5
ipv6 route 2001:DB8:16::/64 Tunnel6
ipv6 route 2001:DB8:17::/64 Tunnel7
ipv6 route 2001:DB8:18::/64 Tunnel8
ipv6 route 2001:DB8:19::/64 Tunnel9
ipv6 route 2001:DB8:1A::/64 Tunnel10
ipv6 route 2001:DB8:1B::/64 Tunnel11
ipv6 route 2001:DB8:1C::/64 Tunnel12
!
```

Checkpoint #1: *call lab assistant to verify the connectivity. Save the configuration as it is on the router – use a separate worksheet, or the workspace at the end of this Module.*

STOP AND WAIT HERE

SCENARIO 2 – 6rd

14. **6rd Tunnels.** The second scenario looks at creating 6rd (IPv6 Rapid Deploy) IPv6 tunnels over an IPv4 infrastructure. The first scenario relies on a lot of manual configuration at both the service provider and the end user. It is quite often too much to ask an end user to set up a complicated configuration – 6rd provides a lot more automation on most home/end-user routers.

    In this case the existing IPv4 network created in the IPv4 version of the lab remains, but the lab instructors will reset the lab router configuration to be IPv4 only as it was at the start of the previous scenario.

15. **End user configuration.** The lab instructors will have explained how 6rd worked during the presentations. But suffice to say, the router configuration on the CPE device is the same for all devices – there is no custom configuration as we saw in Scenario One with the static tunnels.

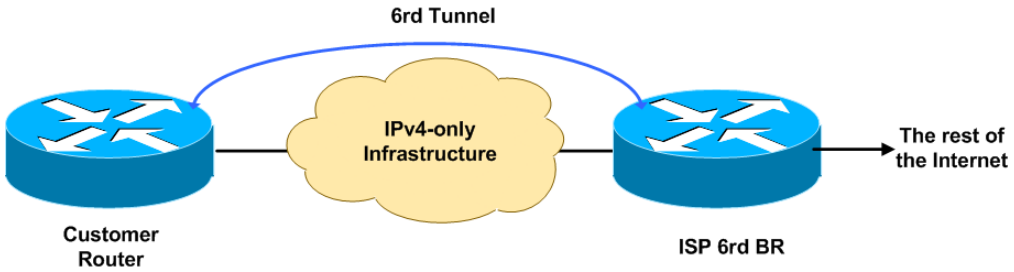

Figure 3 – 6rd Tunnel

16. **6rd Addressing.** We will be using the address block 2001:db8:d00::/40 for 6rd. All end users will be automatically assigned a /48 out of this /40 – the final 8 bits of the address will come from the IPv4 address of the point to point link connecting the Customer Router to the Access Router. For example, for E5, the point to point link address is 10.0.2.18, so we will get 6rd to use the final 8 bits of this IPv4 address to generate the unique /48 for this end-site. 18 in decimal is 12 in hexadecimal, so the /48 for the E5 end-site is 2001:db8:d12::/48. The 6rd BR router will be the core router in the ISP network – the lab instructors have set up the Core Router with the IP address 10.0.2.254 and the 6rd tunnels will terminate here.

17. **Create the 6rd Tunnel.** Each router team will create a tunnel on their customer router. Here is a configuration example for E5:

```
interface Tunnel0
 ipv6 enable
 tunnel source FastEthernet0/0
 tunnel mode ipv6ip 6rd
 tunnel 6rd ipv4 prefix-len 24
 tunnel 6rd prefix 2001:DB8:D00::/40
 tunnel 6rd br 10.0.2.254
```

```
!
```

This example applies to all routers – notice that there is nothing specific to the router in the example above. To explain the configuration:

- `ipv6 enable`: enables IPv6 on the tunnel interface, but only uses link-local addressing. Global unicast addressing is not needed.
- `tunnel source FastEthernet0/0`: the 6rd tunnel uses the point to point link to the ISP as the source – when creating the 6rd address block, it uses part of this IPv4 address.
- `tunnel mode ipv6ip 6rd`: specifies that this is a 6rd tunnel.
- `tunnel 6rd ipv4 prefix-len 24`: drop the first 24 bits, using only the final 8 bits for the 6rd address.
- `tunnel 6rd prefix 2001:db8:d00::/40`: the address block the ISP uses for 6rd – the final 8 bits of the IPv4 address will make this up to the /48.
- `tunnel 6rd br 10.0.2.254`: specifies the address of the 6rd Border Router.

18. **IPv6 Static Routes.** To complete the configuration, each team now needs to add static routes so that IPv6 traffic goes over the 6rd tunnel to all destinations. Again, all edge/customer routers need the following:

```
ipv6 route 2001:DB8:D00::/40 Tunnel0
ipv6 route ::/0 2001:db8:DFE::
```

The first static route points the entire /40 address block to the 6rd tunnel.

The second static route defaults all IPv6 traffic to the IPv6 6rd address of the 6rd BR router (using the same mechanism, namely last 8 bits of the IPv4 address (254) converted to hexadecimal (FE)). It has to be routed towards the 6rd BR, not just the Tunnel interface.

19. **Configuring IPv6 prefixes on local interfaces.** We will use a feature in Cisco IOS called "general-prefix". This allows us to refer to learned addresses (by 6rd, DHCPv6PD etc) without configuring specific addresses on each prefix. The IOS command is very simple:

```
ipv6 general-prefix 6RDLAB 6rd Tunnel0
```

Which says: what ever prefix we learn by 6rd from Tunnel0 (ie the /48 which 6rd creates), we will assign the name "6RDLAB" – and when we need to assign addresses to other interfaces on the router, we can refer to them using this name.

20. **Assigning a subnet to a local interface.** Now we have the general prefix configured, we can use it to apply address to local interfaces. For example, FastEthernet0/1 on the edge/customer routers is a local LAN, and would get this configuration:

```
interface FastEthernet0/1
 description Local LAN
 ipv6 address 6RDLAB ::1:0:0:0:1/64
!
```

And the resulting IPv6 addresses for example, for router E10, are:

```
EDGE10#show ipv6 interface brief
FastEthernet0/0        [up/up]
```

```
    unassigned
FastEthernet0/1        [up/up]
    FE80::C80F:6FF:FE1C:6
    2001:DB8:D26:1::1
Tunnel0                [up/up]
    FE80::A00:226
EDGE10#
```

Note the FastEthernet0/1, which is the local LAN interface for the customer router, now has the 6RD generated subnet.

21. Ping Test #1. Ping the IPv6 address of the 6BR router mentioned above. If there are problems, use the following commands to help determine the problem:

`show ipv6 route`	: see if there is a route for the intended destination
`show ipv6 interface brief`	: see IPv6 status of the tunnel interface

22. Ping Test #2. Ping the IPv6 addresses of the two services routers mentioned in the previous scenario – remember that S1 was on 2001:db8:1::1 and S2 was on 2001:db8:1::2. Again, if there are problems or no responses, troubleshoot what might be wrong. Remember to source the pings from the local LAN address using the extended ping address.

23. Traceroute Test. Once another router team has completed their configuration, ask them for their local LAN IP address, as assigned by the 6rd general prefix. Run a traceroute from your router to their router. What do you see? Here is an example (from E5 to E10):

```
EDGE5>trace 2001:db8:d26:1::1
Type escape sequence to abort.
Tracing the route to 2001:DB8:D26:1::1

  1 2001:DB8:D26:1::1 96 msec 92 msec 68 msec
EDGE5>
```

Why does the trace not go via the 6rd BR as it did in the previous scenario?

24. Aside: 6rd configuration on the 6BR router. This final section is for information only, and gives the 6rd configuration needed on the 6BR router.

```
interface Loopback1
 description 6RD Termination
 ip address 10.0.2.254 255.255.255.255
!
interface Tunnel0
 ipv6 enable
 tunnel source Loopback1
 tunnel mode ipv6ip 6rd
 tunnel 6rd ipv4 prefix-len 24
 tunnel 6rd prefix 2001:DB8:D00::/40
!
ipv6 route 2001:DB8:D00::/40 Tunnel0
```

This configuration creates a dedicated Loopback interface for the tunnel termination/source point, with the termination address. As with the customer router, the interface Tunnel0 is created, in IPv6 over IPv4 mode, but specifying the use of 6rd. And the address block is specified, along with the number of bits to be dropped from the IPv4 prefix configured on the Loopback1 interface.

Also, compare this configuration with the configuration of the core router in the previous scenario, where each static tunnel was configured manually. Quite laborious to set up and maintain compared with using 6rd – which is why 6rd is preferred now over static tunnels in larger scale deployments where the ISP infrastructure is not yet wholly IPv6 capable.

Checkpoint #2: *call lab assistant to verify the connectivity. Save the configuration as it is on the router – use a separate worksheet, or the workspace at the end of this Module.*

STOP AND WAIT HERE

SCENARIO 3 – Native IPv6 (Dual-Stack) with DHCPv6-PD

25. Native Dual Stack. The third scenario represents the final stage of an ISP's transition to provide native IPv6 to its end-users. We could simply turn on IPv6 on the Access Routers and on the link from Access to the Customer Routers; while that represents what happens in the Enterprise world, with a stack dual-stack configuration that resembles the lab exercises we have completed prior to this one, it is more instructive to look at a situation typically used by ISPs who are providing native dual-stack IPv4/IPv6 to existing home/small office users over broadband infrastructure.

The IPv4 configuration remains unchanged from the previous scenarios – but now, the IPv6 addressing will be set up to receive an address block distributed by DHCPv6 Prefix Delegation from the ISP's Access Router. And Cisco IOS's general prefix mechanism will be used as in the previous example to do address assignment on interfaces on the Customer Router.

26. Activating IPv6 on the Access Routers. The lab instructors will have enabled IPv6 on the three access routers in the network. This will include adding IPv6 to the point-to-point link to the Core Router, adding the routers to ISIS, and adding the routers to the IPv6 iBGP mesh. The configuration of the Access Routers will be shared at the completion of the workshop lab.

27. Activating IPv6 on the link to the ISP – Step 1. The lab instructors will now activate IPv6 on the links from the three Access Routers to the 12 Edge/Customer Routers. The link addressing will only be link-local (no global unicast addressing is required), and DHCPv6 Prefix-Delegation will be configured to distribute an address block to each customer. This is the address block to be used by the DHCPv6 PD on each Access Router:

A1: 2001:db8:c00::/40 distributing /48s
A2: 2001:db8:b00::/40 distributing /48s
A3: 2001:db8:a00::/40 distributing /48s

The resulting IOS configuration, for example for the A1 router, is as follows:

```
ipv6 dhcp pool DHCPv6
 prefix-delegation pool dhcpv6-pool1 lifetime 1800 600
 dns-server 2001:DB8:FFFF::1
 domain-name lab.net
!
ipv6 local pool dhcpv6-pool1 2001:DB8:C00::/40 48
```

which creates a unique pool called “DHCPv6”, set up with prefix-delegation using the block 2001:DB8:C00::/40, distributing /48s out of the block. The DNS server and the domain-name are fictitious, but are provided to round out the exercise.

28. **Activating IPv6 on the link to the ISP – Step 2.** The lab instructors will now activate DHCPv6 on each Access Router interface pointing towards the customers. Here is an example of the interface configuration which would go on to Router A1:

```
interface Ethernet1/0
 description Link to Customer1
 ip address 10.0.2.1 255.255.255.252
 ipv6 enable
 ipv6 dhcp server DHCPv6
!
```

The final two lines are new, enabling IPv6 on the interface (link-local only, no global unicast address), and distributing addresses by DHCP according to the definition of the DHCPv6 pool mentioned in the previous step.

29. **Activating IPv6 on the link to the ISP – Step 3.** The final action is for the lab groups, and that is to activate native IPv6 on the link to the ISP Access Routers. Here is a configuration example, as would be used on router E4:

```
interface FastEthernet0/0
 description Link to Access1
 ip address 10.0.2.14 255.255.255.252
 ipv6 address autoconfig default
 ipv6 dhcp client pd PDLAB rapid-commit
!
```

The new commands are in the last two lines:

a) The first command says to get the IPv6 address by autoconfiguration – this does two things, the first enabling IPv6 on the interface, and the second by setting IPv6 address depending on what is configured on the other end of the link. It will result in just a link local address being used. The **`default`** keyword means that the router will install a default route on this interface to whatever the destination is at the other end of the link. This saves the requirement of configuring a static default route else where in the configuration.
b) The second command says that the interface will operate in DHCPv6 client mode, using Prefix Delegation to learn the address block from the neighbouring router, and it will save the address block in PDLAB. **`rapid-commit`** simply speeds up the DHCPv6 process between client and server (2 messages are used rather than 4).

30. **Assigning a subnet to a local interface.** Now we have the general prefix configured via DHCPv6 PD, we can use it to apply address to local interfaces. For example, FastEthernet0/1 on the edge/customer routers is a local LAN, and would get this configuration:

```
interface FastEthernet0/1
 description Local LAN
 ipv6 address PDLAB ::1:0:0:0:1/64
!
```

And the resulting IPv6 addresses for example, for router E10, are:

```
EDGE4#show ipv6 interface brief
FastEthernet0/0        [up/up]
    FE80::C809:3FF:FE24:8
FastEthernet0/1        [up/up]
    FE80::C809:3FF:FE24:6
    2001:DB8:C03:1::1
EDGE4#
```

Note the FastEthernet0/1, which is the local LAN interface for the customer router, now has the DHCPv6 PD sourced subnet. (DHCPv6 hands out /48s in sequence from the first one, and in this case, EDGE4 router got the fourth address in the pool.)

31. **Ping Test #1.** Ping the IPv6 addresses of the two services routers mentioned in the previous scenario – remember that S1 was on 2001:db8:1::1 and S2 was on 2001:db8:1::2. Again, if there are problems or no responses, troubleshoot what might be wrong. Remember to source the pings from the local LAN address using the extended ping address.

32. **Summary.** This lab module has demonstrated three ways of providing IPv6 access to end-users. The first two cover situations where the ISP may not have deployed IPv6 over their entire network as yet, whereas the final scenario covers the situation where the ISP is fully dual-stack across their backbone network.

Checkpoint #3: *call lab assistant to verify the connectivity. Save the configuration as it is on the router – use a separate worksheet, or the workspace at the end of this Module.*