IPv6 Security Lab - 6rd

Exploring 6 “Rapid Deploy"

6rd is a technology which allows network operators to deploy IPv6 to end users without having to upgrade the entire infrastructure between them and their customer to support IPv6. This is typically the case where the ISP has old DSLAMs which don’t support IPv6 framing, or are leasing a third party’s access network which perhaps also is not quite IPv6 ready.

In this lab we will configure a 6rd tunnel between our Customer router and our Border router. To do this we will disable IPv6 on our core network, leaving only IPv6 operational on our Border router.

Disabling IPv6 on the core router

The first step is to disable IPv6 on the core router. Be sure to keep a copy of the configuration first!

The simplest way to do this is simply to turn off IPv6 Unicast Routing, like this:

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ no ipv6 unicast-routing ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

This will remove all the IPv6 routing protocol configuration on the core router, including OSPF/IS-IS and BGP. The IPv4 configuration will still remain, and the IPv4 routing will still function. You will notice that IPv6 addressing will remain on the router interfaces, and configuration like IPv6 packet filters etc.

Confirm that IPv4 routing still works.

And confirm that IPv6 routing no longer works - you should no longer have any IPv6 connectivity between the Border, Peering and Access routers.

Setting up 6rd

Background

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 per end user configuration which makes 6rd a very simple to deploy technology for a service provider, as all their CPE devices can have the same configuration prior to shipping to the end-user.

What we will set up is something similar to the diagram below.

![](diagrams/6rd-diagram.png)

The Customer Router in the diagram is our Customer Router in our network. And the ISP 6rd BR capability will be configured on our autonomous system's Border Router.

Addressing

We will be using the address block 2001:DB8:X:C000::/52 for 6rd. All end users will be automatically assigned a /60 out of this /52 – the final 8 bits of the address will come from the last quad of the IPv4 address of the point-to-point link connecting the Customer Router to the Access Router.

This is how it works: the point-to-point link address on the Customer Router is 100.68.X.**34**, so we will configure 6rd to use the final **8** bits of this IPv4 address to generate the unique /60 for this end-site. **34** in decimal is **22** in hexadecimal. Which makes the /60 for the end-site is 2001:DB8:X:C**22**0::/60[^1].

[^1]: Real world examples are not likely to be as constrained as our lab network. If end-users were to get /56s by 6rd, we could set aside a /40 pool, and determine the IPv6 address for 6rd from that pool. For example, if 2001:0DB8:0D00::/40 is used as the pool, then we could create the /56 the end user gets from the final 16 bits of the IPv4 point to point link address. If this was 10.0.10.18, then 10 in decimal becomes **0A**, and 18 in decimal becomes **12** in hexadecimal, making the unique IPv6 address block for the end-site 2001:DB8:0D**0A**:**12**00::/56.

Creating the Tunnel

Now create the tunnel on the Customer Router. The configuration will look like this:

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~ interface Tunnel0 ipv6 enable tunnel source FastEthernet0/1 tunnel
mode ipv6ip 6rd tunnel 6rd ipv4 prefix-len 24 tunnel 6rd prefix 2001:DB8:X:C000::/52 tunnel 6rd br
100.68.X.1 !
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~
```

To explain this configuration:

`ipv6 enable` - enables IPv6 on the tunnel interface, but only uses link-local addressing. Global unicast addressing is not needed.

`tunnel source FastEthernet0/1` - the 6rd tunnel uses the point-to-point link address to the Access Router 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:X:C000::/52` - the address block the ISP uses for 6rd – the final 8 bits of the IPv4 address will make this up to the /60.

`tunnel 6rd br 100.68.X.1` - specifies the address of the 6rd Border Router (we are using the Loopback address of the Border Router).

Setting up the IPv6 Routes

To complete the configuration, we now add static routes so that IPv6 traffic goes over the 6rd tunnel to all destinations. The Customer router needs the following:

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~ ipv6 route 2001:DB8:X:C000::/52 Tunnel0 ipv6 route ::/0
2001:DB8:X:C010::
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~
```

The first static route points the entire /52 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 (1) converted to hexadecimal (01)). It has to be routed towards the 6rd BR, not just the Tunnel interface.

Configuring 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 /60 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.

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

And the resulting IPv6 addresses for the Customer router would look like this:

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~ Cust5# show ipv6 interface brief -to-be-done-
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~
```

Configuring the 6BR

Now we need to set up the 6BR functionality on the Border router.

As with the customer router, we need to create the tunnel end point:

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ interface Tunnel0  ipv6 enable  tunnel source Loopback0  tunnel mode ipv6ip 6rd  tunnel 6rd ipv4 prefix-len 24  tunnel 6rd prefix 2001:DB8:X:C000::/52 ! ipv6 route 2001:DB8:X:C000::/52 Tunnel0
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

The Loopback0 interface already exists - we created that for the iBGP session anchor point earlier on in the workshop.

Explaining the configuration:

`ipv6 enable` - enables IPv6 on the tunnel interface, but only uses link-local addressing. Global unicast addressing is not needed.

`tunnel source Loopback0` - the 6rd tunnel uses the loopback address of the Border router as the source.

`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:X:C000::/52` - the address block the ISP uses for 6rd – the final 8 bits of the IPv4 address will make this up to the /60.

Testing

With the 6BR and the Customer router now set up to support 6rd, trying some IPv4 and IPv6 traceroutes from the Customer router to the backbone Transit providers and out to the Internet.

What do you see?

From:
https://bgp4all.com/pfs/ - **Philip Smith's Internet Development Site**

Permanent link:
https://bgp4all.com/pfs/training/sanog32/a-6rd?rev=1532341111

Last update: **2018/07/23 10:18**