Prefix Delegation

I'd recommend reading about Prefix Delegation to get a better understanding of it but the gist is that using DHCPv6 it is possible to request a "prefix" where any IPv6 address starting with that will be routed to the router. Then the router can use that to configure clients on the network to each have a unique address instead of the router only one (as in IPv4) and having to share it using a hack like masquerading.

Install A Client

There are a few different DHCPv6 clients you can use that support *Prefix Delegation* but I decided to go with *wide-dhcpv6-client*. I also tried *dhcpcd* but found the configuration syntax to be a little uglier.

```
# apt install wide-dhcpv6-client
```

The config below is doing a few different things that I'll list but you can read about all the possible dhcp6c.conf configuration.

```
# /etc/wide-dhcpv6/dhcp6c.conf
+
+ interface eth0 {
+ #  send rapid-commit;
+   send ia-na 0;
+   send ia-pd 1;
+ };
+
+ id-assoc na 0 {
+
+ };
+
+ id-assoc pd 1 {
+   prefix ::/60 infinity;
+
+   prefix-interface eth1 {
+     sla-id 0;
```

```
+     sla-len 4;
+     ifid 1;
+   };
+
+   prefix-interface eth1.8 {
+     sla-id 1;
+     sla-len 4;
+     ifid 1;
+   };
+
+   prefix-interface eth1.9 {
+     sla-id 2;
+     sla-len 4;
+     ifid 1;
+   };
+ };
```

- **Lines 3-7** use `eth0` to request a "normal address" with `ia-na` and a delegrated prefix range with `ia-pd`
- **Lines 9-11** are required and correspond to the "normal addresss" I asked for, but there is no extra configuration needed
- **Lines 13 and 14** are the start of the prefix delegation block and I am specifying I want a prefix that gives me a subnet of `/60`. I know *Comcast* will give me a `/60` which is 295,147,905,179,352,825,856 (two hundred ninety five quintillion, one hundred forty seven quadrillion, nine hundred five trillion, one hundred seventy nine billion, three hundred fifty two million, eight hundred twenty five thousand, eight hundred fifty six) so that should be more than enough.
- **Lines 16-20** and the other similar blocks just specify what slice of the prefix I was delegated that I want applied to each interface. The lines `sla-id` (Site-Level Aggregation identifier) is just an index to a what is basically an IPv4 subnet, `sla-len` is the size of that subnet (I requested a /60 so if our SLA is /4 then I end up with /64 which is ideal for an IPv6 subnet), and `ifid` just defines that I want the first address available in our subnet assigned to our interface. So if the IPv6 addresses assigned to this interface was `2601:1833:a3a:100::/64` the interface would have `2601:1833:a3a:100::1/64` assigned to it.

The next step was just to enable and run the service.

```
# systemctl enable --now wide-dhcpv6-client
```

Then I was able to verify that I had publicaly accessible IPv6 addresses.

```
# ip -6 addr
1: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 state UP qlen 1000
```

```
    inet6 2001:6020:ae3:1022:a4d3:f031:fb7e:e629/128 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::2b0:c9ff:fe79:cd77/64 scope link
       valid_lft forever preferred_lft forever
2: eth1@if9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 state UP qlen 1000
    inet6 2601:1833:a3a:100::1/64 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::d45a:67ff:fec6:6688/64 scope link
       valid_lft forever preferred_lft forever
3: eth1.8@eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 state UP qlen 1000
    inet6 2601:1833:a3a:101::1/64 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::d45a:67ff:fec6:6688/64 scope link
4: eth1.9@eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 state UP qlen 1000
    inet6 2601:1833:a3a:102::1/64 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::d45a:67ff:fec6:6688/64 scope link
       valid_lft forever preferred_lft forever
...
```

Exactly like I hoped, I can see that using `ia-nd` to request a "normal address" for `eth0` resulted in `2001:6020:ae3:1022:a4d3:f031:fb7e:e629/128` being assigned as my routers public IPv6 address. It also looks like the prefix delegation worked since `eth1`, `eth1.8`, `eth1.9` all have the same prefix with incrementing SLA identifiers that you can see represented by `100`, `101`, `102` in their addresses.

Revision #5
Created 2021-04-01 02:39:30 UTC by dustin@swigg.net
Updated 2021-04-08 13:00:49 UTC by dustin@swigg.net