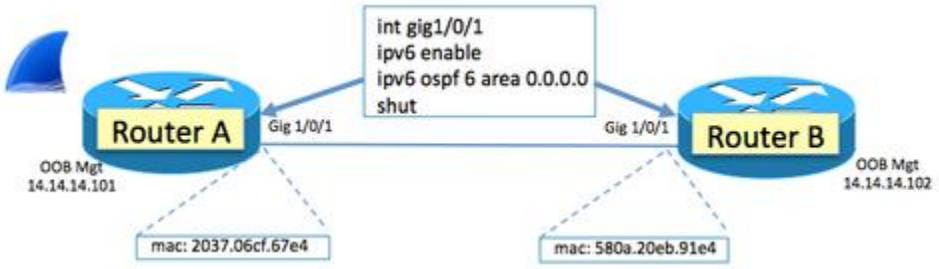
Understanding IPv6: Link-Local 'Magic'

Denise Fishburne performs a little IPv6 sleight of hand in the second post in her series on IPv6.

For those of you new to IPv6, what I am about to show you is going to look a lot like a magic trick. I'm going to bring up an IPv6 IGP neighbor relationship (OSPFv3) between two routers. This doesn't sound like a magic trick, I know. But what if I told you I am going to do this *without* putting any IPv6 addresses into the configurations of either routers?

Like any true magician, I must start my magic act with letting you know I have nothing up my sleeves. So let's review the facts:

- IPv6 unicast routing is globally enabled on both routers
- IPv6 OSPFv3 is enabled via the one global command, "ipv6 router ospf 6"
- Each router has an interface in an out-of-band management network (OOB mgt.) in the subnet 14.14.14.0/24.
- RouterA is 14.14.14.101 and RouterB is 14.14.14.102 in this OOB management network
- The IPv4 addresses for the OOB management interfaces are the only IP addresses in the configurations
- Gig1/0/1 on both routers only has only two IPv6 commands on it, as shown below
- Router A is monitoring the gig1/0/1 interface and sending the traffic to a Spirent TestCenter port for capture



The interfaces are currently, as I mentioned, administratively shut down. So why look at the output to ***show int gig1/0/1*** on Router A? Hint, hint: The MAC address of Router A is highlighted.

```
RouterA#sh int Gig1/0/1
Gig1/0/1 is admin down, line protocol is down (disabled)
  H/W is Gigabit Ethernet, address is 2037.06cf.67e4 (bia 2037.06cf.67e4)
```

Let's "no shut" both Router A's interface and Router B's interface. Wave the magic wand and say a few magic words and….

VOILA! OSPFv3 neighbor!

```
RouterA#sh ipv6 ospf neighbor

            OSPFv3 Router with ID (14.14.14.101) (Process ID 6)

Neighbor ID     Pri   State     Dead Time  Interface ID  Interface
14.14.14.102     1   FULL/DR   00:00:38   63            Gig1/0/1
```

Can you guess?

How did I do that? Let's take a step backward for a second and think about all the clues I've given you thus far.

In my previous blog, I raised a question that might have been posed during the development of IPv6 (IPng at the time):
"IPng addressing on local links: *Why use up precious IPng addresses supporting routing protocols on a local segment just for the purpose of routers talking to each other? They are just communicating on that local segment."*

It's a good question and one I definitely imagine was asked. With the concern about the exhaustion of IPv4 addresses, I can imagine address conservation techniques might have been forefront in people's minds. Why, then, burn up an IPng address on the link between RouterA and RouterB just to have OSPF neighbors come up?

What are the other clues I have given you?

- In the diagram above, I show the MAC addresses of both Router A's interface and Router B's interface
- I'm highlighting the MAC address in the show interface gig1/0/1 output for RouterA.

Sniffer trace
Let's look at the sniffer trace we caught and focus just on the OSPF.

```
12 fe80::5a0a:20ff:feeb:91e4  ff02::5                  OSPF    Hello Packet
15 fe80::2237:6ff:fecf:67e4   ff02::5                  OSPF    Hello Packet
18 fe80::5a0a:20ff:feeb:91e4  fe80::2237:6ff:fecf:67e4 OSPF    Hello Packet
19 fe80::2237:6ff:fecf:67e4   fe80::5a0a:20ff:feeb:91e4 OSPF    Hello Packet
33 fe80::5a0a:20ff:feeb:91e4  ff02::5                  OSPF    Hello Packet
34 fe80::2237:6ff:fecf:67e4   ff02::5                  OSPF    Hello Packet
39 fe80::5a0a:20ff:feeb:91e4  ff02::5                  OSPF    Hello Packet
40 fe80::2237:6ff:fecf:67e4   ff02::5                  OSPF    Hello Packet
43 fe80::5a0a:20ff:feeb:91e4  ff02::5                  OSPF    Hello Packet
44 fe80::2237:6ff:fecf:67e4   ff02::5                  OSPF    Hello Packet
49 fe80::5a0a:20ff:feeb:91e4  ff02::5                  OSPF    Hello Packet
50 fe80::2237:6ff:fecf:67e4   ff02::5                  OSPF    Hello Packet
51 fe80::5a0a:20ff:feeb:91e4  fe80::2237:6ff:fecf:67e4 OSPF    DB Description
52 fe80::2237:6ff:fecf:67e4   fe80::5a0a:20ff:feeb:91e4 OSPF    DB Description
55 fe80::2237:6ff:fecf:67e4   fe80::5a0a:20ff:feeb:91e4 OSPF    DB Description
56 fe80::5a0a:20ff:feeb:91e4  fe80::2237:6ff:fecf:67e4 OSPF    DB Description
57 fe80::2237:6ff:fecf:67e4   fe80::5a0a:20ff:feeb:91e4 OSPF    DB Description
58 fe80::5a0a:20ff:feeb:91e4  fe80::2237:6ff:fecf:67e4 OSPF    DB Description
59 fe80::2237:6ff:fecf:67e4   fe80::5a0a:20ff:feeb:91e4 OSPF    LS Request
```

Lots of FE80:: and FF02::5 stuff, eh? These are all part of the magic trick.

It may look messy and intimidating to read, but it is really just three numbers repeating again and again in our trace, all of which are actually IPv6 addresses critical to making the magic trick work:

- FF02::5
- FE80::5A0A:20FF:FEEB:91E4
- FE80::2237:06FF:FECF:67E4

Examining FF02::5

First, let's take a closer look at the one that stands out a little easier and is a little less intimidating – FF02::5.

```
33 fe80::5a0a:20ff:feeb:91e4  ff02::5    OSPF    Hello Packet
34 fe80::2237:6ff:fecf:67e4   ff02::5    OSPF    Hello Packet
39 fe80::5a0a:20ff:feeb:91e4  ff02::5    OSPF    Hello Packet
40 fe80::2237:6ff:fecf:67e4   ff02::5    OSPF    Hello Packet
43 fe80::5a0a:20ff:feeb:91e4  ff02::5    OSPF    Hello Packet
44 fe80::2237:6ff:fecf:67e4   ff02::5    OSPF    Hello Packet
49 fe80::5a0a:20ff:feeb:91e4  ff02::5    OSPF    Hello Packet
50 fe80::2237:6ff:fecf:67e4   ff02::5    OSPF    Hello Packet
```

Does that "::5" at the end of "FF02::5" remind you of anything? How about the clue that FF02::5 is only a destination IP address and never a source? Anything? Would it remind you of anything if I mentioned the IPv4 address 224.0.0.5?

Let's go to this IANA page, plus this other IANA page and merge together some information from the two.

IPv6		IPv4	
Address(s)	**Description**	**Address(es)**	**Description**
		224.0.0.0	Base Address (Reserved)
FF02:0:0:0:0:0:0:1	All Nodes Address	224.0.0.1	All Systems on this Subnet
FF02:0:0:0:0:0:0:2	All Routers Address	224.0.0.2	All Routers on this Subnet
FF02:0:0:0:0:0:0:3	Unassigned	224.0.0.3	Unassigned
FF02:0:0:0:0:0:0:4	DVMRP Routers	224.0.0.4	DVMRP Routers
FF02:0:0:0:0:0:0:5	OSPFIGP	224.0.0.5	OSPFIGP OSPFIGP All Routers
FF02:0:0:0:0:0:0:6	OSPFIGP Designated Routers	224.0.0.6	OSPFIGP OSPFIGP Designated Routers
FF02:0:0:0:0:0:0:7	ST Routers	224.0.0.7	ST Routers
FF02:0:0:0:0:0:0:8	ST Hosts	224.0.0.8	ST Hosts
FF02:0:0:0:0:0:0:9	RIP Routers	224.0.0.9	RIP2 Routers
FF02:0:0:0:0:0:0:A	EIGRP Routers	224.0.0.10	IGRP Routers
FF02:0:0:0:0:0:0:B	Mobile-Agents	224.0.0.11	Mobile-Agents
FF02:0:0:0:0:0:0:C	SSDP	224.0.0.12	DHCP Server / Relay Agent
FF02:0:0:0:0:0:0:D	All PIM Routers	224.0.0.13	All PIM Routers
FF02:0:0:0:0:0:0:E	RSVP-ENCAPSULATION	224.0.0.14	RSVP-ENCAPSULATION

Mystery solved for FF02::5! It is the IPv6 multicast address equivalent to 224.0.0.5: OSPF.

Understanding IPv6: Link-Local 'Magic'

Denise Fishburne performs a little IPv6 sleight of hand in the second post in her series on IPv6.

So what are all those FE80:: IPv6 addresses and where did they come from? Clearly, they somehow came from RouterA and RouterB because this is a sniffer trace of the packets on the point-to-point link between Router A and Router B.

```
51 fe80::5a0a:20ff:feeb:91e4  fe80::2237:6ff:fecf:67e4  OSPF    DB Description
52 fe80::2237:6ff:fecf:67e4   fe80::5a0a:20ff:feeb:91e4 OSPF    DB Description
55 fe80::2237:6ff:fecf:67e4   fe80::5a0a:20ff:feeb:91e4 OSPF    DB Description
56 fe80::5a0a:20ff:feeb:91e4  fe80::2237:6ff:fecf:67e4  OSPF    DB Description
57 fe80::2237:6ff:fecf:67e4   fe80::5a0a:20ff:feeb:91e4 OSPF    DB Description
58 fe80::5a0a:20ff:feeb:91e4  fe80::2237:6ff:fecf:67e4  OSPF    DB Description
59 fe80::2237:6ff:fecf:67e4   fe80::5a0a:20ff:feeb:91e4 OSPF    LS Request
```

So the question is, how did the routers get IPv6 addresses if I didn't configure any. Allow me to introduce you to the IPv6 link-local unicast address type.

RFC4291 section 2.4 introduces the varying address types. FF00::/8 we met above with FF02::5. Our two FE80:: IPv6 addresses clearly fall under the "link-local unicast" family.

2.4. Address Type Identification

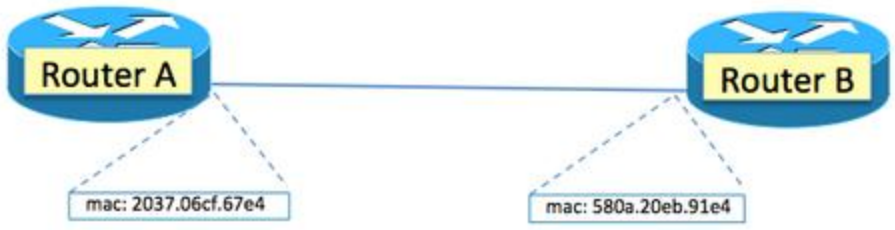
The type of an IPv6 address is identified by the high-order bits of the address, as follows:

Address type	Binary prefix	IPv6 notation	Section
Unspecified	00...0 (128 bits)	::/128	2.5.2
Loopback	00...1 (128 bits)	::1/128	2.5.3
Multicast	11111111	FF00::/8	2.7
Link-Local unicast	1111111010	FE80::/10	2.5.6
Global Unicast	(everything else)		

RFC4291 section 2.8 states, "*A host is required to recognize the following addresses as identifying itself*"and it goes on to have the first bulleted item as "*its required Link-Local address for each interface.*"

So we have just learned that IPv6 hosts must have a link-local unicast IPv6 address for each interface. We also know that this IPv6 link-local unicast address must start with FE80:: as per RFC4291 section 2.4.

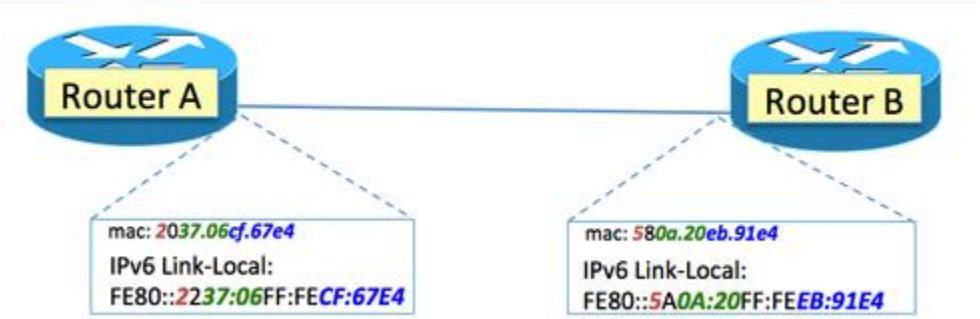
But where DO the rest of the numbers after FE80:: come from?



What were those two IPv6 link-local unicast addresses we saw in the trace again?

- FE80::5A0A:20FF:FEEB:91E4
- FE80::2237:06FF:FECF:67E4

Notice anything? Look at those numbers and look at the picture above. See it? Let me help a little.



The routers and code version I am using for Router A and Router B are defaulting to creating the required IPv6 link-local unicast address via the described method of creating an IEEE EUI-64 identifier from an IEEE 48-bit MAC identifier. I will not go into detail describing this here as there are many helpful links out on the web to explain the magic, including RFC4291 Appendix A.

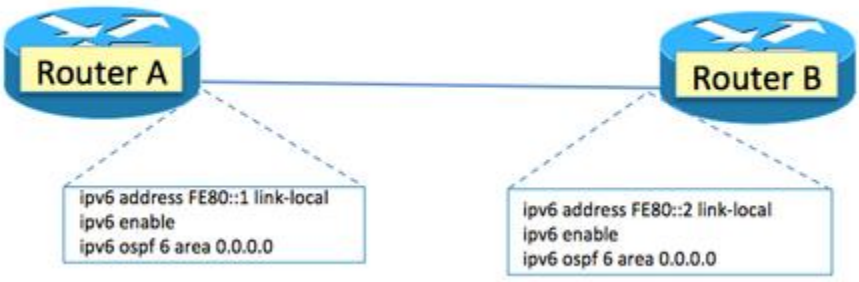
The point I'm trying to make here is not the specifics of how the MAC to IPv6 link-local magic happens. The "takeaway" I want you to sear into your brain is that, by default, the IPv6 link-local address for Router A and Router B that is used to bring up the OSPF will be based on the MAC address for that specific interface.

Can this get complicated? You tell me. Can you easily read the sniffer trace below and know who is who?

```
51 fe80::5a0a:20ff:feeb:91e4  fe80::2237:6ff:fecf:67e4  OSPF    DB Description
52 fe80::2237:6ff:fecf:67e4   fe80::5a0a:20ff:feeb:91e4 OSPF    DB Description
55 fe80::2237:6ff:fecf:67e4   fe80::5a0a:20ff:feeb:91e4 OSPF    DB Description
56 fe80::5a0a:20ff:feeb:91e4  fe80::2237:6ff:fecf:67e4  OSPF    DB Description
57 fe80::2237:6ff:fecf:67e4   fe80::5a0a:20ff:feeb:91e4 OSPF    DB Description
58 fe80::5a0a:20ff:feeb:91e4  fe80::2237:6ff:fecf:67e4  OSPF    DB Description
59 fe80::2237:6ff:fecf:67e4   fe80::5a0a:20ff:feeb:91e4 OSPF    LS Request
```

Statically defined IPv6 link-local

Like most defaults, we can override them. For example, below I am assigning FE80::1 as the IPv6 link-local address for Router A and FE80::2 as the IPv6 link-local address for Router B.



Let's shut/no shut the interface and grab a new sniffer trace. As you can see, this makes the sniffer trace much easier to read. This will become even more important, in my opinion, when we talk about RAs (router advertisements) in the future.

```
46 fe80::1      ff02::5      OSPF    Hello Packet
50 fe80::2      ff02::5      OSPF    Hello Packet
52 fe80::1      ff02::5      OSPF    Hello Packet
53 fe80::1      fe80::2      OSPF    DB Description
54 fe80::2      fe80::1      OSPF    DB Description
55 fe80::1      fe80::2      OSPF    DB Description
56 fe80::2      fe80::1      OSPF    DB Description
57 fe80::1      fe80::2      OSPF    LS Request
58 fe80::1      fe80::2      OSPF    DB Description
59 fe80::2      fe80::1      OSPF    LS Update
60 fe80::2      fe80::1      OSPF    LS Request
61 fe80::1      fe80::2      OSPF    LS Update
62 fe80::1      ff02::6      OSPF    LS Update
63 fe80::2      ff02::5      OSPF    LS Update
64 fe80::1      fe80::2      OSPF    LS Acknowledge
```

Voila!

There is no magic of course. The "rabbit" was "in the hat" all along, so to speak.

The IPv6 IGP neighbor relationship (in this case, OSPF) comes up and doesn't have to use one global unicast IPv6 address.

You might find yourself now wondering: "If we had assigned IPv6 global unicast addresses to those interfaces, would the IGP (OSPFv3) have used the global unicast addresses as the source and destination IPv6 addresses for the IGP neighboring instead of the link-local?"

Good question. Catch you on that one in a future blog.