

## Lab 3: IPv6 Redux

*or: A closer look and a deeper understanding of IPv6 on MS-Win, IOS*

### What You Will Do:

1. Re-do the key elements of Lab 3 using a combination of your laptop and PT. (Ideally you were able to save your config from Lab 3 and can copy + paste most of it.)
2. Ensure you know how to find and correlate MS-Win interface numbering
3. Configure IPv6 static and default routes to achieve end-to-end IPv6 connectivity

### Things that you will need to know or learn:

1. How to access the Properties window for a NIC under MS-Win 10.
2. How to copy & paste inside a CMD window on MS-Win.
3. A clear understanding of the difference between link-local vs Global unicast addresses.
4. Meaning of the written format for IPv6 addresses and masks.
5. Essential IPv6 address ranges:  
Link-local (FE80::/10), Global unicast (e.g. 2001::/64), Site local (FD00::/8), Default (::/0)
6. There's no pre-defined interface (or location) for link-local addresses!
7. Wading through the (longer) output of commands when using IPv6.

### What you need to submit and when:

1. Complete the Lab 3 Post-lab quiz on BrightSpace with answers from your work, **before** your next lab.

### Required Equipment:

- Access to a computer running MS-Win 7 (or higher), either real or a VM.
- Access to a computer running a recent version of Packet Tracer (e.g. ver 8.2).

### Marks:

This post lab is worth 35% of the mark for Lab 3.

10% of your final mark is for labs done during the course of the semester.

### References and Resources:

- Wk02 slide deck on IPv6 (course web site)
- Wk02 IPv6 Reference (course web site)
- Packet Tracer ver 8.2 or higher (available from NetAcad)
- Algonquin's NetLab facility – <http://netlab.algonquincollege.com>



## Task 1: A Closer look at IPv6 with MS-Win 10

Knowing how to identify an interface is essential for link-local pinging.

For this task, you'll need access to a PC (either real or virtual) running MS-Win 7 (or higher). We want to compare Cisco's use of *names* for interfaces (gigabitEthernet0/0) vs MS-Win's use of *numbers* for identifying interfaces. You'll gather two sets of information and see how they correlate.

Step 1. Two items need to be correct or the later steps won't work!

(A) Choose some NIC that will have a connection, either a wifi NIC *with a connection* or an Ethernet NIC that has a cable (real or virtual) attached and is *active*.

(B) Ensure that IPv6 is **enabled** on that NIC (i.e. as per Lab 3 Task 2).

Step 2. Gather the first info: full details on the current IP configuration.

Open a CMD window, and display the complete IP settings: `ipconfig /all`

The listing may have several NICs and you'll need to identify your chosen NIC. Any IPv4 addressing might give some clues: for example, if you're at home, the address will likely be a 192.168.X.0/24 subnet. Other clues are the title for the NIC's section: e.g.

wired connections: *Ethernet adapter Local Area Connection*

wireless connections: *Wireless LAN adapter Wireless Network Connection*

Make a copy of the output.

Step 3. Gather the second info: interfaces and routing table(s).

(Yes, every network-capable device has a routing table, even a PC.)

In the CMD window, get routing information: `netstat -nr`

You'll notice three sections: a list of interfaces, an IPv4 routing table, an IPv6 routing table

Make a copy of the output.

Step 4. Compare the two sets of info to correlate details and answer these questions.

**Q:** In the ipconfig, the actual name and model of the NIC are listed. What is the exact 'title' for that line of config (i.e. what's the name of that field)?

**Q:** In the routing information, the exact same info on NIC name & model appears. Find it. In which of the three sections from step 3 does it appear (i.e. what's the exact title of the section)?

**Q:** Find the NIC's MAC address on the line 'Physical address' in the ipconfig. Observe that it's written in a certain format, but note there are other possible formats. To help with troubleshooting, that MAC address also appears in the info from Step 3. What's the title of the section it's in? Is it written in the exact same format, character for character?

**Q:** In the ipconfig, find the complete link-local address. You know how IPv6 addresses are written and what are valid characters for hexadecimal numbers. That enables you to know:

A. From **Lab 3 from T108**, what three extra characters were added to the link-local address of the Realtek NIC?

B. From *your chosen NIC*, find the corresponding extra characters. What are they?

C. Is the link-local address created using EUI-64 or a randomly generated value?

**Q:** We've identified a key section from Step 3.

A. In that section, how many columns are there (not the number of characters, but the number of fields in each line)?

B. Find the line about your NIC. In what column do you see the numeric value you identified in the previous question?

Now here's the learning objective: if/when you need to troubleshoot, can you (A) identify a given NIC; (B) find that NIC and its MAC address; and (C) identify the number MS-Win has assigned to that interface? Can you correlate this information between the IP config and the routing info?

## Task 2: Topology: Connection and Setup

Using PT, re-create the topology from Lab 3 (see page 2): either make a **copy** of Lab2-postlab and modify it, or start fresh. In either case, if you wish you can use IPv4 addressing to ensure that the wiring, interfaces, etc are properly configured. Verify & troubleshoot as needed.

Don't forget to include the loopback 1 interfaces on each router.

**Q:** The loopback 1 addresses are from a specific IPv6 address range. What is the correct name of that range? What is the RFC defining that range? Hint: Check out the IPv6 reference from Week 2 on the course site.

## Task 3: Link-Local Addresses

This task provides practice on configuring link-local addresses on a router. Remember: IPv6 is enabled on an interface by one of two methods: (a) the command **ipv6 enable** or (b) by setting a unicast IPv6 address (either link-local, global unicast, or Unique local).

Step 1. Enable IPv6 on the G0/1 interface by setting a link-local address:

```
ipv6 address FE80:R:R::R link-local where: R = 1 or 3
```

where: **R** is the router number (1, 3) as per the topology diagram.

Step 2. Verify (ALWAYS!) the setting and value is correct, and fix mistakes as necessary:

```
show ipv6 interface brief
```

## Task 4: Confirming Connectivity with Link-local addresses

This task provides practice on pinging link-local addresses to confirm connectivity.

We're working with PT so it's useful to mention that although it's a very respectable simulator, it's primary purpose is as a learning tool so several adaptations have been made. For example, although MS-Win doesn't allow you to modify the link-local address, you can set the address manually for the PCs in PT (it makes typing some ping commands a lot easier!) Also, there's no firewall in PT that needs to be disabled!

Step 1. [Optional, but recommended!] Let's take advantage of conveniences that PT offers us, and which allow keeping the focus on learning instead of watching for typing mistakes! On each of the PT PCs, click the Config tab, the FastEthernet button (column on left side), and do a manual override of link-local address: FE80:1:1::10 (PC1) or FE80:3:3::30 (PC2). Keep the IPv6 Configuration at it's default setting of Automatic.

Step 2. From each router's CLI, issue a ping to the PC's link-local address

```
ping FE80:R:R::R where: R = 1 or 3
```

The PC is connected to G0/1, so use that when asked about which interface to use. Troubleshooting should focus on typos.

Step 3. From each PC, ping the router's link-local address. (For the sake of learning, Cisco made the ping report the TTL even though MS-Win doesn't show TTL for IPv6.)

Step 4. From each PC, ping the router's loopback address. That confirms that the default gateway on the PC is correct and working properly.

Successful pings in both directions confirm that everything is correctly connected. Global addresses can now be added with confidence that the infrastructure is correct.

## Task 5: SLAAC

We need end-to-end connectivity in later tasks, so we'll get additional practice configuring SLAAC addressing since that is adequate. As a reminder, the only requirements for SLAAC are:

- a Global (or Local) unicast address assigned to the router interface
- routing capability enabled: `ipv6 unicast-routing`
- the PC set to automatic addressing

Step 1. Set up both routers with global unicast addresses as per the topology diagram (redo Lab 3 Task 5, step 2).

Step 2. Using the appropriate command in the CMD window, confirm that the PCs have obtained an address via SLAAC. If one your PC's doesn't get an address automatically and you've spent 20-30 mins troubleshooting, you can proceed to the next step as long as the other PC has a global unicast address.

Step 3. With SLAAC working, you can get confirmation of the full details from the CLI:

```
show ipv6 interface g0/1
```

(I got ~21 lines.) As always, it's a great idea to copy & save the output.

Step 4. From the output of the two commands, find the answers to these questions.

Q: Does the PC use the global unicast or the link-local address for it's gateway?

Q: From the router interface details, what does the very last line of output confirm about SLAAC? What is the exact, full text of that line?

Q: By default, how often are RA messages sent out? They are sent every \_\_\_\_ secs.

## Task 6: IPv6 Default and Static routes

With global unicast addresses between each PC and it's associated router, end to end connectivity is just a few steps away. Whether the addressing is IPv4 or IPv6, the same kind of static routes are needed. We'll use IPv6 for this exercise.

Step 1. Prior to adding routing, the basics are always needed first:

- assign global unicast addresses to R1 G0/0, R2 G0/0 + G0/1, and R3 G0/0 as indicated on the topology diagram
- ensure all interfaces are up/up
- ensure all routers, including R2, are ready for actual IPv6 routing (hint: Task 5)

You can confirm that everything is working by pinging across individual links R1-R2, and R2-R3. Troubleshoot by verifying: the cabling is correct, interfaces are up/up; that you've typed the addresses correctly and assigned them to the designated interfaces.

Step 2. Each of the end routers, R1 + R3, need a default static route pointing to R2.

Regardless of the IP version, it looks like: `[proto] route [default route - all 0's] [next-hop]` where: proto = ip, ipv6; default route written suitably; and next-hop is the address for the router at the *other end* of the link. Configure the static route on each end router.

Step 3. The router in the middle, R2, needs specific routes to the PC subnets on R1, R3.

Regardless of the IP version, it looks like: `[proto] route [subnet ID/mask] [next-hop]` where: proto = ip, ipv6; subnet ID/mask written suitably; and next-hop is the address for the router at the *other end* of the link.

Specifically, R2 needs a route to 2001:1:1::/64 with a next hop of R1's interface, and a route to 2001:3:3::/64 with a next hop of R3's interface.

Step 4. You should be able to ping from R1-R3. If not, troubleshoot by checking that you've typed the routes correctly (there's a total of 4: R1 x 1, R3 x 1, R2 x 2).

Step 5. You should also be able to ping from PC1 to the opposite router (e.g. R3) and opposite PC (e.g. PC2). Conveniently for our learning, the CMD window in PT reports the TTL for IPv6 (unlike a real MS-Win PC).

**Q:** What is the TTL when pinging from a PC to the opposite router?

**Q:** What is the TTL when pinging from one PC to the other?

Finally, here's two additional questions from the Lab activity in class that you can answer in the post-lab quiz:

**Q.** (From Lab 3) Which DHCPv6 command sets the "O" bit? Ref: Wk02Day2 lecture notes

**Q.** (From Lab 3) Which DHCPv6 command sets the "M" bit? Ref: Wk02Day2 lecture notes.

Congratulations! You've finished the post-lab exercise. Well done!

Hopefully, if you hit any problems, you never spent more than ~20 mins before reaching for help.