IPV6 FUNDAMENTALS

LEARN THE BASICS OF HOW IPV6 WORKS, IPV6 ADDRESSES AND IPV6 SUBNETTING

Ramon Nastase

IPv6 for Beginners

Your Quick Guide for Learning the Fundamentals of the IPv6 Protocol in only one sitting

Table of Contents

- Introduction to IPv6
- Let's simplify these addresses
- IPv6 address types
    - IPv6 Global addresses
    - IPv6 Link-Local addresses
    - Three ways of transmitting packets
- IPv6 Subnetting
- IPv4 & IPv6. How can we use the two protocols on the Internet?
- How can we configure IPv6?

From the Author

First of all, I want to thank you for reaching this point in the #LearnNetworking - IPv6 Concepts and Subnetting guide and thank you for deciding to invest in yourself to become better. From this guide, you can expect to get a better understanding of how the Internet works, how it grows, and what are the trends in technology in the future with the rapid expansion of the Internet of Things (IoT)

At the end of each chapter, you can expect practical exercises (subnetting or PC implementation of the elements studied). Each concept studied here can be applied to all existing devices on the market (which actively use IPv6).

As a Gift, I want to give, for FREE, access to my six steps guide on starting (and performing) as an IT Specialist in today's digital world. Claim your FREE guide by accessing this page: **https://bit.ly/IT-GIFT**, or you can scan the QR code:

SCAN ME

You can also check out one of my other books on Amazon.com by clicking this link: https://amzn.to/2zE72Wm

Also, I invite you to check out my IPv6 and Pv4 Subnetting course on Educative.io. By using this link, you'll be able to enjoy it for FREE. Check out the course for FREE on this link: https://www.educative.io/courses/quick-guide-ip-subnetting

Courses by Ramon Nastase

Ramon Nastase

A Quick Guide for IP Subnetting

Beginner

Preview →

Introduction to IPv6

Today, there are more than 20 billion devices connected to the Internet worldwide, and the number just gets bigger and bigger as days pass by. This is a major problem, especially for ISPs (Internet Service Providers), because it exceeds the 4.3 Billion that IPv4 was providing.

So here comes the need for a better, much larger protocol known as IPv6. IPv6 is a new addressing (identification) protocol that introduces a *new address* format (in hexadecimal) and a much, **much larger addressing space has** emerged. IPv6 is 128 bits (that means we have **2 ^ 128** addresses available), which is an infinitely larger space compared to IPv4 (which is only 32 bits long).

Besides these features, IPv6 has many others and will streamline the communication process of devices on the Internet, making it faster and more secure. According to this article on the Wikipedia website (https://en.wikipedia.org/wiki/IPv6), this number looks like this:

*"**340,282,366,920,938,463,463,374,607,431,768,211,456** or **$3.4×10^{38}$** (340 trillions of trillions of trillions)"*

Figure 1.1

An IPv6 can look similar to:

2003:4581:A7C1:EFDB:0000:0000:1327:0001

As you can see, this address is way different than an IPv4 address (ex: 192.168.1.1) that we are used to. The IPv6 addresses are formed out of **hex characters** (16 values, between **0 - 9** and **A - F**)

Let's simplify these addresses

The people who have developed the IPv6 protocol concluded that the addresses are way too long. So, they thought and found a solution to make them shorter by simplifying them in this way:

1) In case we have two or more consecutive fields containing **0s** (0000:0000...), then it can be reduced to **::**

   1234:ABCD:3123:0000:0000:0000:0000:0000 -> 1234:ABCD:3123::

   For example, the **default route** in IPv6 looks like this: **::/0** (and is much shorter compared to the IPv4 one, 0.0.0.0/0).

2) In case we don't have multiple consecutive fields of 0 we have another solution: **we'll write a group of 0000 as 0**. Here's an

example:

*1234:ABCD:3123:00A8:0A31:8000**:0000**:0001 -> 1234:ABCD:3123:00A8:0A31:8000**:0:**001*

3) In the above example we can simplify even more:

   *1234:ABCD:3123:00A8:0A31:8000**:0000**:0001 -> 1234:ABCD:3123:A8:A31:8000**:0:**1*

   And now, you can see that I've **eliminated all the 0s** from the beginning:

   00A8 -> A8, 0001 -> 1

The IPv6 address from the end of the last section

(*2003:4581:A7C1:EFDB:0000:0000:1327:0001)* can be written as:

2003:4581:A7C1:EFDB::1327:1

IPv6 address types

Now that you know what's the deal with IPv6 and what an address looks like, it's time to talk about the types of IPv6 addresses:

I. **Global** - 2000::/3 (*used on the Internet*)

II. **Link-Local** - FE80::/10 (*used in LANs*)

Let's start our discussion with the first type of IPv6 address:

1) IPv6 Global addresses

The IPv6 Global addresses are similar to the public IPv4 addresses (they **can be used on the Internet**, not only on the LAN). You will recognize these global addresses because they always start with the number **2** or **3** (due to its subnet mask -> **2000::/3**).

In the following figure 1.2, you can see the structure of such an IPv6 address:

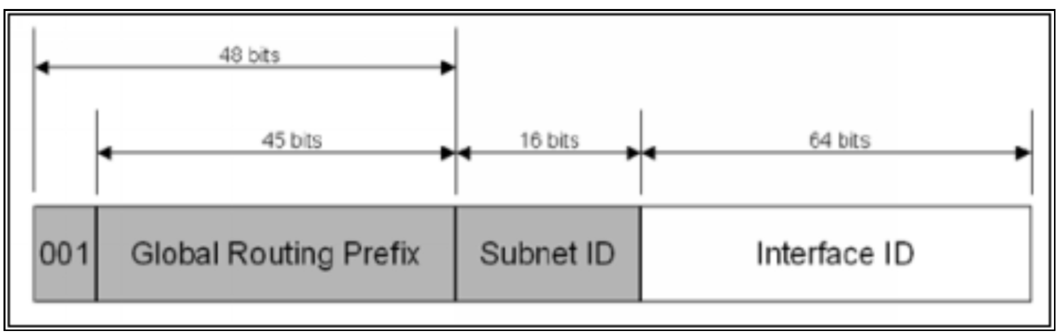


Figure 1.2

First of all, the **first 3 bits** are **reserved** (this is why the **IPv6 Global** addresses start with **2**). After this resides the Global Routing Prefix (or the addressing space that has been reserved for the ISP). The subnet ID represents the ID of the Networks (a total of 65536 = 2^16) and the last field, the Interface ID, is the device's IPv6 address.

Here a few examples of global addresses (NOTE: these are host addresses, not network):

- **2001:DB8:A0B:12F0::1**/64
- **3731:ADE0:9923:23::90**/64

- **2020:ABCD:1:FFF0:84:ADEF**/64

If you are not familiar with the *host/network addresses and/or subnetting*, I recommend you **check out this GUIDE** (https://amzn.to/33LLkzw).

2) IPv6 Link-Local addresses

There are special addresses that make communication between devices (PCs, Laptops, Smartphones, etc.) within a local area network (LAN) possible. The advantage of these types of addresses is that they have configured automatically (**autoconfig**) and do not require any user intervention.

Link-Local IPv6 addresses are in the **FE80:: /10** format. This means that they begin with **FE80** and the *rest of the address is generated* automatically by **EUI-64.** For example, if you're using Windows) and looking at the CMD now, you can see an *IPv6 Link-Local Address*:

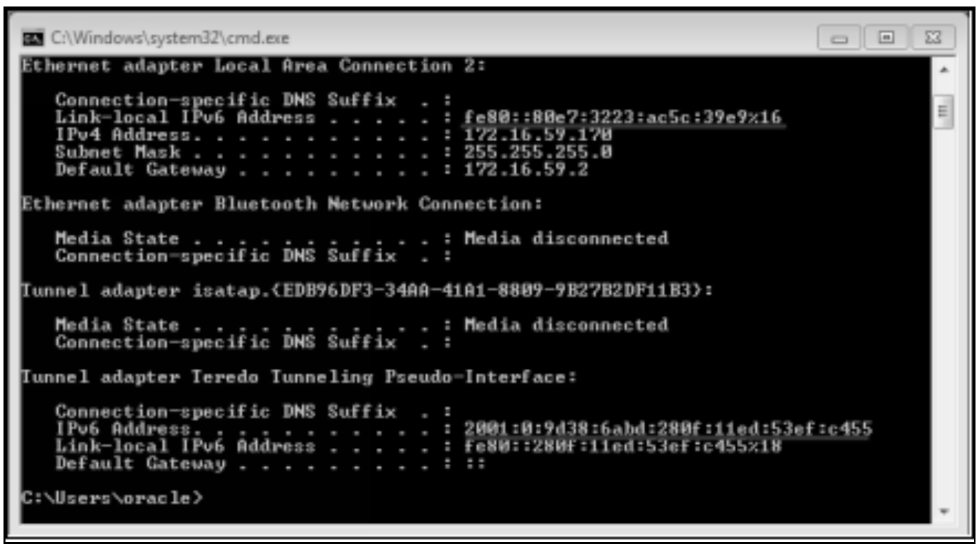


Figure 1.3

The first IPv6 address represents the Link-Local address of the Ethernet interface and the 2nd address that starts with 2001:0… is the global IPv6 address belonging to the Tunnel interface.

As I said before, the Link-Local addresses are generated using the EUI-64 technique. This technique is being used by most vendors out there (e.g. Linux), but within Windows, Microsoft chooses to randomly generate them.

(https://technet.microsoft.com/en-us/library/2007.08.cableguy.aspx) these IPv6 Link-Local addresses. But for the majority of the systems: **EUI-64** is a technique that uses the **MAC** addresses (because it's a unique address) and an **FFFE** Hexa field to form an IPv6 Link-Local address. In Figure 1.4, you can see exactly how it's being done:

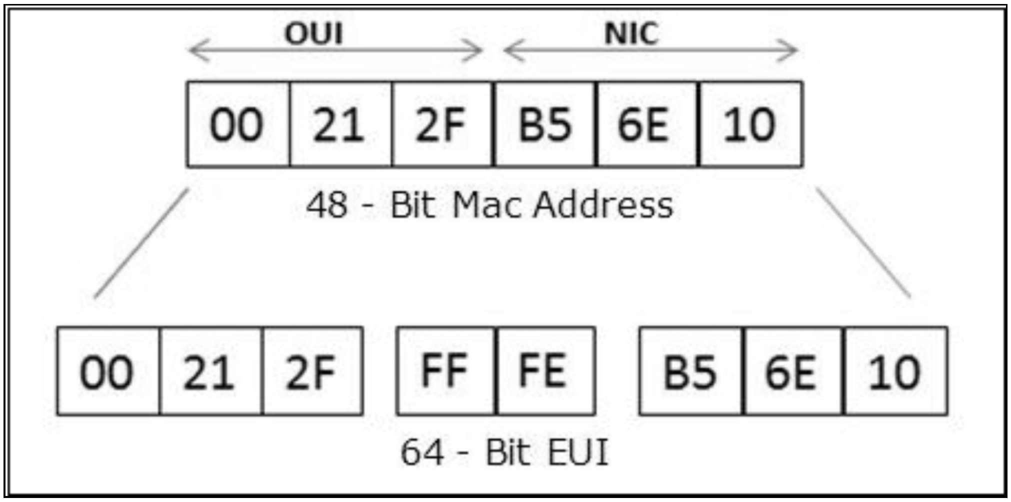


Figure 1.4

The initial MAC address *00:21:2F:B5:6E:10* is separated into 2 parts and in the middle, the FF:FE values are being added. Thus, *0021:2FFF:FEB5:6E10* will represent the host bits of the newly generated IPv6 address:

FE80::0021:2FFF:FEB5:6E10/64

In some cases, the 2nd bit (of the added string) is switched (1 -> 0 or 0 -> 1). In this scenario, our IPv6 address will look similar to:

FE80::**0021**:2FFF:FEB5:6E10/64 -> FE80::**0023**:2FFF:FEB5:6E10/64

Before moving forward, I want to tell you something that you should remember: the **IPv6 Link-local addresses DO NOT leave** the **LAN** area. They **don't have access to the Internet**.

Three ways of transmitting packets

Besides IPv4, in IPv6, there is no such thing as BROADCAST message types. Broadcast traffic has been replaced with Multicast.

If we think about it, that sounds pretty logical. Why? Because if I want to send traffic to a specific group of devices in my network (aka. Multicast), then I can use a similar message type and send it **to all the devices** connected to my network (thus, there will be a specific group formed by all of the devices).

Now, let's take a look at the other ways we can send IPv6 traffic:

- **Unicast** - aka. *one-to-one*
- **Multicast** - aka. *one-to-many* (or one-to-all)
- **Anycast** - aka. *one-to-closest*

Example of Anycast traffic:

If we want to go to Google.com, we must first translate the name into an IP address. To do so, we need a DNS server.

We have multiple DNS servers in the US and an Anycast message will be sent to the closest server to our location. Thus the waiting time will be lower and our browsing experience will appear faster.

IPv6 Subnetting

Ooo, Yes! Subnetting. Many people are scared of IPv6 subnetting, but I always say they have no reason to worry because **IPv6 subnetting** is (in my opinion) simpler than **IPv4 subnetting**. When speaking about IPv6 networks, you'll never see something like this:

"I want an IPv6 network with five or ten addresses"... no.

Why? Because there are enough addresses available and the size of a network, one does not have to worry. You will often see **IPv6 networks that have a / 64, / 80, / 96, or other masks**. Most often, I can say I saw networks / 64.

Now let's see what a /64 network looks like or means? Let's take the following example:

2002:ABCD:1234:9FD8::/64

This address is an excellent start for us to learn subnetting. /64, in this case, refers to the fact that the first **64 bits DO NOT CHANGE** (the first four fields). They remain the same.

So, we can assign a mask of /80 for any network we want. Let's assume that we need four networks, each with a different number of required IP addresses (all fitting into /80).

1st network: **2002:ABCD:1234:9FD8:0000::/80**

(or 2002:ABCD:1234:9FD8::/80)

2nd network: **2002:ABCD:1234:9FD8:0001::/80**

(or 2002:ABCD:1234:9FD8:1::/80)

3rd network: **2002:ABCD:1234:9FD8:0002::/80**

(or 2002:ABCD:1234:9FD8:2::/80)

4th network: **2002:ABCD:1234:9FD8:0003::/80**

(or 2002:ABCD:1234:9FD8:3::/80)

What did I do above? I carefully looked at what **field** I was in (**5th out of 8**) and I **added one** from one network to the other).

And that's it! It doesn't make sense to overstress with subnetting based on how many IP addresses we need in our networks. We have to take the IPv6 address space provided to us by our ISPs and break it down into smaller networks (with a mask of /64, /80, /89, etc.).

After that, we'll ask ourselves: "how many networks do I need ?" We'll start dividing the big network (provided by our ISP) into smaller ones.

In other words, if we receive a /64 network, then we'll use /80 for our smaller ones.

IPv4 & IPv6. How can we use the two protocols on the Internet?

Most of the time, we'll have a network in which IPv4 and IPv6 are configured. By design, the two protocols are **INCOMPATIBLE**. This means that a PC with an IPv4 address can't communicate with another PC with an IPv6 address.

And now this question comes: what are we going to do in this situation? And, of course, we have multiple solutions for this problem:

- **Dual-Stack**
- **Tunnels 6to4 or 4to6**

- **NAT-PT (aka NAT64)**

1. Dual-Stack

The Dual-Stack concept is very simple. We have to configure **2 IP addresses** (one being a **v4** and the other **v6**) on the device (Router, PC, server, etc.). By doing it this way, the PCs will be able to talk to each other by using any of the two protocols (figure 1.5)

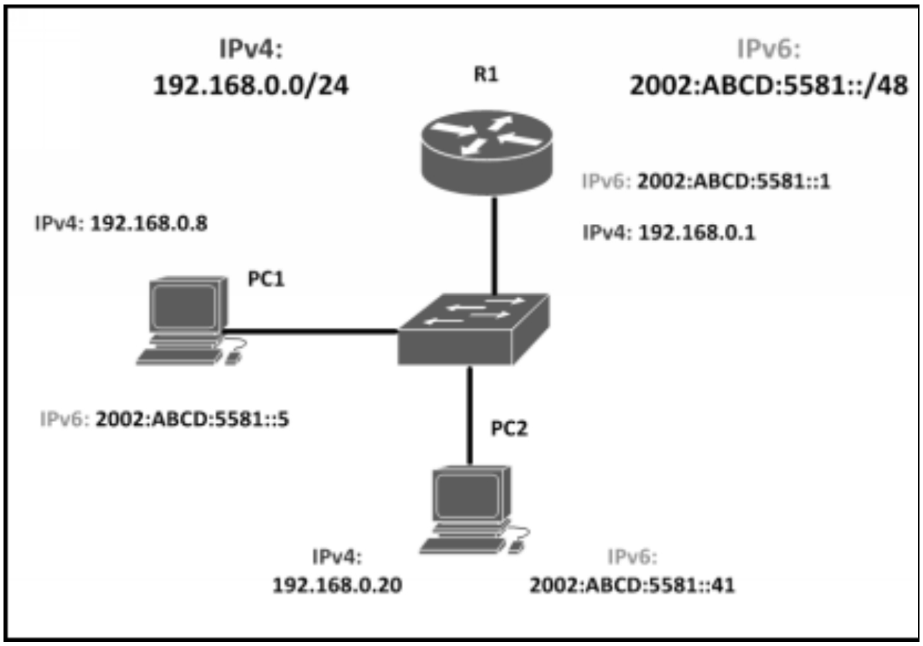


Figure 1.5

2. 6to4 or 4to6 Tunnels

Let's say that we have a scenario similar to Figure 1.6. Two IPv6 networks are connected to two different Routers in separated geographical regions and connected by a large IPv4 network (aka. The Internet). *In* this scenario, the PCs from the two networks **can't communicate** with each other due to the **incompatibility of IPv4 and IPv6**. A solution to this problem is to create a 6to4 tunnel between the Routers, transporting the (encapsulated) IPv6 addresses over the IPv4 network (Internet). By doing things in this way, the 2 PCs will be able to send traffic to each other (via the 6to4 Tunnel). We can have a *similar scenario* in which the 2 Routers are connected to 2 different IPv4 networks and there is an IPv6 network in the middle. The solution would be to form a 4to6 **tunnel** between the 2 Routers over the IPv6 network.

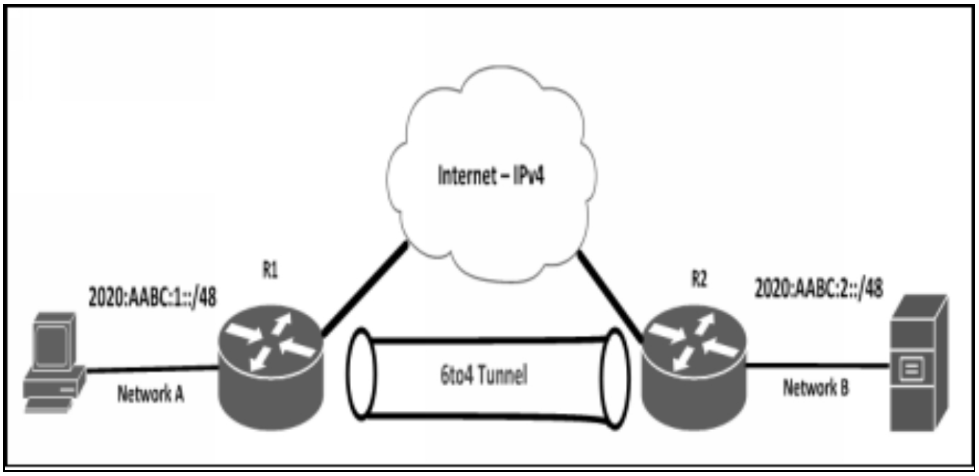


Figure 1.6

3. NAT-PT (Network Address Translation - Protocol Translation)

Routers usually use NAT (Network Address Translation) to translate a private IP address (ex: 192.168.1.10) to a public IP address (ex: 85.13.217.9) for the devices (that are being translated) to reach the Internet.

Well, the **NAT-PT** does the same thing, the only difference being that it **translates an IPv4** address into **an IPv6** address or ***reverse*** (IPv6 -> IPv4)

For example, a Router can be connected to two networks, one using (only) IPv4 addresses and the other using (only) IPv6 addresses. If two end devices from the networks want to send traffic to each other, they won't be able to. So, by using NAT-PT, the Router will translate the IPv4 address to an IPv6 address, thus providing the traffic exchange.

Figure 1.7 is the scenario that we spoke about above:

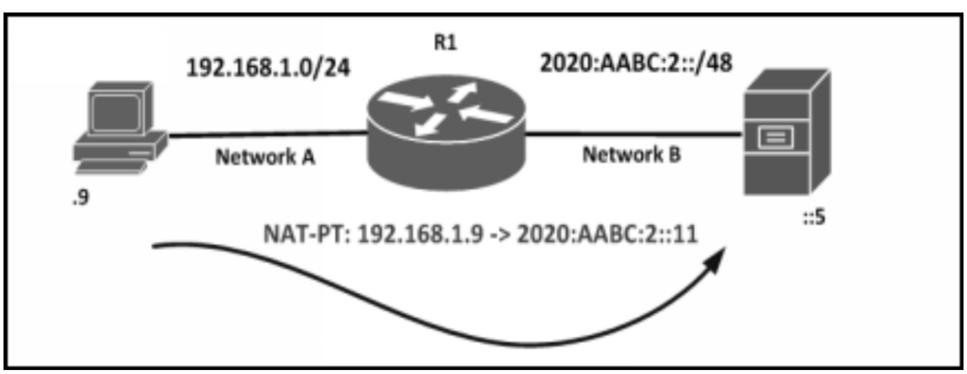


Figure 1.7

NOTE: the Router doesn't have dual-stack configured on any of the interfaces

How can we configure IPv6?

Let's move further and configure IPv6 addresses on the PCs and Router from the topology in figure 1.8 (this process is very similar to the one from the IPv4 config):

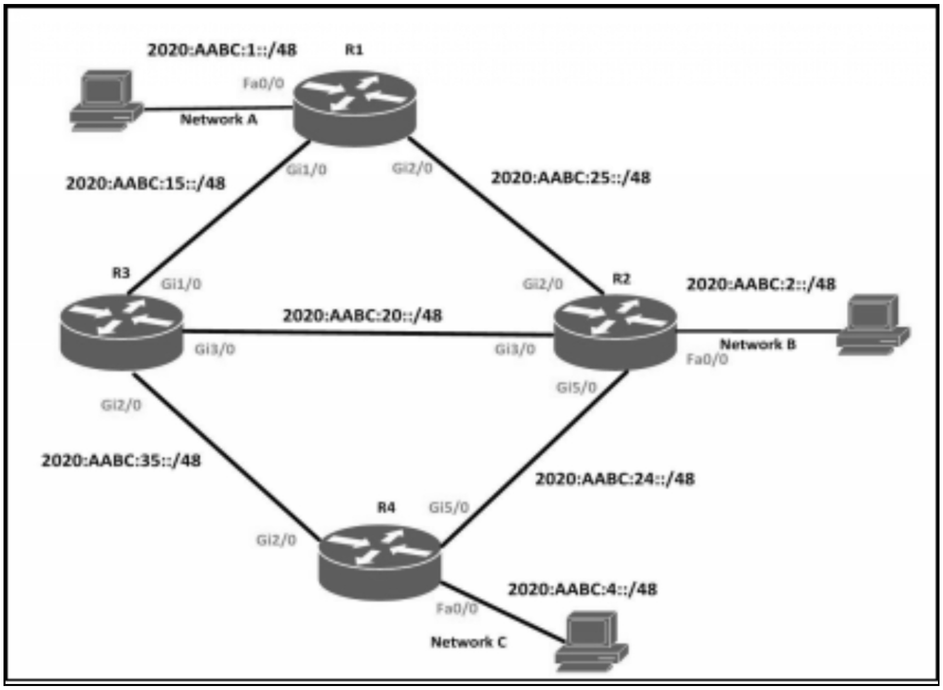


Figure 1.8

Let's start by **setting an IPv6 address on Windows 7** (the steps are similar for Windows 8.1, 10 and 11). Usually, the IPv6 address on a PC (or laptop) doesn't need to be configured (due to the IPv6's auto-configuration feature). Still, there are many situations when we need a static/manual address (for example, if we have a server on the Windows machine).

Here's how we can assign an IPv6 address to a Windows machine. But, first of all, we need to get to the right configuration windows (where our network interfaces reside):

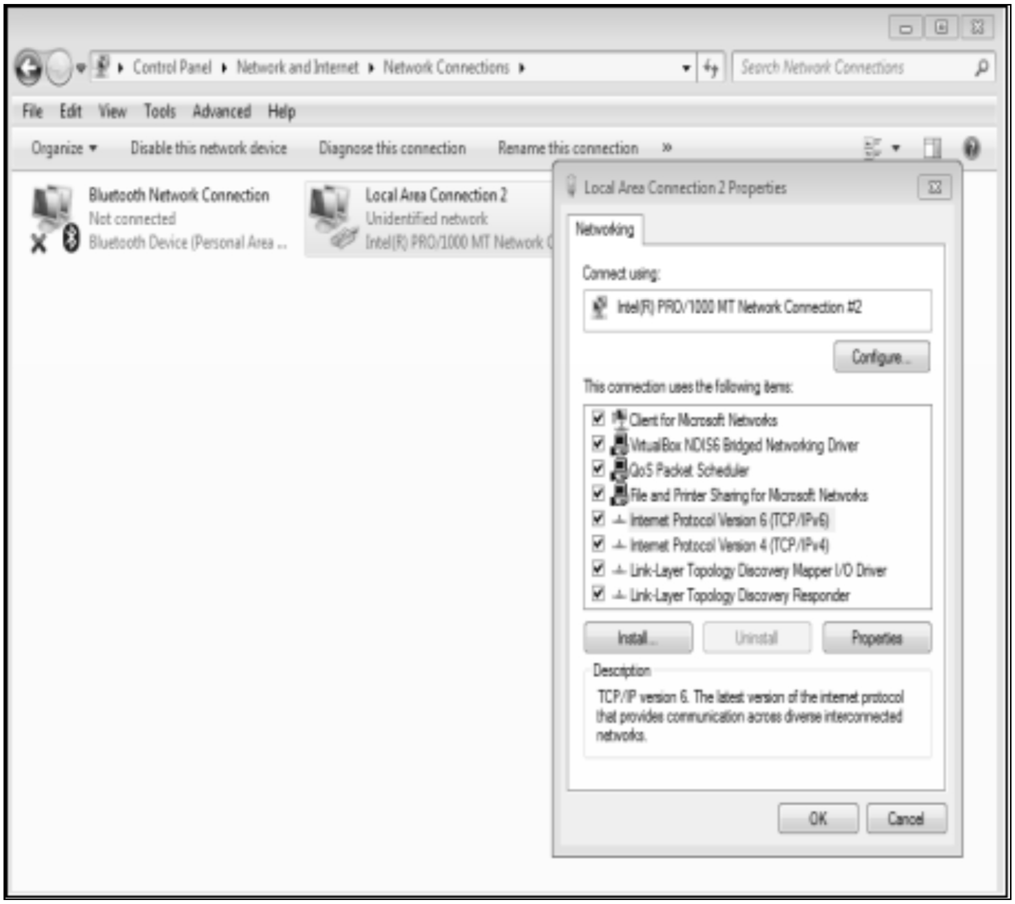


Figure 1.9

Here we are going to select IPv6 and click on "**Properties**". After that, a window will appear and we're going to be able to insert an *IPv6 address, a netmask, a default gateway, and a DNS server.* Here are the IPv6 settings:

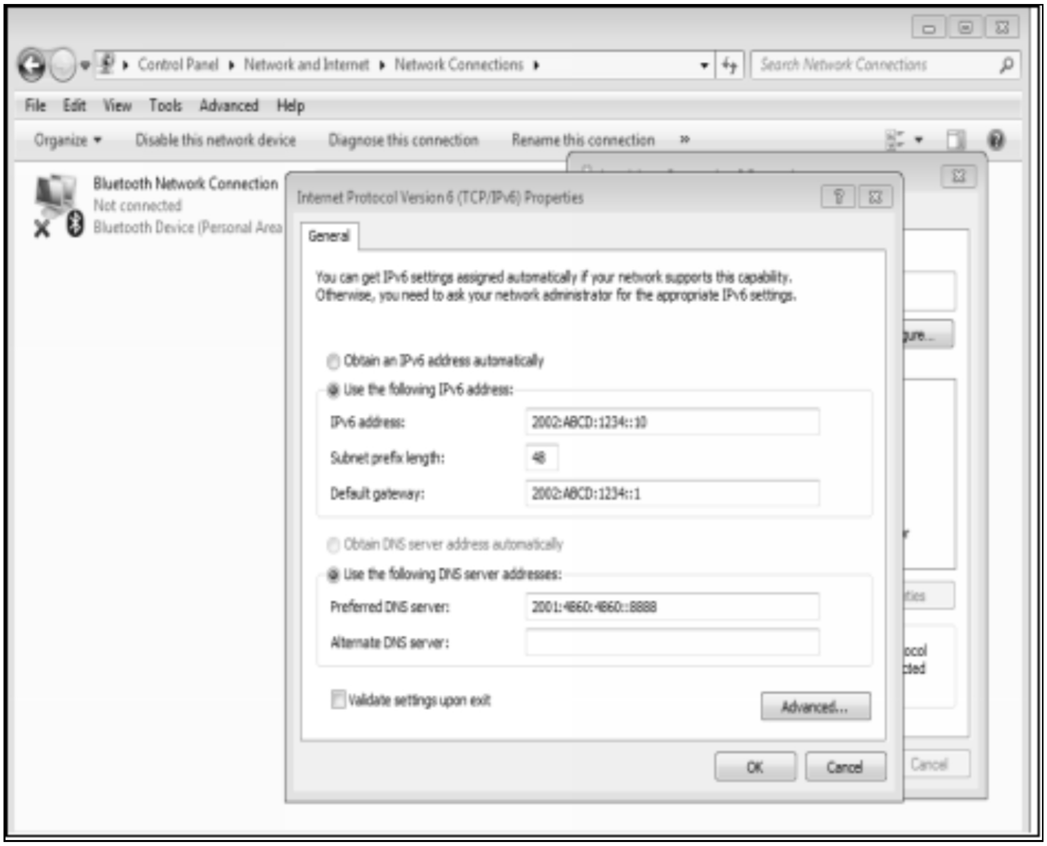


Figure 1.10

Now, let's check the config that we did above in figure 1.10 **>ipconfig**

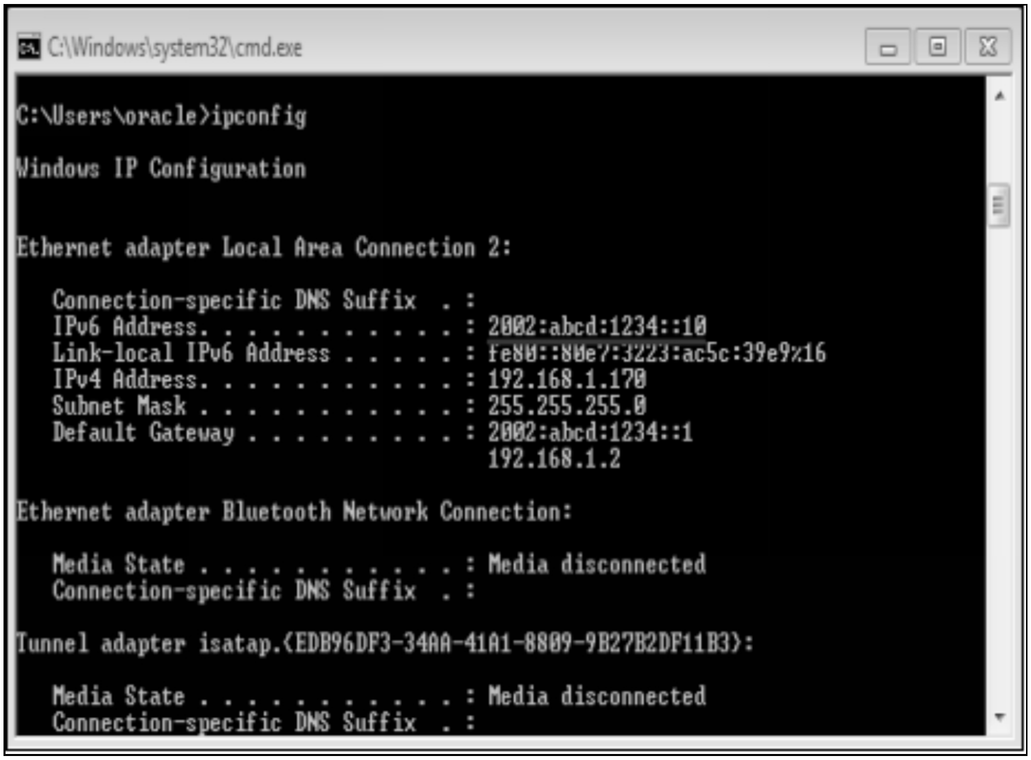


Figure 1.11

Now, after we saw how to set an IPv6 address on Windows, it's time to move further and see how can we do it on professional networking equipment (in this case, a Router from cisco.com):

On R1, the config will be:

```
R1(config)#ipv6 unicast-routing
```

```
R1(config)#interface Fa0/0
R1(config-if)#ipv6 address 2020:AABC:1::1/48
R1(config-if)#no shutdown
```

```
R1(config)#interface Gi1/0
R1(config-if)#ipv6 address 2020:AABC:15::1/48
R1(config-if)#no shutdown
```

```
R1(config)#interface Gi2/0
R1(config-if)#ipv6 address 2020:AABC:25::1/48
R1(config-if)#no shutdown
```

```
R1(config)#
R1(config)#ipv6 unicast-routing
R1(config)#interface fa0/0
R1(config-if)#ipv6 address 2020:AABC:1::1/48
R1(config-if)#exit
R1(config)#interface Gi1/0
R1(config-if)#ipv6 address 2020:AABC:15::1/48
R1(config-if)#exit
R1(config)#interface Gi2/0
R1(config-if)#ipv6 address 2020:AABC:25::1/48
R1(config-if)#exit
R1(config)#
```

On R2, the config will be:

```
R2(config)#ipv6 unicast-routing
```

```
R2(config)#interface Gi1/0
R2(config-if)#ipv6 address 2020:AABC:15::2/48
R2(config-if)#no shutdown
```

```
R2(config)#interface Gi2/0
R2(config-if)#ipv6 address 2020:AABC:35::1/48
R2(config-if)#no shutdown
```

```
R2(config)#interface Gi3/0
R2(config-if)#ipv6 address 2020:AABC:20::1/48
R2(config-if)#no shutdown
```

```
R2#
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#ipv6 unicast-routing
R2(config)#interface Gi1/0
R2(config-if)#ipv6 address 2020:AABC:15::2/48
R2(config-if)#exit
R2(config)#interface Gi2/0
R2(config-if)#ipv6 address 2020:AABC:35::1/48
R2(config-if)#exit
R2(config)#interface Gi3/0
R2(config-if)#ipv6 address 2020:AABC:20::1/48
R2(config-if)#exit
R2(config)#
```

On R3, the config will be:

```
R3(config)#ipv6 unicast-routing
```

```
R3(config)#interface Fa0/0
R3(config-if)#ipv6 address 2020:AABC:2::1/48
R3(config-if)#no shutdown
```

```
R3(config)#interface Gi2/0
R3(config-if)#ipv6 address 2020:AABC:25::2/48
R3(config-if)#no shutdown
```

```
R3(config)#interface Gi3/0
R3(config-if)#ipv6 address 2020:AABC:20::2/48
R3(config-if)#no shutdown
```

```
R3(config)#interface Gi5/0
R3(config-if)#ipv6 address 2020:AABC:24::1/48
R3(config-if)#no shutdown
```

```
R3#
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#ipv6 unicast-routing
R3(config)#interface fa0/0
R3(config-if)#ipv6 address 2020:AABC:2::1/48
R3(config-if)#exit
R3(config)#interface Gi2/0
R3(config-if)#ipv6 address 2020:AABC:25::2/48
R3(config-if)#exit
R3(config)#interface Gi3/0
R3(config-if)#ipv6 address 2020:AABC:20::2/48
R3(config-if)#exit
R3(config)#interface Gi5/0
R3(config-if)#ipv6 address 2020:AABC:24::1/48
R3(config-if)#exit
R3(config)#
```

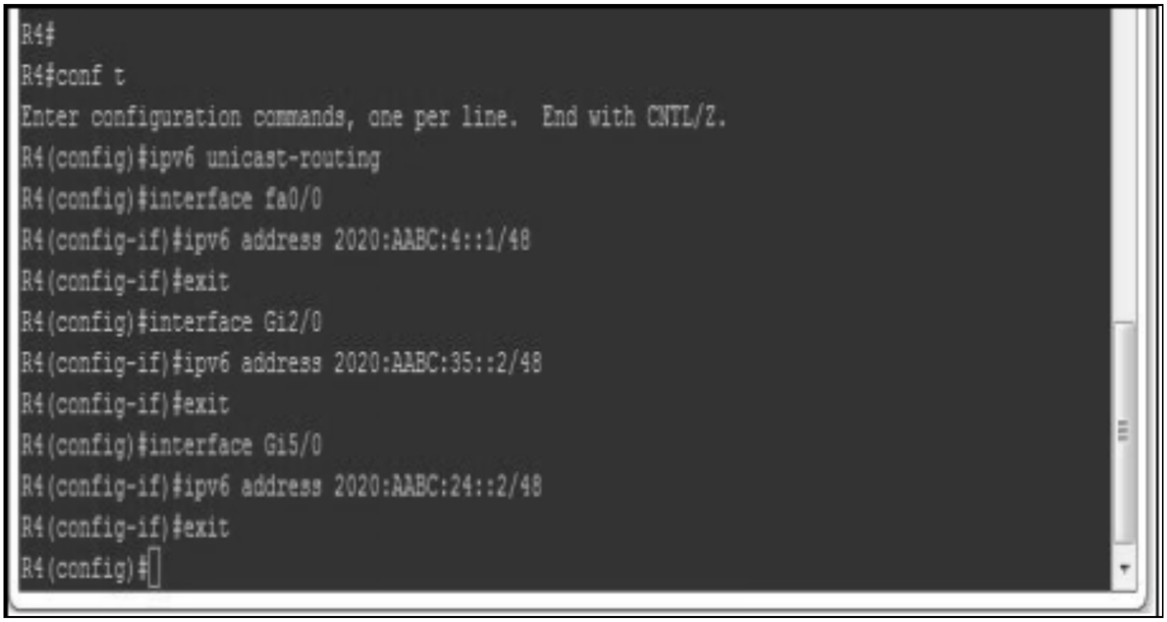
On R4, the config will be:

```
R4(config)#ipv6 unicast-routing
```

```
R4(config)#interface Fa0/0
R4(config-if)#ipv6 address 2020:AABC:4::1/48
R4(config-if)#no shutdown
```

```
R4(config)#interface Gi2/0
R4(config-if)#ipv6 address 2020:AABC:35::2/48
R4(config-if)#no shutdown
```

```
R4(config)#interface Gi5/0
R4(config-if)#ipv6 address 2020:AABC:24::2/48
R4(config-if)#no shutdown
```



The IPv6 address assignment on the Routers and PCs has been completed successfully.

IPv6 recap

And now, as a final recap, I invite you to take the following test and solve the suggested exercises.

IPv6 Protocol – Test

1) Which IPv6 address is valid?

- 2031:0:130F::9C0:876A:130B
- 2001:0DB8:0000:130F:0000:0000:08GC:140B
- 2001:0DB8:0:130H::87C:140B
- 2031::130F::9C0:876A:130B

2) Which IPv6 address is the equivalent of the IPv4 interface loopback address 127.0.0.1?

- ::1
- ::
- 2000::/3
- 0::/10

3) How many bits are contained in each field of an IPv6 address?

- 24
- 4
- 8
- 16

4) Which two are features of IPv6? (Choose two)

- Multicast
- B. Broadcast
- C. Allcast
- D. Podcast

- E. Anycast

5) Which two statements describe the characteristics of IPv6 unicast addressing? (Choose two)

- Global addresses start with 2000::/3
- Link-local addresses start with FE00:/12
- Link-local addresses start with FF00::/10

- D. There is only one loopback address and it is ::1
- E. If a global address is assigned to an interface, that is the only allowable address for the interface.

IPv6 Exercises

1) What is another format for the IPv6 address:

1080:0000:0000:0000:0000:0000:1267:01A2

2) What is another format for the IPv6 address:

1234:ABCD:3123:0000:0000:0000:0000:0000

3) What is another format for the IPv6 address:

2009:D814:A7C1:EFDB:0000:0000:AC29:0001

4) What is another format for the IPv6 address

2002:ABCD:1234:9FD8:0000::

5) What is another format for the IPv6 address

2002::ABCD:1234:9FD8:0000

6) To which network does the following address belong

2020:AABC:24::1234:5678

7) To which network does the following address belong

FDFC:16AE:4583:908F:0000:0000:0000:0000

8) To which network does the following address belong

2003:DDD0:1181::1

9) To which network does the following address belong

2999:FFFF:9999:FFFF:9999:FFFF:0000:0001

IPv6 Protocol - Test - Solutions

1. A
2. A
3. D
4. A, E
5. A, D

IPv6 Exercises - Solutions

1. 1080::1267:01A2
2. 1234:ABCD:3123::
3. 2009:D814:A7C1:EFDB::AC29:0001
4. 2002:ABCD:1234:9FD8:0000:0000:0000:0000
5. 2002:0000:0000:0000:ABCD:1234:9FD8:0000
6. 2020:AABC:24::/48
7. FDFC:16AE:4583:908F::/64
8. 2003:DDD0:1181::/48
9. 2999:FFFF:9999:FFFF:9999:FFFF::/96

PS: On the next page you'll be able to see the solutions of the questions above, but before going there, I strongly suggest you try to solve the questions on your own.

Enjoyed the book? Leave a review!

I would like to thank you for completing this book and reaching the end of it (the majority of people don't get this far). If you received value from this book, I'd like to ask you for a favour. Would you be kind enough to leave a positive review? Thank you!

You can also check out one of my other books on Amazon.com by clicking this link: https://amzn.to/2zE72Wm

Wishing you the best of luck!
Ramon Nastase
IT, Education and Optimism

PS: don't forget to claim your FREE gift here: https://bit.ly/IT-GIFT
PPS: Don’t forget to check out the FREE IPv6 and IPv4 Subnetting course on Educative.io. Click here and start learning for FREE: https://www.educative.io/courses/quick-guide-ip-subnetting