Chapter 7

IP Addressing

THE FOLLOWING COMPTIA NETWORK+ EXAM OBJECTIVES ARE COVERED IN THIS CHAPTER:

- ✓ **1.3 Identify the following address formats**
  - IPv6
  - IPv4
- ✓ **1.4 Given a scenario, evaluate the proper use of the following addressing technologies and addressing schemes**
  - Addressing Technologies
    - Public vs. private
    - DHCP (static, dynamic APIPA)
  - Addressing schemes
    - Unicast
    - Multicast
    - Broadcast

One of the most important topics in any discussion of TCP/IP is IP addressing. An *IP address* is a numeric identifier assigned to each machine on an IP network. It designates the specific location of a device on the network.

An IP address is a software address, not a hardware address—the latter is hard-coded on a Network Interface Card (NIC) and used for finding hosts on a local network. IP addressing was designed to allow hosts on one network to communicate with a host on a different network regardless of the type of LANs the hosts are participating in.

Before we get into the more complicated aspects of IP addressing, you need to understand some of the basics. First I'm going to explain some of the fundamentals of IP addressing and its terminology. Then you'll learn about the hierarchical IP addressing scheme and private IP addresses.

I'll define unicast, multicast, and broadcast addresses, and then finish the chapter with a discussion on IPv6. And I promise to make it all as painless as possible.

The reason that we would even discuss IPv6 (besides to cover the objectives, of course) is because of the lack of IPv4 addresses available for use in the future networks, which we need to keep our corporate and private networks and even the Internet running. Basically, we're running out of addresses for all our new hosts! IPv6 will fix this for us.

For up-to-the-minute updates for this chapter, please see `www.lammle.com` or `www.sybex.com/go/comptianetwork+studyguide`.

IP Terminology

Throughout this chapter, you'll learn several important terms vital to your understanding of the Internet Protocol. Here are a few to get you started:

Bit A *bit* is one digit, either a 1 or a 0.

Byte A *byte* is 7 or 8 bits, depending on whether parity is used. For the rest of this chapter, always assume a byte is 8 bits.

Octet An octet, made up of 8 bits, is just an ordinary 8-bit binary number. In this chapter, the terms *byte* and *octet* are completely interchangeable.

Network address This is the designation used in routing to send packets to a remote network—for example, 10.0.0.0, 172.16.0.0, and 192.168.10.0.

Broadcast address The *broadcast address* is used by applications and hosts to send information to all hosts on a network. Examples include 255.255.255.255, which designates all networks and all hosts; 172.16.255.255, which specifies all subnets and hosts on network 172.16.0.0; and 10.255.255.255, which broadcasts to all subnets and hosts on network 10.0.0.0.

You will find the terms *subnet mask* and *slash notation* (for example, /24) used a few times in this chapter. These terms will be fully defined and used in Chapter 8, "IP Subnetting, Troubleshooting IP, and Introduction to NAT."

The Hierarchical IP Addressing Scheme

An IP address consists of 32 bits of information. These bits are divided into four sections, referred to as *octets* or bytes, and four octets sum up to 32 bits (8×4=32). You can depict an IP address using one of three methods:

- Dotted-decimal, as in 172.16.30.56
- Binary, as in 10101100.00010000.00011110.00111000
- Hexadecimal, as in AC.10.1E.38

Each of these examples validly represents the same IP address. Hexadecimal isn't used as often as dotted-decimal or binary concerning IP addressing, but you still might find an IP address stored in hexadecimal in some programs. The Windows Registry is a good example of a program that stores a machine's IP address in hex.

The 32-bit IP address is known as a structured or hierarchical address, as opposed to a flat, or nonhierarchical address. Although either type of addressing scheme can be used, *hierarchical addressing* has been chosen for a very important reason. The major advantage of this scheme is that it can handle a large number of addresses, namely 4.3 billion (a 32-bit address space with two possible values for each position—either 0 or 1—gives you 2^{32}, or 4,294,967,296). The disadvantage of the flat-addressing scheme, and the reason it's not used for IP addressing, relates to routing. If every address were unique, all routers on the Internet would need to store the address of each and every machine on the Internet. This would make efficient routing impossible, even if only a fraction of all possible addresses were used.

The solution to this problem is to use a two- or three-level hierarchical addressing scheme that is structured by network and host or by network, subnet, and host.

This two- or three-level scheme is comparable to a telephone number. The first section, the area code, designates a very large area. The second section, the prefix, narrows the scope to a local calling area. The final segment, the customer number, zooms in on the specific connection. IP addresses use the same type of layered structure. Rather than all 32 bits being treated as a unique identifier, as in flat addressing, a part of the address is designated as the

network address and the other part is designated as either the subnet and host or just the host address.

Next, I'm going to cover IP network addressing and the different classes of address used to address our networks.

Network Addressing

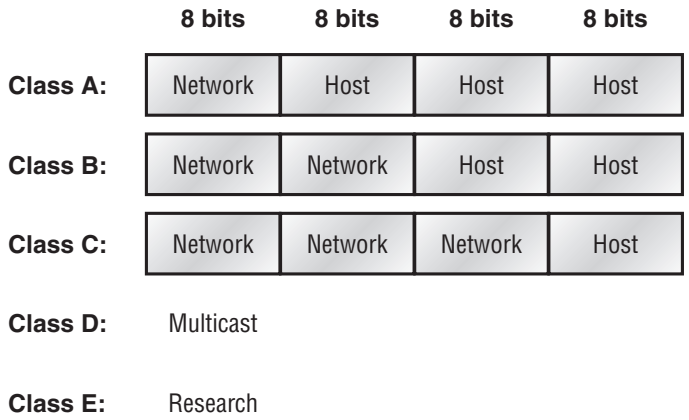
The *network address*—also called the network number—uniquely identifies each network. Every machine on the same network shares that network address as part of its IP address. In the IP address 172.16.30.56, for example, 172.16 is the network address.

The *host address* is assigned to, and uniquely identifies, each machine on a network. This part of the address must be unique because it identifies a particular machine—an individual—as opposed to a network, which is a group. This number can also be referred to as a *host address*. So in the sample IP address 172.16.30.56, the 30.56 is the host address.

The designers of the Internet decided to create classes of networks based on network size. For the small number of networks possessing a very large number of hosts, they created the rank *Class A network*. At the other extreme is the *Class C network*, which is reserved for the numerous networks with a small number of hosts. The class distinction for networks between very large and very small is predictably called the *Class B network*.

Subdividing an IP address into a network and host address is determined by the class designation of your network. Figure 7.1 summarizes the classes of networks—a subject I'll explain in much greater detail throughout this chapter.

FIGURE 7.1 Summary of the three classes of networks



To ensure efficient routing, Internet designers defined a mandate for the leading-bits section of the address for each different network class. For example, since a router knows that a Class A network address always starts with a 0, the router might be able to speed a packet on its way after reading only the first bit of its address. This is where the address schemes define the difference between a Class A, a Class B, and a Class C address. Coming up, I'll discuss the differences between these three classes, followed by a discussion of the Class D and Class E addresses. For now, know that Classes A, B, and C are the only ranges that are used to address hosts in our networks.

Class A Addresses

In a Class A network address, the first byte is assigned to the network address and the three remaining bytes are used for the host addresses. The Class A format is as follows:

network.host.host.host

For example, in the IP address 49.22.102.70, the 49 is the network address and 22.102.70 is the host address. Every machine on this particular network would begin with the distinctive network address of 49.

Class A network addresses are 1 byte long, with the first bit of that byte reserved, and the 7 remaining bits available for manipulation, or addressing. As a result, the maximum number of Class A networks that can be created is 128. Why? Well, each of the 7 bit positions can be either a 0 or a 1, and 2^7 gives you 128.

The designers of the IP address scheme said that the first bit of the first byte in a Class A network address must always be off, or 0. This means a Class A address must be between 0 and 127 in the first byte, inclusive.

Consider the following network address:

0xxxxxxx

If we turn the other 7 bits all off and then turn them all on, we'll find the Class A range of network addresses:

00000000 = 0
01111111 = 127

So, a Class A network is defined in the first octet between 0 and 127, and it can't be less or more.

To complicate matters further, the network address of all 0s (0000 0000) is reserved to designate the default route (see Table 7.1). Additionally, the address 127, which is reserved for diagnostics, can't be used either, which means that you can really only use the numbers 1 to 126 to designate Class A network addresses. This means the actual number of usable Class A network addresses is 128 minus 2, or 126.

Each Class A address has 3 bytes (24-bit positions) for the host address of a machine. This means there are 2^{24}—or 16,777,216—unique combinations and, therefore, precisely that many potential unique host addresses for each Class A network. Because host addresses with the two patterns of all 0s and all 1s are reserved, the actual maximum usable number of hosts for a Class A network is 2^{24} minus 2, which equals 16,777,214. Either way, you can see that's a seriously huge number of hosts to have a network segment!

Here's an example of how to figure out the valid host IDs in a Class A network address:

- All host bits off is the network address: 10.0.0.0.
- All host bits on is the broadcast address: 10.255.255.255.

The valid hosts are the numbers in between the network address and the broadcast address: 10.0.0.1 through 10.255.255.254. Notice that 0s and 255s can be valid host IDs. All you need to remember when trying to find valid host addresses is that the host bits can't ever be all be turned off or all turned on at the same time.

TABLE 7.1 Reserved IP Addresses

Address	Function
Network address of all 0s	Interpreted to mean "this network or segment."
Network address of all 1s	Interpreted to mean "all networks."
Network 127.0.0.1	Reserved for loopback tests. Designates the local host and allows that host to send a test packet to itself without generating network traffic.
Host address of all 0s	Interpreted to mean "network address" or any host on specified network.
Host address of all 1s	Interpreted to mean "all hosts" on the specified network; for example, 128.2.255.255 means "all hosts" on network 128.2 (Class B address).
Entire IP address set to all 0s	Used by Cisco routers to designate the default route. Could also mean "any network."
Entire IP address set to all 1s (same as 255.255.255.255)	Broadcast to all hosts on the current network; sometimes called an "all 1s broadcast" or limited broadcast.

Class B Addresses

In a Class B network address, the first 2 bytes are assigned to the network address and the remaining 2 bytes are used for host addresses. The format is as follows:

network.network.host.host

For example, in the IP address 172.16.30.56, the network address is 172.16 and the host address is 30.56.

With a network address being 2 bytes (8 bits each), we're left with 2^{16} unique combinations. But the Internet designers decided that all Class B network addresses should start with the binary digit 1, then 0. This leaves 14 bit positions available to manipulate, so in reality, we get 16,384 (that is, 2^{14}) unique Class B network addresses.

In a Class B network, the RFCs state that the first bit of the first byte must always be turned on but the second bit must always be turned off. If we turn the other 6 bits all off and then all on, we will find the range for a Class B network:

10000000 = 128
10111111 = 191

As you can see, a Class B network is defined when the first byte is configured from 128 to 191.

A Class B address uses 2 bytes for host addresses. This is 2^{16} minus the two reserved patterns (all 0s and all 1s), for a total of 65,534 possible host addresses for each Class B network.

Here's an example of how to find the valid hosts in a Class B network:

- All host bits turned off is the network address: 172.16.0.0.
- All host bits turned on is the broadcast address: 172.16.255.255.

The valid hosts would be the numbers in between the network address and the broadcast address: 172.16.0.1 through 172.16.255.254.

Class C Addresses

The first 3 bytes of a Class C network address are dedicated to the network portion of the address, with only 1 measly byte remaining for the host address. Here's the format:

network.network.network.host

Using the example IP address 192.168.100.102, the network address is 192.168.100 and the host address is 102.

In a Class C network address, the first three bit positions are always the binary 110. The calculation is as follows: 3 bytes, or 24 bits, minus 3 reserved positions leaves 21 positions. Hence, there are 2^{21}, or 2,097,152, possible Class C networks.

For Class C networks, the RFCs define the first 2 bits of the first octet as always turned on, but the third bit can never be on. Following the same process as the previous classes, convert from binary to decimal to find the range. Here's the range for a Class C network:

```
11000000 = 192
11011111 = 223
```

So, if you see an IP address with a range from 192 up to 223, you'll know it's a Class C IP address.

Each unique Class C network has 1 byte to use for host addresses. This gets us to 2^8 or 256, minus the two reserved patterns of all 0s and all 1s, for a total of 254 available host addresses for each Class C network.

Here's an example of how to find a valid host ID in a Class C network:

- All host bits turned off is the network ID: 192.168.100.0.
- All host bits turned on is the broadcast address: 192.168.100.255.

The valid hosts would be the numbers in between the network address and the broadcast address: 192.168.100.1 through 192.168.100.254.

Class D and E Addresses

The addresses 224 to 255 are reserved for Class D and E networks. Class D (224–239) is used for multicast addresses and Class E (240–255) for scientific purposes. But they're really beyond the scope of this book, so I'm not going to go into detail about them here. But you do need to know that the multicast range is from 224.0.0.0 through 239.255.255.255.

Special Purposes of Network Addresses

Some IP addresses are reserved for special purposes, so network administrators can't ever assign these addresses to hosts. Table 7.1 lists the members of this exclusive little club and the reasons why they're included in it.

Private IP Addresses

The people who created the IP addressing scheme also created what we call *private IP addresses*. These addresses can be used on a private network, but they're not routable through the Internet. This is designed for the purpose of creating a measure of much-needed security, but it also conveniently saves valuable IP address space.

If every host on every network had to have real routable IP addresses, we would have run out of available IP addresses to hand out years ago. But by using private IP addresses, ISPs, corporations, and home users only need a relatively tiny group of bona fide IP addresses to connect their networks to the Internet. This is economical because they can use private IP addresses on their inside networks and get along just fine.

To accomplish this task, the ISP and the corporation—the end users, no matter who they are—need to use something called *Network Address Translation (NAT)*, which basically takes a private IP address and converts it for use on the Internet. Many people can use the same real IP address to transmit out onto the Internet. Doing things this way saves megatons of address space—a very good thing for us all!

Real World Scenario

So, What Private IP Address Should I Use?

That's a really great question: Should you use Class A, Class B, or even Class C private addressing when setting up your network? Let's take Acme Corporation in SF as an example. This company is moving into a new building and needs a whole new network (what a treat this is!). It has 14 departments, with about 70 users in each. You could probably squeeze one or two Class C addresses to use, or maybe you could use a Class B, or even a Class A just for fun.

The rule of thumb in the consulting world is, when you're setting up a corporate network—regardless of how small it is—you should use a Class A network address because it gives you the most flexibility and growth options. For example, if you used the 10.0.0.0 network address with a /24 mask, then you'd have 65,536 networks, each with 254 hosts. Lots of room for growth with that network!

(A /24 tells you that a subnet mask has 24 bits out of 32 bits turned on for network subnetting a network. This will be covered in more detail in Chapter 8.)

But if you're setting up a home network, you'd opt for a Class C address because it is the easiest for people to understand and configure. Using the default Class C mask gives you one network with 254 hosts—plenty for a home network.

With the Acme Corporation, a nice 10.1.*x*.0 with a /24 mask (the *x* is the subnet for each department) makes this easy to design, install, and troubleshoot.

The reserved private addresses are listed in Table 7.2.

TABLE 7.2 Reserved IP Address Space

Address Class	Reserved Address Space
Class A	10.0.0.0 through 10.255.255.255
Class B	172.16.0.0 through 172.31.255.255
Class C	192.168.0.0 through 192.168.255.255

APIPA

I discussed this in Chapter 6, "Introduction to Internet Protocol (IP)," but it is worth repeating here. What happens if you have a few hosts connected together with a switch or hub and you don't have a DHCP server? You can add static IP information to a host or you can let Windows provides what is called Automatic Private IP Addressing (APIPA). I don't recommend this, but APIPA is a "feature" so you do need to remember it, hence mentioning it two chapters in a row!

With APIPA, clients can automatically self-configure an IP address and subnet mask, which is the minimum information needed for hosts to communicate when a DHCP server isn't available.

The IP address range for APIPA is 169.254.0.1 through 169.254.255.254. The client also configures itself with a default class B subnet mask of 255.255.0.0.

Broadcast Addresses

Most people use the term *broadcast* as a generic term, and most of the time, we understand what they mean. But not always. For example, you might say, "The host broadcasted through a router to a DHCP server," but, well, it's pretty unlikely that this would ever

really happen. What you probably mean—using the correct technical jargon—is, "The host broadcasted for an IP address; a router then forwarded this as a unicast packet to the DHCP server." Oh, and remember that with IPv4, broadcasts are pretty important, but with IPv6, there aren't any broadcasts sent at all—now there's something to look forward to finding out about in the next section on IPv6!

Okay, I've referred to broadcast addresses throughout some of the earlier chapters and even showed you some examples. But I really haven't gone into the different terms and uses associated with them yet. It's about time I did, so here are the four different broadcast (generic term *broadcast)* types that I'd like to define for you:

Layer 2 broadcasts These are sent to all hosts on a LAN.

Broadcasts (Layer 3) These are sent to all hosts on the network.

Unicast These are sent to a single destination host.

Multicast These are packets sent from a single source and transmitted to many devices on different networks.

First, understand that Layer 2 broadcasts are also known as *hardware broadcasts*—they only go out on a LAN, and they don't go past the LAN boundary (router). The typical hardware address is 6 bytes (48 bits) and looks something like 0c.43.a4.f3.12.c2. The broadcast would be all 1s in binary, which would be all *F*s in hexadecimal, as in FF.FF.FF.FF.FF.FF.

Then there are the plain old broadcast addresses at Layer 3. Broadcast messages are meant to reach all hosts on a broadcast domain. These are the network broadcasts that have all host bits on. Here's an example that you're already familiar with: The network address of 172.16.0.0 255.255.0.0 would have a broadcast address of 172.16.255.255—all host bits on. Broadcasts can also be "all networks and all hosts," as indicated by 255.255.255.255. A good example of a broadcast message is an Address Resolution Protocol (ARP) request. When a host has a packet, it knows the logical address (IP) of the destination. To get the packet to the destination, the host needs to forward the packet to a default gateway if the destination resides on a different IP network. If the destination is on the local network, the source will forward the packet directly to the destination. Because the source doesn't have the MAC address to which it needs to forward the frame, it sends out a broadcast, something that every device in the local broadcast domain will listen to. This broadcast says, in essence, "If you are the owner of IP address 192.168.2.3, please forward your MAC address to me," with the source giving the appropriate information.

A unicast is different because it's a broadcast packet that goes from 255.255.255.255 to an actual destination IP address—in other words, it's directed to a specific host. A DHCP client request is a good example of how a unicast works. Here's an example: Your host on a LAN sends out an FF.FF.FF.FF.FF.FF Layer 2 broadcast and 255.255.255.255 Layer 3 destination broadcast looking for a DHCP server on the LAN. The router will see that this is a broadcast meant for the DHCP server because it has a destination port number of 67 (BootP server) and will forward the request to the IP address of the DHCP server on another LAN. So, basically, if your DHCP server IP address is 172.16.10.1, your host just sends out a 255.255.255.255 DHCP client broadcast request, and the router changes that broadcast to the specific destination address of 172.16.10.1. (In order for the router

to provide this service, you need to configure the interfaces with the `ip helper-address` command—this is not a default service.)

Multicast is a different beast entirely. At first glance, it appears to be a hybrid of unicast and broadcast communication, but that isn't quite the case. Multicast does allow point-to-multipoint communication, which is similar to broadcasts, but it happens in a different manner. The crux of multicast is that it enables multiple recipients to receive messages without flooding the messages to all hosts on a broadcast domain.

Multicast works by sending messages or data to IP multicast group addresses. Routers then forward copies (unlike broadcasts, which are not forwarded) of the packet out every interface that has hosts subscribed to that group address. This is where multicast differs from broadcast messages—with multicast communication, copies of packets, in theory, are sent only to subscribed hosts. When I say "in theory," this means that the hosts will receive, for example, a multicast packet destined for 224.0.0.10 (this is a Routing Information Protocol [RIP] packet, and only a router running the RIP protocol will read these). All hosts on the broadcast LAN (Ethernet is a broadcast multi-access LAN technology) will pick up the frame, read the destination address, and immediately discard the frame, unless they are in the multicast group. This saves PC processing, not LAN bandwidth. Multicasting can cause severe LAN congestion, in some instances, if not implemented carefully.

There are several different groups that users or applications can subscribe to. The range of multicast addresses starts with 224.0.0.0 and goes through 239.255.255.255. As you can see, this range of addresses falls within IP Class D address space based on classful IP assignment.

Internet Protocol Version 6 (IPv6)

People refer to IPv6 as "the next-generation Internet protocol," and it was originally created as the answer to IPv4's inevitable, looming address-exhaustion crisis. Though you've probably heard a thing or two about IPv6 already, it has been improved even further in the quest to bring us the flexibility, efficiency, capability, and optimized functionality that can truly meet our ever-increasing needs. The capacity of its predecessor, IPv4, pales in comparison—and that's the reason it will eventually fade into history completely.

The IPv6 header and address structure has been completely overhauled, and many of the features that were basically just afterthoughts and addendums in IPv4 are now included as full-blown standards in IPv6. It's well equipped, poised, and ready to manage the mind-blowing demands of the Internet to come.

Why Do We Need IPv6?

Well, the short answer is, because we need to communicate, and our current system isn't really cutting it anymore—kind of like how the Pony Express can't compete with airmail. Just look at how much time and effort we've invested in coming up with slick new ways to conserve bandwidth and IP addresses. We've even come up with Variable Length Subnet Masks (VLSMs) in our struggle to overcome the worsening address drought.

It's reality, the number of people and devices that connect to networks increases each and every day. That's not a bad thing at all—we're finding new and exciting ways to communicate to more people all the time; something that's become integral to our culture today. In fact, it's now pretty much a basic human need. But the forecast isn't exactly blue skies and sunshine because, as I alluded to in this chapter's introduction, IPv4, upon which our ability to communicate is presently dependent, is going to run out of addresses for us to use. IPv4 has only about 4.3 billion addresses available—in theory—and we know that we don't even get to use all of those. There really are only about 250 million addresses that can be assigned to devices. Sure, the use of Classless Inter-Domain Routing (CIDR) and NAT has helped to extend the inevitable dearth of addresses, but the truth is we will run out of them, and it's going to happen within a few years. China is barely online, and we know a huge population of people and corporations there surely want to be. There are a lot of reports that give us all kinds of numbers, but all you really need to think about to convince yourself that I'm not just being an alarmist is the fact that there are about 6.5 billion people in the world today, and it's estimated that just over 10 percent of that population is connected to the Internet—wow! IPv6 to the rescue!

That statistic is basically screaming at us the ugly truth that, based on IPv4's capacity, every person can't even have a computer with an IP address—let alone all the other devices we use with them. I have more than one computer, and it's pretty likely you do too. And I'm not even including in the mix phones, laptops, game consoles, fax machines, routers, switches, and a mother lode of other devices we use every day! So I think I've made it pretty clear that we've got to do something before we run out of addresses and lose the ability to connect with each other as we know it. And that "something" just happens to be implementing IPv6.

The Benefits of and Uses for IPv6

What's so fabulous about IPv6? Is it really the answer to our coming dilemma? Is it really worth it to upgrade from IPv4? All good questions—you may even think of a few more. Of course, there's going to be that group of people with the time-tested and well-known "resistance to change syndrome," but don't listen to them. If we had done that years ago, we'd still be waiting weeks, even months for our mail to arrive via horseback. Instead, just know that the answer is a resounding YES! Not only does IPv6 give us lots of addresses (3.4×10^{38} = definitely enough), but there are many other features built into this version that make it well worth the cost, time, and effort required to migrate to it.

Today's networks, as well as the Internet, have a ton of unforeseen requirements that simply were not considerations when IPv4 was created. We've tried to compensate with a collection of add-ons that can actually make implementing them more difficult than they would be if they were mandated by a standard. By default, IPv6 has improved upon and included many of those features as standard and mandatory. One of these sweet new standards is IPSec—a feature that provides end-to-end security and that I'll cover in Chapter 16, "Wide Area Networks." Another little beauty is known as *mobility*, and as its name suggests, it allows a device to roam from one network to another without dropping connections.

But it's the efficiency features that are really going to rock the house! For starters, the header in an IPv6 packet has half the fields, and they are aligned to 64 bits, which gives us some seriously souped-up processing speed—compared to IPv4, lookups happen at light speed. Most of the information that used to be bound into the IPv4 header was taken out, and now you can choose to put it, or parts of it, back into the header in the form of optional extension headers that follow the basic header fields.

And of course there's that whole new universe of addresses (3.4×10^{38}) we talked about already. But where did we get them? Did that Chris Angel–Mindfreak dude just show up and, blammo, they all materialized? The obvious answer is no; but that huge proliferation of address had to come from somewhere, right? Well, it just so happens that IPv6 gives us a substantially larger address space, meaning the address is a whole lot bigger—four times bigger, as a matter of fact! An IPv6 address is actually 128 bits in length, and no worries—I'm going to break down the address piece by piece and show you exactly what it looks like coming up in the section "IPv6 Addressing and Expressions." For now, let me just say that all that additional room permits more levels of hierarchy inside the address space and a more flexible address architecture. It also makes routing much more efficient and scalable because the addresses can be aggregated a lot more effectively. And IPv6 also allows multiple addresses for hosts and networks. Plus, the new version of IP now includes an expanded use of multicast communication (one device sending to many hosts or to a select group), which will also join in to boost efficiency on networks because communications will be more specific.

IPv4 uses broadcasts very prolifically, causing a bunch of problems, the worst of which is of course the dreaded broadcast storm—an uncontrolled deluge of forwarded broadcast traffic that can bring an entire network to its knees and devour every last bit of bandwidth. Another nasty thing about broadcast traffic is that it interrupts each and every device on the network. When a broadcast is sent out, every machine has to stop what it's doing and analyze the traffic, whether the broadcast is meant for it or not.

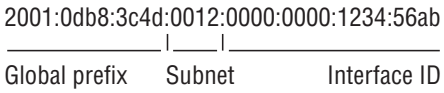
But smile, everyone: There is no such thing as a broadcast in IPv6 because it uses multicast traffic instead. And there are two other types of communication as well: unicast, which is the same as it is in IPv4, and a new type called *anycast*. Anycast communication allows the same address to be placed on more than one device so that when traffic is sent to one device addressed in this way, it is routed to the nearest host that shares the same address. This is just the beginning—we'll get more into the various types of communication in the section "Address Types."

IPv6 Addressing and Expressions

Just as understanding how IP addresses are structured and used is critical with IPv4 addressing, it's also vital when it comes to IPv6. You've already read about the fact that at 128 bits, an IPv6 address is much larger than an IPv4 address. Because of this, as well as the new ways the addresses can be used, you've probably guessed that IPv6 will be more complicated to manage. But no worries! As I said, I'll break down the basics and show you what the address looks like, how you can write it, and what many of its common uses are. It's going to be a little weird at first, but before you know it, you'll have it nailed.

So let's take a look at Figure 7.2, which has a sample IPv6 address broken down into sections.

FIGURE 7.2 IPv6 address example



As you can now see, the address is truly much larger—but what else is different? Well, first, notice that it has eight groups of numbers instead of four and also that those groups are separated by colons instead of periods. And hey, wait a second… there are letters in that address! Yep, the address is expressed in hexadecimal just like a MAC address is, so you could say this address has eight 16-bit hexadecimal colon-delimited blocks. That's already quite a mouthful, and you probably haven't even tried to say the address out loud yet.

One other thing I want to point out is for when you set up your test network to play with IPv6, because I know you're going to want to do that. When you use a web browser to make an HTTP connection to an IPv6 device, you have to type the address into the browser with brackets around the literal address. Why? Well, a colon is already being used by the browser for specifying a port number. So basically, if you don't enclose the address in brackets, the browser will have no way to identify the information.

Here's an example of how this looks:

```
http://[2001:0db8:3c4d:0012:0000:0000:1234:56ab]/default.html
```

Now obviously, if you could, you would rather use names to specify a destination (like `www.lammle.com`); but even though it's definitely going to be a pain in the rear, you just have to accept the fact that sometimes you have to bite the bullet and type in the address number. It should be pretty clear that DNS is going to become extremely important when implementing IPv6.

Shortened Expression

The good news is, there are a few tricks to help rescue you when writing these monster addresses. For one thing, you can actually leave out parts of the address to abbreviate it, but to get away with doing that you have to follow a couple of rules. First, you can drop any leading zeros in each of the individual blocks. After you do that, the sample address from earlier would then look like this:

```
2001:db8:3c4d:12:0:0:1234:56ab
```

Okay, that's a definite improvement—at least you don't have to write all of those extra zeros! But what about whole blocks that don't have anything in them except zeros? Well, you can kind of lose those too—at least some of them. Again referring to our sample address, you can remove the two blocks of zeros by replacing them with double colons, like this:

```
2001:db8:3c4d:12::1234:56ab
```

Cool—you replaced the blocks of all zeros with double colons. The rule you have to follow to get away with this is that you can only replace one contiguous block of zeros in an address. So if my address has four blocks of zeros and each of them is separated, I don't get to replace them all. Check out this example:

```
2001:0000:0000:0012:0000:0000:1234:56ab
```

And just know that you *can't* do this:

```
2001::12::1234:56ab
```

Instead, this is the best that you can do:

```
2001::12:0:0:1234:56ab
```

The reason why this example is your best shot is that if you remove two sets of zeros, the device looking at the address will have no way of knowing where the zeros go back in. Basically, the router would look at the incorrect address and say, "Well, do I place two blocks into the first set of double colons and two into the second set, or do I place three blocks into the first set and one block into the second set?" And on and on it would go, because the information the router needs just isn't there.

Address Types

We're all familiar with IPv4's unicast, broadcast, and multicast addresses, which basically define who or at least how many other devices we're talking to. But as I mentioned, IPv6 adds to that trio and introduces the anycast. Broadcasts, as we know them, have been eliminated in IPv6 because of their cumbersome inefficiency.

Let's find out what each of these types of IPv6 addressing and communication methods do for us:

Unicast Packets addressed to a unicast address are delivered to a single interface. For load balancing, multiple interfaces can use the same address. There are a few different types of unicast addresses, but we don't need to get into that here.

Global unicast addresses These are your typical publicly routable addresses, and they're the same as they are in IPv4.

Link-local addresses These are like the private addresses in IPv4 in that they're not meant to be routed. Think of them as a handy tool that gives you the ability to throw a temporary LAN together for meetings or for creating a small LAN that's not going to be routed but still needs to share and access files and services locally.

Unique local addresses These addresses are also intended for non-routing purposes, but they are nearly globally unique, so it's unlikely you'll ever have one of them overlap with any other address. Unique local addresses were designed to replace site-local addresses, so they basically do almost exactly what IPv4 private addresses do—allow communication throughout a site while being routable to multiple local networks.

Multicast Again, as in IPv4, packets addressed to a multicast address are delivered to all interfaces identified by the multicast address. Sometimes people call them *one-to-many addresses*. It's really easy to spot multicast addresses in IPv6 because they always start with *FF*."

Anycast Like multicast addresses, an anycast address identifies multiple interfaces, but there's a big difference: the anycast packet is delivered to only one address—actually, to the first IPv6 address it finds defined in terms of routing distance. And again, this address is special because you can apply a single address to more than one interface. You could call them one-to-one-of-many addresses, but just saying "anycast" is a lot easier.

You're probably wondering if there are any special, reserved addresses in IPv6 because you know they're there in IPv4. Well, there are—plenty of them! Let's go over them now.

Special Addresses

I'm going to list some of the addresses and address ranges that you should definitely make a point to remember because you'll eventually use them. They're all special or reserved for specific use; but unlike IPv4, IPv6 gives us a galaxy of addresses, so reserving a few here and there doesn't hurt a thing.

0:0:0:0:0:0:0:0 Equals ::. This is the equivalent of IPv4's 0.0.0.0 and is typically the source address of a host when you're using stateful configuration.

0:0:0:0:0:0:0:1 Equals ::1. The equivalent of 127.0.0.1 in IPv4.

0:0:0:0:0:0:192.168.100.1 This is how an IPv4 address would be written in a mixed IPv6/IPv4 network environment.

2000::/3 The global unicast address range.

FC00::/7 The unique local unicast range.

FE80::/10 The link-local unicast range.

FF00::/8 The multicast range.

3FFF:FFFF::/32 Reserved for examples and documentation.

2001:0DB8::/32 Also reserved for examples and documentation.

2002::/16 Used with 6to4, which is the transition system—the structure that allows IPv6 packets to be transmitted over an IPv4 network without the need to configure explicit tunnels.

Summary

In this chapter, I covered the very basics of both IPv4 and IPv6 and how they work in an internetwork (remember that if the word *IP* is used alone, it is referring to just IPv4). As you now know by reading this chapter, even when discussing and configuring the basics,

there is a lot to understand—and we just scratched the surface. But trust me when I say this—you now know more than you'll need to meet the Network+ objectives.

I discussed in detail the difference between each class of address and how to find a network address, broadcast address, and valid host range.

Last, I talked about why we need IPv6 and the benefits associated with it. I followed that up by covering addressing with IPv6 as well as how to use the shortened expressions. And during the talk on addressing with IPv6, I showed you the different address types, plus the special addresses reserved in IPv6.

This next chapter is very important, but it's one that some people find rather challenging; so take a break and get ready for a really fun, but long chapter on IP subnetting. I promise not to torture you too much!

Exam Essentials

Remember the Class A range. The IP range for a Class A network is 1–126. This provides 8 bits of network addressing and 24 bits of host addressing by default.

Remember the Class B range. The IP range for a Class B network is 128–191. Class B addressing provides 16 bits of network addressing and 16 bits of host addressing by default.

Remember the Class C range. The IP range for a Class C network is 192–223. Class C addressing provides 24 bits of network addressing and 8 bits of host addressing by default.

Remember the Private IP ranges. The Class A private address range is 10.0.0.0 through 10.255.255.255.

The Class B private address range is 172.16.0.0 through 172.31.255.255.

The Class C private address range is 192.168.0.0 through 192.168.255.255.

Understand why we need IPv6. Without IPv6, the world would soon be depleted of IP addresses.

Understand link-local. Link-local is like an IPv4 private IP address, but it can't be routed at all, not even in your organization.

Understand unique local. This, like link-local, is like private IP addresses in IPv4 and cannot be routed to the Internet. However, the difference between link-local and unique local is that unique local can be routed within your organization or company.

Remember IPv6 addressing. IPv6 addressing is not like IPv4 addressing. IPv6 addressing has much more address space and is 128 bits long, represented in hexadecimal, unlike IPv4, which is only 32 bits long and represented in decimals.

Written Lab

1. What is the valid range of values that may appear in an IPv4 octet? Give your answer in decimal as well as binary.
2. Name some of the benefits of IPv6 over IPv4.
3. What is the term for the auto-configuration technology responsible for addresses that start with 169.254?
4. What does the IP Properties selection Obtain an IP Address Automatically indicate?
5. What effect will an inappropriate DHCP server have on hosts using static IP addresses?
6. What is the name for a 48-bit (6-byte) numerical address physically assigned to a network interface, such as a NIC?
7. What gives IPv6 the ability to reference more addresses than IPv4?
8. What predecessor to DHCP, on which DHCP is based, was used to assign a workstation its IP information and to supply it with a boot image?
9. What is the Class C range of values for the first octet in decimal and in binary?
10. What is the 127.0.0.1 address used for?

(The answers to the Written Lab can be found following the answers to the Review Questions for this chapter.)

Review Questions

1. Which of the following addresses is not allowed on the Internet?
   - **A.** 191.192.168.1
   - **B.** 191.168.169.254
   - **C.** 172.32.255.0
   - **D.** 172.31.12.251

2. A host automatically configured with an address from which of the following ranges indicates an inability to contact a DHCP server?
   - **A.** 169.254.0.X with a mask of 255.255.255.0
   - **B.** 169.254.X.X with a mask of 255.255.0.0
   - **C.** 169.254.X.X with a mask of 255.255.255.0
   - **D.** 169.255.X.X with a mask of 255.255.0.0

3. Which statement regarding private IP addresses is most accurate?
   - **A.** Private addresses cannot be used in intranets that require routing.
   - **B.** Private addresses must be assigned by a registrar or ISP.
   - **C.** A remote host across the Internet cannot ping your host if it has a private address.
   - **D.** Private addresses can only be used by a single administrative domain.

4. Which of the following is a valid Class A address?
   - **A.** 191.10.0.1 255.0.0.0
   - **B.** 127.10.0.1 255.0.0.0
   - **C.** 128.10.0.1 255.0.0.0
   - **D.** 126.10.0.1 255.0.0.0

5. Which of the following is a valid Class B address?
   - **A.** 10.1.1.1 255.255.0.0
   - **B.** 126.1.1.1 255.255.0.0
   - **C.** 129.1.1.1 255.255.0.0
   - **D.** 192.168.1.1 255.255.0.0

6. Which of the following describes a broadcast address?
   - **A.** All network bits are on (1s).
   - **B.** All host bits are on (1s).
   - **C.** All network bits are off (0s).
   - **D.** All host bits are off (0s).

7. Which of the following is a Layer 2 broadcast?
   - **A.** FF:FF:FF:EE:EE:EE
   - **B.** FF:FF:FF:FF:FF:FF
   - **C.** 255.255.255.255
   - **D.** 255.0.0.0

8. In a class C IP address, how long is the network address?
   - **A.** 8 bits
   - **B.** 16 bits
   - **C.** 24 bits
   - **D.** 32 bits

9. Which of the following is true when describing a unicast address?
   - **A.** Packets addressed to a unicast address are delivered to a single interface.
   - **B.** These are your typical publicly routable addresses, just like regular publicly routable addresses in IPv4.
   - **C.** These are like private addresses in IPv4 in that they are not meant to be routed.
   - **D.** These addresses are meant for nonrouting purposes, but they are almost globally unique so it is unlikely they will have an address overlap.

10. A host is rebooted and you view the IP address that it was assigned. The address is 169.123.13.34. Which of the following happened?
    - **A.** The host received an APIPA address
    - **B.** The host received a multicast address
    - **C.** The host received a public address
    - **D.** The host received a private address

11. An IPv4 addresses uses 32 bits. How many bits is an IPv6 address?
    - **A.** 64
    - **B.** 128
    - **C.** 192
    - **D.** 255

12. Which of the following is true when describing a multicast address?
    - **A.** Packets addressed to a unicast address from a multicast address are delivered to a single interface.
    - **B.** Packets are delivered to all interfaces identified by the address. This is also called a one-to-many address.
    - **C.** It identifies multiple interfaces and is delivered to only one address. This address can also be called one-to-one-of-many.
    - **D.** These addresses are meant for nonrouting purposes, but they are almost globally unique so it is unlikely they will have an address overlap.

13. Which of the following is true when describing an anycast address?
    A. Packets addressed to a unicast address from an anycast address are delivered to a single interface.
    B. Packets are delivered to all interfaces identified by the address. This is also called a one-to-many address.
    C. This address identifies multiple interfaces, and the anycast packet is delivered to only one address. This address can also be called one-to-one-of-many.
    D. These addresses are meant for nonrouting purposes, but they are almost globally unique so it is unlikely they will have an address overlap.

14. You want to ping the loopback address of your local host. Which two addresses could you type?
    A. `ping 127.0.0.1`
    B. `ping 0.0.0.0`
    C. `ping ::1`
    D. `trace 0.0.::1`

15. What two statements about IPv6 addresses are true?
    A. Leading zeros are required.
    B. Two colons (::) are used to represent successive hexadecimal fields of zeros.
    C. Two colons (::) are used to separate fields.
    D. A single interface will have multiple IPv6 addresses of different types.

16. What two statements about IPv4 and IPv6 addresses are true?
    A. An IPv6 address is 32 bits long, represented in hexadecimal.
    B. An IPv6 address is 128 bits long, represented in decimal.
    C. An IPv4 address is 32 bits long, represented in decimal.
    D. An IPv6 address is 128 bits long, represented in hexadecimal.

17. Which of the following is a Class C network address?
    A. 10.10.10.0 255.255.255.0
    B. 127.0.0.1 255.255.255.0
    C. 128.0.0.0 255.255.0.0
    D. 192.255.254.0 255.255.255.0

18. Which two of the following are private IP addresses? (Choose two.)
    A. 12.0.0.1
    B. 168.172.19.39
    C. 172.20.14.36
    D. 172.33.194.30
    E. 192.168.24.43

19. Which of the following is a valid IP address that can be used on the Internet (meaning the public addressing scheme)?

A. 10.10.1.1

B. 168.16.1.1

C. 234.1.1.1

D. 172.30.1.1

20. Which of the following is an invalid IP address for a host?

A. 10.0.0.1

B. 128.0.0.1

C. 224.0.0.1

D. 172.0.0.1

Answers to Review Questions

1. D. The addresses in the range 172.16.0.0 through 172.31.255.255 are all considered private, based on RFC 1918. Use of these addresses on the Internet is prohibited so that they can be used simultaneously in different administrative domains without concern for conflict. Some experts in the industry believe these addresses are not routable, which is not true.

2. B. APIPA uses the link-local private address range of 169.254.0.0 through 169.254.255.255 and a subnet mask of 255,255,0,0 (see RFC 3330). APIPA addresses are used by DHCP clients that cannot contact a DHCP server and have no static alternate configuration. These addresses are not Internet-routable and cannot, by default, be used across routers on an internetwork.

3. C. Private IP addresses are not routable over the Internet, as either source or destination addresses. Because of that fact, any entity that wishes to use such addresses internally can do so without causing conflicts with other entities and without asking permission of any registrar or service provider. Despite not being allowed on the Internet, private IP addresses are fully routable on private intranets.

4. D. The Class A range is 1–126 in the first octet/byte, so this makes answer B incorrect. Only answer D is a valid Class A address.

5. C. The Class B range is 129–191 in the first octet/byte. Only answer C is a valid Class B address.

6. B. If you turned on all host bits (all of the host bits are 1s), this would be a broadcast address for that network.

7. B. A Layer 2 broadcast is also referred to as a MAC address broadcast, which is in hexadecimal and is FF:FF:FF:FF:FF:FF

8. C. A default class C subnet mask is 255.255.255.0, which means that the first three octets are the network number, or first 24 bits.

9. A. Packets addressed to a unicast address are delivered to a single interface. For load balancing, multiple interfaces can use the same address.

10. C. I wonder how many picked APIPA address as your answer? An APIPA address is 169.254.X.X. The host address in this question is a public address. Somewhat of a tricky question if you did not read carefully.

11. B. An IPv6 address is 128 bits in size.

12. B. Packets addressed to a multicast address are delivered to all interfaces identified by the multicast address, the same as in IPv4. A multicast address is also called a one-to-many address. You can tell multicast addresses in IPv6 because they always start with FF.

13. C. Anycast addresses identify multiple interfaces, which is the same as multicast; however, the big difference is that the anycast packet is delivered to only one address: the first one it finds defined in the terms of routing distance. This address can also be called one-to-one-of-many.

14. A, C. The loopback address with IPv4 is 127.0.0.1. With IPv6, that address is ::1.

15. B, D. In order to shorten the written length of an IPv6 address, successive fields of zeros may be replaced by double colons. In trying to shorten the address further, leading zeros may also be removed. Just as with IPv4, a single device's interface can have more than one address; with IPv6 there are more types of addresses and the same rule applies. There can be link-local, global unicast, and multicast addresses all assigned to the same interface.

16. C, D. IPv4 addresses are 32 bits long and are represented in decimal format. IPv6 addresses are 128 bits long and represented in hexadecimal format.

17. D. Only answer D is in the Class C range of 192–224. It might look wrong because there is a 255 in the address, but this is not wrong—you can have a 255 in a network address.

18. C, E. The Class A private address range is 10.0.0.0 through 10.255.255.255. The Class B private address range is 172.16.0.0 through 172.31.255.255, and the Class C private address range is 192.168.0.0 through 192.168.255.255.

19. B. The private address range is 10.0.0.0 through 10.255.255.255, 172.16.0.0 through 172.31.255.255 and 192.168.0.0 through 192.168.255.255. Also, 224.0.0.0 through 239.255.255.255 is reserved for multicast addressing.

20. C. Answer C is a multicast address and cannot be used to address hosts.

Answers to Written Lab

1. The range of values that an IPv4 octet can take on is 0 through 255 in decimal, which stems from the values 00000000 through 11111111 in binary.
2. IPv6 has the following characteristics, among others, that make it preferable to IPv4: more available addresses, simpler header, options for authentication and other security.
3. Automatic Private IP Addressing (APIPA) is the technology that results in hosts automatically configuring themselves with addresses that begin with 169.254.
4. Filling in the Obtain an IP Address Automatically radio button in IP Properties configures the host as a DHCP client.
5. None; DHCP servers cannot override statically assigned IP information.
6. A MAC address, sometimes called a hardware address or even a burned-in address.
7. The fact that it has 128-bit (16-octet) addresses, compared to IPv4's 32-bit (4-octet) addresses.
8. BootP, the Bootstrap Protocol, used the same port numbers as DHCP but supplied more simplified information to a diskless workstation and allowed the workstation to download a remote boot image.
9. 192–223, 110*xxxxx*.
10. Loopback or diagnostics.