Application OpenedApplication Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

**Module**

**3**

**Addressing**

* [Module Introduction](javascript://)
* **3-1**[Addressing Overview](javascript://)
  + **3-1a**[MAC Addresses](javascript://)
* **3-2**[IP Addresses](javascript://)
  + **3-2a**[IPv4 Addresses](javascript://)
  + **3-2b**[IPv6 Addresses](javascript://)
  + **3-2c**[Types of IPv6 Addresses](javascript://)
* **3-3**[Ports and Sockets](javascript://)
* **3-4**[Domain Names and DNS](javascript://)
  + **3-4a**[Namespace Databases](javascript://)
  + **3-4b**[Name Servers](javascript://)
  + **3-4c**[Resource Records in a DNS Database](javascript://)
  + **3-4d**[DNS Server Software](javascript://)
* **3-5**[Troubleshooting Address Problems](javascript://)
  + **3-5a**[Troubleshooting Tools](javascript://)
  + **3-5b**[Common Network Issues](javascript://)
* **3-6**[Module Review](javascript://)
  + **3-6a**[Module Summary](javascript://)
  + **3-6b**[Key Terms](javascript://)
  + **3-6c**[Review Questions](javascript://)
  + **3-6d**[Hands-On Projects](javascript://)
  + **3-6e**[Capstone Projects](javascript://)

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# Module Introduction

### Objectives

After reading this module and completing the exercises, you should be able to:

* **1**Work with MAC addresses
* **2**Configure TCP/IP settings on a computer, including IP address, subnet mask, default gateway, and DNS servers
* **3**Identify the ports of several common network protocols
* **4**Describe domain names and the name resolution process
* **5**Use command-line tools to troubleshoot common network problems

**On the Job**

I woke up to a message from an on-call engineer, Bill, saying, “Help, I am out of ideas for DNS troubleshooting!” Twenty minutes later, as I walked into the office, he recited a chaotic list of all the troubleshooting steps he took and every possible problem that could have caused the issue at hand. We took a walk to the vending machines so I could get caffeine and the story.

Dying server hardware forced Bill to move a number of services to new hardware. DNS was scheduled to be last, as the configuration was simple, and moving it was supposed to be a quick and easy task. Everything seemed to work fine, but queries for all of the Internet and a test internal domain were not being answered. The OS configuration and DNS server settings all seemed fine, but no matter what we tweaked, the service did not work right.

Because Bill knew more about DNS than I did, there was little reason for a detailed walk-through of the configurations. I took a quick look, in hope of finding something obvious that he had missed, but the configuration was sound. Because no trivial fix was available, I reverted to basic troubleshooting mode and started to work through a simple list of items to check: “ping localhost, ping the interface, ping the router, and a host beyond it, etc.”

The last check returned “connect: Network is unreachable.” A quick glance at the routing table explained the issue: There was no default route. Without a way to forward traffic, no host outside of a few statically defined internal networks were reachable, including all the root DNS servers.

The fix was simple and, once the service was restored, I helped a bit with moving other services. Another set of eyes is an invaluable asset during late-night work, and I had to work off all that caffeine.

**Marcin Antkiewicz**

In [Module 1](javascript://), you learned that the OSI model can be used to describe just about every aspect of networking. You saw firsthand the usefulness of working your way up or down the seven layers of the OSI model to troubleshoot networking problems. In [Module 2](javascript://), you toured the elements of a typical network infrastructure and saw the importance of documentation in maintaining and troubleshooting a network. In this module, you will learn the several methods used to address and find software, files, computers, and other devices on a network. You’ll take a bottom-up approach to the OSI model to explore these topics, starting at the data link layer and working your way up to the application layer. (The lowest OSI layer, the physical layer, does not require a network address.) At the end of this module, you will learn how to troubleshoot addressing problems by using common command-line utilities.

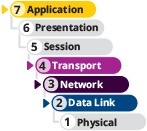
Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# 3-1Addressing Overview



### Certification

* 2.3

Given a scenario, configure and deploy common Ethernet switching features.

Average reading time: 11 minutes

In [Module 1](javascript://), you learned that addressing methods operate at the data link, network, transport, and application layers of the OSI model so that one host or node can find another on a network. Here’s a quick overview of the four addressing methods, starting at the lowest layer:

* **Data link layer MAC (Media Access Control) address**—A MAC address is embedded on every NIC on the globe and is assumed to be unique to that NIC. A MAC address is 48 bits, written as six hex (hexadecimal) numbers separated by colons, as in 00:60:8C:00:54:99. Nodes on a LAN find each other using their MAC addresses. Switches, which function at layer 2, check MAC addresses to determine where to send messages on the LAN.

**Note 3-1**

A hexadecimal number (also called a hex number) is a number written in the base-16 number system, which uses the 16 numerals 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F.

* **Network layer IP (Internet Protocol) address**—An IP address is assigned to nearly every [**interface**](javascript://), which is a network connection made by a node on a network. An IP address can be used to find any computer in the world if the IP address is public on the Internet. Applications such as browsers can store and retrieve IP addresses. But for routing purposes, an IP address is used only at the network layer. Routers, which function primarily at layer 3, check IP addresses to determine which network a message is destined for. There are two types of IP addresses:
  + [**IPv4 (Internet Protocol version 4)**](javascript://) addresses have 32 bits and are written as four decimal numbers called [**octets**](javascript://), for example, 92.106.50.200. Each octet, when written in binary, consists of exactly 8 bits. For example, the octet 92 can be written as 0101 1100.

**Note 3-2**

A binary number is a number written in the base-2 number system, which uses only the numerals 0 and 1.

* + [**IPv6 (Internet Protocol version 6)**](javascript://) addresses have 128 bits and are written as eight blocks of hex numbers, for example, 2001:0DB8:0B80:0000:0000:00D3:9C5A:00CC. Each block, when written in binary, contains 16 bits.
* **Transport layer ports**—A port is a number used by the transport layer to find an application. It identifies one application among several that might be running on a host. For example, a web server application is usually configured to listen for incoming requests at port 80 or port 443.
* **Application layer domain names, computer names, and host names**—Every host on a network is assigned a unique character-based name called the [**FQDN (fully qualified domain name)**](javascript://), for example, susan.mycompany.com, ftp.mycompany.com, and www.mycompany.com. Collectively, the last two parts of a host’s name (for example, mycompany.com) are called the [**domain name**](javascript://), which matches the name of the organization’s domain or network. The first part (for example, susan, ftp, and www) is the [**host name**](javascript://), which identifies the individual computer on the network. Ftp is the host name usually given to an FTP server, and www is often the host name assigned to a computer running a web server. When technicians refer to a “host name,” you can usually assume they’re referring to the FQDN unless stated otherwise.

**Note 3-3**

Technically, an FQDN ends in a period:

* ftp.mycompany.com.

However, in most applications, the terminal period is understood even when it is not typed or shown on the screen.

The organization responsible for tracking the assignments of IP addresses, port numbers, and domain names is [**IANA (Internet Assigned Numbers Authority)**](javascript://) (pronounced “I–anna”). IANA is a department of [**ICANN (Internet Corporation for Assigned Names and Numbers)**](javascript://). ICANN is a nonprofit organization charged with setting many policies that guide how the Internet works. For more information, see [iana.org](http://iana.org/" \t "_blank) and [icann.org](http://icann.org/" \t "_blank). At [icann.org](http://icann.org/" \t "_blank), you can download helpful white papers that explain how the Internet works.

Now that you have the big picture of how addressing happens at each layer of the OSI model, you’re ready to dig into the details, beginning with MAC addresses at the bottom of the model.

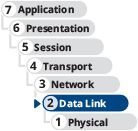
Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 3-1aMAC Addresses



You can often find a network adapter’s MAC address stamped directly onto the NIC’s circuit board or on a sticker attached to some part of the NIC, as shown in [Figure 3-1](javascript://). Later in this module, you’ll learn to use TCP/IP utilities to report the MAC address.

**Figure 3-1**

NIC with MAC address



Source: D-Link of North America

MAC addresses contain two parts, are 48 bits long, and are written as hexadecimal numbers separated by colons—for example, 00:60:8C:00:54:99. The first 24 bits (six hex characters, such as 00:60:8C in this example) are known as the [**OUI (Organizationally Unique Identifier)**](javascript://), which identifies the NIC’s manufacturer. A manufacturer’s OUI is assigned by IEEE (Institute of Electrical and Electronics Engineers). If you know a computer’s MAC address, you can determine which company manufactured its NIC by looking up its OUI. IEEE maintains a database of OUIs and their manufacturers, which is accessible via the web. At the time of this writing, the database search page could be found at [standards-oui.ieee.org/oui.txt](http://standards-oui.ieee.org/oui.txt" \t "_blank). You can also use an OUI lookup tool, such as Wireshark’s at [wireshark.org/tools/oui-lookup](http://wireshark.org/tools/oui-lookup" \t "_blank).

**Note 3-4**

Links to websites given in this course might become outdated as websites change. If a given link doesn’t work, try to search for the item online to find the new link.

The last 24 bits make up the [**extension identifier**](javascript://) or [**device ID**](javascript://) and identify the device itself. Manufacturers assign each NIC a unique extension identifier, based on the NIC’s model and manufacture date, so that, in theory, no two NICs share the same MAC address.

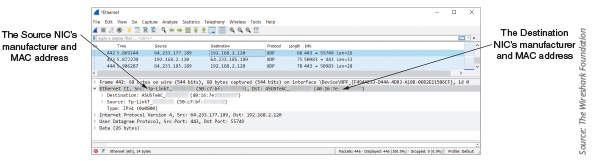
**Applying Concepts 3-1**

### Identify a NIC Manufacturer

Most network packets include the MAC address of the sender, the receiver, or both. When collecting network data on Wireshark using the default settings, some OUIs are automatically resolved, telling you the manufacturer of each device. In [Figure 3-2](javascript://), you can see where Wireshark has identified the manufacturers—TP-Link and ASUS—of two NICs on this network.

**Figure 3-2**

Wireshark capture shows the manufacturers of the Source and Destination nodes’ NICs



Enlarge Image

Source: The Wireshark Foundation

Sometimes, however, you might be working with physical addresses provided by a command output, or you might need a little more information than what is provided by a Wireshark capture. For these situations, use an online MAC address lookup table such as Wireshark’s OUI Lookup Tool. Complete the following steps:

1. 1

In your browser, go to [wireshark.org/tools/oui-lookup](http://wireshark.org/tools/oui-lookup" \t "_blank).

1. 2

Notice earlier in [Figure 3-2](javascript://) that the MAC addresses of the Source and Destination devices are listed in the Ethernet frame. The first three bytes of the Destination device’s MAC address, 40:16:7e, make up the OUI of the device’s manufacturer. Type those numbers into Wireshark’s OUI Lookup Tool and click **Find**. What results did you get?

**Note 3-5**

If you are pulling OUIs from your own Wireshark capture or command-line output, you can copy and paste one or more OUIs into the website search box.

You can perform the same lookup using output from a PowerShell or Command Prompt window:

1. 3

Open a PowerShell or Command Prompt window and enter **ipconfig /all** to identify your NIC’s MAC address.

1. 4

From your command output, select the first three bytes of the physical address for the active network connection and press **Ctrl**+**C**. Note: You might need to first press **Ctrl**+**M** to enable marking.

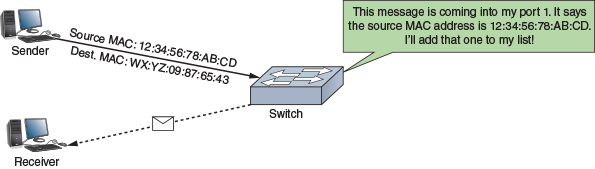
1. 5

Click in the search box on Wireshark’s website, press **Ctrl**+**V** to paste the information into the Wireshark Lookup Tool, and click **Find**. Who is the manufacturer of your NIC?

Switches use MAC addresses to identify devices on the local area network. As each device communicates on the network, the switch identifies the sending device’s MAC address from its transmitted message as shown in [Figure 3-3](javascript://). The MAC address is stored in a [**MAC address table**](javascript://) that maps each MAC address to a physical port on the switch.

**Figure 3-3**

The switch learns the sending device’s MAC address

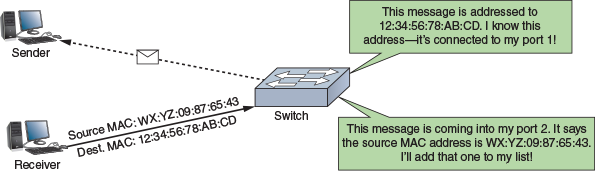


Enlarge Image

Later, when the switch sees a message destined for that MAC address, the switch checks its MAC address table and determines which port leads to the correct device, as shown in [Figure 3-4](javascript://). The information in a MAC address table expires after a short period of time, so the switch is constantly relearning where devices are located on the network. This balances a switch’s sensitivity to changes in the network with the switch’s ability to quickly and efficiently direct network traffic to its destination. At the end of this module in [Capstone Project 3-2](javascript://), you’ll work with a switch in Packet Tracer as it builds a MAC address table.

**Figure 3-4**

The switch learned earlier which port the Destination MAC address is connected to



Enlarge Image

**Remember This…**

* Describe addressing systems used at the data link, network, transport, and application layers.
* Identify a device’s MAC address.
* Interpret the information given in a MAC address.

**Self-Check**

1. What numbering system do humans use to write MAC addresses?

Answer

* 1. Decimal
  2. Binary
  3. Base-2
  4. Hexadecimal

1. What Windows command outputs a computer’s MAC address?

Answer

* 1. ipconfig
  2. ping
  3. ipconfig /all
  4. oui-lookup

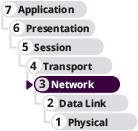
Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# 3-2IP Addresses



### Certification

* 1.4

Given a scenario, configure a subnet and use appropriate IP addressing schemes.

* 1.6

Explain the use and purpose of network services.

* 5.5

Given a scenario, troubleshoot general networking issues.

Average reading time: 34 minutes

As you move up to layer 3, recall that IP addresses identify nodes at the network layer. Whereas MAC addresses are used by switches for communication inside a local network, an IP address is required for a device to communicate outside its local network through a gateway device such as a router. While switches need MAC addresses to identify devices in a network, routers rely on IP addresses to locate devices across networks.

You can assign a persistent or [**static IP address**](javascript://) to a device, or you can configure the device to request and receive (or lease) a [**dynamic IP address**](javascript://) from a DHCP server each time it connects to the network. A [**DHCP (Dynamic Host Configuration Protocol)**](javascript://) server manages the dynamic distribution of IP addresses to devices on a network. You’ll learn more about DHCP shortly.

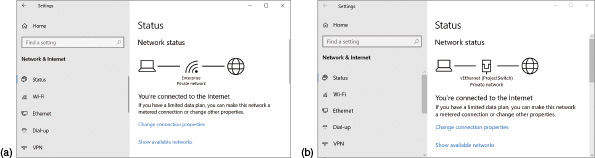
**Applying Concepts 3-2**

### Windows TCP/IP Settings

Let’s begin with a look at IP addresses and related TCP/IP settings on a Windows 10 computer. Take note that, if you’re using a computer with Hyper-V enabled, you might see a few interesting variations as you click through the screens in this module. For example, a computer not running Hyper-V will likely show a direct connection to your LAN, as shown in [Figure 3-5a](javascript://). A computer that is running Hyper-V will likely show a connection to your virtual switch that you created in [Module 1](javascript://), as shown in [Figure 3-5b](javascript://). You’ll learn more about why this is the case in a later module when you study virtualization technologies more closely.

**Figure 3-5**

(a) A computer not running Hyper-V shows a connection directly to the LAN; (b) A computer running Hyper-V shows a connection with a virtual switch



Enlarge Image

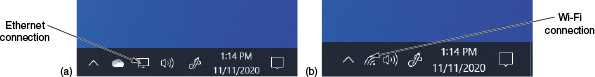
To check TCP/IP settings on your Windows computer, complete the following steps:

1. 1

Click **Start** and then click the **Settings** gear icon. Click **Network & Internet**. Alternatively, you can right-click the active network connection icon on the right side of your taskbar near the date and time (see [Figure 3-6](javascript://)) and then click **Open Network & Internet settings**.

**Figure 3-6**

Right-click the active network connection to see its properties



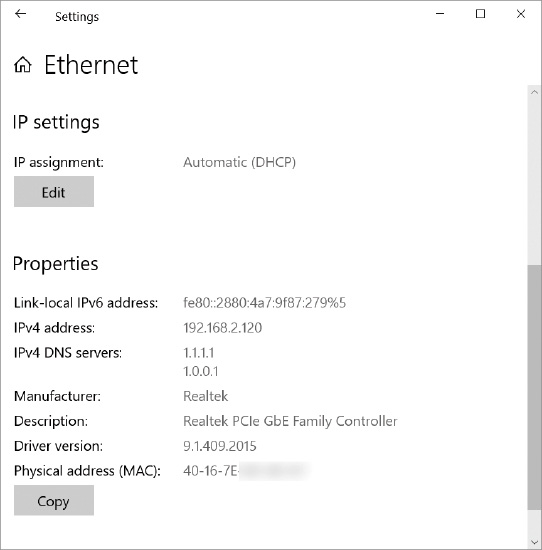
Enlarge Image

1. 2

Click **Change connection properties** and scroll down to the IP settings and Properties sections. [Figure 3-7](javascript://) shows the TCP/IP settings, including IP assignment source (Automatic from DHCP), IPv6 and IPv4 addresses, DNS servers, and Physical address (MAC).

**Figure 3-7**

View TCP/IP settings



Enlarge Image

1. 3

You probably have the Automatic (DHCP) option enabled, which dynamically assigns an IP address from a DHCP server. The Properties section shows your IP address, MAC address, and DNS (Domain Name Service) servers. DNS servers are responsible for tracking computer names and their IP addresses. Later in the module, you’ll learn more about the various types of DNS servers and how they work together.

You can find similar information and more using the [**ipconfig**](javascript://) utility, which you first used in Project 2-3 while working with Nmap. You’ll learn more about this utility later in this module. Network technicians need to be comfortable with the CLI (command line interface) because it is quicker and often more powerful and flexible than a GUI (graphical user interface). To see the additional information ipconfig reports, complete the following steps:

1. 4

Open a PowerShell or Command Prompt window and enter **ipconfig**. What are your IPv4 address, subnet mask, and default gateway settings for your active network connection?

**Note 3-6**

Notice that ipconfig by itself does not output the MAC address. You must use the /all parameter to see the MAC address, which you did earlier in [Applying Concepts 3-1](javascript://).

Here’s a brief explanation of the subnet mask and default gateway settings:

* A [**subnet mask**](javascript://), also called a netmask, is a 32-bit number that helps one computer find another. The 32 bits are used to indicate what part of an IP address’s bits are the network portion, called the [**network ID**](javascript://) or network address, and which bits consist of the host portion, called the [**host ID**](javascript://) or [**node ID**](javascript://). Using this information, a computer can determine if another computer with a given IP address is on its own or a different network.
* A [**gateway**](javascript://) is a computer, router, firewall, or other device that a host uses to access another network. The [**default gateway**](javascript://) is the routing device that nodes on the network turn to for access to the outside world. In the [On the Job](javascript://) story at the beginning of this module, you read about a problem that appeared to be a DNS issue but was, in fact, a missing default route that prevented network nodes from reaching DNS servers outside the local network. The default gateway provides a connection to all resources outside the local network when static routes aren’t available (which is most of the time).

**Note 3-7**

Technically, there is a subtle distinction between the meanings of the terms subnet mask and netmask. A [**subnet**](javascript://) is a smaller network within a larger network. A netmask indicates the bits of an IP address that identify the larger network, while the subnet mask indicates the bits of an IP address that identify a smaller subnet within the larger network. Most of the time, however, these two terms are used interchangeably. You’ll learn more about subnets in a later module.

Recall that networks may use two types of IP addresses: IPv4 addresses, which have 32 bits, and IPv6 addresses, which have 128 bits. You’ll first learn about how IPv4 addresses are formatted and assigned, and then you’ll explore how IPv6 addresses are designed to solve some limitations of IPv4.

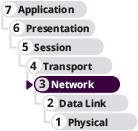
Go to pg.

[**help**](javascript://)

Application OpenedApplication Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 3-2bIPv6 Addresses



As IPv4 address supplies started running low, the IPv6 standards were developed to allow for more public IP addresses on the Internet. IPv6 designers also worked to improve routing capabilities and speed of communication over the established IPv4 standards. Let’s begin this discussion by looking at how IPv6 addresses are written, as follows:

* Recall that an IPv6 address has 128 bits that are written as eight blocks (also called quartets) of hexadecimal numbers separated by colons, like this: 2001:0000:0B80:0000:0000:00D3:9C5A:00CC
* Each block is 16 bits long. For example, the first block in the preceding IP address is the hexadecimal number 2001, which can be written as 0010 0000 0000 0001 in binary.
* Because IPv6 addresses are so long, shorthand notation helps to make these addresses easier to read and write. For example, leading zeroes in a four-character hex block can be eliminated. This means the sample IP address can be written as 2001:0000:B80:0000:0000:D3:9C5A:CC.
* If blocks contain all zeroes, they can be eliminated and replaced by double colons (::). To avoid confusion, only one set of double colons is used in an IPv6 address. This means the sample IP address can be written two ways:
  + 2001::B80:0000:0000:D3:9C5A:CC
  + 2001:0000:B80::D3:9C5A:CC

In this example, the preferred method is the second one (2001:0000:B80::D3:9C5A:CC) because this way, the address contains the fewest zeroes.

The way computers communicate using IPv6 has changed the terminology used to describe TCP/IP communication. Here are a few terms used in the IPv6 standards:

* A [**link**](javascript://), sometimes called the local link, is any LAN bounded by routers. [**Neighbors**](javascript://) are two or more nodes on the same link.
* When a network is configured to use both IPv4 and IPv6 protocols, the network is said to be [**dual stacked**](javascript://). However, if packets on this network must traverse other networks where dual stacking is not used, the solution is to use [**tunneling**](javascript://), which is a method of transporting IPv6 packets over an IPv4 network.
* The last 64 bits, or four blocks, of an IPv6 address identify the interface and are called the [**interface ID**](javascript://) or interface identifier. These 64 bits uniquely identify an interface on the local link.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

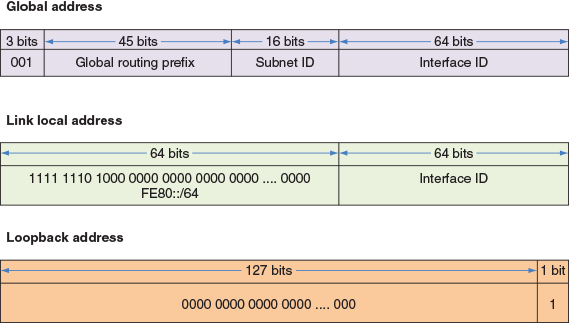
## 3-2cTypes of IPv6 Addresses

IPv6 classifies IP addresses differently than IPv4 does. IPv6 supports these three types of IP addresses, classified by how the address is used, as follows:

* [**Unicast address**](javascript://)—Specifies a single node on a network. [Figure 3-16](javascript://) diagrams three types of unicast addresses, including the following:

**Figure 3-16**

Three types of IPv6 addresses



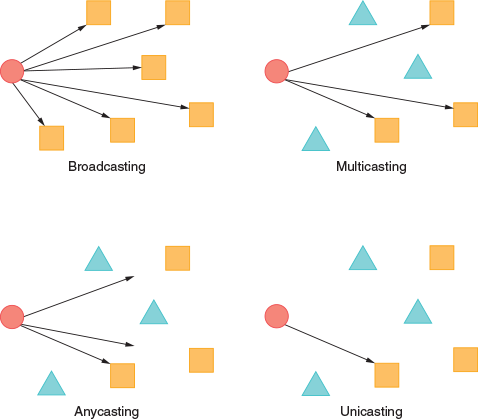
* + [**Global address**](javascript://)—Can be routed on the Internet and is similar to public IPv4 addresses. Most begin with the prefix 2000::/3, although other prefixes are being released. The /3 indicates that the first three bits are fixed and are always 001. Looking at [Figure 3-16](javascript://), notice the 16 bits reserved for the [**subnet ID**](javascript://), which can be used to identify a segment of a large corporate network.
  + [**Link local address**](javascript://)—Can be used for communicating with nodes in the same link and is similar to an autoconfigured APIPA address in IPv4. It begins with FE80::/10. The first 10 bits (indicated by /10) of the reserved prefix are fixed (1111 1110 10), and the remaining 54 bits in the 64-bit prefix are all zeroes. Therefore, a link local address prefix is sometimes written as FE80::/64, as shown in [Figure 3-16](javascript://). Link local addresses are not allowed past the local link or on the Internet.
  + **Loopback address**—Similar to the IPv4 loopback address, can be used to test that an interface and supporting protocol stack are functioning properly. Consists of 127 zeros followed by a 1 and is written ::1/128.

* **[Multicast address](javascript://)**—Delivers packets to all nodes in a targeted, multicast group.
* [**Anycast address**](javascript://)—Identifies multiple destinations, with packets delivered to the closest destination. For example, a DNS server might send a DNS request to a group of DNS servers that have all been assigned the same anycast address. A router handling the request examines routes to all the DNS servers in the group and routes the request to the closest server.

Recall that with IPv4 broadcasting, messages are sent to every node on a network. However, IPv6 reduces network traffic by eliminating broadcasting. The concepts of broadcasting, multicasting, anycasting, and unicasting are depicted in [Figure 3-17](javascript://) for easy comparison. In the figure, each pink circle is the sending node. The yellow squares are the intended recipients. The blue triangles are other nodes on the network and do not receive the transmission.

**Figure 3-17**

Concepts of broadcasting, multicasting, anycasting, and unicasting



Interestingly, due to the way switches learn MAC addresses of connected devices, multicasting can cause traffic congestion problems called [**multicast flooding**](javascript://). The multicast group is assigned a single IP address, which means no particular MAC address can be associated with that IP address. Instead, when the switch receives a multicast message, it must flood all its interfaces with the transmission, which defeats the purpose of multicasting.

To fix this problem, a switch that must handle multicast traffic should have IGMP snooping enabled on it. IGMP (Internet Group Management Protocol) is a network-layer protocol used to manage multicast group memberships and direct multicast traffic to the correct devices. However, because switches are layer 2 devices, they miss out on the IGMP information that identifies group members. IGMP snooping, then, gives switches the ability to detect IGMP messages on the network and gather information from those messages to add accurate entries in their MAC address tables.

[Table 3-3](javascript://) lists some currently used address prefixes for IPv6 addresses. Notice in the table the unique local unicast addresses, which work on local links and are similar to IPv4 private IP addresses. You can expect more prefixes to be assigned as they are needed.

**Table 3-3**

### Address prefixes for types of IPv6 addresses

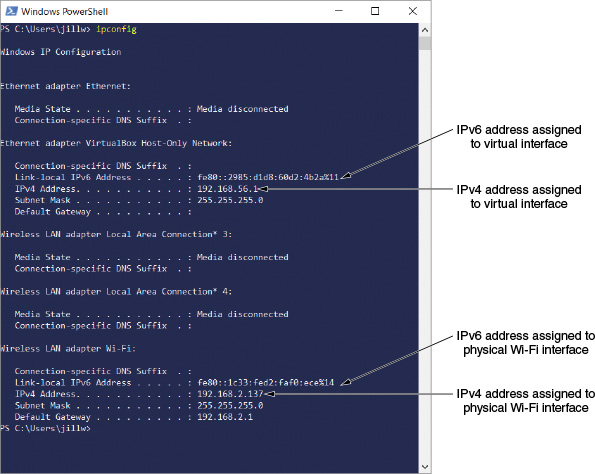
| **IP address type** | **Address prefix** | **Notes** |
| --- | --- | --- |
| Global unicast | 2000::/3 | First 3 bits are always 001 |
| Link local unicast | FE80::/10 (also written as FE80::/64) | First 10 bits are always 1111 1110 10 followed by 54 zeroes |
| Unique local unicast | FC00::/7 | First 7 bits are always 1111 110 |
| FD00::/8 | First 8 bits are always 1111 1101 |
| Multicast | FF00::/8 | First 8 bits are always 1111 1111 |

Enlarge Table

You can use the ipconfig command to view IPv4 and IPv6 addresses assigned to all network connections on a computer. For example, in [Figure 3-18](javascript://), four IP addresses have been assigned to the physical and virtual connections on this computer.

**Figure 3-18**

The ipconfig command shows IPv4 and IPv6 addresses assigned to this computer



Enlarge Image

### IPv6 Autoconfiguration

IPv6 addressing is designed so that a computer can autoconfigure its own link local IP address without the help of a DHCPv6 server. This process is called [**SLAAC (stateless address autoconfiguration)**](javascript://) and is similar to how IPv4 uses an APIPA address but results in an address the computer can continue to use on the network. The following SLAAC steps describe how a computer using IPv6 makes a network connection:

1. Step 1

The computer creates its IPv6 address. It uses FE80::/64 as the first 64 bits, called the prefix. Depending on how the OS is configured, the last 64 bits (called the interface ID) can be generated in one of two ways:

* + **The 64 bits are randomly generated**—In this case, the IP address is called a temporary address and is never registered in DNS or used to generate global addresses for use on the Internet. The IP address changes often to help prevent hackers from discovering the computer. This is the default method used by Windows 10.
  + **The 64 bits are generated from the network adapter’s MAC address**—MAC addresses consist of 48 bits (formally called EUI-48) and must be converted to the 64-bit standard, called the [**EUI-64 (Extended Unique Identifier-64)**](javascript://) standard. To generate the interface ID, the OS takes the 48 bits of the device’s MAC address, inserts a fixed 16-bit value in the middle of the 48 bits, and inverts the value of the seventh bit.

1. Step 2

The computer checks to make sure its IP address is unique on the network. It does this by sending a message to the IP address and, if there’s a reply, then the address is a duplicate and the computer tries again with a different address.

1. Step 3

The computer asks if a router on the network can provide configuration information. This message is called an [**RS (router solicitation)**](javascript://) message. If a router responds with DHCP information in what’s called an **[RA (router advertisement)](javascript://)** message, the computer uses whatever information this might be, such as the IP addresses of DNS servers or the network prefix. The process is called prefix discovery; the computer then uses the prefix to generate its own link local or global IPv6 address by appending its interface ID to the prefix.

Because a computer can generate its own link local or global IP address, a DHCPv6 server usually serves up only global IPv6 addresses to hosts that require static address assignments. For example, web servers and DNS servers can receive their static IPv6 addresses from a DHCPv6 server.

**Note 3-9**

On larger networks, IP address infrastructure can quickly become overwhelming. An [**IPAM (IP address management)**](javascript://) system, whether as a standalone product or embedded in another product such as Windows Server, provides a way to plan, deploy, and monitor a network’s IP address space. IPAM tools can automatically detect IP address ranges, assignments, reservations, and exclusions, integrate this information with data from DNS records, and provide constant monitoring for growth, security, and troubleshooting purposes. In a project at the end of this module, you’ll use an IP scanning tool similar in concept to Nmap that can identify information about all devices connected to a network.

Now you’re ready to take another step up the OSI model to layer 4, where ports are used to identify an application when it receives communication from a remote host.

**Remember This…**

* Describe the role of a default gateway.
* Memorize the IP ranges of the IPv4 classes.
* Distinguish between RFC1918 addresses and public IP addresses.
* Describe the role of DHCP and various scope options.
* Explain the NAT process.
* Identify types of IPv4 and IPv6 addresses, including multicast, unicast, anycast, broadcast, link local, and loopback.
* Explain key IPv6 concepts, including tunneling, dual stack, shorthand notation, router advertisement, and SLAAC.

**Self-Check**

1. What command shows you a computer’s TCP/IP configuration?

Answer

* 1. ping
  2. ipconfig
  3. ssh
  4. nmap

1. Which of the following IPv4 addresses is a public IP address?

Answer

* 1. 10.0.2.14
  2. 172.16.156.254
  3. 192.168.72.73
  4. 64.233.177.189

1. Which IPv6 prefix can be routed on the Internet?

Answer

* 1. 2000::/3
  2. FE80::/10
  3. ::1/128
  4. FC00::/7

**You’re Ready**

You’re now ready to complete [Project 3-1: Create a NAT Translation Table Entry](javascript://), or you can wait until you’ve finished reading this module.

**You’re Ready**

You’re now ready to complete [Project 3-2: Change IPv6 Autoconfiguration Settings](javascript://), or you can wait until you’ve finished reading this module.

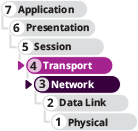
Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# 3-3Ports and Sockets



### Certification

* 1.5

Explain common ports and protocols, their application, and encrypted alternatives.

* 1.6

Explain the use and purpose of network services.

* 2.1

Compare and contrast various devices, their features, and their appropriate placement on the network.

Average reading time: 10 minutes

A port is a number assigned to a [**process**](javascript://), such as an application or a service, that can receive data. Whereas an IP address is used to find a computer, a port is used to find a process running on that computer. TCP and UDP ports ensure that data is transmitted to the correct process among multiple processes running on a single device. If you compare network addressing with the addressing system used by the postal service, and you equate a host’s IP address to the address of a building, then a port is similar to an apartment number within that building.

A [**socket**](javascript://) consists of both a host’s IP address and a process’s TCP or UDP port, with a colon separating the two values. For example, the standard port for the Telnet service is TCP 23. If a host has an IP address of 10.43.3.87, the socket for Telnet running on that host is 10.43.3.87:23.

When the host receives a request to communicate on TCP port 23, it establishes or opens a [**session**](javascript://), which is an ongoing conversation, with the Telnet service. At that point, the socket is said to be open. When the TCP session is complete, the socket is closed or dissolved. You can think of a socket as a virtual conversation or circuit between a server and client (see [Figure 3-19](javascript://)).

**Figure 3-19**

A virtual connection for the Telnet service



Enlarge Image

Port numbers range from 0 to 65535 and are categorized by IANA into three types:

* [**Well-known ports**](javascript://)—Range from 0 to 1023 and are assigned by IANA to widely used and well-known utilities and applications, such as Telnet, FTP, and HTTP. [Table 3-4](javascript://) lists some of the most common well-known ports and registered ports used by TCP and/or UDP.
* [**Registered ports**](javascript://)—Range from 1024 to 49151 and can be used temporarily by processes for nonstandard assignments to increase security. Default assignments of these registered ports must be registered with IANA.
* **Dynamic and private ports**—Range from 49152 to 65535 and are open for use without restriction.
  + **Dynamic port**—Number assigned by a client or server as the need arises. For example, if a client application has several open sockets with multiple servers, it can use a different dynamic port number for each socket.
  + **Private port**—Number assigned by a network administrator that is different from the well-known port number for that service. For example, the administrator might assign a private port number other than the standard port 80 to a web server on the Internet so that several people can test the site before it’s made available to the public. To reach the web server, a tester must enter the private port number in the browser address box along with the web server’s IP address.

**Table 3-4**

### Common well-known and registered TCP and UDP ports

| **Port** | **Process name** | **Protocol used** | **Used for:** |
| --- | --- | --- | --- |
| 20 | FTP-DATA | TCP | File transfer—data |
| 21 | FTP | TCP | File transfer—control (an FTP server listens at port 21 and sends/receives data at port 20) |
| 22 | SSH | TCP | Secure communications between Linux computers or, if installed, between Windows computers |
| 22 | SFTP | TCP | Encrypted file transfer using SSH |
| 23 | TELNET | TCP | Unencrypted control of remote computers |
| 25 | SMTP | TCP | Outgoing email messages |
| 53 | DNS | TCP or UDP | Name resolution |
| 67 | DHCP | UDP | Distribution of IP addresses on a network—client to server messages |
| 68 | DHCP | UDP | Distribution of IP addresses on a network—server to client messages |
| 69 | TFTP | UDP | Simple file transfer |
| 80 | HTTP | TCP or UDP | Requests between web servers and web clients |
| 110 | POP3 | TCP | Incoming email messages (downloaded messages) |
| 123 | NTP | UDP | Network time synchronization |
| 143 | IMAP4 | TCP | Incoming email messages (messages stored on server) |
| 161 | SNMP | TCP or UDP | Messages sent to managed network devices from SNMP manager |
| 162 | SNMP | Typically UDP | Responses or unsolicited information sent from network devices to manager |
| 389 | LDAP | TCP or UDP | Access to network-based directories |
| 443 | HTTPS | TCP | Secure implementation of HTTP over SSL or TLS |
| 445 | SMB | TCP | Network file sharing |
| 514 | Syslog | UDP | Manages and stores information about system events |
| 587 | SMTP TLS | TCP | SMTP encrypted by TLS |
| 636 | LDAPS | TCP or UDP | Secure access to network-based directories |
| 993 | IMAP4 over SSL | TCP or UDP | IMAP4 encrypted by SSL or TLS |
| 995 | POP3 over SSL | TCP or UDP | POP3 encrypted by SSL or TLS |
| 1433 | SQL Server | TCP | Connections to installation of Microsoft SQL Server from other databases or applications |
| 1521 | SQLnet, also called Oracle Net Services | TCP | Connections to installation of Oracle Database from other databases or applications |
| 3306 | MySQL |  | Connections to installation of MySQL Server from other databases or applications |
| 3389 | RDP | TCP | Encrypted control of remote Windows computers |
| 5060 | SIP | UDP | Creation of unencrypted connections for multimedia session |
| 5061 | SIP | UDP | Creation of encrypted connections for multimedia session |

Enlarge Table

### Exam Tip

To prepare for the CompTIA Network+ exam, you need to memorize all the well-known and registered ports listed in [Table 3-4](javascript://). Some of these protocols are discussed in detail in later modules. They’re collected in [Table 3-4](javascript://) for easy reference and last-minute review before taking your exam. Pay close attention to secure alternatives for common protocols. For example, what is the secure port for SMTP? If you answered 25, that is incorrect. SMTP over TLS is encrypted and runs on port 587. The Network+ exam is known to test on this type of distinction.

In [Module 1](javascript://), you learned about most of the protocols listed in [Table 3-4](javascript://). A few of them have already been covered in this module. Here’s a brief description of the ones not yet covered:

* [**TFTP (Trivial File Transfer Protocol)**](javascript://)—Most commonly used by computers (without user intervention) as they are booting up to request configuration files from another computer on the local network. TFTP uses UDP, whereas normal FTP uses TCP.
* [**NTP (Network Time Protocol)**](javascript://)—A simple protocol used to synchronize clocks on computers throughout a network. The genius of NTP is how it can almost completely account for the variable delays across a network, even on the open Internet. It does this through a hierarchy of time servers where stratum-1 servers communicate directly with a primary time source, such as GPS (Global Positioning System) or Galileo (Europe’s version of GPS). These servers track UTC (Coordinated Universal Time) and provide this information to lower strata servers. Each hop between NTP servers increases the [**stratum**](javascript://) number by 1 up to 16. NTP servers at any stratum can then convert the provided UTC into its local time zone. Not every network has its own time server, but those that do can maintain accuracy for its NTP clients to within a millisecond of each other and are closely synced to the UTC.
* [**LDAP (Lightweight Directory Access Protocol)**](javascript://)—A standard protocol for accessing network-based directories. [**LDAPS (Lightweight Directory Access Protocol over SSL)**](javascript://) uses SSL to encrypt its communications. You’ll learn more about LDAP and LDAPS in a later module.
* [**SMB (Server Message Block)**](javascript://)—First used by earlier Windows OSs for file sharing on a network. UNIX uses a version of SMB in its Samba software, which can share files with other operating systems, including Windows systems.
* [**Syslog (system log)**](javascript://)—A Linux or UNIX standard for generating, storing, and processing messages about events on a system. It describes methods for detecting and reporting events and specifies the format and contents of messages. The syslog utility does not alert a user to problems—it only keeps a history of messages issued by the system.
* [**SQLnet**](javascript://), also known as Oracle Net Services—Used by Oracle Database to communicate with other Oracle Databases or with database clients. This interconnection allows applications and databases to be distributed across different machines and still communicate as if they were on the same machine.

Which port a protocol communicates over becomes especially relevant when configuring firewalls. Recall that a firewall is a dedicated device or software on a computer that selectively filters or blocks traffic between networks. It works by blocking traffic on all ports and between all IP addresses except those that are specifically approved by the network admin. You will learn more about firewalls in later modules. For now, note that all firewalls are porous to some degree in that they always let some traffic through; the question is what kind of traffic they allow—some of this filtering is accomplished by opening or closing ports. For example, if you have a SQL Server database that provides data to clients throughout your network, you’ll need to allow port 1433 (and a few others) on the firewall protecting that database.

**Remember This…**

* Explain how a port (as part of a socket) identifies a running process.
* Memorize the given list of well-known ports.
* Choose secure ports when possible.

**Self-Check**

1. When hosting a secure email server for access from the Internet, which port should be open on the corporate firewall?

Answer

* 1. 25
  2. 110
  3. 443
  4. 587

1. Which port should be open so you can remote into the corporate office’s Linux Server from a branch office?

Answer

* 1. 22
  2. 23
  3. 1433
  4. 3389

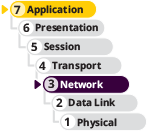
Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# 3-4Domain Names and DNS



### Certification

* 1.6

Explain the use and purpose of network services.

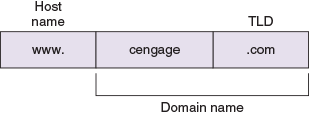
Average reading time: 25 minutes

When is the last time you entered a website’s IP address into your browser’s address bar so you could go to that website? For that matter, do you know the IP address for any website you visit frequently? Probably not. Instead, you type in a website address, such as cengage.com. Your browser adds some more information, such as https://www.cengage.com, which is called a URL. A [**URL (uniform resource locator)**](javascript://) is an application layer addressing scheme that identifies where to find a particular resource on a network or across networks. Application layer addressing was created because character-based names are easier for humans to remember than numeric IP addresses. The first part of the URL, https, identifies the protocol to be used, in this case, HTTPS. The next part is the FQDN. Recall that an FQDN is a host name and a domain name together, as shown in [Figure 3-20](javascript://) and described next:

* The host name is determined by the website developer or administrator.
* Domain names must be registered with an Internet naming authority that works on behalf of ICANN.
* The last part of an FQDN (com in this example) is called the [**TLD (top-level domain)**](javascript://).

**Figure 3-20**

A fully qualified domain name



[Table 3-5](javascript://) lists some well-known ICANN-approved TLDs. No restrictions exist on the use of the .com, .org, and .net TLDs, and ICANN restricts what type of hosts can be associated with the .arpa, .mil, .edu, and .gov TLDs. Other TLDs are dedicated to hosts in specific countries, such as .us, .eu (for countries in the European Union), .ca (Canada), and .au (Australia). Companies can also register their own TLD, although this is not cheap. A complete list of current TLDs can be found at [iana.org/domains/root/db/](http://iana.org/domains/root/db/" \t "_blank).

**Table 3-5**

### Some well-known top-level domains

| **Domain suffix** | **Type of organization** |
| --- | --- |
| ARPA | Reverse lookup domain (special Internet function) |
| COM | Commercial |
| EDU | Educational |
| GOV | Government |
| ORG | Noncommercial organization (such as a nonprofit agency) |
| NET | Network (such as an ISP) |
| MIL | United States military organization |
| BIZ | Businesses |
| INFO | Unrestricted use |

While FQDNs are convenient for humans, a computer must convert the FQDN to an IP address before it can find the referenced computer. Suppose you type an FQDN into a browser address bar; how does your computer figure out the IP address for that web server? To answer this question, you need to learn about [**name resolution**](javascript://), which is an application layer process of discovering the IP address of a host when its FQDN is known.

In the mid-1980s, DNS (Domain Name System) was designed to associate computer names with IP addresses. DNS is an application layer client-server system of computers and databases made up of these elements:

* **Namespace**—The entire collection of computer names and their associated IP addresses stored in databases on DNS name servers around the globe
* **Name servers**—Computers that hold these databases, organized in a hierarchical structure
* **Resolvers**—A DNS client that requests information from DNS name servers

**Note 3-10**

A registry, also known as a domain name registry operator, is an organization or country that is responsible for one or more TLDs and that maintains a database or registry of TLD information. A domain name registrar such as GoDaddy ([godaddy.com](http://godaddy.com/" \t "_blank)) is an organization accredited by registries and ICANN to lease domain names to companies or individuals, following the guidelines of the TLD registry operators.

Let’s take a closer look at each of these components.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 3-4aNamespace Databases

DNS namespace databases are stored on thousands of servers around the world, rather than being centralized on a single server or group of servers. In other words, DNS doesn’t follow a centralized database model, but rather a distributed database model. Because data is distributed over thousands of servers, DNS will not fail catastrophically if one or a handful of servers experiences errors.

Each organization that provides host services (for example, websites or email) on the public Internet is responsible for providing and maintaining its own DNS authoritative servers for public access, or they can use a third-party or cloud-hosted DNS server. An [**authoritative name server**](javascript://) is the authority on computer names and their IP addresses for computers in their domains. The domains (for example, cengage.com) that the organization is responsible for managing are collectively called a [**DNS zone**](javascript://). A large organization can keep all its domains in a single zone, or it can subdivide its domains into multiple zones to make each zone easier to manage.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 3-4bName Servers

An organization might have these four common types of DNS servers:

* [**Primary DNS server**](javascript://)—The authoritative name server for the organization, which holds the authoritative DNS database for the organization’s zones. This server is contacted by clients, both local and over the Internet, to resolve DNS queries for the organization’s domains.
* [**Secondary DNS server**](javascript://)—The backup authoritative name server for the organization. When a secondary DNS server needs to update its database, it makes the request to the primary server for the update; this update process is called a [**zone transfer**](javascript://).
* [**Caching DNS server**](javascript://)—A server that accesses public DNS data and caches the DNS information it collects. This server receives DNS queries from local network clients and works to resolve them by contacting other DNS servers for information. Caching DNS servers do not store zone files (which is why they must rely on their caches and resolution efforts), and, therefore, do not participate in zone transfers, which further helps to reduce network traffic on the intranet.
* [**Forwarding DNS server**](javascript://)—An optional server that receives queries from local clients but doesn’t work to resolve the queries. Typically, a forwarding server will maintain its own DNS cache from previous queries, and so it might already have the information the client needs. If not, the forwarding server forwards the query to another server to resolve. Several forwarding servers might be strategically placed throughout the organization’s network to reduce network traffic on slow links.

**Note 3-11**

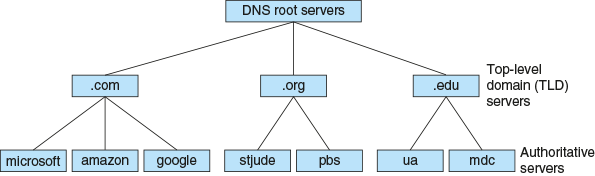
The primary and secondary DNS servers listed in a client’s IP configuration, like what you saw in [Applying Concepts 3-2](javascript://), are not the same thing as an organization’s primary and secondary authoritative DNS servers. The client’s configuration is referring to the network’s caching or forwarding servers.

Any of these DNS server types can coexist on the same machine, depending on the needs of the network. For example, a primary DNS server for one zone might be a secondary DNS server for a different zone within the organization. A primary DNS server might also serve as a caching server for its local network clients (although for security purposes, this is not recommended). A caching server might also rely on forwarding for certain clients or certain types of traffic.

DNS name servers are organized in a global hierarchical structure shown in [Figure 3-21](javascript://). At the root level, 13 clusters of [**root DNS servers**](javascript://) hold information used to locate the TLD (top-level domain) servers. These TLD servers hold information about the authoritative name servers owned by various organizations and how to find them.

**Figure 3-21**

Hierarchy of name servers



Enlarge Image

To understand how these servers interact, let’s look at an example. Suppose an employee at Cengage, using a computer in the cengage.com domain, enters [www.mdc.edu](http://www.mdc.edu/" \t "_blank) in their web browser address box. The browser makes an API call to the DNS resolver, a TCP/IP component in the OS, for the IP address of the [www.mdc.edu](http://www.mdc.edu/" \t "_blank) host.

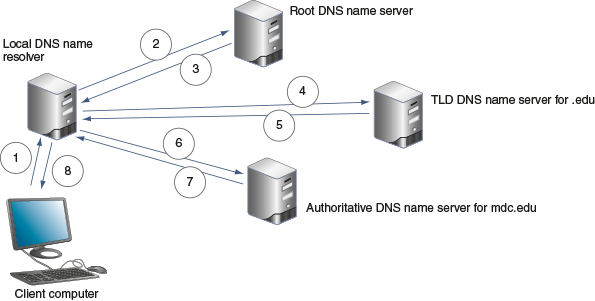
**Note 3-12**

Recall that an application uses an API call to request the operating system perform a service or task.

Here are the steps to resolve the name, which are also illustrated in [Figure 3-22](javascript://):

**Figure 3-22**

Queries for name resolution of [www.mdc.edu](http://www.mdc.edu/" \t "_blank)



Enlarge Image

1. Step 1

The resolver on the client computer first searches its DNS cache, a database stored on the local computer, for the match. If it can’t find the information there, the resolver sends a DNS message or query to its local DNS server. In this example, let’s assume this caching server doesn’t yet know the IP address of the [www.mdc.edu](http://www.mdc.edu/" \t "_blank) host.

**Note 3-13**

DNS messages are application layer messages that use UDP at the transport layer. Communication with DNS servers occur through port 53.

1. Steps 2 and 3

The local name server queries a root server with the request. The root server responds to the local name server with a list of IP addresses of TLD name servers responsible for the .edu suffix.

1. Steps 4 and 5

The local name server makes the same request to one of the TLD name servers responsible for the .edu suffix. The TLD name server responds with the IP address of the [mdc.edu](http://mdc.edu/" \t "_blank) authoritative server.

1. Steps 6 and 7

The local name server makes the request to the authoritative name server at Miami Dade Community College, which responds to the Cengage name server with the IP address of the [www.mdc.edu](http://www.mdc.edu/" \t "_blank) host.

1. Step 8

The local name server responds to the client resolver with the requested IP address. Both the Cengage name server and the Cengage client computer store the information in their DNS caches, and, therefore, don’t need to ask again until that information expires.

Requests sometimes involve additional name servers. Following are a few ways the process can get more complex:

* A client’s local caching server typically is not the same machine as the authoritative name server for the organization’s domain. Instead, the caching server exists only to resolve names for its own local clients.
* Name servers within a company might not have access to root servers. The local name server might forward the query to the name server at the company’s ISP, which might forward the query to a name server elsewhere on the Internet. This name server might query a root server; however, if any name server in the process has the requested information, it responds without the involvement of a root server, TLD name server, or authoritative name server.
* A TLD name server might be aware of an intermediate name server rather than the authoritative name server. When the local name server queries this intermediate name server, it might respond with the IP address of the authoritative name server.

Notice in these steps, the local name server kept working until the FQDN resolution was made, but other servers only aided in the process. As you can see, there are two types of DNS requests:

* [**Recursive lookup**](javascript://)—A query that demands a resolution or the answer “It can’t be found.” For example, the initial request the resolver makes to the local server is a recursive query. The local server must provide the information requested by the resolver, as in “The buck stops here.” If the local server doesn’t have that information already, it reaches out to other servers to find the information or determine the information is not available.
* [**Iterative lookup**](javascript://)—A query that does not demand resolution. For example, when the local server issues queries to other servers, the other servers only provide information if they have it.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 3-4cResource Records in a DNS Database

If you’ve ever worked with a database, you’re already familiar with some typical database components such as records and fields. DNS information, being a database, is similarly organized. Namespace databases are stored in DNS zone files, which are simple text files consisting of [**resource records**](javascript://) that each store specific kinds of information about the zone. These records consist of fields specific to the kind of information that record type should hold, such as the zone name, class (these days, that’s always Internet class), and record type. Other fields in each record vary according to the purpose of that record. A DNS administrator (including network administrators and website administrators) needs to be familiar with these record types. The following list describes some of the most common DNS record types found in a zone file:

* [**SOA (start of authority) record**](javascript://)—Listed at the beginning of the zone file and gives important information about the zone, such as a contact email address, when the zone was last updated, how long the zone information is valid until it should be refreshed, and necessary information for how to perform a zone transfer.
* [**A (address) record**](javascript://)—Stores the name-to-address mapping for a host. This resource record provides the primary function of DNS, which is to match a given FQDN to its IPv4 address in response to a [**forward lookup**](javascript://) request.
* [**AAAA (address) record**](javascript://) (called a “quad-A record”)—Holds the name-to-address mapping for IPv6 addresses.
* [**CNAME (canonical name) record**](javascript://)—Holds alternative names for a host, such as blog.mycompany.com or shop.mycompany.com. These names can be used in place of the [**canonical name**](javascript://), which is the complete and properly formatted name, such as www.mycompany.com. The web server at www.mycompany.com can then detect the request for the Blog or Shop page and direct the client to the appropriate page on the website.
* [**PTR (pointer) record**](javascript://)—Used for a [**reverse lookup**](javascript://), also called **[rDNS (reverse DNS)](javascript://)**, which provides a host name when you know its IP address. PTR records are usually created by ISPs and stored in a specially formatted reverse lookup zone file, or reverse zone. Reverse zones differ from a typical forward lookup zone file, or forward zone, that holds A records. In a reverse zone, the IP addresses must instead be stored in reverse—with the last octet listed first—plus the domain .in-addr.arpa. For example, the IP address 1.2.3.4 would be stored in a PTR record as 4.3.2.1.in-addr.arpa.
* [**NS (name server) record**](javascript://)—Indicates the authoritative name server for a domain. It’s mostly used for delegating subdomains to other name servers.
* [**MX (mail exchanger) record**](javascript://)—Identifies an email server and is used for email traffic.
* [**SRV (service) record**](javascript://)—Identifies the hostname and port of a computer that hosts a specific network service besides email, such as FTP or SIP.
* [**TXT (text) record**](javascript://)—Holds any type of free-form text. It might contain text designed to be read by humans regarding network, server, or accounting issues. Most often it’s used by the following:
  + **SPF (Sender Policy Framework)**—A validation system that helps fight spam by identifying the email servers allowed to send email on behalf of a domain.
  + **DKIM (DomainKeys Identified Mail)**—An authentication method that uses encryption to verify the domain name of an email’s sender.

### Exam Tip

While there are more than 100 types of DNS resource records, the CompTIA Network+ exam expects you to know about the nine types in the preceding list. Make sure you can explain the basic purpose of each DNS record type listed. Study the examples given in [Table 3-6](javascript://) to help you better understand each record type.

[Table 3-6](javascript://) lists some sample zone file entries. Each line, or record, contains the text IN, which refers to the class and indicates the record can be used by DNS servers on the Internet.

**Table 3-6**

### Records in a DNS zone file

| **Record** | **Description** |
| --- | --- |
| www.example.com IN SOA admin.example.com 2022052403 86400 7200 3600000 | Lists email contact information, date of latest revision, and time specifications indicating how long the information is valid; notice the @ symbol in the email is not allowed and so a period is used instead |
| www.example.com IN A 92.100.80.40 | Maps the server named www in the example.com domain to the IP address 92.100.80.40 |
| www.example.com IN AAAA 2001:db8:cafe:f9::d3 | Maps a name to an IPv6 address |
| demo.example.com IN CNAME www.example.com | Says that the www.example.com host can also be addressed by its alias name demo.example.com |
| 40.80.100.92.in-addr-arpa IN PTR www.example.com | Used for reverse lookup—that is, to find the name when you know the IP address; note the IP address is reversed and in-addr-arpa is appended to it |
| www.example.com IN NS [ns1.otherdns.com](http://ns1.otherdns.com/" \t "_blank) | Directs DNS queries to a third-party, authoritative DNS server |
| example.com IN MX 10  mail.us.example.com  example.com IN MX 20  mail2.us.example.com | Tells email servers the preferred routes to take, ordered by best route, when sending email to the example.com domain |
| \_sip.\_udp.example.com IN  SRV 0 75 5060  fastsip.example.com  \_sip.\_udp.example.com IN  SRV 0 25 5060  slowsip.example.com | Directs SIP traffic (\_sip.) to two SIP servers (fastsip.example.com and slowsip.example.com) using UDP (\_udp.) at the transport layer and the registered SIP port (5060); while the priority for both is 0 (the highest priority), the traffic load is distributed more heavily on the faster server (75) and more lightly on the slower server (25) |
| example.com IN TXT v=spf1 include:[outlook.com](http://outlook.com/" \t "_blank) ~all | Adds the [outlook.com](http://outlook.com/" \t "_blank) email server as an approved sender for the example.com domain; the phrase v=spf1 defines the SPF version |

Enlarge Table

An actual DNS zone file begins with a [**TTL (Time to Live)**](javascript://) directive that identifies how long the information in the file should be saved in a cache on a server. Administrators can set the TTL based on how frequently they expect the IP addresses to change. TTL information is included in zone transfers.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 3-4dDNS Server Software

By far, the most popular DNS server software is BIND (Berkeley Internet Name Domain), which is free, open-source software that runs on Linux, UNIX, and Windows platforms. You can download the BIND software from [isc.org](http://isc.org/" \t "_blank). Most Linux and UNIX distributions include BIND in the distribution.

Many other DNS server software products exist. For example, Windows Server has a built-in DNS service called DNS Server, which partners closely with AD (Active Directory) services. This server role enables AD clients to find the domain controller for initial authentication and, later, to find other computers that are identified by name on the network. DNS Server can even be configured to provide different information to external clients, who are typically accessing network resources over the Internet. Previously, it was necessary to run two different DNS servers, one for internal clients and one for external clients. Today, Windows Server is capable of more nuanced configuration, called split-brain or split-horizon deployment, to handle these two populations differently. For example, consider a fictional website, careers.example.com, that lists internal job postings for employees and different job postings and information posted to the public. DNS policies differentiate between internal and external clients, sending their requests for name resolution on the same FQDN to different IP addresses and, therefore, different information on the website.

**Applying Concepts 3-5**

### Change DNS Servers

In [Applying Concepts 3-2](javascript://), you practiced finding TCP/IP settings on a Windows 10 computer using the Settings app and the CLI. You might have noticed that, in the Settings app, you could only change the DNS settings if you turned off DHCP. While it’s possible to set preferred DNS servers without disabling DHCP, you must do so in Control Panel. It’s a bit of a challenge to find Control Panel in Windows 10. Here are a few options:

* In the Search box, start typing **control** and press **Enter** when the Control Panel app appears.
* Press **Win+R**, type **control**, and press **Enter**.
* Click **Start**, scroll down and click **Windows System**, and then click **Control Panel**.

You can pin Control Panel to the Start menu or to your taskbar at the bottom of your screen to make it more accessible in the future. With Control Panel open, right-click its icon in the taskbar, and click **Pin to taskbar**.

When Control Panel is used for projects in this course, steps are written for the Large icons view unless stated otherwise. This generally makes the most important items to technicians easier to access. To change the view in Control Panel, click the **View by** drop-down menu in the top right corner of the window. Windows will usually remember the last view you used the next time you open Control Panel.

As for DNS servers, in most cases, your computer is probably using the DNS servers your ISP provides. For various reasons (such as performance improvements, filter options, or outage problems), you might want to change your DNS servers to third-party options such as Google Public DNS, OpenDNS, Cloudflare, or Verisign DNS. You can search online for these DNS servers’ IP addresses. Then complete the following steps to configure your Windows 10 computer to refer to one of these DNS providers instead:

1. 1

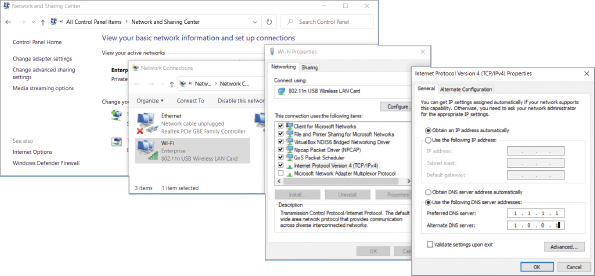
In Control Panel, click **Network and Sharing Center**. In the left pane, click **Change adapter settings**.

1. 2

Right-click the active network connection and click **Properties**. In the connection’s properties dialog box, click **Internet Protocol Version 4 (TCP/IPv4)**, and then click **Properties**. See [Figure 3-23](javascript://).

**Figure 3-23**

Configure TCP/IP for a network interface by using static or dynamic IP addressing



Enlarge Image

1. 3

Select Use the following DNS server addresses to manually assign DNS server addresses. For example, if you want to use Cloudflare’s DNS servers, you would enter 1.1.1.1 as the Preferred DNS server and 1.0.0.1 as the Alternate DNS server. Then click **OK**. Which DNS servers did you decide to use?

**Remember This…**

* Describe DNS’s global hierarchy of servers.
* Identify lookup types, including forward lookup, reverse lookup, recursive lookup, and iterative lookup.
* Explain the purpose of each DNS resource record type, including SOA, A, AAAA, CNAME, PTR, NS, MX, SRV, and TXT.

**Self-Check**

1. When your computer requests a DNS lookup, which DNS server holds the most reliable information for that DNS record?

Answer

* 1. Caching DNS server
  2. Forwarding DNS server
  3. Primary DNS server
  4. Root DNS server

1. Which DNS record type is listed first in a zone file?

Answer

* 1. A
  2. AAAA
  3. CNAME
  4. SOA

1. Which DNS record type is used to find an FQDN from a given IP address?

Answer

* 1. A
  2. CNAME
  3. PTR
  4. MX

**You’re Ready**

You’re now ready to complete [Project 3-3: Manage a DNS Cache](javascript://), or you can wait until you’ve finished reading this module.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# 3-5Troubleshooting Address Problems

### Certification

* 1.5

Explain common ports and protocols, their application, and encrypted alternatives.

* 5.3

Given a scenario, use the appropriate network software tools and commands.

* 5.5

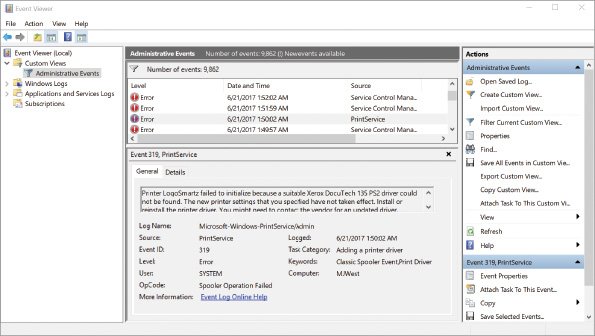
Given a scenario, troubleshoot general networking issues.

Average reading time: 30 minutes

Now that you are familiar with the basics of network addressing, you can learn how to solve problems with network connections. Event Viewer is one of the first places to start looking for clues when something goes wrong with a computer. It can provide valuable information about the problems the computer is experiencing, and it might even make suggestions for what to do next. For example, consider the printer error shown in [Figure 3-24](javascript://).

**Figure 3-24**

Event Viewer provided the diagnosis of a printer problem and recommended steps to fix the problem



Enlarge Image

When Event Viewer doesn’t give the information you need, move on to the TCP/IP troubleshooting commands discussed next.

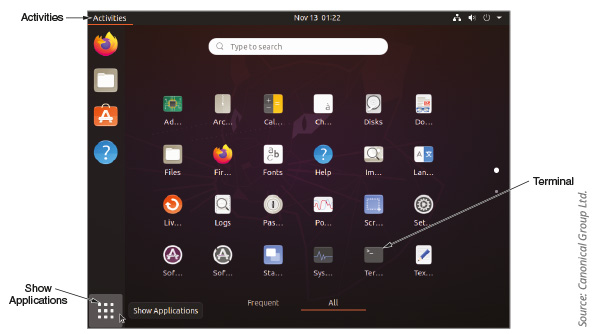
**Note 3-14**

You’ve already seen some basics of how to work with PowerShell or Command Prompt in Windows. As a network professional, you’ll also need to know how to work with Linux commands—you’ll get some practice with Linux in projects later in this course. When working on a Linux system, you’ll need to open a shell prompt. The steps for accessing a shell prompt vary depending on the Linux distribution that you’re using. For Ubuntu Desktop, use any of the following options to open Terminal:

* Click **Activities** at the top of the left sidebar, type **terminal**, and click **Terminal** (see [Figure 3-25](javascript://)).
* Click the **Show Applications** icon in the bottom left corner and click **Terminal** (see [Figure 3-25](javascript://)).
* Press **Ctrl**+**Alt**+**T** on your keyboard.

**Figure 3-25**

In Ubuntu, Terminal provides a CLI



Enlarge Image

Source: Canonical Group Ltd.

To close the shell prompt, click the red **X** icon or enter the **exit** command.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

## 3-5aTroubleshooting Tools

Command-line tools are a great way to get a look “under the hood” when something is going wrong on your network. Some of the most helpful tools are ping, ipconfig (Windows only), ip and ifconfig (Linux only), nslookup, and dig (Linux only). Let’s see what each of these tools can do. As you read, consider practicing the commands at a Windows command prompt or Linux shell prompt.

### ping

The [**ping (Packet Internet Groper)**](javascript://) utility is used to verify that TCP/IP is installed, bound to the NIC, configured correctly, and communicating with the network. Think about how a whale sends out a signal and listens for the echo. The nature of the echo can tell the whale a lot of information about the object the original signal bumped into. The ping utility starts by sending out a signal called an echo request to another computer, which is simply a request for a response. The other computer then responds to the request in the form of an echo reply. The protocol used by the echo request and echo reply is [**ICMP (Internet Control Message Protocol)**](javascript://), a lightweight protocol used to carry error messages and information about a network.

**Note 3-15**

When looking at command options, brackets indicate parameters you can add. Italics indicate information you would fill in with specific data, such as the target computer’s actual IP address.

Generally, the first tool you should use to test basic connectivity to the network, Internet, and specific hosts is ping. The ping command has several options, called parameters or switches. [Table 3-7](javascript://) gives some examples of how these parameters can be used. A few of them are listed below:

The ping command with all parameters and switches. To use the command, ping is typed followed by any switches or parameters that are required. They include the following. Hyphen a. Hyphen t. Hyphen n. Hyphen question mark. I P address. Host name. Forward slash, question mark.

**Table 3-7**

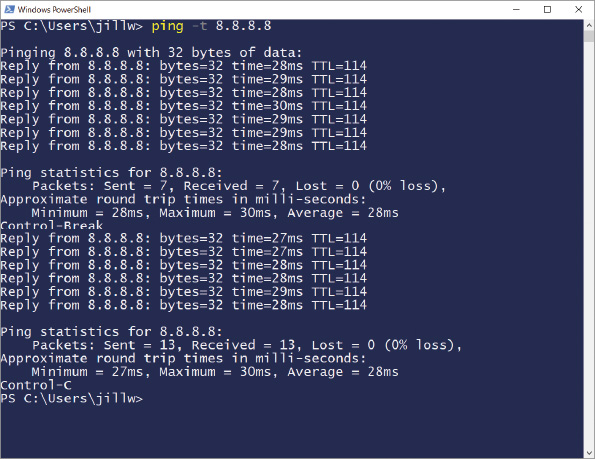
### Options for the ping command

| **Sample ping commands** | **Description** |
| --- | --- |
| ping [google.com](http://google.com/" \t "_blank) | Ping a host using its host name to verify you have Internet access and name resolution. [Google.com](http://google.com/" \t "_blank) is a reliable site to use for testing. |
| ping 8.8.8.8 | Ping an IP address on the Internet to verify you have basic Internet access without involving name resolution. The address 8.8.8.8, which is easy to remember, points to Google’s public DNS servers. |
| ping -a 8.8.8.8 | Test for name resolution and display the host name to verify DNS is working. |
| ping 92.10.11.200 | In this example, 92.10.11.200 is the address of a host on another subnet in your corporate network. This ping shows if you can reach that subnet. |
| ping 192.168.1.1 | In this example, 192.168.1.1 is the address of your default gateway. This ping shows if you can reach it. |
| ping 127.0.0.1 | Ping the loopback address, 127.0.0.1, to determine whether your workstation’s TCP/IP services are running. |
| ping localhost | This is another way of pinging your loopback address. |
| ping -? or ping /? | Display the help text for the ping command, including its syntax and a full list of parameters. |
| ping -t 192.168.1.1 | Continue pinging until interrupted. To display statistics while the ping continues running, press CTRL+Break (the Break key is usually paired with the Pause key). To stop pinging, press CTRL+C. Checking statistics while pinging a target outside your local network can provide insightful information on how well your Internet connection is performing. For example, you might ping one of Google’s public DNS servers at 8.8.8.8. [Figure 3-26](javascript://) shows a ping to this IP address with the -t parameter where statistics were displayed during the ping and again after the ping was cancelled. |
| ping -n 2 192.168.1.1 | Define the number of echo requests to send. By default, ping sends four echo requests. This example limits that number to two, or you could increase the number of requests. |

Enlarge Table

**Figure 3-26**

Results of a successful ping -t showing statistics while the ping is running and again after it’s complete



Enlarge Image

IPv6 networks use a version of ICMP called ICMPv6. Here are two variations of ping for different operating systems, which can be used with IPv6 addresses:

* ping6—On Linux computers running IPv6, use ping6 to verify whether an IPv6 host is available. When you ping a multicast address with ping6, you get responses from all IPv6 hosts on that subnet.
* ping -6—On Windows computers, use ping with the -6 parameter. The ping -6 command verifies connectivity on IPv6 networks.

**Note 3-16**

In Windows, the -6 parameter is not necessary when pinging an IPv6 address (as opposed to pinging a host name) because the format of the address itself specifies that an IPv6 host is being pinged.

For the ping6 and ping -6 commands to work over the Internet, you must have access to the IPv6 Internet. Your ISP might provide native IPv6 connectivity, or you might be able to use an IPv6 tunnel provided by an IPv6 tunnel broker service, such as IPv6 Tunnel Broker ([tunnelbroker.net](http://tunnelbroker.net/" \t "_blank)), offered by Hurricane Electric, or SixXS (Six Access at [sixxs.net/main](http://sixxs.net/main" \t "_blank)).

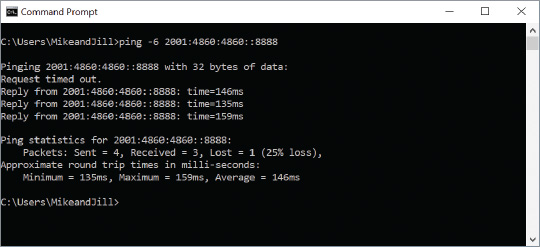
Try pinging Google’s IPv6 DNS server, as follows:

The ping command is used on Google's I P v 6 D N S server. The command used is the following. Ping hyphen 6 2001 colon 4860 colon 4860 colon colon 8888.

[Figure 3-27](javascript://) shows the results on a computer with an ISP that does provide access to the IPv6 Internet; the IPv6 ping was successful after a short delay.

**Figure 3-27**

After an initial delay, the ping -6 was successful



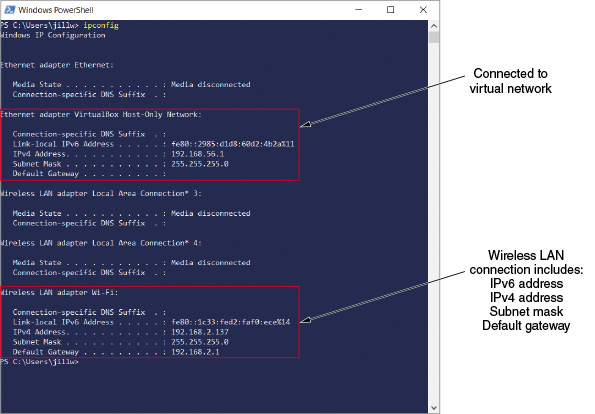
### ipconfig

The ipconfig command shows current TCP/IP addressing and domain name information on a Windows computer. You can also use ipconfig to change some of these settings. Here are two ways to use ipconfig:

* In a PowerShell or Command Prompt window, enter **ipconfig** to view IP configuration information (see [Figure 3-28](javascript://)). Notice which local connections are available on your computer and which ones are currently connected. Also locate your active connection’s IPv4 or IPv6 address, subnet mask, and default gateway.

**Figure 3-28**

This computer is connected to two different network interfaces, one of which is a virtual network from VirtualBox

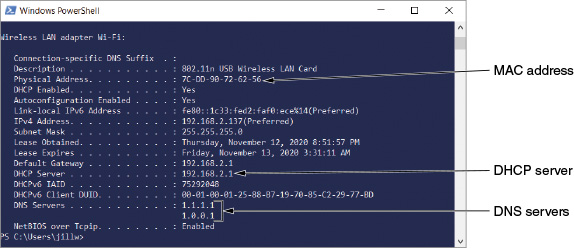


Enlarge Image

* The ipconfig command shows an abbreviated summary of configuration information. To see a more complete summary, enter the command **ipconfig /all**. See [Figure 3-29](javascript://) for an example.

**Figure 3-29**

ipconfig /all gives more information than ipconfig by itself



[Table 3-8](javascript://) describes some helpful parameters for the ipconfig command. Notice that, with the ipconfig command, you need to type a forward slash (/) before a parameter, rather than the hyphen you use with the ping command.

**Table 3-8**

### Examples of the ipconfig command

| **ipconfig command** | **Description** |
| --- | --- |
| ipconfig /? or ipconfig -? | Display the help text for the ipconfig command, including its syntax and a full list of parameters. |
| ipconfig /all | Display TCP/IP configuration information for each network adapter. |
| ipconfig /release | Release the IP address when dynamic IP addressing is being used. Releasing the IP address effectively disables the computer’s communications with the network until a new IP address is assigned. |
| ipconfig /release6 | Release an IPv6 IP address. |
| ipconfig /renew | Lease a new IP address (often the same one you just released) from a DHCP server. To solve problems with duplicate IP addresses, misconfigured DHCP, or misconfigured DNS, reset the TCP/IP connection by consecutively entering these two commands:  ipconfig /release  ipconfig /renew |
| ipconfig /renew6 | Lease a new IPv6 address from a DHCPv6 server. |
| ipconfig /displaydns | Display information about name resolutions that Windows currently holds in the DNS resolver cache. |
| ipconfig /flushdns | Flush—or clear—the name resolver cache, which might solve a problem when the browser cannot find a host on the Internet or when a misconfigured DNS server has sent wrong information to the resolver cache. |

Enlarge Table

### ip

On UNIX and Linux systems, use the **[ip](javascript://)** utility to view and manage TCP/IP settings. As with ipconfig on Windows systems, you can use ip to view and modify TCP/IP settings and to release and renew the DHCP configuration. The ip command is built in the following format, where options define variations of an action; objects can refer to a link, address, route, rule, etc.; and command defines what to do to the object.

The usage of the i p command which is used to view the T C P I P settings on a Unix or Linux computer. The command is as follows. i p is typed followed by required options and followed by the object. Optional parameters include various commands such as help.

When entering ip commands, you can abbreviate to save time. For example, the command ip link can be abbreviated ip l. The full spellout of a command is sometimes written inside brackets to show which part of the command is optional: ip l[ink].

Any ip commands that change the state of a link require elevated privileges. In Linux and UNIX systems, this is accomplished by logging in as the root user or by temporarily elevating the current user’s privileges with the sudo (superuser do) command. Also, any changes to link configurations are not saved by default—they will be lost upon the next restart unless the configuration changes are added to a startup script or edits are made to underlying configuration files.

**Note 3-17**

Remember that Linux and UNIX commands are case sensitive. Be sure to type ip and not Ip.

If your Linux or UNIX system provides a GUI (graphical user interface), first open a shell prompt from the desktop. At the shell prompt, you can use the ip commands listed in [Table 3-9](javascript://).

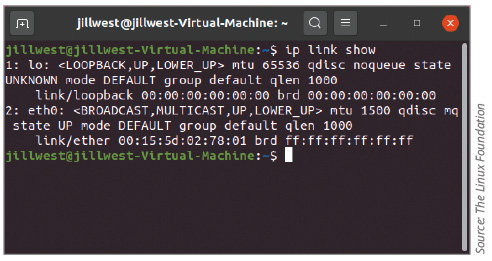
**Table 3-9**

### Some ip commands

| **ip command** | **Description** |
| --- | --- |
| ip link show | Display basic data link layer information, including the MAC address of the NIC, as shown in [Figure 3-30](javascript://). |
| sudo ip link set eth0 up | Bring the eth0 interface to an “up” state. Notice that elevated privileges and a password are required to execute this command. |
| sudo ip link set eth0 down | Bring the eth0 interface to a “down” state. Notice that elevated privileges and a password are required to execute this command. |
| ip address show | Display all IP addresses associated with a device. |
| sudo ip address delete 192.168.201.191/24 device eth0 | Remove an existing class C IP address from the device’s eth0 interface. Notice that elevated privileges and a password are required to execute this command. |
| sudo ip address add 192.168.201.191/24 device eth0 | Add a static class C IP address to the device’s eth0 interface. Notice that elevated privileges and a password are required to execute this command. |
| ip help | Display the full list of available objects and commands. |
| ip link help | Display options specific to the ip link command. |

**Figure 3-30**

Basic link information available through ip link show



Source: The Linux Foundation

### ifconfig

Similar to ip, the **[ifconfig](javascript://)** utility allows you to view and manage TCP/IP settings on UNIX and Linux systems. As with ipconfig on Windows systems and ip on UNIX/Linux systems, you can use ifconfig to view and modify TCP/IP settings; however, ifconfig is older and more limited than ip. Note that ifconfig has been deprecated in many Linux distributions, including Ubuntu. This means it’s not installed by default, and users are intended to use the ip command instead. However, you can install ifconfig by running the command sudo apt install net-tools.

If your Linux or UNIX system provides a GUI (graphical user interface), first open a shell prompt from the desktop. At the shell prompt, you can use the ifconfig commands listed in [Table 3-10](javascript://).

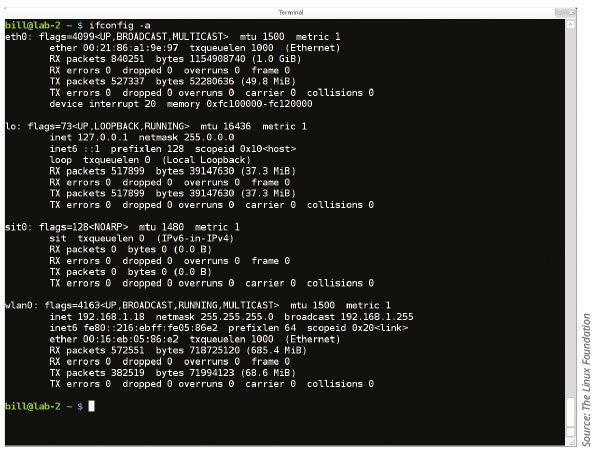
**Table 3-10**

### Some ifconfig commands

| **ifconfig command** | **Description** |
| --- | --- |
| ifconfig | Display basic TCP/IP information and network information, including the MAC address of the NIC. |
| ifconfig -a | Display TCP/IP information associated with every interface on a Linux device; can be used with other parameters. See [Figure 3-31](javascript://). |
| sudo ifconfig eth0 down | Mark the eth0 interface, or network connection, as unavailable to the network. Notice that elevated privileges and a password are required to execute this command. |
| sudo ifconfig eth0 up | Reinitialize the eth0 interface after it has been taken down (via the ifconfig eth0 down command), so that it is once again available to the network. Notice that elevated privileges and a password are required to execute this command. |
| ifconfig eth0 netmask 255.255.255.224 | Change the eth0 interface’s subnet mask to 255.255.255.224. |
| man ifconfig | Display the manual pages, called man pages, for the ifconfig command, which tells you how to use the command and about command parameters (similar to the ipconfig /? command in Windows). |

**Figure 3-31**

Detailed information available through ifconfig –a



Enlarge Image

Source: The Linux Foundation

**Note 3-18**

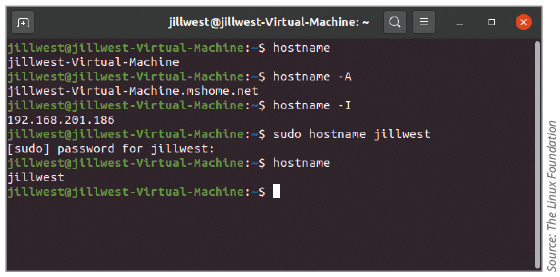
Other ifconfig parameters, such as those that apply to DHCP settings, vary according to the type and version of the UNIX or Linux system you use.

### hostname

The [**hostname**](javascript://) utility offers very basic commands used to display a device’s host name, either in Windows, UNIX, or Linux systems. In Windows, hostname has no additional parameters. In UNIX or Linux, hostname offers a few more options including the ability to change the computer’s name, as shown in [Figure 3-32](javascript://). [Table 3-11](javascript://) explains some of the options.

**Figure 3-32**

Use hostname to view or change a device’s host name



Source: The Linux Foundation

**Table 3-11**

### Some hostname commands

| **hostname command** | **Description** |
| --- | --- |
| hostname | Display a device’s hostname. |
| hostname -A | Display a device’s FQDN. |
| hostname -I | Resolve a device’s hostname with its IP address on the network. |
| sudo hostname jillwest | Set a new hostname on a device. Notice that elevated privileges and a password are required to execute this command. |

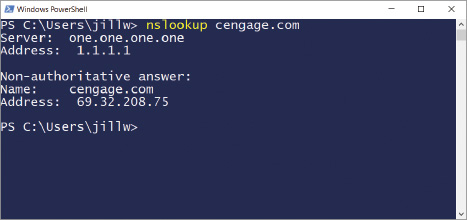
### nslookup

The **[nslookup (name space lookup)](javascript://)** utility allows you to query the DNS database from any computer on the network and find the host name of a device by specifying its IP address, or vice versa. This is useful for verifying that a host is configured correctly or for troubleshooting DNS resolution problems. For example, if you want to find out whether the host named cengage.com is operational, enter the command nslookup cengage.com.

[Figure 3-33](javascript://) shows the result of running a simple nslookup command. Notice that the command provides the target host’s IP address as well as the name and address of the primary DNS server that provided the information.

**Figure 3-33**

nslookup shows DNS server and web host information



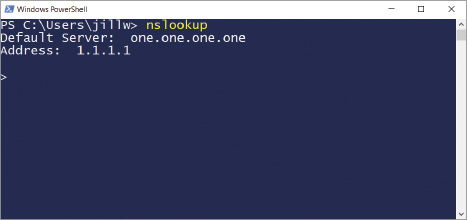
To find the host name of a device whose IP address you know, you need to perform a reverse DNS lookup: nslookup 69.32.208.74. In this case, the response would include the FQDN of the target host and the name and address of the primary DNS server that made the response.

The nslookup utility is available in two modes: interactive and noninteractive. Nslookup in noninteractive mode gives a response for a single nslookup command. This is fine when you’re investigating only one server, or when you’re retrieving single items of information at a time. To test multiple DNS servers at one time, use the nslookup utility in interactive mode, which makes available more of the utility’s options. To launch interactive mode, enter the **nslookup** command without any parameters.

As shown in [Figure 3-34](javascript://), after you enter this command, the command prompt changes to a greater-than symbol (>). You can then use additional commands to find out more about the contents of the DNS database. For example, on a computer running UNIX, you could view a list of all the host name and IP address correlations on a particular DNS server by entering the command ls.

**Figure 3-34**

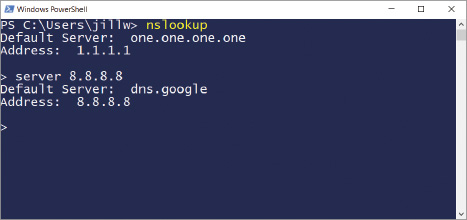
Interactive mode of the nslookup utility



You can change DNS servers from within interactive mode with the server subcommand and specifying the IP address of the new DNS server. Assign a new DNS server, such as Google’s public DNS server, with the command: server 8.8.8.8 (see [Figure 3-35](javascript://)).

**Figure 3-35**

The server subcommand can be used to change DNS servers



To exit nslookup interactive mode and return to the normal command prompt, enter exit.

Many other nslookup options exist. To see these options on a UNIX or Linux system, use the man nslookup command. On a Windows-based system, use the nslookup /? command.

### dig

The [**dig (domain information groper)**](javascript://) utility is available on Linux and macOS and provides more detailed domain information than nslookup. It’s installed by default in Ubuntu. You can install dig on other distributions with the command apt-get install dnsutils. Use dig to query DNS nameservers for information about host addresses and other DNS records. The dig utility is newer than nslookup; it uses more reliable sources of information to output its results and makes more advanced options available for complex queries. For a time, nslookup in Linux was deprecated in favor of dig (and a related command, host), but has since been resurrected because it’s considered easier to use than dig. Some sample dig commands are covered in [Table 3-12](javascript://).

**Table 3-12**

### Sample dig commands

| **Sample dig commands** | **Description** |
| --- | --- |
| dig google.com | Perform a DNS lookup on a domain name. |
| dig @8.8.8.8 google.com | Specify a name server (found at 8.8.8.8) to resolve the [google.com](http://google.com/" \t "_blank) domain. |
| dig @8.8.8.8 google.com MX | Request a list of all MX records in the [google.com](http://google.com/" \t "_blank) domain using a specific name server (found at 8.8.8.8). |
| dig google.com ANY | Request a list of all record types in the [google.com](http://google.com/" \t "_blank) domain. |
| dig -x 74.125.21.102 | Perform a reverse lookup on a Google IP address. |
| man dig | Display the man page for the dig command. |

### IP Scanner

While not a command-line tool, an [**IP scanner**](javascript://) can be used to gather information about all devices connected to a network, including host names, manufacturer names, operating systems, IP addresses, MAC addresses, interfaces used, and open ports with running services. You already practiced using Nmap in an earlier module, and you’re learning to use ipconfig in this module. While Nmap and ipconfig are powerful tools, they can lack efficiency in a large, corporate environment. More sophisticated IP scanners can help detect and manage large numbers of devices throughout a complex network.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# Module Review

## 3-6a**Module Summary**

### Addressing Overview

* Addressing methods operate at the data link, network, transport, and application layers of the OSI model.
* A MAC address is embedded on every NIC on the globe and is assumed to be unique to that NIC.
* An IP address is assigned to nearly every interface and can be used to find any computer in the world if the IP address is public on the Internet.
* A port is a number used by the transport layer to identify one application among several that might be running on a host.
* Every host on a network is assigned a unique character-based name called the FQDN (fully qualified domain name), consisting of a host name and a domain name.
* MAC addresses contain two parts, are 48 bits long, and are written as hexadecimal numbers separated by colons. The first 24 bits (six hex characters) are known as the OUI (Organizationally Unique Identifier), which identifies the NIC’s manufacturer. The last 24 bits make up the extension identifier or device ID and identify the device itself.

### IP Addresses

* You can permanently assign a static IP address to a device, or you can configure the device to request and receive (or lease) a dynamic IP address from a DHCP server each time it connects to the network. A DHCP (Dynamic Host Configuration Protocol) server manages the dynamic distribution of IP addresses to devices on a network.
* Network technicians need to be comfortable with the CLI (command line interface) because it is quicker and often more powerful and flexible than a GUI (graphical user interface).
* A subnet mask indicates what portion of an IP address is the network portion, called the network ID or network address, and what part is the host portion, called the host ID or node ID.
* When using classful addressing, which is an older method of managing IP address ranges, the dividing line between the network and host portions is determined by the numerical range the IP address falls in. In contrast to classful addressing, classless addressing allows the dividing line between network and host portions to fall anywhere along the string of binary bits in an IP address. Moving this dividing line allows for segmenting networks within networks in a process called subnetting.
* Class A (1.x.y.z to 126.x.y.z), class B (128.0.x.y to 191.255.x.y), and class C (192.0.0.x to 223.255.255.x) addresses, for the most part, can be used to connect to and access Internet resources. Class D (begins with octets 224 through 239) and class E (begins with 240 through 254) IP addresses are not available for general use.
* To conserve its public IP addresses, a company can instead use private IP addresses for devices on its private networks—that is, devices that do not directly connect to the Internet but instead communicate through a representative device such as a router. These addresses were set aside for private use by IANA’s RFC1918 (Request for Comment 1918) document, released in 1996.
* A broadcast message is read by every node on the network. A LAN, which consists of all the nodes a broadcast reaches, can be referred to as a broadcast domain. Routers don’t forward broadcast messages, thus creating a boundary for a LAN.
* A DHCP scope, or DHCP pool, defines a range of IP addresses to be assigned to clients when they request an address. Other scope options include limiting the lease time and identifying the default gateway and DNS servers.
* When private devices need access to other networks or the Internet, a public-facing gateway (such as a router or firewall) substitutes the private IP addresses used by computers on the private network with its own public IP address. This process is called NAT (Network Address Translation). Besides requiring only a single public IP address for the entire private network, another advantage of NAT is security; the gateway hides the entire private network behind this one address.
* IPv6 offers a lot more public IP addresses than does IPv4 and also provides improvements to routing capabilities and speed of communication. In IPv6, a link, sometimes called the local link, is any LAN bounded by routers. Neighbors are two or more nodes on the same link.
* IPv6 supports three types of IP addresses, classified by how the address is used: unicast address (including global address, link local address, and loopback address), multicast address, and anycast address. IPv6 reduces network traffic by eliminating broadcasting.
* IPv6 addressing uses a process called SLAAC (stateless address autoconfiguration) so that a computer can autoconfigure its own link local IP address without the help of a DHCPv6 server. After generating an IPv6 address and confirming the address is unique on the network, the computer asks if a router on the network can provide configuration information. This message is called an RS (router solicitation) message. If a router responds with DHCP information in what’s called an RA (router advertisement) message, the computer uses whatever information this might be, such as the IP addresses of DNS servers or the network prefix.

### Ports and Sockets

* A port is a number assigned to a process, such as an application or a service, that can receive data. Whereas an IP address is used to find a computer, a port is used to find a process running on that computer. A socket consists of both a host’s IP address and a process’s TCP or UDP port, with a colon separating the two values.
* Well-known ports range from 0 to 1023 and are assigned by IANA to widely used and well-known utilities and applications. Registered ports range from 1024 to 49151 and can be used temporarily by processes for non-standard assignments to increase security. Dynamic and private ports range from 49152 to 65535 and are open for use without restriction.
* Which port a protocol communicates over becomes especially relevant when configuring firewalls. A firewall works by blocking traffic on all ports except those that are specifically approved by the network admin.

### Domain Names and DNS

* Name resolution is the process of discovering the IP address of a host when its FQDN is known. An authoritative name server is the authority on computer names and their IP addresses for computers in their domains. The domains an organization is responsible for managing are called a DNS zone. DNS (Domain Name System) associates computer names with IP addresses. DNS namespace databases are stored on thousands of servers around the world, rather than being centralized on a single server or group of servers.
* The primary DNS server holds the authoritative DNS database for the organization’s zones. When a secondary DNS server needs to update its database, it makes the request to the primary server for the update; this update process is called a zone transfer.
* A caching DNS server accesses public DNS data and caches the DNS information it collects. This server receives DNS queries from local network clients and works to resolve them by contacting other DNS servers for information. Caching DNS servers do not store zone files, which is why they must rely on their caches and resolution efforts.
* DNS name servers are organized in a global hierarchical structure. At the root level, 13 clusters of root DNS servers hold information used to locate the TLD (top-level domain) servers. These TLD servers hold information about the authoritative servers owned by various organizations.
* There are two types of DNS requests: recursive lookups, which demand a resolution or the answer that the information can’t be found, and iterative lookups, which do not demand resolution and only provide information if the server already has it.
* Namespace databases are stored in DNS zone files, which are simple text files consisting of resource records that each store specific kinds of information about the zone. These records consist of fields specific to the kind of information that record type should hold, such as the zone name, class (these days, that’s always Internet class), and record type. Other fields in each record vary according to the purpose of that record.

### Troubleshooting Address Problems

* The ping (Packet Internet Groper) utility is used to verify that TCP/IP is installed, bound to the NIC, configured correctly, and communicating with the network. The ping utility starts by sending out a signal called an echo request to another computer, which is simply a request for a response. The other computer then responds to the request in the form of an echo reply. The protocol used by the echo request and echo reply is ICMP (Internet Control Message Protocol), a lightweight protocol used to carry error messages and information about a network.
* The ipconfig command shows current TCP/IP addressing and domain name information on a Windows computer. You can also use ipconfig to change some of these settings.
* On UNIX and Linux systems, use the ip utility to view and manage TCP/IP settings. As with ipconfig on Windows systems, you can use ip to view and modify TCP/IP settings and to release and renew the DHCP configuration. Any ip commands that change the state of a link require elevated privileges. In Linux and UNIX systems, this is accomplished by logging in as the root user or by temporarily elevating the current user’s privileges with the sudo (superuser do) command. Also, any changes to link configurations are not saved by default—they will be lost upon the next restart unless the configuration changes are added to a startup script or edits are made to underlying configuration files.
* Similar to ip, use the ifconfig utility to view and manage TCP/IP settings on UNIX and Linux systems. The ifconfig utility has been deprecated in many Linux distributions, including Ubuntu. This means it’s not installed by default, and users are intended to use the ip command instead. However, you can install it by running the command sudo apt install net-tools.
* The hostname utility offers very basic commands used to display a device’s host name, either in Windows, UNIX, or Linux system. In Windows, hostname has no additional parameters. In UNIX or Linux, hostname offers a few more options including the ability to change the computer’s name.
* The nslookup (name space lookup) utility allows you to query the DNS database from any computer on the network and find the host name of a device by specifying its IP address, or vice versa. This is useful for verifying that a host is configured correctly or for troubleshooting DNS resolution problems.
* The dig (domain information groper) utility is available on Linux and macOS and provides more detailed domain information than nslookup. It’s installed by default in Ubuntu. You can install dig on other distributions with the command apt-get install dnsutils. Use dig to query DNS nameservers for information about host addresses and other DNS records.
* While not a command-line tool, an IP scanner can be used to gather information about all devices connected to a network, including host names, manufacturer names, operating systems, IP addresses, MAC addresses, interfaces used, and open ports with running services. Sophisticated IP scanners can help detect and manage large numbers of devices throughout a complex network.

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# Module Review

## 3-6b**Key Terms**

* [**A (address) record**](javascript://)
* [**AAAA (address) record**](javascript://)
* [**anycast address**](javascript://)
* [**APIPA (Automatic Private IP Addressing)**](javascript://)
* [**authoritative name server**](javascript://)
* [**broadcast**](javascript://)
* [**broadcast domain**](javascript://)
* [**caching DNS server**](javascript://)
* [**canonical name**](javascript://)
* [**CIDR (Classless Interdomain Routing) notation**](javascript://)
* [**classful addressing**](javascript://)
* [**classless addressing**](javascript://)
* [**CNAME (canonical name) record**](javascript://)
* [**default gateway**](javascript://)
* [**device ID**](javascript://)
* [**DHCP (Dynamic Host Configuration Protocol)**](javascript://)
* [**DHCP reservation**](javascript://)
* [**DHCP scope**](javascript://)
* [**DHCP scope exhaustion**](javascript://)
* [**dig (domain information groper)**](javascript://)
* [**DNS zone**](javascript://)
* [**domain name**](javascript://)
* [**dual stacked**](javascript://)
* [**dynamic IP address**](javascript://)
* [**EUI-64 (Extended Unique Identifier-64)**](javascript://)
* [**exclusion range**](javascript://)
* [**extension identifier**](javascript://)
* [**forward lookup**](javascript://)
* [**forwarding DNS server**](javascript://)
* [**FQDN (fully qualified domain name)**](javascript://)
* [**gateway**](javascript://)
* [**global address**](javascript://)
* [**host ID**](javascript://)
* [**host name**](javascript://)
* [**hostname**](javascript://)
* [**IANA (Internet Assigned Numbers Authority)**](javascript://)
* [**ICANN (Internet Corporation for Assigned Names and Numbers)**](javascript://)
* [**ICMP (Internet Control Message Protocol)**](javascript://)
* [**ifconfig**](javascript://)
* [**interface**](javascript://)
* [**interface ID**](javascript://)
* [**ip**](javascript://)
* [**IP reservation**](javascript://)
* [**IP scanner**](javascript://)
* [**IPAM (IP address management)**](javascript://)
* [**ipconfig**](javascript://)
* [**IPv4 (Internet Protocol version 4)**](javascript://)
* [**IPv6 (Internet Protocol version 6)**](javascript://)
* [**iterative lookup**](javascript://)
* [**LDAP (Lightweight Directory Access Protocol)**](javascript://)
* [**LDAPS (Lightweight Directory Access Protocol over SSL)**](javascript://)
* [**lease time**](javascript://)
* [**link**](javascript://)
* [**link local address**](javascript://)
* [**loopback address**](javascript://)
* [**MAC address table**](javascript://)
* [**MAC reservation**](javascript://)
* [**multicast**](javascript://)
* [**multicast address**](javascript://)
* [**multicast flooding**](javascript://)
* [**MX (mail exchanger) record**](javascript://)
* [**name resolution**](javascript://)
* [**NAT (Network Address Translation)**](javascript://)
* [**neighbor**](javascript://)
* [**network ID**](javascript://)
* [**node ID**](javascript://)
* [**NS (name server) record**](javascript://)
* [**nslookup (name space lookup)**](javascript://)
* [**NTP (Network Time Protocol)**](javascript://)
* [**octet**](javascript://)
* [**OUI (Organizationally Unique Identifier)**](javascript://)
* [**PAT (Port Address Translation)**](javascript://)
* [**ping (Packet Internet Groper)**](javascript://)
* [**primary DNS server**](javascript://)
* [**private IP address**](javascript://)
* [**process**](javascript://)
* [**PTR (pointer) record**](javascript://)
* [**public IP address**](javascript://)
* [**RA (router advertisement)**](javascript://)
* [**rDNS (reverse DNS)**](javascript://)
* [**recursive lookup**](javascript://)
* [**registered port**](javascript://)
* [**resource record**](javascript://)
* [**reverse lookup**](javascript://)
* [**RFC1918 (Request for Comment 1918)**](javascript://)
* [**root DNS server**](javascript://)
* [**RS (router solicitation)**](javascript://)
* [**scope option**](javascript://)
* [**secondary DNS server**](javascript://)
* [**session**](javascript://)
* [**SLAAC (stateless address autoconfiguration)**](javascript://)
* [**SMB (Server Message Block)**](javascript://)
* [**SOA (start of authority) record**](javascript://)
* [**socket**](javascript://)
* [**SQLnet**](javascript://)
* [**SRV (service) record**](javascript://)
* [**static IP address**](javascript://)
* [**stratum**](javascript://)
* [**subnet**](javascript://)
* [**subnet ID**](javascript://)
* [**subnet mask**](javascript://)
* [**subnetting**](javascript://)
* [**syslog (system log)**](javascript://)
* [**TFTP (Trivial File Transfer Protocol)**](javascript://)
* [**TLD (top-level domain)**](javascript://)
* [**TTL (Time to Live)**](javascript://)
* [**tunneling**](javascript://)
* [**TXT (text) record**](javascript://)
* [**unicast address**](javascript://)
* [**URL (uniform resource locator)**](javascript://)
* [**well-known port**](javascript://)
* [**zone transfer**](javascript://)

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# Module Review

## 3-6c**Review Questions**

1. Which part of a MAC address is unique to each manufacturer?
   1. The network identifier
   2. The OUI
   3. The device identifier
   4. The physical address
2. What type of device does a computer turn to when attempting to connect with a host with a known IP address on another network?
   1. Default gateway
   2. DNS server
   3. Root server
   4. DHCP server
3. What decimal number corresponds to the binary number 11111111?
   1. 255
   2. 256
   3. 127
   4. 11,111,111
4. Suppose you send data to the 11111111 11111111 11111111 11111111 IP address on an IPv4 network. To which device(s) are you transmitting?
   1. All devices on the Internet
   2. All devices on your local network
   3. The one device that is configured with this IP address
   4. No devices
5. When your computer first joins an IPv6 LAN, what is the prefix of the IPv6 address the computer first configures for itself?
   1. FF00::/8
   2. ::1/128
   3. 2000::/3
   4. FE80::/64
6. If you are connected to a network that uses DHCP, and you need to terminate your Windows workstation’s DHCP lease, which command would you use?
   1. ipconfig /release
   2. ipconfig /renew
   3. ifconfig /release
   4. ifconfig /renew
7. Which of these commands has no parameters in Windows?
   1. ping
   2. ipconfig
   3. hostname
   4. nslookup
8. Which DNS server offers the most current resolution to a DNS query?
   1. Primary DNS server
   2. Root DNS server
   3. Caching DNS server
   4. TLD DNS server
9. You have just brought online a new secondary DNS server and notice your network-monitoring software reports a significant increase in network traffic. Which two hosts on your network are likely to be causing the increased traffic and why?
   1. The caching and primary DNS servers because the caching server is requesting zone transfers from the primary server
   2. The secondary and primary DNS servers because the secondary server is requesting zone transfers from the primary server
   3. The root and primary DNS servers because the primary server is requesting zone transfers from the root server
   4. The web server and primary DNS server because the web server is requesting zone transfers from the primary DNS server
10. Which type of DNS record identifies an email server?
    1. AAAA record
    2. CNAME record
    3. MX record
    4. PTR record
11. What is the range of addresses that might be assigned by APIPA?
12. You are the network manager for a computer training center that allows students to bring their own laptops to class for learning and taking notes. Students need access to the Internet, so you have configured your network’s DHCP server to issue IP addresses automatically. Which DHCP option should you modify to make sure you are not wasting addresses used by students who have left for the day?
13. You have decided to use SNAT and PAT on your small office network. At minimum, how many IP addresses must you obtain from your ISP for all five clients in your office to be able to access servers on the Internet?
14. Explain how the bits of an IPv6 address are organized and describe IPv6 shorthand notation.
15. FTP sometimes uses a random port for data transfer, but an FTP server always, unless programmed otherwise, listens to the same port for session requests from clients. What port does an FTP server listen on?
16. You issue a transmission from your workstation to the following socket on your LAN: 10.1.1.145:53. Assuming your network uses standard port designations, what application layer protocol handles your transmission?
17. Suppose you want to change the default port for RDP as a security precaution. What port does RDP use by default, and from what range of numbers should you select a private port number?
18. You have just set up a new wireless network at your house, and you want to determine whether your Linux laptop has connected to it and obtained a valid IP address. What command will give you the information you need?
19. While troubleshooting a network connection problem for a coworker, you discover the computer is querying a nonexistent DNS server. What command-line utility can you use to assign the correct DNS server IP address?
20. When running a scan on your computer, you find that a session has been established with a host at the address 208.85.40.44:443. Which application layer protocol is in use for this session? What command-line utility might you use to determine the domain name of the other computer?

Go to pg.

[**help**](javascript://)

Application Opened

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?#header)

# Module Review

## 3-6d**Hands-On Projects**

**Note 3-19**

Websites and applications change often. While the instructions given in these projects were accurate at the time of writing, you might need to adjust the steps or options according to later changes.

**Note to Instructors and Students:** A rubric is provided for evaluating student performance on these projects. Please see Appendix D.

**Project 3-1**

### Create a NAT Translation Table Entry

* **Estimated Time:** 20 minutes
* **Objective:** Given a scenario, configure a subnet and use appropriate IP addressing schemes. (Obj. 1.4)
* **Resources:**
  + No special resources required
* **Context:** Your corporation hosts a website at the static public IP address 92.110.30.123. A router directs this traffic to a web server at the private IP address 192.168.11.100. However, the web server needs a hardware upgrade and will be down for two days. Your network administrator has asked you to configure the router so that requests to the IP address 92.110.30.123 are redirected to the backup server for the website, which has the private IP address 192.168.11.110. The router’s inside Ethernet interface uses IP address 192.168.11.254, and its outside interface uses the IP address 92.110.30.65. Answer the following questions about the new static route you’ll be creating:
  + 1

What is the router’s outside interface IP address?

* + 2

What is the router’s inside interface IP address?

* + 3

What is the website’s public IP address?

* Use the example given in [Figure 3-15](javascript://) earlier in the module as a template to create the NAT translation table entries for the address translation. For the subnet masks, use the default subnet mask for a Class C IP address license. Include appropriate comment lines in your table. **Take a screenshot of your NAT translation table**; submit this visual with your answers to this project’s questions.

**Project 3-2**

### Change IPv6 Autoconfiguration Settings

* **Estimated Time:** 20 minutes (+15 minutes for group work, if assigned)
* **Objective:** Given a scenario, configure a subnet and use appropriate IP addressing schemes. (Obj. 1.4)
* **Group Work:** This project includes enhancements when assigned as a group project.
* **Resources:**
  + Windows 10 computer with administrative access
  + Internet access
* **Context:** By default, when configuring an IPv6 address, Windows 10 generates a random number to fill out the bits needed for the NIC portion of the IPv6 address. This security measure helps conceal your device’s MAC address, and it further protects your privacy by generating a new number every so often. There may be times, however, when you need your system to maintain a static IPv6 address. To do this, you can disable IPv6 autoconfiguration using the netsh utility in an elevated PowerShell or Command Prompt window. Forcing the computer to use SLAAC to generate its IPv6 address will result in the same IPv6 address every time. Complete the following steps:
  + 1

In this project, you’ll use the netsh utility. Do some research online about this tool and answer the following questions:

* + 1. What is netsh used for?
    2. What is the role of a netsh context?
    3. What netsh command access the interface context for managing network connections?
  + 2

Open an elevated PowerShell or Command Prompt window.

* + 3

Enter **ipconfig /all** and find the TCP/IP information for the active network connection. **Take a screenshot** of this information; submit this visual with your answers to this project’s questions. What is your computer’s current IPv6 address and MAC address? Carefully compare the two addresses. Are they in any way numerically related?

* + 4

To disable the random IP address generation feature, enter the command:

**netsh interface ipv6 set global randomizeidentifiers**=**disabled**

* + 5

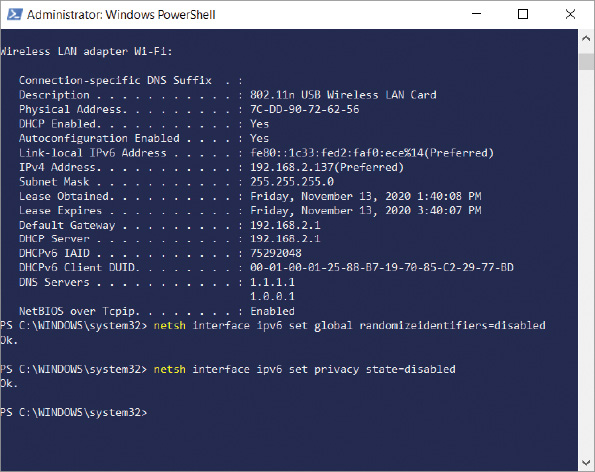
To instruct Windows to use the EUI-64 standard instead of the default settings, enter the command:

**netsh interface ipv6 set privacy state**=**disabled**

[Figure 3-38](javascript://) shows where both commands were entered and accepted.

**Figure 3-38**

PowerShell outputs a confirmation for each netsh command entered



Enlarge Image

* + 6

Enter **ipconfig /all** again. What is your computer’s new IPv6 address? How closely does this number resemble the MAC address? Notice after FE80:: that the fixed value FF FE has been inserted halfway through the MAC address values. The host portion of the IPv6 address might use a slightly different value than the OUI in the MAC address because the seventh bit of the MAC address is inverted.

* + 7

**For group assignments:** Complete the following steps:

* + 1. Attempt to ping each other’s devices using ping -6 and IPv6 addresses. What response did you get?
    2. Attempt to ping Google’s IPv6 DNS address on the Internet: 2001:4860:4860::8888. What response did you get?
    3. There are many reasons why pinging an IPv6 address on your local network might not work even if the network and its devices are functioning properly. For example, your LAN might not support IPv6. In some cases, you might have successfully pinged an IPv6 address on your local network but not on the Internet. If one or both of your pings did not work, spend a few moments with your group doing some investigating and troubleshooting to see if you can determine where the IPv6 ping is failing and what you would need to do to fix it. What possibilities did you come up with?
  + 8

Re-enable random IPv6 address generation with these two commands:

**netsh interface ipv6 set global randomizeidentifiers**=**enabled**

**netsh interface ipv6 set privacy state**=**enabled**

**Project 3-3**

### Manage a DNS Cache

* **Estimated Time:** 10 minutes
* **Objective:** Explain the use and purpose of network services. (Obj. 1.6)
* **Resources:**
  + Windows 10 computer with administrative access
  + Internet access
* **Context:** You have learned that clients as well as name servers store DNS information to associate names with IP addresses. In this project, you view the contents of a local DNS cache, clear it, and view it again after performing some DNS lookups. Then you change DNS servers and view the DNS cache once again. Complete the following steps:
  + 1

To view the DNS cache, open an elevated PowerShell or Command Prompt window and enter the following command: **ipconfig /displaydns**

* + 2

If this computer has been used to resolve host names with IP addresses—for example, if it has been used to retrieve email or browse the web—a list of locally cached resource records appears. Read the file to see what kinds of records have been saved, using the scroll bar if necessary. What is the most common record type in this list?

* + 3

Clear the DNS cache with this command: **ipconfig /flushdns**

The operating system confirms that the DNS resolver cache has been flushed. One circumstance in which you might want to empty a client’s DNS cache is if the client needs to reach a host whose IP address has changed (for example, a website whose server was moved to a different hosting company). If the DNS information is locally cached, the client will continue to look for the host at the old location. Clearing the cache allows the client to retrieve the new IP address for the host.

* + 4

View the DNS cache again with the command: **ipconfig /displaydns**

* + 5

Open a browser window and navigate to three websites you have not recently visited, such as [howstuffworks.com](http://howstuffworks.com/" \t "_blank), [nautil.us](http://nautil.us/" \t "_blank), and [mapcrunch.com](http://mapcrunch.com/" \t "_blank).

* + 6

Return to the PowerShell or Command Prompt window and view the DNS cache containing the new list of resource records. **Take a screenshot** of one of these records that was collected in response to your browser activity; submit this visual with your answers to this project’s questions.

### Exam Tip

* The Network+ exam includes two types of questions: performance-based questions (called PBQs) and multiple-choice questions. You’re already familiar with multiple-choice questions. PBQs give you a scenario of some kind and require you to interact with tools such as a network diagram, user interface, or command-line interface to complete tasks related to the scenario. A common performance-based question on the Network+ exam requires you to know how to manage DNS from a CLI using ipconfig. You might also see a multiple-choice question that covers similar information.

**Project 3-4**

### Download and Use an IP Scanner

* **Estimated Time:** 20 minutes
* **Objective:** Given a scenario, use the appropriate network software tools and commands. (Obj. 5.3)
* **Group Work:** This project includes enhancements when assigned as a group project.
* **Resources:**
  + Windows 10 computer with administrative access
  + Internet access
* **Context:** You’ve already seen the kind of information you can detect about your network devices using Nmap. In this project, you will use a free, popular tool called Advanced IP Scanner to detect devices on your network and determine what additional information an IP scanner can collect. Complete the following steps:

### Caution

Take note that scanning a network you don’t own or don’t have permission to scan is illegal. Do not use an IP scanner on public Wi-Fi networks at all. Also don’t use an IP scanner on any network you don’t own unless you have written permission from the owner to scan the network.

* + 1

Go to [advanced-ip-scanner.com](http://advanced-ip-scanner.com/" \t "_blank). Download and install the free application Advanced IP Scanner. When the installation is finished, run Advanced IP Scanner.

* + 2

In the user interface, notice you have two options for the scan: Scan the local machine’s subnet or scan a class C subnet. Depending on your network configuration, there might be no difference between these two options. If there is, choose the option that is most appropriate for your network. Also, if you are using the computer on which you installed your VMs from earlier Capstone Projects, the scan might target two IP ranges. What IP range(s) will you be scanning?

* + 3

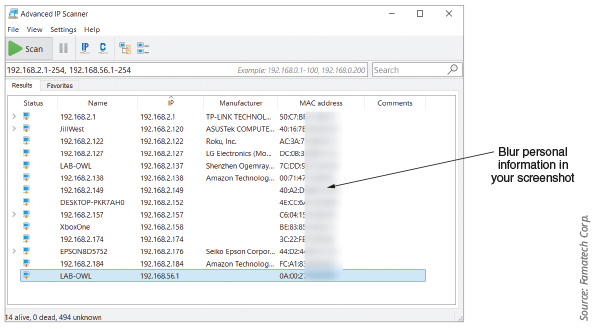
When you’re ready, click **Scan**. Give the scanner a few minutes to complete the scan.

* + 4

When the scan is complete, **take a screenshot** of the results. Blur out any private information and submit this visual with your answers to this project’s questions. [Figure 3-39](javascript://) shows the results of one network scan. Even some devices that do not have names were identifiable by manufacturer, such as the Roku device.

**Figure 3-39**

Advanced IP Scanner found and identified several devices on this LAN



Enlarge Image

Source: Famatech Corp.

* + 5

What surprises you about the results of your scan? Are there any devices you can’t identify that you need to research further? Keep in mind that it’s important to track which devices are connected to your network to ensure no one is using your network without permission and to ensure no devices are leaking sensitive information to the Internet or being maliciously controlled over the Internet.

* + 6

Depending on the device, Advanced IP Scanner offers some remote-control options. For example, right-click one of the devices in your scan results and explore the options in the pop-up menu. Some of these tools require Radmin, which is a free remote access application. You can experiment with this tool if you want to. You also have some command-line tools in Advanced IP Scanner. Use the Ping tool to check the connection with one of the devices on your network. How does this ping function differently than the pings you’ve run in other projects? When you’re ready, press **Ctrl+C** to stop the ping.

* + 7

**For group assignments:** One option for remotely accessing a computer through Advanced IP Scanner is RDP for Windows computers. Find another group member’s computer in your scan results, right-click, point to **Tools**, and click **RDP**. The other person will need to enter their user account credentials. Alternatively, each group member can create a guest account and share those credentials with group members to use with RDP from Advanced IP Scanner.

* + 8

In your wiki, add a new page titled **Applications:AdvancedIPScanner**. Indicate the module and project number for this installation, the computer you used for this project, a brief description of what you learned, and any other information you might find helpful when using Advanced IP Scanner later.

Go to pg.

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/fullbook.html?" \l "header)

## 3-2aIPv4 Addresses

A 32-bit IP address is organized into four groups of 8 bits each, which are presented as four decimal numbers separated by periods, such as 72.56.105.12. Each of these four groups is called an octet. The largest possible 8-bit number is 11111111, which is equal to 255 in decimal. So, the largest possible IP address in decimal is 255.255.255.255. In binary, this number is written 11111111.11111111.11111111.11111111. Each of the four octets can be any number from 0 to 255, making a total of nearly 4.3 billion IPv4 addresses (256 × 256 × 256 × 256). Some IP addresses are reserved, so these numbers are approximations. How are IPv4 addresses determined, and what information do they offer?

### Format of IPv4 Addresses

The first part of an IP address identifies the network, and the last part identifies the host. Where this dividing line falls between network and host bits varies according to several factors. When using [**classful addressing**](javascript://), which is an older method of managing IP address ranges, the dividing line between the network and host portions is determined by the numerical range the IP address falls in. Classful IPv4 addresses are categorized into five classes: class A, class B, class C, class D, and class E.

Classes A, B, and C addresses, for the most part, can be used to connect to and access Internet resources. [Table 3-1](javascript://) shows the numerical ranges for these classes of IPv4 addresses. Class D and class E IPv4 addresses are not available for general use. Class D addresses begin with octets 224 through 239 and are used for [**multicast**](javascript://) transmissions, in which one host sends messages to multiple hosts. An example of this is when a host transmits a videoconference over the Internet to multiple participants. Class E addresses, which begin with 240 through 254, are reserved for research.

**Table 3-1**

### IP address classes

| **Class** | **Network octets** | **Approximate number of possible networks** | **Approximate number of possible hosts in each network** |
| --- | --- | --- | --- |
| A | 1.x.y.z to 126.x.y.z with subnet mask of 255.0.0.0 | 126 | 16 million |
| B | 128.0.x.y to 191.255.x.y with subnet mask of 255.255.0.0 | 16,000 | 65,000 |
| C | 192.0.0.x to 223.255.255.x with subnet mask of 255.255.255.0 | 2 million | 254 |

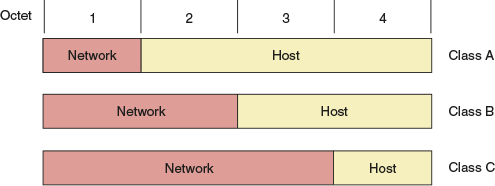
### Exam Tip

The CompTIA Network+ exam expects you to be able to identify the class of any IPv4 address. For the exam, memorize the second column in [Table 3-1](javascript://).

In [Table 3-1](javascript://), notice how each class of addresses defines a different number of octets in its numerical range. For class A, the first octet must fall within a certain range, while the last three octets are undefined. For class B, the first two octets must fall within a certain range, and so on. Each of these three classes—A, B, and C—dedicate different numbers of octets to the network and host portions. [Figure 3-8](javascript://) shows where this dividing line falls for each of these three classes. As you can see, a value of 255 in the corresponding octet of the subnet mask indicates that octet of the IP address identifies the network. A value of 0 in the corresponding octet of the subnet mask indicates that octet of the IP address identifies the host. For example, a class A subnet mask (255.0.0.0) contains a value of 255 in the first octet only. Simultaneously, a class A IP address uses information in its first octet to identify the network.

**Figure 3-8**

The network portion and host portion for each class of IP addresses



Class A, B, and C licensed IP addresses are available for use on the Internet and are, therefore, called [**public IP addresses**](javascript://). However, even with nearly 4.3 billion IP addresses, the number of devices communicating on the Internet far exceeds this number. To conserve its public IP addresses, a company can instead use [**private IP addresses**](javascript://) for devices on its private networks—that is, devices that do not directly connect to the Internet but instead communicate through a representative device such as a router. These addresses were set aside for private use by IANA’s [**RFC1918 (Request for Comment 1918)**](javascript://) document, released in 1996. IANA allocated the following IP addresses for private networks:

* 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

Under these guidelines, a router and a web server might have a public IP address, but laptops, desktops, smartphones, and IoT (Internet of Things) devices might all have private IP addresses. The private devices communicate through the router or similar device as their public representative connected to the Internet. You’ll learn more about how this works shortly.

Private IP addresses are not routable on the Internet. In other words, routers will not forward messages addressed to an IP address within one of these private ranges. In addition, the IP addresses listed in [Table 3-2](javascript://) are reserved for special use by TCP/IP and should not be assigned to a public or private device on a network.

**Table 3-2**

### Reserved IP addresses

| **IP address(es)** | **Function** |
| --- | --- |
| 255.255.255.255 | Used for broadcast messages by TCP/IP background processes. A [**broadcast**](javascript://) message is read by every node on the network. Recall that a LAN is defined as a group of computers and other devices that can directly address each other without going through a router. Technically, a LAN, which consists of all the nodes a broadcast reaches, can be referred to as a [**broadcast domain**](javascript://). Routers don’t forward broadcast messages, thus, creating a boundary for a LAN. |
| 0.0.0.0 | Currently unassigned |
| 127.0.0.1 through 127.255.255.254 | Used for research or can indicate your own computer, in which case it is called the [**loopback address**](javascript://). Later in this module, you will learn to use the loopback address to verify that TCP/IP is configured correctly on a computer when it can talk to and hear itself on the loopback interface. |
| 169.254.0.1 through 169.254.255.254 | Used to create an [**APIPA (Automatic Private IP Addressing)**](javascript://) address when a computer configured for DHCP first connects to the network and is unable to lease an IPv4 address from the DHCP server. Notice that nearly any IP address starting with 169.254. is identified as an APIPA address. |

### Exam Tip

APIPA-related questions are known to appear often on the CompTIA Network+ exam.

In contrast to classful addressing, [**classless addressing**](javascript://) allows the dividing line between network and host portions to fall anywhere along the string of binary bits in an IP address. Shifting this dividing line allows for segmenting networks within networks in a process called [**subnetting**](javascript://).

With classless addressing, you can’t just look at an IP address’s numerical range and know how many octets identify the network and how many octets identify the host. Instead, you rely on a variety of subnet mask values to communicate any number of bits used for the network or host portions. Another option is to use [**CIDR (Classless Interdomain Routing) notation**](javascript://), devised by the IETF in 1993. (Note that CIDR is pronounced cider.) This shorthand method for identifying network and host bits in an IP address is also known as slash notation.

CIDR notation takes the network ID or a host’s IP address and follows it with a forward slash (/), which is then followed by the number of bits that are used for the network ID. For example, a private IP address could be written as 192.168.89.127/24, where 24 represents the number of bits in the network ID. In CIDR terminology, the forward slash with the number of bits used for the network ID—for example, /24—is known as a CIDR block.You’ll learn how to calculate subnets and their various subnet masks in a later module.

### DHCP (Dynamic Host Configuration Protocol)

Recall that static IP addresses are manually assigned by the network administrator, whereas dynamic IP addresses are automatically assigned by a DHCP server each time a computer connects to the network. Because it can become unmanageable to keep up with static IP address assignments, most network administrators choose to use dynamic IP addressing by running a DHCP server. Let’s see what decisions must be made when configuring DHCP on a network.

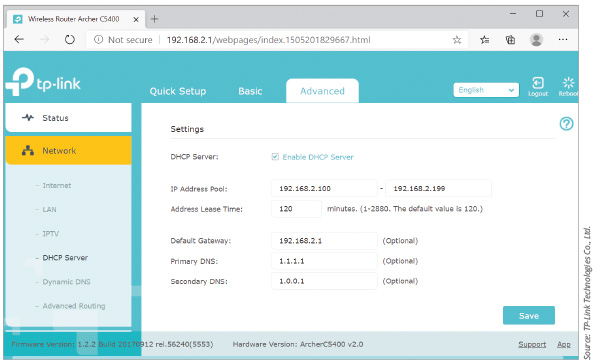
**Applying Concepts 3-3**

### Configure a DHCP Server

While each type of DHCP server software is configured differently, they offer many options in common. Generally, you define a range of IP addresses, called a [**DHCP scope**](javascript://) or DHCP pool, to be assigned to clients when they request an address. For example, [Figure 3-9](javascript://) shows a screen provided by the firmware utility for a home router, which is also a DHCP server.

**Figure 3-9**

Set a range of IP addresses on a DHCP server



Enlarge Image

Source: TP-Link Technologies Co., Ltd.

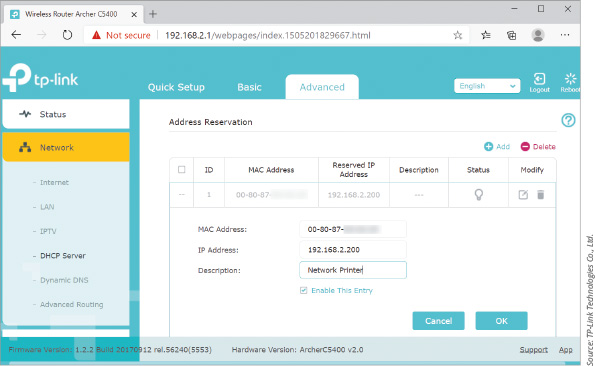
Using this screen, you set the starting IP address (192.168.2.100 in the figure) and the ending IP address (192.168.2.199 in the figure) of the DHCP scope. The scope includes the following additional information, called [**scope options**](javascript://):

* A time limit, called a [**lease time**](javascript://), which restricts the amount of time a network host can keep the IP address before it must request a renewal. When the lease time expires without renewal, the DHCP server returns the IP address to the available address pool.
* The default gateway’s IP address, which each client must know to send messages to hosts on other networks. The default gateway is typically a router or firewall.
* The primary and secondary DNS server addresses, which clients use to match computer names with IP addresses. DNS servers might be internal to the local network or external and accessed through the default gateway, like you saw in the [On the Job](javascript://) story at the beginning of this module.

When other nodes on the network frequently need to know the IP address of a particular client, you can have DHCP offer that client the same IP address every time it requests one. The DHCP server recognizes this client based on its MAC address, so this reserved IP address is called a variety of names: [**MAC reservation**](javascript://), [**IP reservation**](javascript://), or [**DHCP reservation**](javascript://). For example, a network printer should consistently use the same IP address so that computers on the network can always find it. In [Figure 3-10](javascript://), which shows the management interface for a TP-Link SOHO router, an OKI Data network printer has a reserved IP address of 192.168.2.200.

**Figure 3-10**

Reserve an IP address for one or more network clients, such as a network printer



Enlarge Image

Source: TP-Link Technologies Co., Ltd.

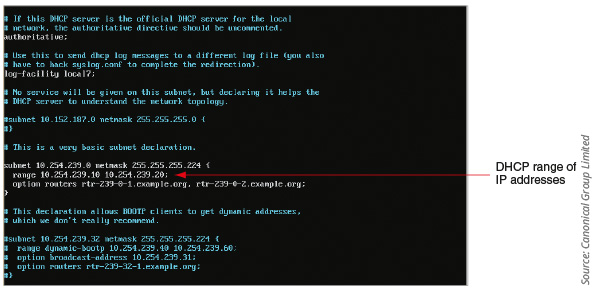
**Note 3-8**

A reserved IP address is not quite the same thing as a static IP address. A reserved IP address is offered to the client by DHCP when the client requests an IP address. A static IP address is configured on the client itself so that the client never requests an IP address from DHCP in the first place. If you have one or more clients on the network with static IP addresses, you need to configure an IP [**exclusion range**](javascript://) on the DHCP server. This excludes one or more IP addresses from the IP address pool so the server doesn’t offer those IP addresses to other clients.

In Linux systems, you configure the DHCP software by editing a configuration file in a text editor. For example, the configuration file for one Linux distro’s DHCP server is dhcpd.conf (notice the .conf file extension), which is stored in the /etc/dhcp directory. [Figure 3-11](javascript://) shows the configuration file as it appears in vim, which is a Linux text editor. A hash symbol (#) at the beginning of a line identifies the line as a comment line (a line that is not executed). The range of IP addresses that will be assigned to clients in [Figure 3-11](javascript://) is 10.254.239.10 to 10.254.239.20, which consists of 11 IP addresses.

**Figure 3-11**

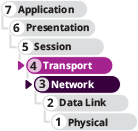
Edit a configuration file in a text editor to set an IP address range for a Linux-based DHCP server



Enlarge Image

Source: Canonical Group Limited

### Address Translation

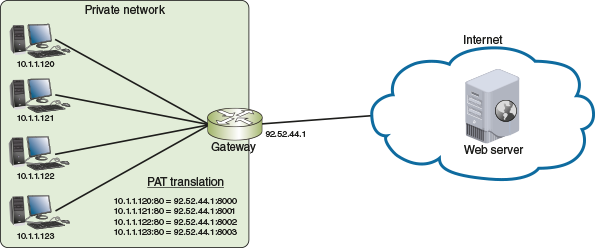


Recall that most devices on a network are assigned a private IP address, while only a few, representative devices (such as a router) receive a public IP address for communicating directly on the Internet. When private devices need access to other networks or the Internet, a public-facing gateway (such as a router or firewall) substitutes the private IP addresses used by computers on the private network with its own public IP address. This process is called [**NAT (Network Address Translation)**](javascript://). Besides requiring only a single public IP address for the entire private network, another advantage of NAT is security; the gateway hides the private network’s hosts behind this one address.

How does the gateway keep track of which local host is to receive a response from a web server on the Internet? [**PAT (Port Address Translation)**](javascript://) assigns a separate TCP port to each session between a local host and an Internet host. See [Figure 3-12](javascript://). When the Internet host responds to the local host, the gateway uses PAT to determine which local host is the intended recipient.

**Figure 3-12**

PAT (Port Address Translation)



Enlarge Image

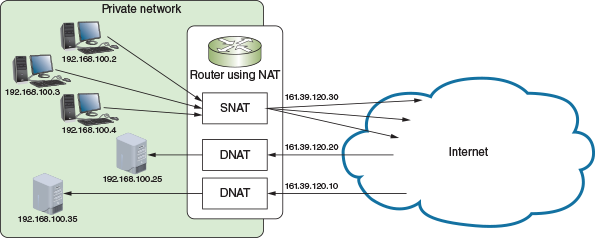
This discussion of ports edges into transport layer addressing, which you’ll read about shortly. For now, let’s get back to NAT. Two variations of NAT you need to be aware of include the following:

* **SNAT**—Using SNAT (Source Network Address Translation), the gateway assigns the same public IP address to a host each time it makes a request to access the Internet. Small home networks with only a single public IP address provided by its ISP use SNAT.
* **DNAT**—Using DNAT (Destination Network Address Translation), hosts outside the network address a computer inside the network (such as a web server or an email server) by a predefined public IP address. When a message sent to the public IP address reaches the router managing DNAT, the destination IP address is changed to the private IP address of the host inside the network. The router must maintain a translation table of public IP addresses mapped to various hosts inside the network.

[Figure 3-13](javascript://) contrasts SNAT and DNAT. SNAT changes the source IP addresses of outgoing messages and is used to reduce the number of public IP addresses needed by a network. DNAT changes the destination IP address of incoming messages and is often used by organizations that provide services to the Internet. The various servers can use private IP addresses for security and also to allow network administrators more freedom to manage these servers. For example, they can switch a web server to a backup computer while doing maintenance on the primary server by simply making a change in the router’s DNAT settings, redirecting a public IP address to the backup computer.

**Figure 3-13**

SNAT for outgoing messages, and DNAT for incoming messages



Enlarge Image

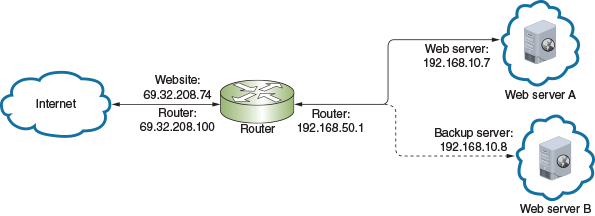
**Applying Concepts 3-4**

### Configure Address Translation Using NAT

For simple default gateways such as a home router, configuring address translation means making sure NAT is turned on. That’s about all you can do. However, for more advanced gateways, such as an industrial-grade Cisco router or Linux server, you configure the NAT software by editing NAT translation tables stored on the device. For example, suppose your network supports a web server available to the Internet, as shown in [Figure 3-14](javascript://).

**Figure 3-14**

Messages to the website are being routed to web server A



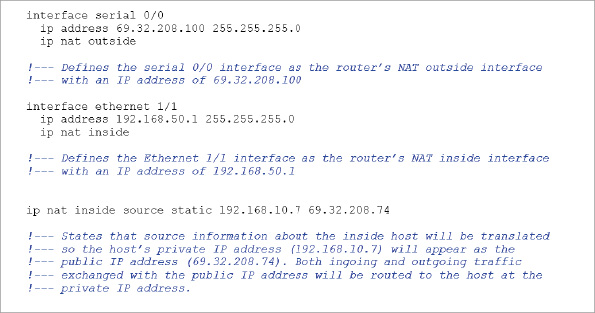
Enlarge Image

Source: Cengage Learning

On the web, the website is known by the public IP address 69.32.208.74. [Figure 3-15](javascript://) shows the sample text file required to set up the translation tables for DNAT to direct traffic to the web server at private IP address 192.168.10.7. Note that any line beginning with an exclamation mark (!) is a comment.

**Figure 3-15**

NAT translation table entry in Linux



Enlarge Image

The first section of code defines the router’s outside interface, which connects with the outside network and is called the serial interface. The second section defines the router’s inside Ethernet interface. The last line that is not a comment line says that, when clients from the Internet send a request to IP address 69.32.208.74, the request is translated to the IP address 192.168.10.7.

At the end of this module, you’ll create your own NAT translation table entry using this example as a template. To help you better understand where the IP addresses in a translation table entry come from, answer the following questions about the information in [Figures 3-14](javascript://) and [Figure 3-15](javascript://):

1. What is the router’s outside interface IP address?
2. What is the router’s inside interface IP address?
3. What is the website’s public IP address?
4. What is the private IP address of the active web server?

Go to pg.

[**help**](javascript://)