2. Which of the following address types is best suited for a DNS server?
   - **A.** DHCP-assigned address
   - **B.** APIPA address
   - **C.** Alternate configuration address
   - **D.** Manual address

Lesson 2: Understanding IPv4 Addressing

IPv4 is by far the most popular networking protocol in use. Although connecting computers to an established IPv4 network is straightforward (and often entirely automatic), to implement, configure, and troubleshoot IPv4, you need to understand basic concepts about IPv4 addressing.

After this lesson, you will be able to:

- Understand the structure of an IPv4 address, including the network ID and host ID.
- Understand the function of a subnet mask.
- Understand the function of a default gateway in IP routing.
- Understand and recognize the private IPv4 address ranges.
- Understand the concept of an address block.
- Determine the number of addresses in a given address block.
- Determine the address block size needed for a given number of addresses.
- Understand the benefits of subnetting.
- Enumerate subnets based on network needs and a given address space.
- Determine whether two addresses are configured in the same subnet.

Estimated lesson time: 180 minutes

The Structure of IPv4 Addresses

IPv4 addresses are 32 bits in length and are composed of 4 *octets* of 8 bits apiece. The usual representation of an IPv4 address is in *dotted-decimal* notation, with each of the four numbers—for example, 192.168.23.245—representing an octet separated from another by a period (dot). This common dotted-decimal notation, however, is only ever displayed for human benefit. Computers actually read IPv4 addresses in their native 32-bit binary notation such as the following:

    11000000 10101000 00010111 11110101

This point becomes important if you want to understand how IPv4 works. IPv4 is an addressing system—a system to help *find* devices—and not merely an identification system. Every IPv4 address on a network must be unique, but an address cannot be assigned randomly to a networked device because that would provide no way of finding the device. The way that IPv4 achieves both uniqueness and "findability" is by dividing addresses into two parts: the network ID and the host ID.

Network ID and Host ID

The first part of an IPv4 address is the *network ID*. The job of the network ID is to identify a particular network within a larger IPv4 internetwork (such as the Internet). The last part of an IPv4 address is the host ID. The *host ID* identifies an IPv4 host (a computer, router, or other IPv4 device) within the network defined by the network ID.

> ***NOTE*** **NETWORK ID + HOST ID = 32 BITS**
>
> **If n = the number of bits in the network ID and h = the number of bits in the host ID, n + h is equal to 32.**

Figure 1-30 shows a sample view of an IPv4 address (131.107.16.200) as it is divided into network ID and host ID sections. The letters w, x, y, and z are often used to designate the four octets within an IPv4 address. In this example, the network ID portion (131.107) is indicated by octets w and x. The host ID portion (16.200) is indicated by octets y and z.

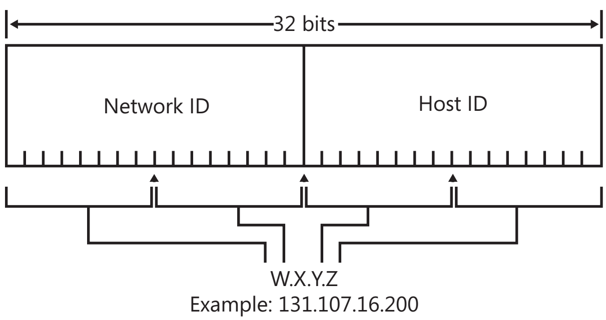

FIGURE 1-30 Network and host IDs

IPV4 ADDRESSES AND ZIP+4 COMPARED

This system of dividing the IPv4 address into a network ID and a host ID is reminiscent of the "ZIP+4" system used by most post offices in the United States Postal System. This system is used to route and deliver mail to individual post office boxes across the country.

> ***NOTE*** **ZIP+4**
>
> **For the purposes of our analogy, assume that the +4 digits only ever represent individual post office boxes.**

Taken together, the 5-digit ZIP code (also known as a postal code) and the 4-digit box number represent a unique 9-digit ZIP+4 address similar in structure and function to the 32-bit IPv4 address. The first part of the ZIP+4 address—the 5-digit zip code—represents a findable area, not a unique address. The second part represents a specific 4-digit mailbox within the 5-digit ZIP code area, a mailbox to which the post office represented by the ZIP code has the responsibility to deliver mail.

However, ZIP+4 addresses are much simpler than IPv4 addresses in one respect. When you look at a ZIP+4 address, you know for certain which part of the address represents the post office (the ZIP code) and which part represents the individual mailbox (the +4). The dividing line between them never changes. The first five digits and the last four digits always have the same function.

The tricky thing about IPv4 addresses is that the size of the network ID and the size of the host ID vary. Just by looking at an IPv4 address such as 192.168.23.245, you cannot determine which of the 32 bits are used for the network ID and which are used for the host ID. To do this, you need an additional piece of information. That piece of information is the subnet mask.

Subnet Masks

The subnet mask is used to determine which part of a 32-bit IPv4 address should be considered its network ID. For example, when you write 192.168.23.245/24, the /24 represents the subnet mask and indicates that the first 24 of the 32 bits in that IPv4 address should be considered its network ID. For the IPv4 address 131.107.16.200 shown in Figure 1-29 earlier, the first 16 bits according to the picture are used for the network ID. Therefore, the appropriate subnet mask to be used by a host assigned that address is /16.

The two subnet masks just mentioned—/16 and /24—are relatively easy to interpret. Because their values are divisible by 8, these subnet masks indicate that the network ID is composed of, respectively, the first two complete octets and the first three complete octets of an IPv4 address. In other words, the network ID of a host assigned the address 131.107.16.200 /16 is 131.107, and the host's network address is therefore 131.107.0.0. The network ID of a host assigned the address 192.168.23.245/24 is 192.168.23, and host's network address is therefore 192.168.23.0. However, subnet masks are not always divisible by 8 and are not always so easy to interpret, as we shall see.

REAL WORLD

J.C. Mackin

This lesson describes how to perform IP addressing calculations by hand. These techniques help you develop an understanding of IP addressing, but otherwise, they're mostly useful in exam situations. In the real world, you can use software tools to perform all the calculations related to addressing that are described in this lesson, including finding subnet masks, determining block sizes, and enumerating subnet ranges. Search for terms such as "subnet calculator" or "VLSM calculator," and you will find many tools that make these tasks easy.

SUBNET MASK NOTATIONS

So far, the discussion has focused on subnet masks in slash notation—also known as Classless Inter Domain Routing (CIDR) notation or network prefix notation. Slash notation is a common way of referring to subnet masks both on the 70-642 exam and in the real world. However, subnet masks are represented just as commonly in 32-bit dotted-decimal notation.

In dotted-decimal notation, the subnet mask takes the form of a 32-bit IPv4 address. For example, the subnet mask /16 is represented in dotted-decimal notation as 255.255.0.0, and the subnet mask /24 is represented in dotted-decimal notation as 255.255.255.0.

To understand the connection between a subnet mask expressed by its slash notation and its dotted-decimal equivalent, you first have to translate the slash notation to binary notation. To begin, take the value after the slash in slash notation—for example, the 16 in /16—and represent it as an equivalent number of ones in binary notation, with a space after each 8 bits, or octet.

11111111 11111111

Then, to complete the 32-bit subnet mask in binary notation, add a string of 0s until the values of all 32 bits are represented (again with a space after each 8 bits):

11111111 11111111 00000000 00000000

Finally, convert this binary notation into dotted-decimal notation. Because 11111111 is the binary equivalent of the decimal 255, and 00000000 is the binary equivalent of the decimal 0, you can represent each octet as either 255 or 0. For this reason, /16 is equivalent to 255.255.0.0.

IMPORTANT **WHAT HAPPENED TO ADDRESS CLASSES?**

You might occasionally hear that a /8 address is called *Class A*, a /16 address is called *Class B*, and a /24 address is called *Class C*. These terms refer to an older system of IPv4 routing that is no longer used, even though its vocabulary is sometimes used informally. The 70-642 exam does not use these terms because they are technically defunct.

Converting Between Binary and Decimal Notations

It's not often that you need to convert between base-two and base-ten notations, and if you do, you could use a scientific calculator. However, when you don't have access to a calculator, it's good to know how to perform these conversions manually. It will certainly also help you understand the logic of IP addressing.

The key to understanding binary notation is to understand the value of each bit place. As with the base ten system, in which each place holds different values such as ones, tens, hundreds, and so on, a base two system holds potential values in each bit place, increasing from right to left.

Table 1-1 shows the scientific and decimal notation associated with each bit place within a binary octet. Notice that, as you move from right to left and begin with the eighth bit's potential value of 1, each successive bit represents double the potential value of the previous bit, with a maximum value of 128 for the leftmost bit. Knowing this pattern allows you to recall easily the potential value of each bit place.

TABLE 1-1 Potential Values in a Binary Octet

BIT PLACE	1ST BIT	2ND BIT	3RD BIT	4TH BIT	5TH BIT	6TH BIT	7TH BIT	8TH BIT
SCIENTIFIC NOTATION	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
DECIMAL NOTATION	128	64	32	16	8	4	2	1

Note that these numbers represent only the values that are held when the bit places contain a 1. When an octet contains a 0 in any bit place, the value of the bit is null. For example, if the first (leftmost) bit place is filled with a bit value of 1, the equivalent decimal value is 128. Where the bit value is 0, the equivalent decimal value is 0 as well. If all the bit places in an octet are filled with ones (1), the equivalent decimal value is 255. If all the bit places are filled with zeroes (0), the equivalent decimal value is 0.

Binary-to-Decimal Conversion Example

The following binary string represents an octet that could be used in an IPv4 address:

10000011

To understand the decimal equivalent of this binary octet, draw a simple conversion table, such as the one that follows, in which to enter the bit values of the octet.

128	64	32	16	8	4	2	1
1	0	0	0	0	0	1	1

By using this table as a reference, you can perform simple addition of each bit place's decimal equivalent value to find the decimal sum for this octet string, as follows:

128 + 2 + 1 = 131

Because the sum is 131, the octet with the binary value 10000011 is expressed as 131 in decimal form.

Decimal-to-Binary Conversion Example

You convert an octet from decimal to binary form by drawing the conversion chart and then adding a 1 in the octet's bit places from left to right until the desired target decimal value is achieved. If, by adding a 1, your total would exceed the target decimal value, simply note a 0 in that bit place instead and move to the next bit place. There is always exactly one combination of 1s and 0s that will yield the target value.

For example, suppose you want to convert the octet value 209 into binary form. First, draw the conversion table on scratch paper, as shown here:

128	64	32	16	8	4	2	1

Next, consider the potential value of the first (leftmost) bit place. Is 128 less than 209? Because it is, you should write a 1 beneath the 128 on your scratch paper and then write a 128 off to the side to keep tally of the running subtotal.

128	64	32	16	8	4	2	1	SUBTOTAL
1								128

Move to the next potential value. Is 128 + 64 less than 209? The sum of these values is only 192, so again, you should write a 1 beneath the 64 and then a 64 to your running subtotal.

128	64	32	16	8	4	2	1	SUBTOTAL
1	1							128 +64 =192

The next potential value is 32, but if you were to add a 1 here, you would achieve a subtotal of 224. This exceeds the target total of 209, so you must place a zero in the third bit place of the octet and not add anything to your running subtotal.

128	64	32	16	8	4	2	1	SUBTOTAL
1	1	0						128 +64 =192

Next, the fourth bit potential value is 16; adding this value to 192 results in a subtotal of 208. Is 208 less than 209? Because it is, you should add a 1 beneath the 16 and a 16 to your running subtotal.

128	64	32	16	8	4	2	1	SUBTOTAL
1	1	0	1					128 64 +16 =208

Because you need to add a value of only 1 to achieve the target value of 209, placing a 1 in the eighth bit place will complete the translation of the octet.

128	64	32	16	8	4	2	1	SUBTOTAL
1	1	0	1	0	0	0	1	128 64 16 +1 =209

The octet with the decimal value 209 is therefore written in binary notation as 11010001.

SUBNET MASK MIDRANGE VALUES

The subnet masks we have been looking at in dotted-decimal notation have octets whose values are represented as either 255 or 0. This limits our discussion to only three possible subnet masks: /8 (255.0.0.0), /16 (255.255.0.0), and /24 (255.255.255.0). In fact, these are the most common subnet masks used for addresses on the Internet (especially /24 or 255.255.255.0).

However, both on the 70-642 exam and in the real world, you will also encounter subnet masks such as /25 or /22 which, when expressed in dotted-decimal notation, include a mid-range value octet such as 128 or 252. This situation arises whenever the length of a network ID (expressed in bits) is not divisible by 8.

Figure 1-31 shows the binary representation of the IPv4 address 192.168.14.222 with a typical subnet mask of /24 or 255.255.255.0. For this address, the network ID is represented by the first 24 bits (first three octets), and the host ID is represented by the last 8 bits (the last octet).

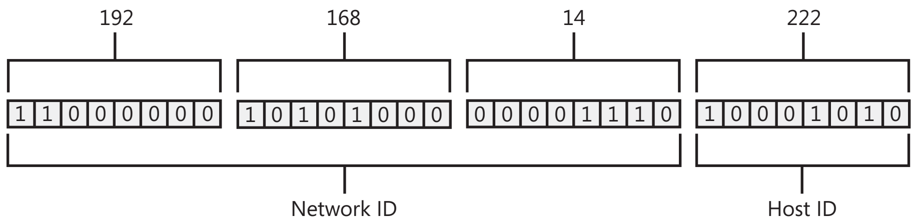

FIGURE 1-31 An IPv4 address with a /24 subnet mask

Now, consider the same IPv4 address with a 26-bit subnet mask, as shown in Figure 1-32. In this example, the network ID uses the first two bits from the last octet. The last octet is therefore dedicated partially to the network ID and dedicated partially to the host ID. In binary, the network ID is simply a 26-bit number, whereas the host ID is a 6-bit number.

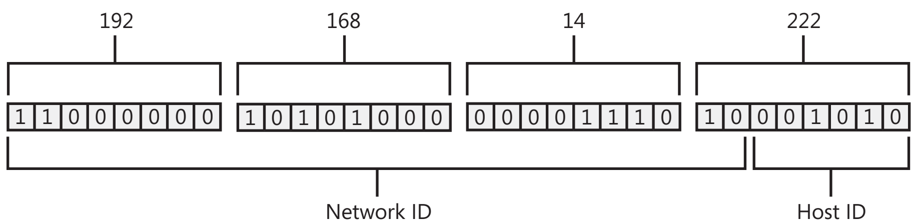

FIGURE 1-32 The same IPv4 address with a /26 subnet mask

Table 1-2 compares the slash, binary, and dotted-decimal notations for all subnet masks from /8 to /30. These are the only subnet masks you are ever likely to see. However, the subnet masks you will encounter most frequently (both on the 70-642 exam and in the real world) are in the /20 to /30 range.

IMPORTANT **STUDY THIS TABLE**

This table presents information that most network administrators are expected to understand. Be sure to spend as much time as necessary browsing this table until you are comfortable with subnet mask values and how the three notations relate to one another.

TABLE 1-2 Subnet Mask Notations Compared

SLASH NOTATION	BINARY NOTATION	DOTTED DECIMAL NOTATION
/8	11111111 00000000 00000000 00000000	255.0.0.0
/9	11111111 10000000 00000000 00000000	255.128.0.0
/10	11111111 11000000 00000000 00000000	255.192.0.0
/11	11111111 11100000 00000000 00000000	255.224.0.0
/12	11111111 11110000 00000000 00000000	255.240.0.0
/13	11111111 11111000 00000000 00000000	255.248.0.0
/14	11111111 11111100 00000000 00000000	255.252.0.0
/15	11111111 11111110 00000000 00000000	255.254.0.0
/16	11111111 11111111 00000000 00000000	255.255.0.0
/17	11111111 11111111 10000000 00000000	255.255.128.0
/18	11111111 11111111 11000000 00000000	255.255.192.0
/19	11111111 11111111 11100000 00000000	255.255.224.0
/20	11111111 11111111 11110000 00000000	255.255.240.0
/21	11111111 11111111 11111000 00000000	255.255.248.0
/22	11111111 11111111 11111100 00000000	255.255.252.0
/23	11111111 11111111 11111110 00000000	255.255.254.0
/24	11111111 11111111 11111111 00000000	255.255.255.0
/25	11111111 11111111 11111111 10000000	255.255.255.128
/26	11111111 11111111 11111111 11000000	255.255.255.192
/27	11111111 11111111 11111111 11100000	255.255.255.224
/28	11111111 11111111 11111111 11110000	255.255.255.240
/29	11111111 11111111 11111111 11111000	255.255.255.248
/30	11111111 11111111 11111111 11111100	255.255.255.252

SUBNET MASK OCTET VALUES

Learning the sequence of nine possible octet values for a subnet mask will help you when you need to determine the size of an existing or planned network. To a large degree, in fact, the ability to perform such calculations in one's head is expected of a good network administrator.

Use Table 1-3 to help you memorize the values. Begin by covering the top row of the table. After you can recite without hesitation the decimal value associated with any number of 1-bits or binary values chosen at random from the bottom two rows, proceed to cover up the bottom two rows. When you can recite without hesitation the number of 1-bits associated with any decimal value chosen at random from the top row, proceed to memorize the sequence of decimal values from left to right and right to left.

TABLE 1-3 Subnet Mask Octet Values

DECIMAL VALUE	0	128	192	224	240	248	252	254	255
# OF 1-BITS	0	1	2	3	4	5	6	7	8
BINARY VALUE	00000000	100000000	11000000	11100000	11110000	11111000	11111100	1111110	11111111

Converting Subnet Masks Between Slash Notation and Dotted-Decimal Notation

You should normally use either memorization or a reference chart to convert between subnet mask notations when no software tools are available. However, you might be interested to know how to perform the conversion without relying on memorized values, a reference chart, or binary math. This method requires some practice and should not be your first choice for converting subnet mask notations on the 70-642 exam.

To convert a subnet mask expressed as */n* to a subnet mask in dotted decimal notation, do the following:

1. **Write out an octet of 255 for each time that 8 goes into *n*.**

   For example, for a /19 network, 8 goes into 19 twice. You should therefore write out "255.255.".

2. **Subtract the remainder of the previous operation from 8, raise two to the power of this difference, and then subtract this new value from 256. This calculation can be expressed mathematically as $256 - 2^{[8 - (n \bmod 8)]}$, where *n* is your */n* value and *mod* is the operation that yields a remainder after dividing the preceding value by the following value. Place the result of this operation into the next octet. If any other octets remain, set them to 0.**

   Following with the last example, $256 - 2^{[8 - (19 \bmod 8)]} = 256 - 2^{[8 - 3]} = 256 - 2^5 = 256 - 32 = 224$. You should therefore write out "224" for the third octet and "0" as the fourth octet. A /19 network therefore has a subnet mask of 255.255.224.0.

If you need to convert a dotted-decimal subnet mask to a subnet mask expressed as */n*, do the following:

1. Multiply 8 times the number of octets in the subnet mask that are set to 255. If the other octets are all set to 0, put a slash in front of this value, and you are finished. Otherwise, keep this value as a subtotal, and proceed to the next step.

   For example, the octet 255.255.255.192 has three subnets of 255. 8 * 3 = 24. Because the fourth subnet has a midrange value, you will keep 24 as a subtotal.

2. For an octet with a value between 0 and 255, subtract the value of the octet from 256, convert the resulting value to an expression of 2 to an exponent, and finally subtract this exponent from 8. This calculation can be expressed mathematically as $8 - \log_2(256 - y)$, where y is the value of the octet and $\log_2$ is the binary logarithm, or the exponent required for 2 that would yield a value equal to the number that follows. Add the result of this operation to the subtotal from step 1, and put a slash in front of this value to obtain final the */n* notation.

   Following with the previous example, 192 is the value of the midrange octet. $8 - \log_2(256 - 192) = 8 - \log_2(64) = 8 - 6 = 2$. Add 24 from the previous step to 2, and you have a final result of /26. Therefore, a subnet mask of 255.255.255.192 is equivalent to /26.

Understanding Routing and Default Gateways

The determination of the network ID by using the subnet mask is a vital step in IPv4 communication. This network ID essentially tells a computer how to handle an IPv4 packet. When a computer on a network needs to send an IPv4 packet to a remote address, the computer first compares its own network ID to that of the destination network ID specified in the IPv4 packet. (To determine these network IDs, the computer always uses its locally configured subnet mask.) If the two network IDs match, the message is determined to be local and is broadcast to the local subnet. If the two network IDs do not match, the computer sends the packet to a local router at the address known as the *default gateway*. The router at this default gateway address then forwards the IPv4 datagram in a manner determined by its routing tables.

Figure 1-33 illustrates this process of IP routing. In the figure, a computer whose address is 192.168.100.5/24 needs to send an IP packet destined for the address 192.168.1.10. Because the network IDs of the two addresses do not match, the computer sends the packet to the router specified by the default gateway address. This router consults its routing tables and sends the packet to the router connected to the 192.168.1.0 network. When the router connected to this network receives the packet, the router broadcasts the packet over the local subnet. The destination computer at the address 192.168.1.10 responds to the broadcast and receives the packet for internal processing.

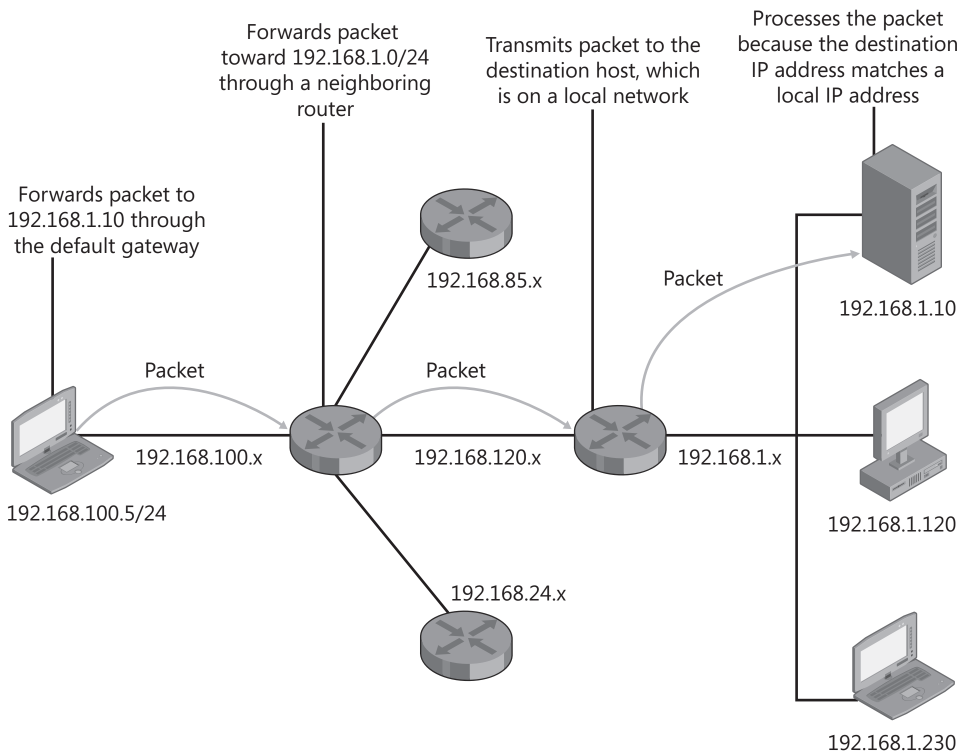

FIGURE 1-33 Routing an IP packet over an internetwork

Remember also these essential points about routing and default gateways:

- A default gateway must share the same network ID and be located within the same broadcast domain as the hosts it is serving.
- If a host has no default gateway setting configured, that host will be unable to connect to the Internet or to any computers beyond broadcast range. For example, a private internal server that occasionally needs to download content from the Internet needs to have a default gateway configured.
- Leaving the default gateway setting unconfigured on a host prevents access to that host from all points beyond the local subnet. In certain situations, therefore, you might in fact want to leave the default gateway setting unconfigured for security reasons.

Understanding IPv4 Address Ranges

The vast majority of IPv4 addresses are *unicast*, which means they are assigned only to one host at a time. These IPv4 unicast addresses can be divided into three categories: APIPA (introduced in Lesson 1, "Introducing Windows Networking"), Public, and Private ranges. Whereas APIPA addresses are used only for temporary addresses or isolated computers, public and private ranges are divided into blocks that can be assigned to entire networks. These public and private ranges, along with the concept of address blocks in general, are described in the following sections.

Using Public IPv4 Addresses

Every IPv4 address on the public Internet is unique. To allow networks to obtain unique addresses for the Internet, the Internet Assigned Numbers Authority (IANA) divides up the non-reserved portion of the IPv4 address space and delegates responsibility for address allocation to a number of regional registries throughout the world. These registries include Asia-Pacific Network Information Center (APNIC), American Registry for Internet Numbers (ARIN), and Réseaux IP Européens Network Coordination Centre (RIPE NCC). The regional registries then allocate *blocks* of addresses to a small number of large Internet service providers (ISPs) that then assign smaller blocks to customers and smaller ISPs, who then provide public addresses to organizations paying for Internet service. Within these organizations, public addresses are typically reserved for Internet-facing routers and servers.

Using Private IPv4 Addresses

The IANA has also reserved a certain number of IPv4 addresses that are never used on the global Internet. These private IPv4 addresses are used for hosts that require IPv4 connectivity but that do not need to be seen on the public network. For example, you should not obtain public IPv4 addresses for use inside a private network, such as a home network or an internal business network. Private networks should use the address ranges shown in Table 1-4 to provide addresses for hosts on the network.

TABLE 1-4 Private Address Ranges

STARTING ADDRESS	ENDING ADDRESS
10.0.0.0	10.255.255.254
172.16.0.0	172.31.255.254
192.168.0.0	192.168.255.254

Hosts addressed with a private IPv4 address connect to the Internet through a server or router performing Network Address Translation (NAT). The router performing NAT can be a computer running Windows Server 2008 R2 or a dedicated routing device.

Understanding Address Blocks and Subnets

Most organizations use a combination of public and private addresses. Public addresses are usually reserved for Internet-facing routers and servers such as mail and web servers that require external access. The internal network composed of internal routers, servers, and clients should use private address ranges. What is most certain is that every organization that wants to communicate on the Internet must have at least one public address. This one public address can then be leveraged by many clients through NAT.

You obtain public IPv4 addresses from your ISP for all routers and servers that are directly facing the Internet. Although small organizations might be able to get by with only a single public IPv4 address, many organizations need far more than that. Organizations needing more than one public address usually purchase those addresses from their ISP as a block.

An *address block* is the complete group of contiguous IP addresses that shares any single network ID. For example, an organization may purchase from an ISP a /24 address block with network ID 206.73.118. The range of addresses associated with this address block is 206.73.118.0–206.73.118.255.

> ***NOTE*** **WHAT IS ADDRESS SPACE?**
>
> **The range of addresses associated with a given address block is also known as the block's *address space*.**

It is essential to understand that the addresses within an address block constitute a single network, and unless the network is subnetted—a possibility we will consider later in this lesson—that address block will serve a *single broadcast domain* with a single router, or way out of the network. The *default gateway* is the address assigned to that router within the same broadcast domain.

Stated another way, an address block by default is designed to serve a single subnet. A *subnet* is a group of hosts within a single broadcast domain that share the same network ID and the same default gateway address.

Figure 1-34 displays a network served by the address block 206.73.118.0/24.

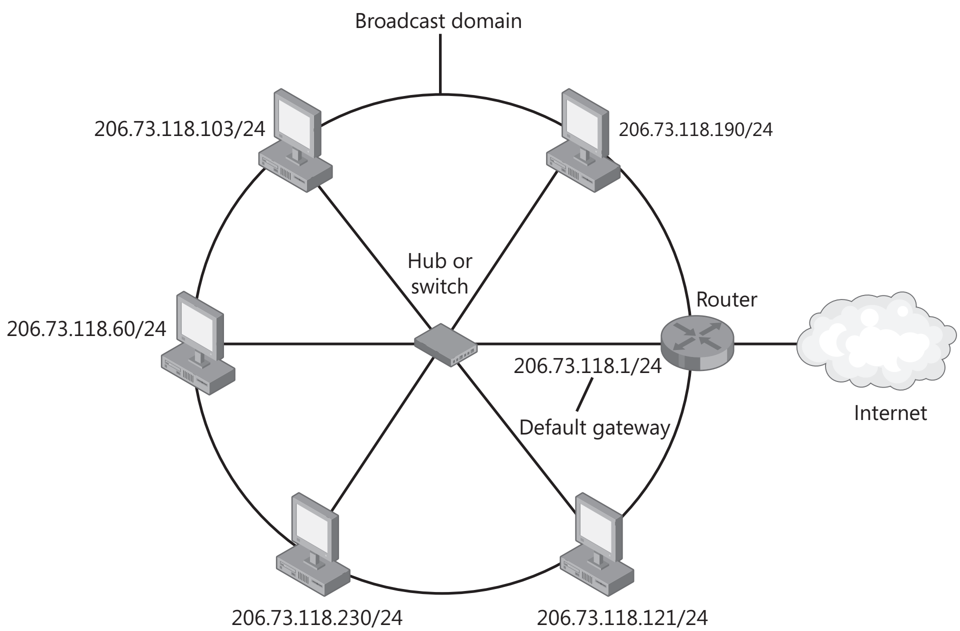

FIGURE 1-34 A single-subnet network

NOTE **WHAT'S THE DIFFERENCE BETWEEN A NETWORK AND A SUBNET?**

The terms *network* and *subnet* are often used interchangeably. The difference between them is that a subnet always refers to a single broadcast domain that is undivided. The term network, however, can refer to a single subnet or a group of interconnected subnets.

Creating an IPv4 Address Block Reference Chart

You should create a reference chart whenever you have to perform calculations related to IPv4 address blocks without the help of software tools. The chart will enable you to quickly solve problems related to addressing and is especially useful—even essential—in exam situations.

EXAM TIP

For the Microsoft exam, it's a good idea to draw this reference chart on the scratch pad provided by the testing site even before you click Start to begin the test. Doing so will give you the time you need to fill in the correct values without having to worry about the exam clock.

Rotate your scratch pad so that the longest side is horizontal, and then begin your reference chart by writing a 1 on the right side. Working from right to left, keep doubling and writing the values until you reach 256. Leave plenty of room on the left side of the scratch pad because you will be filling in more values later.

256	128	64	32	16	8	4	2	1

Next, draw a horizontal line above the row of numbers you have written and a vertical line to the left of 256. Label the row "Block size" to the far left, as shown here:

Block Size	256	128	64	32	16	8	4	2	1

Then, write the values /24 through /32 from left to right above the block size numbers you have just written. Your chart should now look like this:

	/24	/25	/26	/27	/28	/29	/30	/31	/32
Block Size	256	128	64	32	16	8	4	2	1

Each column in the chart represents a network of a distinct size. The block size value informs you of the number of addresses available within a given size network. For example, according to the chart, a /28 network provides 16 addresses.

Now, draw a horizontal line beneath the block size numbers. Subtract these same values from 256 and write the result in a row below the new horizontal line. (The results of this operation are the same subnet mask octet values shown in Table 1-3. If you have memorized these values, this step will be much faster.) Label the new row "Mask."

	/24	/25	/26	/27	/28	/29	/30	/31	/32
Block Size	256 -256	256 -128	256 -64	256 -32	256 -16	256 -8	256 -4	256 -2	256 -1
Mask	0	128	192	224	240	248	252	254	255

The Mask row value informs you of the last octet in the subnet mask for each network. The first three octets are 255.255.255 for all networks to the right of the vertical line. To determine the subnet mask of a /29 network, for example, begin with 255.255.255 and then add the Mask value for /29 shown on the chart (248) for the last octet. Therefore, the full subnet mask for a /29 network is 255.255.255.248.

At this point, you should fill in three or four more columns to the left of the vertical line, as desired. The networks to the left of this line are distinct because they have a value less than 255 in the third octet of the subnet mask.

In this example, you can fill in values for four more networks. To begin, write /20 through /23 to the left of /24 on the top row.

	/20	/21	/22	/23	/24	/25	/26	/27	/28	/29	/30	/31	/32
Block Size					256	128	64	32	16	8	4	2	1
Mask					0	128	192	224	240	248	252	254	255

In the Block Size row, continue to fill in each space by doubling the values from right to left. However, you should also now include a second expression of each block size value as a multiple of 256, as shown. You will use the factors paired with 256 for calculations related to the third octet of addresses. For example, you will use these factors to calculate address ranges on large block sizes and to fill in the Mask values in the next step.

	/20	/21	/22	/23	/24	/25	/26	/27	/28	/29	/30	/31	/32
Block Size	4096 256 x 16	2048 256 x 8	1024 256 x 4	512 256 x 2	256	128	64	32	16	8	4	2	1
Mask					0	128	192	224	240	248	252	254	255

For each factor you just paired with 256 in the Block Size row, subtract that factor from 256. (You can imagine replacing the multiplication sign with a subtraction sign in the multiple-of-256 expressions you have just written.) In other words, perform 256 – 16, 256 – 8, 256 – 4, and 256 – 2, and put the results of these four operations in the remaining spaces in the Mask row.

The completed reference chart will look as it does in Figure 1-35.

	/20	/21	/22	/23	/24	/25	/26	/27	/28	/29	/30	/31	/32
Block Size	4096 256 x 16	2048 256 x 8	1024 256 x 4	512 256 x 2	256	128	64	32	16	8	4	2	1
Mask	240	248	252	254	0	128	192	224	240	248	252	254	255

FIGURE 1-35 The IPv4 address block reference chart

To determine the dotted-decimal subnet mask for a network to the left of the vertical line, note that the Mask value in the chart for these networks refers to the value of the third octet in the subnet mask. The first two octets in these networks is always 255.255, and the last octet is always 0. For example, a /22 network has a Mask value of 252 in the chart. Therefore, the subnet mask for a /22 network in dotted-decimal notation is 255.255.252.0.

After you have completed the chart, be sure to refer to it for all questions related to IP addressing.

EXAM TIP

For manual calculations related to IP addressing, it's very useful to memorize the powers of 2 up to 2^{12}. If you have not memorized these values, you should create a power of 2 chart during the exam on the scratch pad provided by the testing site. These values are shown in the following list. The chart is easy and quick to create because each value is double the previous value. Note also that these values are the same as the block sizes, so if you prefer, you can add these 2^x expressions above the block size values in your reference chart.

$2^1 = 2$

$2^2 = 4$

$2^3 = 8$

$2^4 = 16$

$2^5 = 32$

$2^6 = 64$

$2^7 = 128$

$2^8 = 256$

$2^9 = 512$

$2^{10} = 1024$

$2^{11} = 2048$

$2^{12} = 4096$

Determining the Block Size for a /*n* Network

If your company purchases a block of addresses from an ISP, the size of that address block is often expressed by its subnet mask in slash notation. If you have not memorized these sizes, the easiest way to determine the number of addresses in a network expressed as /*n* is to use a reference chart such as the one shown in Figure 1-35. However, if you are in a location where you cannot create a chart, you can use the following formula:

$2^{(32-n)}$ = *number of addresses*

For example, a /27 network includes $2^{(32-27)} = 2^5 = 32$ addresses.

> ***NOTE*** **KNOW YOUR POWER OF 2 VALUES!**
>
> **If you have the power of 2 memorized, you should be able to perform this and other similar calculations in your head.**

Figure 1-36 illustrates this mathematical relationship between the */n* value and the number of addresses. Remember that every IPv4 address is a 32-bit address. If the first 24 bits are fixed as the network ID, 8 bits remain to be used as addresses in that network. You determine how many possible values exist within these 8 bits by calculating 2^8. Therefore 2^8, or 256, is the number of unique addresses for a /24 network.

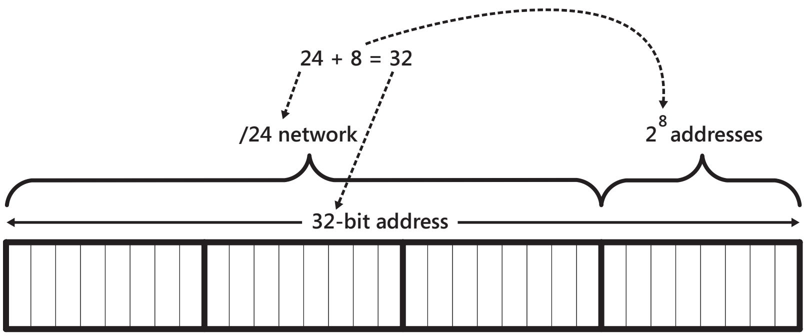

FIGURE 1-36 A /24 network leaves 8 bits for addresses.

Figures 1-37 and 1-38 illustrate the same concept for a /23 and /25 network, respectively.

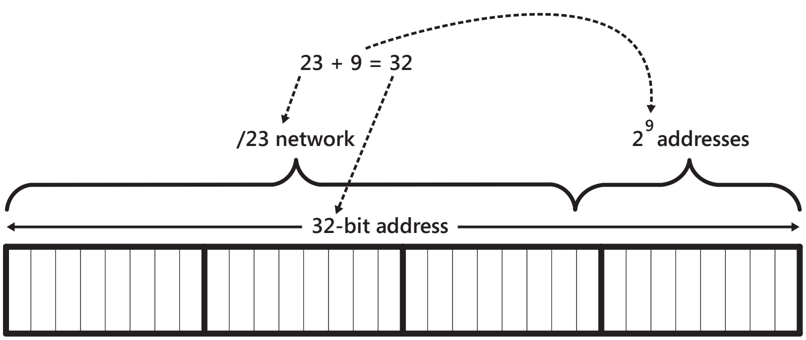

FIGURE 1-37 A /23 network allows for 29 or 512 addresses.

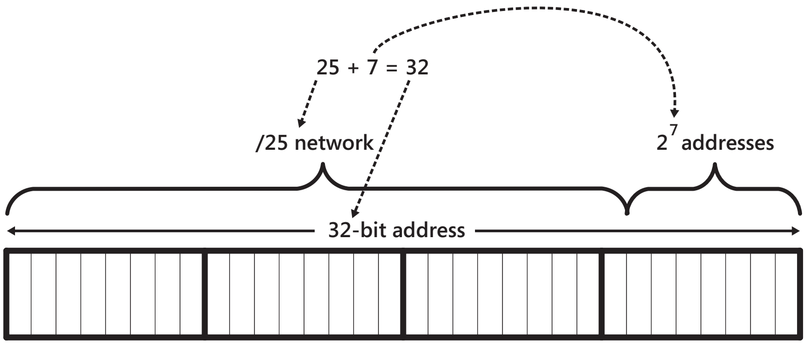

FIGURE 1-38 A /25 network allows for 27 or 128 addresses.

Determining the Block Size for a Dotted-Decimal Subnet Mask

Often you need to determine the size of an address block when the subnet mask for that network is given in a dotted-decimal format. Using memorized knowledge or a reference chart is the easiest way to solve this type of problem without software tools, but if you have not memorized the values and are in a location where you cannot create a chart, you can use the following method.

If the subnet mask value provided is 255.255.255.0 or greater, the calculation is fortunately very easy. Just use the following formula, where *z* is the value of the last octet:

256 – *z* = *number of addresses*

For example, if a network has a subnet mask of 255.255.255.240, the block size is 256 – 240 = 16 addresses. If a network has a subnet mask of 255.255.255.192, the block size is 256 – 192 = 64 addresses. If a network has a subnet mask of 255.255.255.0, the block size is 256 – 0 = 256 addresses. Remember that the block size will always be a power of 2, so if you have the powers of 2 memorized, you should be able to perform the calculation in your head.

If the subnet mask value for a network is between 255.255.0.0 and 255.255.255.0, the calculation is still fairly easy. Just use the following formula, where *y* is the value of the third octet:

(256 – *y*) * 256 = *number of addresses*

For example, if a network has a subnet mask of 255.255.252.0, the block size is (256 – 252) * 256 = 4 * 256 = 1024 addresses. If a network has a subnet mask of 255.255.240.0, the block size is (256 – 240) * 256 = 16 * 256 = 4096 addresses. Again, the block size will always be a power of 2, so if you have the powers of 2 memorized, you might still be able to perform the calculation in your head.

Network administrators rarely need to determine the address block size for a network with a subnet mask between 255.0.0.0 and 255.255.0.0, and you will not need to perform such a calculation on the 70-642 exam. However, for completeness, the formula is presented here (where *x* is the value of the second octet):

(256 – *x*) * 256 * 256 = *number of addresses*

Quick Check

- **Does an address block get bigger or smaller when its subnet mask is lengthened? By how much?**

Quick Check Answer

- **An address block gets smaller by half for each 1-bit increment in the subnet mask.**

DETERMINING HOST CAPACITY PER BLOCK

The host capacity of an address block is the number of addresses within that block that can be assigned to computers, routers, and other devices. In every address block assigned to a single broadcast domain and subnet, exactly two addresses are reserved for special use: the first, or all-(binary)-zeroes address, in the block, which is reserved as the address of the entire subnet; and the last, or all-(binary)-ones address in the block, which is reserved for the broadcast address. This means that the maximum host capacity of an address block is always two fewer than the number of addresses in that network.

For example, the network 192.168.10.0/24 has 256 addresses. The specific address 192.168.10.0 is reserved for the network address, and 192.168.10.255 is reserved for the network broadcast address. This leaves 254 addresses that can be assigned to network hosts.

Determining Block Size Requirements

If you are designing a network for a given number of computers, you often have to determine an appropriate subnet mask for that network. For example, if you are creating a new LAN that will be connected to a larger corporate network, you could request a block size expressed as */n* from the central IT department of the company.

To determine block size requirements expressed in slash notation, first add 2 to the number of computers on the network you are planning for. This resulting value can be called *a*, or the number of addresses you need in the block. Then use either memorized knowledge of address block sizes or a reference chart such as the one in Figure 1-35 to determine the smallest network, expressed as */n*, that has a block size of *a* or greater.

If you have not memorized the address block sizes and are in a location where it is impractical to create a reference chart, perform the following steps:

1. Determine the smallest value of x such that $2^x \geq a$. To express this step another way, ask yourself the following question: What is the smallest exponent for 2 that would yield a value equal to or greater than the number of addresses you need?
2. Subtract the x (exponent) value from 32 to get the n value of the subnet mask expressed as */n*.

This procedure is illustrated on the following page.

Step 1: **# of computers + 2 = *a* addresses**

Step 2: $2^{?} \geq a$

Step 3: $32 - x = /n$

For example, if you are designing a new network with 48 computers, you will need 48 + 2, or 50, addresses for the subnet. Because $2^5 = 32$ and $2^6 = 64$, the value 6 is the smallest exponent for 2 that would yield a value equal to or greater than 50. Finally, because 32 – 6 = 26, you can determine that you need a /26 block to accommodate your new network.

To determine block size requirements expressed in terms of a dotted-decimal subnet mask, begin as always by adding 2 to the number of computers on the network you are planning for. We can call this result *a*. Then use memorized values of subnet masks or a reference chart such as the one in Figure 1-35 to determine the smallest network, expressed as a dotted-decimal subnet mask, that has a block size of *a* or greater.

If you have not memorized the address block sizes and are in a location where it is impractical to create a reference chart, perform the following steps:

1. Determine the smallest value that is a power of two (such as 128, 256, 512, and so on) and that is also greater than or equal to *a*. You can refer to this value as *p*.
2. If $p \leq 256$, set the first three octets to 255. Subtract *p* from 256, and then place this new value in the fourth octet.

This procedure when $p \leq 256$ is summarized here:

Step 1: **# of computers + 2 = *a* addresses**

Step 2: $\underline{2^{?}} \geq a$

Step 3: $\begin{array}{r} 256 \\ \underline{-\,p} \end{array}$

255.255.255.

For example, if you are designing a new network with 30 computers, you need 30 + 2, or 32, addresses for the subnet. Because $2^5 = 32$, the value 32 is the smallest power of 2 that is big enough to accommodate your needs. 256 – 32 = 224, so you need a subnet mask of 255.255.255.224 to accommodate your new network.

If $p \geq 256$, set the first two octets to 255 and the fourth octet to 0. Then determine the following value and place it in the third octet: 256 – (*p* / 256).

This procedure when $p \geq 256$ is summarized as follows:

Step 1: **# of computers + 2 = *a* addresses**

Step 2: $2^{?} \geq a$

Step 3:

$$\begin{array}{r} 256 \\ -\ (p/256) \\ \hline 255.255.\_\_.0 \end{array}$$

For example, suppose you are designing a network for 2000 computers. In this case, you need 2002 addresses for the subnet. The smallest power of 2 that is equal to or greater than 2002 is 2048. 2048 / 256 = 8, and 256 – 8 = 248. Therefore, you need a network with a subnet mask of 255.255.248.0 to accommodate your needs.

EXAM TIP

Expect to see at least one question on the 70-642 exam in which you are given a specific number of computers and need to determine a subnet mask that will accommodate those computers. The answer choices might present subnet masks in either dotted-decimal or slash notation.

What Is Subnetting?

Subnetting refers to the practice of logically subdividing a network address space by extending the string of 1-bits used in the subnet mask of a network. This extension enables you to create multiple logical subnets or broadcast domains within the original network address space.

Assume that you have purchased from your ISP the address block 131.107.0.0 /16 for use within your organization. Externally, the ISP then uses the /16 (255.255.0.0) subnet mask on its routers to forward to your organization IPv4 packets that have been addressed to 131.107.*y.z*.

In a first scenario, suppose that you configure the subnet mask at its original 255.255.0.0 value on all internal hosts. In this case, all IPv4 addresses within the address space, such as 131.107.1.11 and 131.107.2.11, are logically seen by hosts to share the same network ID (131.107) and to belong to the same subnet. All hosts within this address space therefore attempt to communicate with one another by means of a broadcast. The configuration in this first scenario requires that internal to the network, only layer 1 and layer 2 devices such as hubs, switches, and wireless bridges that do not block broadcasts can be used.

In a second scenario, you decide to alter the subnet mask used within your organization to /24 or 255.255.255.0. In this case, internal hosts will read the addresses 131.107.1.11 and 131.107.2.11 as having different network IDs (131.107.1 vs. 131.107.2) and consider these addresses as belonging to different subnets.

To communicate with each other, the hosts assigned the addresses 131.107.1.11/24 and 131.107.2.11/24 send IPv4 packets to their respective default gateways, an address that must lie within the same broadcast domain. The router owning the default gateway address is then responsible for routing the IP packet toward the destination subnet. Hosts external to the organization continue to use the /16 subnet mask to communicate with hosts within the network.

Figure 1-39 and Figure 1-40 illustrate these two possible versions of the network.

Whereas the original /16 network address space in Figure 1-34 consisted of a single subnet including up to 65,534 ($2^{16} - 2$) hosts, using a /24 subnet mask everywhere internally, as shown in Figure 1-35, allows you to subdivide this original space into 256 (2^8) subnets with as many as 254 ($2^8 - 2$) hosts each.

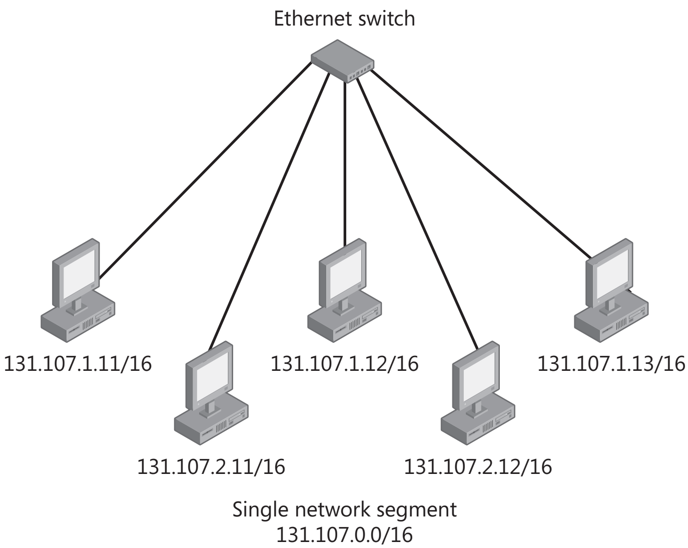

FIGURE 1-39 A /16 address space, not subnetted

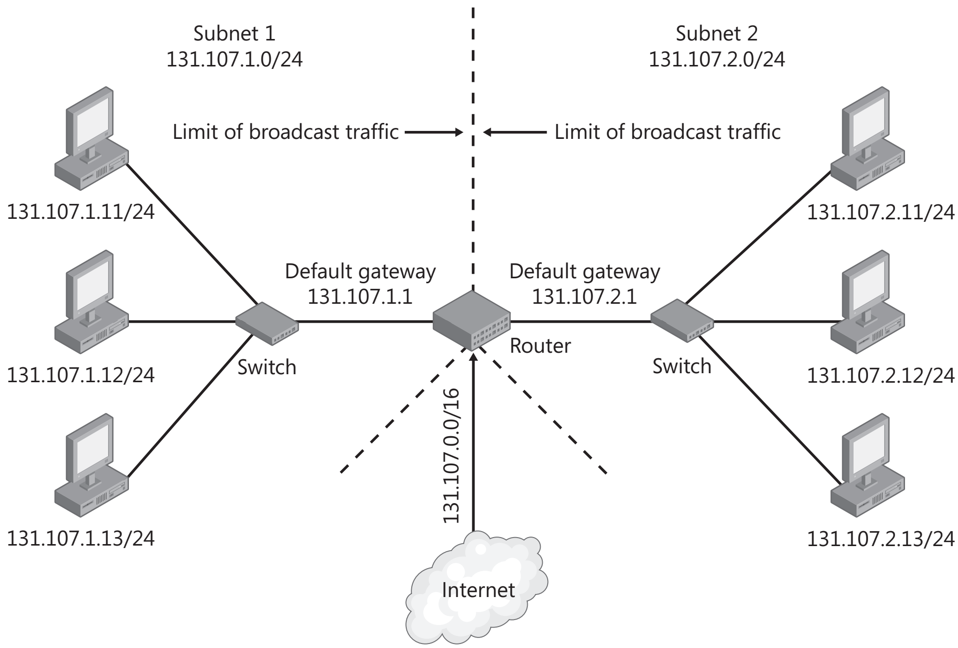

FIGURE 1-40 Subnetted /16 address space

Advantages of Subnetting

Subnetting is often used to accommodate a divided physical topology or to restrict broadcast traffic on a network.

Accommodating Physical Topology

Suppose you are designing or redesigning the address space for a campus network that has four public servers housed in each of four buildings. Each of the four buildings includes a router that connects to the rest of the campus network. If your ISP has allocated to you the /26 network 208.147.66.0, your address space includes 64 public addresses in the range from 208.147.66.0 through 208.147.66.63. Because of the physical topology, however, you cannot put all of these addresses in a single subnet. As illustrated in Figure 1-41, the routers in each building create eight separate physical segments that you have to accommodate with your IP address space design.

By extending the subnet mask in all four locations to /29, you can accommodate this physical topology by creating eight logical subnets of 8 addresses apiece.

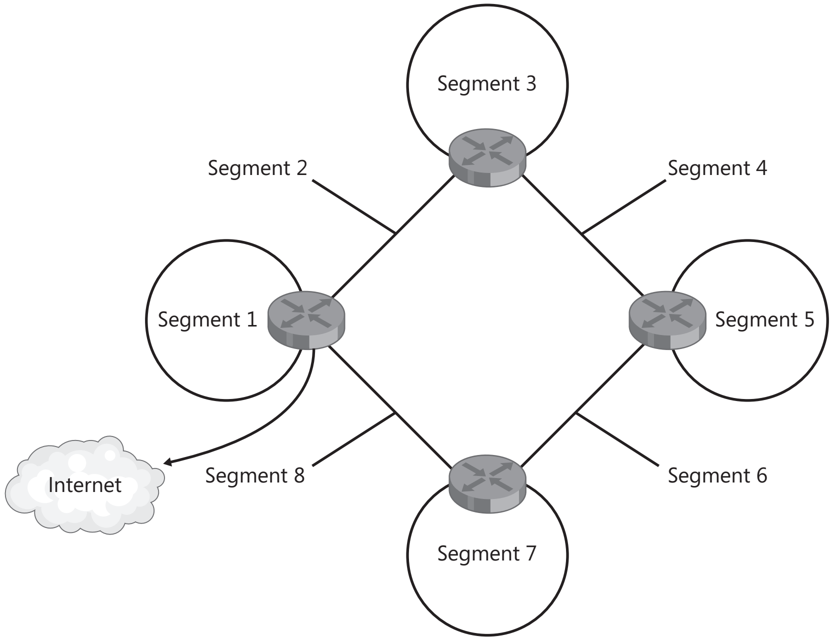

FIGURE 1-41 Each physical network segment requires its own logical subnet.

Restricting Broadcast Traffic

A broadcast is a network message sent from a single computer and propagated to all other devices on the same physical network segment. Broadcasts are resource-intensive because they use up network bandwidth and request the attention of every network adapter and processor on the LAN.

Routers block broadcasts and protect networks from becoming overburdened with unnecessary traffic. Because routers also define the logical limits of subnets, subnetting a network allows you to limit the propagation of broadcast traffic within that network.

> ***NOTE*** **VLANS ARE AN ALTERNATIVE TO SUBNETTING**
>
> **As a means to restrict broadcast traffic in large networks, virtual LAN (VLAN) switches are becoming an increasingly popular alternative to subnetting. Through VLAN software that integrates all the VLAN switches on the network, you can design broadcast domains in any manner, independent of the network's physical topology.**

The Subnet ID

Every 32-bit IPv4 address consists of a host ID and a network ID. When you obtain an address block from your ISP (or from your central network administrator in a large network), that address block contains a single network ID that cannot be changed. If you are given a /16 network, for example, the values of the first 16 bits of your address block are not configurable. It is only the remaining portion—the portion reserved for the host ID—that represents your configurable address space.

When you decide to subnet your network, you are essentially taking some of your configurable address space from the host ID and moving it to the network ID, as shown in Figure 1-42. This string of bits you use to extend your network ID internally within your organization (relative to the original address block) is known as the subnet ID.

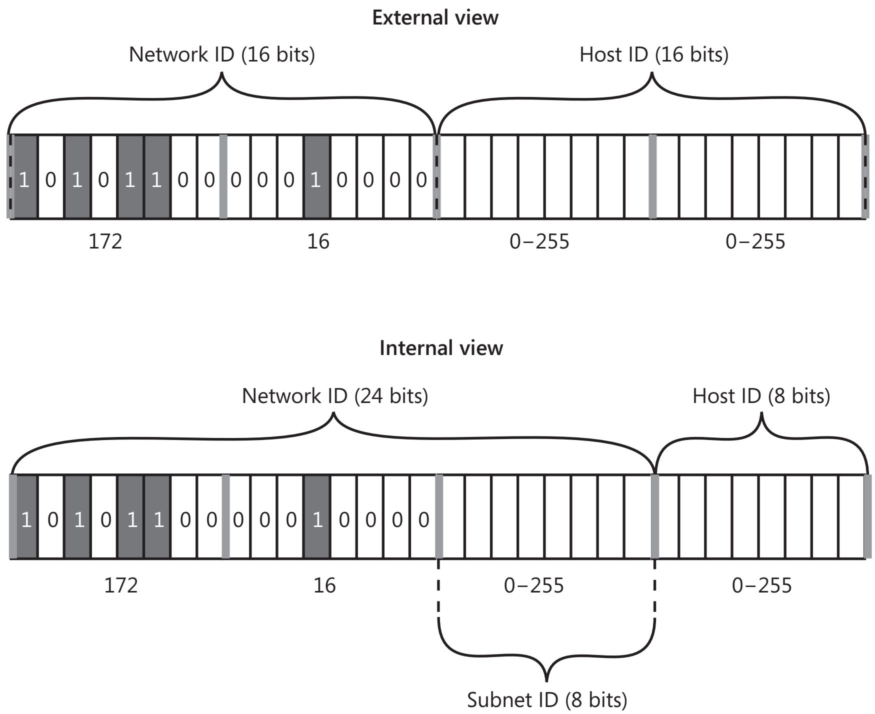

FIGURE 1-42 The Subnet ID is taken from the Host ID.

Creating Equally Sized Subnets

The easiest way to subnet a network is to use one new and extended subnet mask on all computers within your internal address space. Doing so generates a number of subnets of equal size. When you subnet your network in this way, you can determine how many logical subnets have been created by using the formula

$$2^{(n2-n1)} = \textit{number of subnets}$$

where *n2* is the length (in bits) of the new network ID used internally within the organization, and *n1* is the length of the original network ID assigned externally to refer to the entire address block. For example, if you subnet a 10.0.100.0 /24 address space by using a /27 subnet mask on all hosts in your internal network, you generate $2^{(27-24)} = 2^3 = 8$ subnets. Each of these 8 subnets includes $2^{(32-27)} = 2^5 = 32$ addresses.

Figure 1-43 and Figure 1-44 provide a visualization of this address space and how it is affected by the subnetting. The address block shown includes 256 addresses, with each number in the large square representing one of the possible values of the last octet in the 10.0.100.0/24 address space. Figure 1-43 shows the address space before subnetting. The undivided address block represents a single broadcast domain. The 0 address (10.0.100.0) is reserved for the network address, and the 255 address (10.0.100.255) is reserved for the broadcast address. The network will use a single default gateway, typically at the 1 address.

10.0.100.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

$\left.\right\}$ **One /24 subnet**
256 addresses
Network address: 0
Broadcast address: 255

FIGURE 1-43 A /24 address space before subnetting

If all the computers within the address space change their subnet mask to /27, however, the eight subnets shown in Figure 1-44 are generated.

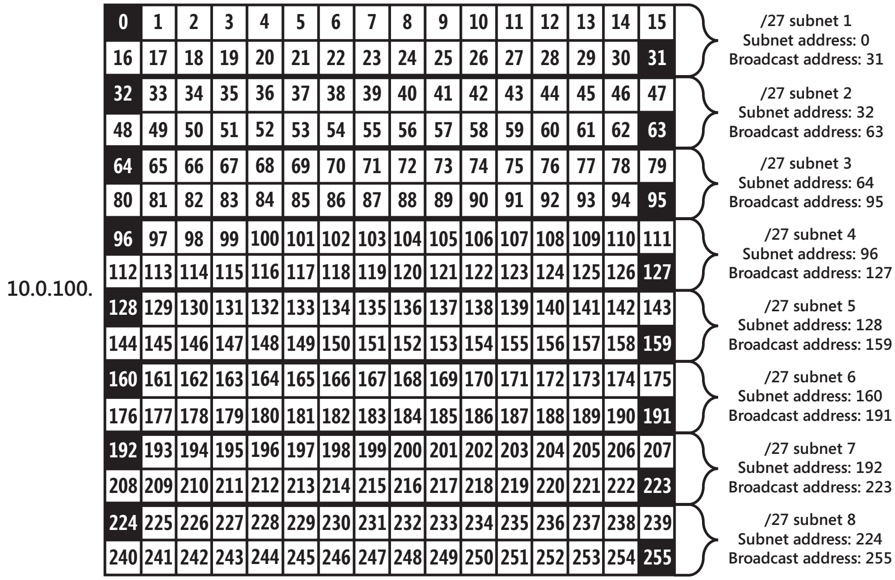

FIGURE 1-44 A /24 address space subnetted globally with a /27 subnet mask

A /27 network can accommodate only 32 distinct addresses, so as a result of subnetting, addresses such as 10.0.100.30 and 10.0.100.33 now belong to different networks. In addition, as shown in Figure 1-44, the first address in each subnet range acts as the network address, and the last address in each range acts as the broadcast address. Each range also requires one address to act as a default gateway.

Using Variable-Length Subnet Masks

It is common to configure an address space so that multiple subnet masks are used internally. Doing so enables you to use your network address space most efficiently.

For example, if you need one subnet to accommodate 100 computers, a second subnet to accommodate 50 computers, and a third subnet to accommodate 20 computers, this arrangement cannot be accommodated by any single internal subnet mask for a /24 address space. As Table 1-5 shows, any single default mask fails to accommodate either enough subnets or enough hosts per subnet to meet all your network needs.

TABLE 1-5 Subnetting a /24 Address Block with a Single Internal Subnet Mask

NETWORK ADDRESS	SUBNETS	HOSTS PER SUBNET
Internal subnet mask: 255.255.255.0	1	254
Internal subnet mask: 255.255.255.128	2	126
Internal subnet mask: 255.255.255.192	4	62
Internal subnet mask: 255.255.255.224	8	30

However, if you use different subnet masks for each subnet, you will be able to divide up the /24 address space in a way that accommodates your needs. This option prevents you from having to acquire new address space from your provider.

Figure 1-45 illustrates how you can use subnet masks of various lengths to accommodate three subnets of 100, 50, and 20 hosts, respectively. This particular network configuration leaves some space unused so that more subnets can be added later.

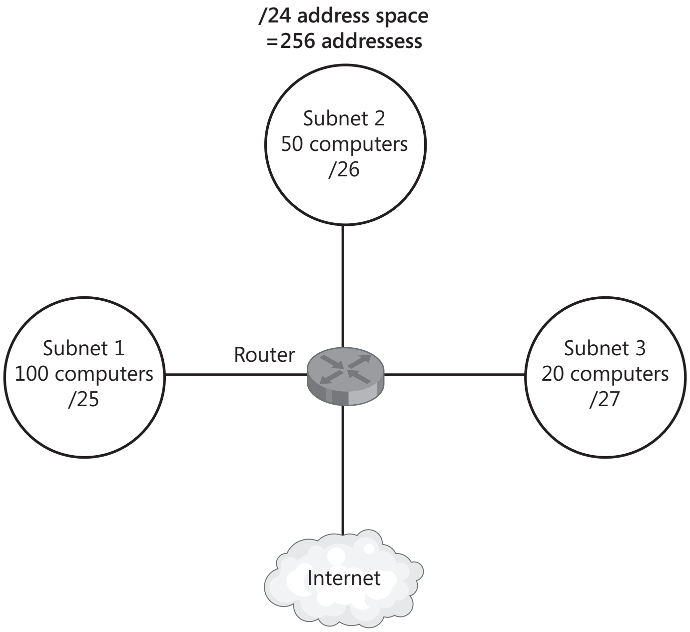

FIGURE 1-45 Using variable-length subnet masks (VLSMs) for flexible subnetting

Enumerating Subnets in an Address Space

When you subnet an address space, you need to know the address ranges of the subnets that are created by this subnetting process. For example, you might have a 192.168.10.0/24 address space and requirements for separate subnets of 20, 50, and 100 hosts. When you divide up your address space to accommodate your needs, you naturally need to determine the specific addresses and subnet masks that will be assigned to each of the three subnets.

To divide an address space and enumerate the address ranges of your subnets, first make sure you have written out an IP address block reference chart such as the one in Figure 1-35.

Then, create a subnet table that gives row headings on the left labeled from top to bottom as Entire Space, Subnet 1, Subnet 2, Subnet 3, and so on, respectively, for every network segment in the network, from largest to smallest. On the top of the table, create column headings from left to right for Addresses Needed, Block Size, First Address, Last Address, and Subnet Mask. The table should look like the one in Figure 1-46.

	Addresses needed	Block size	First address	Last address	Subnet mask
Entire space					
Subnet 1					
Subnet 2					
Subnet 3					
…					

FIGURE 1-46 Create a table to divide your address space into subnets.

After you create the table, simply proceed to fill in the data beginning with the "Entire Space" row. Remember that when filling in block size values > 256, fill in the value expressed as 256 multiplied by some factor, such as "512 = 256 x 2" or "2048 = 256 x 8." The factors will be used for calculations on the third octet.

As an example, we can use the requirements for the 192.168.10.0/24 address space mentioned earlier. The host requirements for its subnets are 100, 50, and 20 computers, or 102, 52, and 22 addresses. We can transfer this information to the table as shown.

	Addresses needed	Block size	First address	Last address	Subnet mask
Entire space	n/a		192.168.10.0		/24 255.255.255.0
Subnet 1	102		192.168.10.0		
Subnet 2	52				
Subnet 3	22				

In the first row, the block size of the entire space is 256 addresses for a /24 network. The last address of the entire space is therefore 192.168.10.255. This last address of the entire space is important. Use this value as a limit to let you know when your subnet requirements go beyond the address space you have available.

	Addresses needed	Block size	First address	Last address	Subnet mask
Entire space	n/a	256	192.168.10.0	192.168.10.255	/24 255.255.255.0
Subnet 1	102		192.168.10.0		
Subnet 2	52				
Subnet 3	22				

After the Entire Space row, you can fill in the remaining data in whichever order is easiest for you. You might find it easiest to fill in all the block sizes first and then continue to fill in the columns from left to right. In this example, we continue to fill the rows in from top to bottom. The method used to fill in any cell in the table is the same in either case.

This brings us to the second row, for Subnet 1. If 102 addresses are needed, you need a block size of 128. The associated subnet mask is /25 or 255.255.255.128, as shown in this table.

	Addresses needed	Block size	First address	Last address	Subnet mask
Entire space	n/a	256	192.168.10.0	192.168.10.255	/24 255.255.255.0
Subnet 1	102	128	192.168.10.0		/25 255.255.255.128
Subnet 2	52				
Subnet 3	22				

Now, add 128 to the last octet of the first address in Subnet 1. This will give you the first address of Subnet 2. To then get the last address of Subnet 1, subtract 1 from this last value, as shown here.

	Addresses needed	Block size	First address	Last address	Subnet mask
Entire space	n/a	256	192.168.10.0	192.168.10.255	/24 255.255.255.0
Subnet 1	102	128	192.168.10.0 +128	192.168.10.127	/25 255.255.255.128
Subnet 2	52		192.168.10.128	-1	
Subnet 3	22				

Next, you can fill in Subnet 2. For 52 addresses, a block size of 64 is needed. The associated subnet mask is /26 or 255.255.255.192. If you add the block size to the first address of Subnet 2, you get the first address of Subnet 3. Subtracting 1 from this last address gives you the last address of Subnet 2, as you can see in the following table. (Note that you can also add the block size to the last address in the previous subnet to give you the last address in the current subnet, but the values are usually not as easy to work with.)

	Addresses needed	Block size	First address	Last address	Subnet mask
Entire space	n/a	256	192.168.10.0	192.168.10.255	/24 255.255.255.0
Subnet 1	102	128	192.168.10.0	192.168.10.127	/25 255.255.255.128
Subnet 2	52	64	192.168.10.128	192.168.10.191	/26 255.255.255.192
Subnet 3	22		192.168.10.192		

Next, you can fill in the data for Subnet 3. For 22 addresses, a block size of 32 is needed. The associated subnet mask is /27 or 255.255.255.224. When you add the block size to the first address in Subnet 3, you get 192.168.10.224. Because you have no more subnets, this address is the first in the range of unused addresses that lies at the end of the entire /24 address space. You can add a row named "Unused" to account for this space, and subtract 1 from the first unused address to determine the last address of Subnet 3.

The finished version of the subnet table is shown here.

	Addresses needed	Block size	First address	Last address	Subnet mask
Entire space	n/a	256	192.168.10.0	192.168.10.255	/24 255.255.255.0
Subnet 1	102	128	192.168.10.0	192.168.10.127	/25 255.255.255.128
Subnet 2	52	64	192.168.10.128	192.168.10.191	/26 255.255.255.192
Subnet 3	22	32	192.168.10.192	192.168.10.223	/27 255.255.255.224
Unused			192.168.10.224	192.168.10.255	

The division of this address space is illustrated in Figure 1-47. The black boxes represent the network addresses and broadcast addresses that cannot be assigned to individual hosts.

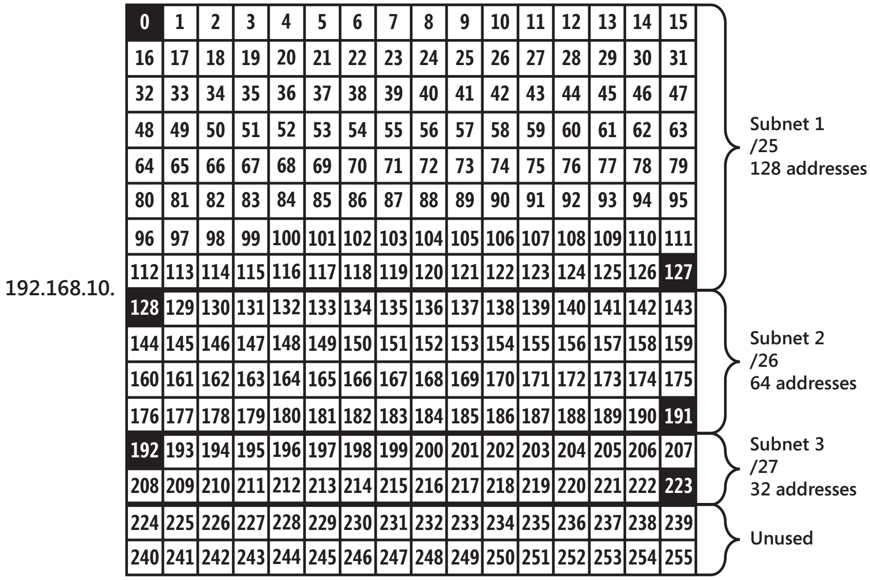

FIGURE 1-47 A /24 network subdivided into subnets of varying sizes

Next, we'll try a second, more complicated example. In this second example, assume that you have been allocated the 10.20.40.0 /22 address space and that you have to accommodate network segments of 300 nodes, 150 nodes, 75 nodes, and 22 nodes.

After you create a subnet table and fill in the information already given, the table should look like the following.

	Addresses needed	Block size	First address	Last address	Subnet mask
Entire space	n/a		10.20.40.0		/22 (255.255.252.0)
Subnet 1	302				
Subnet 2	152				
Subnet 3	77				
Subnet 4	24				

Now you can fill out the first row. A /22 network has a block size of 1024 addresses. Whenever the block size is greater than 256, you should also express the size as 256 times some factor and then use that factor to perform your calculations. In this case, you should write the block size as "256 x 4." Use the factor 4 in the 256 x 4 block size to assist you in determining the upper value of this address range. The factor 4 tells you that the very first address after the entire range must be 10.20.40 + 4.0, or 10.20.44.0. Consequently, the last address of the entire space must be the one before it, or 10.20.43.255. These values are entered into the following table.

	Addresses needed	Block size	First address	Last address	Subnet mask
Entire space	n/a	1024 256x4	10.20.40.0 +4.0 44.0	10.20.43.255 -0.1	/22 (255.255.252.0)
Subnet 1	302				
Subnet 2	152				
Subnet 3	77				
Subnet 4	24				

For Subnet 1, the block size required is 512, or 256 x 2. The associated subnet mask is /23, or 255.255.254.0. To calculate the address range, use the block size factor of 2. The first address of Subnet 1 must be the same as the first address of the entire range, or 10.20.40.0. The factor 2 tells you that the first address of Subnet 2 is 10.20.40 + 2.0, or 10.20.42.0. Consequently, the last address of Subnet 1 is 10.20.41.255, as shown in this table.

	Addresses needed	Block size	First address	Last address	Subnet mask
Entire space	n/a	1024 256x 4	10.20.40.0	10.20.43.255	/22 (255.255.252.0)
Subnet 1	302	512 256x2	10.20.40.0 +2.0	10.20.41.255	/23 255.255.254.0
Subnet 2	152		10.20.42.0	-0.1	
Subnet 3	77				
Subnet 4	24				

For Subnet 2, the block size required is 256. The subnet mask is /24, or 255.255.255.0. This block size is easy to work with: You know that the last address must be 10.20.42.255, and the first address of Subnet 3 must increment the third octet by 1.0 to 10.20.43.0, like the following.

	Addresses needed	Block size	First address	Last address	Subnet mask
Entire space	n/a	1024 256x 4	10.20.40.0	10.20.43.255	/22 (255.255.252.0)
Subnet 1	302	512 256x 2	10.20.40.0	10.20.41.255	/23 255.255.254.0
Subnet 2	152	256x1	10.20.42.0 +1.0	10.20.42.255	/24 255.255.255.0
Subnet 3	77		10.20.43.0	-0.1	
Subnet 4	24				

For Subnet 3, 77 addresses are needed, so the block size required is 128. This block size is associated with a /25 or 255.255.255.128 subnet mask. By adding the block size of 128 to the first address in Subnet 3 of 10.20.42.0, you learn the first address in Subnet 4 is 10.20.43.128 and that the last address in Subnet 3 is 10.20.43.127, as shown in this table.

	Addresses needed	Block size	First address	Last address	Subnet mask
Entire space	n/a	1024 256x 4	10.20.40.0	10.20.43.255	/22 (255.255.252.0)
Subnet 1	302	512 256x 2	10.20.40.0	10.20.41.255	/23 255.255.254.0
Subnet 2	152	256	10.20.42.0	10.20.42.255	/24 255.255.255.0
Subnet 3	77	128	10.20.43.0 +128	10.20.43.127	/25 255.255.255.128
Subnet 4	24		10.20.43.128	-1	

For Subnet 4, 24 addresses are needed, so the block size required is 32. This block size is associated with a /27, or 255.255.255.224 subnet mask. Adding the block size to 10.20.43.128 reveals that there is unused space in the subnet; the resulting value of 10.20.43.160 is still lower than the last address in the entire space of 10.20.43.255. You can add a row named Unused to account for this first unused address, and you can then subtract 1 to arrive at the last address in Subnet 4 of 10.20.43.159.

The final version of the subnet table is shown here.

	Addresses needed	Block size	First address	Last address	Subnet mask
Entire space	n/a	1024 256x 4	10.20.40.0	10.20.43.255	/22 (255.255.252.0)
Subnet 1	302	512 256x 2	10.20.40.0	10.20.41.255	/23 255.255.254.0
Subnet 2	152	256	10.20.42.0	10.20.42.255	/24 255.255.255.0
Subnet 3	77	128	10.20.43.0	10.20.43.127	/25 255.255.255.128
Subnet 4	24	32	10.20.43.128 +32	10.20.43.159	/27 255.255.255.224
Unused			10.20.43.160	-1 10.20.43.255	

The division of this address space is illustrated in Figure 1-48.

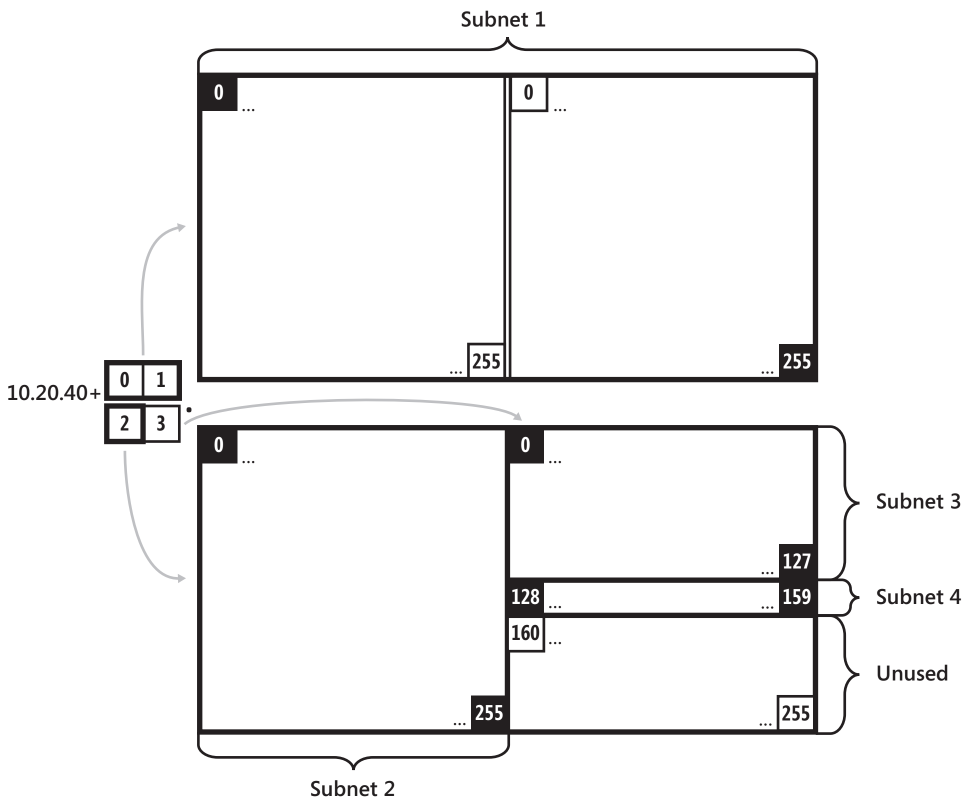

FIGURE 1-48 A /22 address space divided into four subnets of varying sizes

> ***EXAM TIP***
>
> **Questions such as these about dividing up an address space are by far the most time-intensive on the test, but they are not worth more than any other questions. Consequently, you should leave these questions for the end of the exam, after you have answered all other questions.**

Verifying Subnet Ownership and Configuration

As an administrator of a multisubnet network, you need to ensure that individual computers are assigned an IP configuration that places them in the correct subnet with the correct default gateway. To verify the proper IP configuration for hosts, you need to be able to compare different addresses—such as a server address and its default gateway address—and determine whether they are on the same subnet.

To determine whether IP addresses are on the same subnet, first ensure that the hosts you are comparing have the same subnet mask configured. Then, compare the network IDs of the addresses.

For /8, /16, and /24 subnet masks, this comparison is easy: simply compare the IP address values of the first, the first two, or the first three octets, respectively. If and only if the values are identical, the computers are configured on the same subnet. For example, the addresses 192.168.5.1 /24, 192.168.5.32 /24, and 192.168.5.64 /24 are all on the same subnet because they all share the network ID 192.168.5.

For subnet masks of /25 and higher, divide the value of the last octet in each address by the address block size, and drop any remainder so that you are left with a whole number such as 0, 1, or 2. If and only if the resulting whole numbers are the same, the addresses are on the same subnet. For example, 192.168.5.1 /26 and 192.168.5.32 /26 are on the same subnet because the block size of a /26 network is 64, and if you discount the remainder, both 1 ÷ 64 and 32 ÷ 64 equal zero. However, 192.168.5.64 is on a different subnet because 64 ÷ 64 = 1.

For subnet masks between /16 and /24, first convert the subnet mask to dotted-decimal notation by using a reference chart or by memorization. Subtract the value of the third octet in the subnet mask from 256, and then divide the value of the third octet in the IP addresses you want to compare by this resulting difference, dropping any remainders. If and only if the resulting values are the same, the addresses are on the same subnet. For example, if you want to compare 10.0.40.100 /21 and 10.0.41.1 /21, first determine that the dotted-decimal equivalent of /21 is 255.255.248.0, and then subtract 248 from 256 to obtain a value of 8. Finally, because 40 ÷ 8 = 5 and 41 ÷ 8 = 5 with some remainder, the two addresses are located on the same subnet.

PRACTICE Learning to Work with Address Blocks

In this practice, you perform exercises that help solidify your understanding of address blocks, subnet masks, and host capacity.

EXERCISE 1 Choosing an Appropriate Subnet Mask

You are adding a new server to each of the following subnets. Given the addresses of the existing computers on that subnet, determine which subnet mask you should assign the new server.

Question: Which subnet mask would you assign to the new server

SUBNET 1: EXISTING COMPUTERS
10.2.12.1
10.2.41.23
10.2.41.100
10.2.41.101

Answer Choices:

A. 255.0.0.0 (/8)

B. 255.255.0.0 (/16)

C. 255.255.255.0 (/24)

Answer: B

Question: Which subnet mask would you assign to the new server?

SUBNET 2: EXISTING COMPUTERS
192.168.34.1
192.168.34.55
192.168.34.223
192.168.34.5

Answer Choices:

A. 255.0.0.0 (/8)

B. 255.255.0.0 (/16)

C. 255.255.255.0 (/24)

Answer: C

EXERCISE 2 **Converting Subnet Masks to Dotted-Decimal Notation**

Convert the following subnet masks in slash notation to dotted-decimal by using your familiarity with the /16 subnet mask, the /24 subnet mask, and the nine possible subnet mask octet values. Write the final answer in each space provided.

SLASH NOTATION	DOTTED-DECIMAL
/18	
/28	
/21	
/30	
/19	
/26	
/22	
/27	
/17	
/20	
/29	
/23	
/25	

Answer:

SLASH NOTATION	DOTTED-DECIMAL
/18	255.255.192.0
/28	255.255.255.240
/21	255.255.248.0
/30	255.255.255.252
/19	255.255.224.0
/26	255.255.255.192
/22	255.255.252.0
/27	255.255.255.224
/17	255.255.128.0
/20	255.255.240.0
/29	255.255.255.248
/23	255.255.254.0
/25	255.255.255.128

EXERCISE 3 **Converting Subnet Masks to Slash Notation**

Using your familiarity with 255.255.0.0, 255.255.255.0, and with the nine possible values in a subnet mask octet, convert the following subnet masks in dotted-decimal notation to slash notation. Write the final answer in each space provided.

DOTTED-DECIMAL	SLASH NOTATION
255.255.240.0	
255.255.255.248	
255.255.192.0	
255.255.255.128	
255.255.248.0	
255.255.255.224	
255.255.252.0	
255.255.128.0	
255.255.255.252	
255.255.224.0	
255.255.254.0	
255.255.255.192	
255.255.255.240	

Answer:

DOTTED-DECIMAL	SLASH NOTATION
255.255.240.0	/20
255.255.255.248	/29
255.255.192.0	/18
255.255.255.128	/25
255.255.248.0	/21
255.255.255.224	/27
255.255.252.0	/22
255.255.128.0	/17
255.255.255.252	/30
255.255.224.0	/19
255.255.254.0	/23
255.255.255.192	/26
255.255.255.240	/28

EXERCISE 4 **Determining the Host Capacity of Networks**

For each of the given address blocks, determine the number of hosts that can be supported. Write down the answer in the space provided in the right column. (Hint: Remember to subtract 2 from the total number of addresses to determine the number of supported hosts.)

ADDRESS BLOCK	NUMBER OF SUPPORTED HOSTS
131.107.16.0/20	
10.10.128.0 Subnet mask: 255.255.254.0	
206.73.118.0/26	
192.168.23.64 Subnet mask: 255.255.255.224	
131.107.0.0 Subnet mask: 255.255.255.0	
206.73.118.24/29	
10.4.32.0/21	
172.16.12.0/22	
192.168.1.32 Subnet mask: 255.255.255.128	
131.107.100.48/28	
206.73.118.12 Subnet mask: 255.255.255.252	
10.12.200.128/25	
192.168.0.0 Subnet mask: 255.255.248.0	
172.20.43.0/24	
131.107.32.0 Subnet mask 255.255.255.240	
10.200.48.0 Subnet mask: 255.255.240.0	
192.168.244.0/23	
10.0.0.0 /30	
172.31.3.24 Subnet mask: 255.255.255.248	
206.73.118.32/27	
131.107.8.0 Subnet mask: 255.255.252.0	
192.168.0.64 Subnet mask: 255.255.255.192	

Answer:

ADDRESS BLOCK	NUMBER OF SUPPORTED HOSTS
131.107.16.0/20	4,094
10.10.128.0 Subnet mask: 255.255.254.0	510
206.73.118.0/26	62
192.168.23.64 Subnet mask: 255.255.255.224	30
131.107.0.0 Subnet mask: 255.255.255.0	254
206.73.118.24/29	6
10.4.32.0/21	2046
172.16.12.0/22	1022
192.168.1.32 Subnet mask: 255.255.255.128	126
131.107.100.48/28	14
206.73.118.12 Subnet mask: 255.255.255.252	2
10.12.200.128/25	126
192.168.0.0 Subnet mask: 255.255.248.0	2046
172.20.43.0/24	254
131.107.32.0 Subnet mask 255.255.255.240	14
10.200.48.0 Subnet mask: 255.255.240.0	4094
192.168.244.0/23	510
10.0.0.0 /30	2
172.31.3.24 Subnet mask: 255.255.255.248	6
206.73.118.32/27	30
131.107.8.0 Subnet mask: 255.255.252.0	1022
192.168.0.64 Subnet mask: 255.255.255.192	62

EXERCISE 5 **Determining Network Size Requirements in Slash Notation Terms**

Each of the values in the left column of the following table refers to a number of computers that a given network must support. In the corresponding space in the right column, specify with a subnet mask in slash notation the smallest network address size that will accommodate those computers. The first row is provided as an example. (Hint: Remember to add 2 to the number of hosts to determine the number of addresses needed.)

NUMBER OF NETWORK HOSTS	SUBNET MASK (/N)
18	/27
125	
400	
127	
650	
7	
2000	
4	
3500	
20	
32	

Answer:

NUMBER OF NETWORK HOSTS	SUBNET MASK (/N)
125	/25
400	/23
127	/24
650	/22
7	/28
2000	/21
4	/29
3500	/20
20	/27
32	/26

EXERCISE 6 Determining Network Size Requirements in Terms of a Dotted-Decimal Subnet Mask

Each of the values in the left column of the following table refers to a number of computers that a given network must support. In the corresponding space in the right column, specify with a subnet mask in dotted-decimal notation the smallest network size that will accommodate those computers.

The first row is provided as an example. (Hint: Remember to add two to the number of hosts in order to determine the number of addresses needed. Then, use the halving-and-doubling or subtract-from-256 technique.)

NUMBER OF NETWORK HOSTS	SUBNET MASK (W.X.Y.Z)
100	255.255.255.128
63	
1022	
6	
1100	
12	
150	
2500	
20	
300	
35	

Answer:

NUMBER OF NETWORK HOSTS	SUBNET MASK (W.X.Y.Z)
63	255.255.255.128
1022	255.255.252.0
6	255.255.255.248
1100	255.255.248.0
12	255.255.255.240
150	255.255.255.0
2500	255.255.240.0
20	255.255.255.224
300	255.255.254.0
35	255.255.255.192

Lesson Summary

- An IPv4 address is a 32-bit number divided into four octets. One part of the IPv4 address represents a network ID, and the other part represents the host ID.
- The subnet mask is used by an IP host to separate the network ID from the host ID in every IP address. The subnet mask can appear in slash notation, such as /24, or in dotted-decimal notation, such as 255.255.255.0. As a network administrator, you need to be able to translate between these two forms of the IPv4 subnet mask.
- The calculation of the network ID by using the subnet mask tells a computer what to do with an IP packet. If the destination network ID of an IP packet is local, the computer broadcasts the packet on the local network. If the destination network ID is remote, the computer sends the packet to the default gateway.
- The IANA has reserved certain ranges of IP addresses to be used only within private networks. These ranges include 10.0.0.0 through 10.255.255.254, 17.16.0.0 through 17.31.255.254, and 192.168.0.0 through 192.168.255.254.
- You can obtain blocks of IP addresses from your provider. The block will be defined as a single address with a subnet mask, such as 131.107.1.0/24. As a network administrator, you need to be able to determine how many addresses are contained in address blocks defined in this manner. To meet your own needs for addresses, you also need to specify an appropriately sized address block in these terms.
- An address block can be subdivided into multiple subnets, each with its own router. To achieve this, you need to lengthen the subnet mask within your subnets so that computers see their subnet IDs as distinct.

Lesson Review

You can use the following questions to test your knowledge of the information in Lesson 2, “Understanding IPv4 Addressing.” The questions are also available on the companion CD if you prefer to review them in electronic form.

> ***NOTE*** **ANSWERS**
>
> **Answers to these questions and explanations of why each answer choice is correct or incorrect are located in the “Answers” section at the end of the book.**

1. How many computers can you host in an IPv4 network whose address is 172.16.0.0/22?
   - **A.** 512
   - **B.** 1024
   - **C.** 510
   - **D.** 1022

2. You work as a network administrator for a research lab in a large company. The research lab includes six computers, for which central computing services has allocated the address space 172.16.1.0/29. You now plan to add 10 new computers to the research network. Company policy states that each network is granted address space only according to its needs. What should you do?

   A. Ask to expand the network to a /28 address block.

   B. Ask to expand the network to a /27 address block.

   C. Ask to expand the network to a /26 address block.

   D. You do not need to expand the network because a /29 network is large enough to support your needs.

3. You are a network administrator for your company. You have recently deployed a server that runs Windows Server 2008 R2. The server is provided with the following IP configuration:

   Address: 192.168.1.66

   Subnet Mask: 255.255.255.224

   Default Gateway: 192.168.1.1

   Users on remote subnets complain that they are not able to access the new server. Which of the following actions is most likely to resolve the problem?

   A. Change the server address to 192.168.1.62.

   B. Change the server address to 192.168.1.34.

   C. Switch to a 26-bit subnet mask.

   D. Switch to a 25-bit subnet mask.

4. Your company has obtained the 131.107.168.0 /21 address space from an ISP. You now need to design this address space to accommodate the following network segments.

   Segment A: 600 hosts

   Segment B: 300 hosts

   Segment C: 150 hosts

   Segment D: 75 hosts

   Which of the following addresses should you assign the networks?

   A. Segment A: 131.107.168.0/22, Segment B: 131.107.172.0/23, Segment C: 131.107.174.0/24, Segment D: 131.107.175.0/25

   B. Segment A: 131.107.168.0/22, Segment B: 131.107.174.0/23, Segment C: 131.107.175.0/24, Segment D: 131.107.176.0/25.

   C. Segment A: 131.107.168.0/23, Segment B: 131.107.172.0/24, Segment C: 131.107.174.0/25, Segment D: 131.107.175.0/26

   D. Segment A: 131.107.168.0/22, Segment B: 131.107.172.0/23, Segment C: 131.107.174.0/24, Segment D: 131.107.175.128/25