

# Chapter 3

## Subnetting, Variable Length Subnet Masks (VLSMs), and Troubleshooting TCP/IP

---

**THE CCNA EXAM TOPICS COVERED IN THIS CHAPTER INCLUDE THE FOLLOWING:**

- ✓ **Describe how a network works**
  - Interpret network
- ✓ **Implement an IP addressing scheme and IP Services to meet network requirements in a medium-size Enterprise branch office network**
  - Describe the operation and benefits of using private and public IP addressing
  - Implement static and dynamic addressing services for hosts in a LAN environment
  - Calculate and apply an addressing scheme including VLSM IP addressing design to a network
  - Determine the appropriate classless addressing scheme using VLSM and summarization to satisfy addressing requirements in a LAN/WAN environment
  - Identify and correct common problems associated with IP addressing and host configurations



This chapter will pick up right where we left off in the last chapter. We will continue our discussion of IP addressing.

We'll start with subnetting an IP network. You're going to have to really apply yourself, because subnetting takes time and practice in order to nail it. So be patient. Do whatever it takes to get this stuff dialed in. This chapter truly is important—possibly the most important chapter in this book for you to understand.

I'll thoroughly cover IP subnetting from the very beginning. I know this might sound weird to you, but I think you'll be much better off if you can try to forget everything you've learned about subnetting before reading this chapter—especially if you've been to a Microsoft class!

After our discussion of IP subnetting, I'm going to tell you all about Variable Length Subnet Masks (VLSMs), as well as show you how to design and implement a network using VLSM networks.

Once you have mastered VLSM design and implementation, I'll show you how to summarize classful boundaries. We'll go into this further in Chapter 7, “Enhanced IGRP (EIGRP) and Open Shortest Path First (OSPF),” where I'll demonstrate summarizing using EIGRP and OSPF routing protocols.

I'll wrap up the chapter by going over IP address troubleshooting and take you through the steps Cisco recommends when troubleshooting an IP network.

So get psyched—you're about to go for quite a ride! This chapter will truly help you understand IP addressing and networking, so don't get discouraged or give up. If you stick with it, I promise that one day you'll look back on this and you'll be really glad you decided to hang on. It's one of those things that after you understand it, you'll wonder why you once thought it was so hard. Ready? Let's go!



For up-to-the-minute updates for this chapter, please see [www.lammle.com](http://www.lammle.com) and/or [www.sybex.com](http://www.sybex.com).

## Subnetting Basics

In Chapter 2, you learned how to define and find the valid host ranges used in a Class A, Class B, and Class C network address by turning the host bits all off and then all on. This is very good, but here's the catch: You were defining only one network. What happens if you wanted to take one network address and create six networks from it? You would have to do something called *subnetting*, because that's what allows you to take one larger network and break it into a bunch of smaller networks.

There are loads of reasons in favor of subnetting, including the following benefits:

**Reduced network traffic** We all appreciate less traffic of any kind. Networks are no different. Without trusty routers, packet traffic could grind the entire network down to a near standstill. With routers, most traffic will stay on the local network; only packets destined for other networks will pass through the router. Routers create broadcast domains. The more broadcast domains you create, the smaller the broadcast domains and the less network traffic on each network segment.

**Optimized network performance** This is a result of reduced network traffic.

**Simplified management** It's easier to identify and isolate network problems in a group of smaller connected networks than within one gigantic network.

**Facilitated spanning of large geographical distances** Because WAN links are considerably slower and more expensive than LAN links, a single large network that spans long distances can create problems in every area previously listed. Connecting multiple smaller networks makes the system more efficient.

In the following sections, I am going to move to subnetting a network address. This is the good part—ready?

## IP Subnet-Zero

`IP subnet-zero` is not a new command, but in the past, Cisco courseware, and Cisco exam objectives, didn't cover it—but it certainly does now! This command allows you to use the first and last subnet in your network design. For example, the Class C mask of 192 provides subnets 64 and 128 (discussed thoroughly later in this chapter), but with the `ip subnet-zero` command, you now get to use subnets 0, 64, 128, and 192. That is two more subnets for every subnet mask we use.

Even though we don't discuss the command line interface (CLI) until the next chapter, "Cisco's Internetworking Operating System (IOS) and Security Device Manager (SDM)," it's important for you to be familiar with this command:

```
P1R1#sh running-config
Building configuration...
Current configuration : 827 bytes
!
hostname Pod1R1
!
ip subnet-zero
!
```

This router output shows that the command `ip subnet-zero` is enabled on the router. Cisco has turned this command on by default starting with Cisco IOS version 12.x.



When studying for your Cisco exams, make sure you read very carefully and understand if Cisco is asking you *not* to use `ip subnet-zero`. There are instances where this may happen.

## How to Create Subnets

To create subnetworks, you take bits from the host portion of the IP address and reserve them to define the subnet address. This means fewer bits for hosts, so the more subnets, the fewer bits available for defining hosts.

Later in this chapter, you'll learn how to create subnets, starting with Class C addresses. But before you actually implement subnetting, you need to determine your current requirements as well as plan for future conditions.



Before we move on to designing and creating a subnet mask, you need to understand that in this first section, we will be discussing classful routing, which means that all hosts (all nodes) in the network use the exact same subnet mask. When we move on to Variable Length Subnet Masks (VLSMs), I'll discuss classless routing, which means that each network segment *can* use a different subnet mask.

To create a subnet follow these steps:

1. Determine the number of required network IDs:
  - One for each subnet
  - One for each wide area network connection
2. Determine the number of required host IDs per subnet:
  - One for each TCP/IP host
  - One for each router interface
3. Based on the above requirements, create the following:
  - One subnet mask for your entire network
  - A unique subnet ID for each physical segment
  - A range of host IDs for each subnet

### Understanding the Powers of 2

Powers of 2 are important to understand and memorize for use with IP subnetting. To review powers of 2, remember that when you see a number with another number to its upper right (called an exponent), this means you should multiply the number by itself as many times as the upper number specifies. For example,  $2^3$  is  $2 \times 2 \times 2$ , which equals 8. Here's a list of powers of 2 that you should commit to memory:

$$2^1 = 2$$

$$2^2 = 4$$

$$2^3 = 8$$

$$2^4 = 16$$

$$2^5 = 32$$

$$2^6 = 64$$

$$2^7 = 128$$

$$2^8 = 256$$

$$2^9 = 512$$

$$2^{10} = 1,024$$

$$2^{11} = 2,048$$

$$2^{12} = 4,096$$

$$2^{13} = 8,192$$

$$2^{14} = 16,384$$

Before you get stressed out about knowing all these exponents, remember that it's helpful to know them, but it's not absolutely necessary. Here's a little trick since you're working with 2s: Each successive power of 2 is double the previous one.

For example, all you have to do to remember the value of  $2^9$  is to first know that  $2^8 = 256$ . Why? Because when you double 2 to the eighth power (256), you get  $2^9$  (or 512). To determine the value of  $2^{10}$ , simply start at  $2^8 = 256$ , and then double it twice.

You can go the other way as well. If you needed to know what  $2^6$  is, for example, you just cut 256 in half two times: once to reach  $2^7$  and then one more time to reach  $2^6$ .

## Subnet Masks

For the subnet address scheme to work, every machine on the network must know which part of the host address will be used as the subnet address. This is accomplished by assigning a *subnet mask* to each machine. A subnet mask is a 32-bit value that allows the recipient of IP packets to distinguish the network ID portion of the IP address from the host ID portion of the IP address.

The network administrator creates a 32-bit subnet mask composed of 1s and 0s. The 1s in the subnet mask represent the positions that refer to the network or subnet addresses.

Not all networks need subnets, meaning they use the default subnet mask. This is basically the same as saying that a network doesn't have a subnet address. Table 3.1 shows the default subnet masks for Classes A, B, and C. These default masks cannot change. In other words, you can't make a Class B subnet mask read 255.0.0.0. If you try, the host will read that address as invalid and usually won't even let you type it in. For a Class A network, you can't change the first byte in a subnet mask; it must read 255.0.0.0 at a minimum. Similarly, you cannot assign 255.255.255.255, as this is all 1s—a broadcast address. A Class B address must start with 255.255.0.0, and a Class C has to start with 255.255.255.0.

**TABLE 3.1** Default Subnet Mask

Class	Format	Default Subnet Mask
A	<i>network.node.node.node</i>	255.0.0.0
B	<i>network.network.node.node</i>	255.255.0.0
C	<i>network.network.network.node</i>	255.255.255.0

## Classless Inter-Domain Routing (CIDR)

Another term you need to familiarize yourself with is *Classless Inter-Domain Routing (CIDR)*. It's basically the method that ISPs (Internet service providers) use to allocate a number of addresses to a company, a home—a customer. They provide addresses in a certain block size, something I'll be going into in greater detail later in this chapter.

When you receive a block of addresses from an ISP, what you get will look something like this: 192.168.10.32/28. This is telling you what your subnet mask is. The slash notation (/) means how many bits are turned on (1s). Obviously, the maximum could only be /32 because a byte is 8 bits and there are 4 bytes in an IP address: ( $4 \times 8 = 32$ ). But keep in mind that the largest subnet mask available (regardless of the class of address) can only be a /30 because you've got to keep at least 2 bits for host bits.

Take, for example, a Class A default subnet mask, which is 255.0.0.0. This means that the first byte of the subnet mask is all ones (1s), or 11111111. When referring to a slash notation, you need to count all the 1s bits to figure out your mask. The 255.0.0.0 is considered a /8 because it has 8 bits that are 1s—that is, 8 bits that are turned on.

A Class B default mask would be 255.255.0.0, which is a /16 because 16 bits are ones (1s): 11111111.11111111.00000000.00000000.

Table 3.2 has a listing of every available subnet mask and its equivalent CIDR slash notation.

**TABLE 3.2** CIDR Values

Subnet Mask	CIDR Value
255.0.0.0	/8
255.128.0.0	/9
255.192.0.0	/10
255.224.0.0	/11
255.240.0.0	/12

**TABLE 3.2** CIDR Values (*continued*)

<b>Subnet Mask</b>	<b>CIDR Value</b>
255.248.0.0	/13
255.252.0.0	/14
255.254.0.0	/15
255.255.0.0	/16
255.255.128.0	/17
255.255.192.0	/18
255.255.224.0	/19
255.255.240.0	/20
255.255.248.0	/21
255.255.252.0	/22
255.255.254.0	/23
255.255.255.0	/24
255.255.255.128	/25
255.255.255.192	/26
255.255.255.224	/27
255.255.255.240	/28
255.255.255.248	/29
255.255.255.252	/30

The /8 through /15 can only be used with Class A network addresses. /16 through /23 can be used by Class A and B network addresses. /24 through /30 can be used by Class A, B, and C network addresses. This is a big reason why most companies use Class A network addresses. Since they can use all subnet masks, they get the maximum flexibility in network design.



No, you cannot configure a Cisco router using this slash format. But wouldn't that be nice? Nevertheless, it's *really* important for you to know subnet masks in the slash notation (CIDR).

## Subnetting Class C Addresses

There are many different ways to subnet a network. The right way is the way that works best for you. In a Class C address, only 8 bits are available for defining the hosts. Remember that subnet bits start at the left and go to the right, without skipping bits. This means that the only Class C subnet masks can be the following:

Binary	Decimal	CIDR
00000000 = 0		/24
10000000 = 128		/25
11000000 = 192		/26
11100000 = 224		/27
11110000 = 240		/28
11111000 = 248		/29
11111100 = 252		/30

We can't use a /31 or /32 because we have to have at least 2 host bits for assigning IP addresses to hosts. In the past, I never discussed the /25 in a Class C network. Cisco always had been concerned with having at least 2 subnet bits, but now, because of Cisco recognizing the `ip subnet-zero` command in its curriculum and exam objectives, we can use just 1 subnet bit.

In the following sections, I'm going to teach you an alternate method of subnetting that makes it easier to subnet larger numbers in no time. Trust me, you need to be able to subnet fast!

### Subnetting a Class C Address: The Fast Way!

When you've chosen a possible subnet mask for your network and need to determine the number of subnets, valid hosts, and broadcast addresses of a subnet that the mask provides, all you need to do is answer five simple questions:

- How many subnets does the chosen subnet mask produce?
- How many valid hosts per subnet are available?
- What are the valid subnets?
- What's the broadcast address of each subnet?
- What are the valid hosts in each subnet?

At this point, it's important that you both understand and have memorized your powers of 2. Please refer to the sidebar "Understanding the Powers of 2" earlier in this chapter if you need some help. Here's how you get the answers to those five big questions:

- *How many subnets?*  $2^x$  = number of subnets.  $x$  is the number of masked bits, or the 1s. For example, in 11000000, the number of 1s gives us  $2^2$  subnets. In this example, there are 4 subnets.
- *How many hosts per subnet?*  $2^y - 2$  = number of hosts per subnet.  $y$  is the number of unmasked bits, or the 0s. For example, in 11000000, the number of 0s gives us  $2^6 - 2$  hosts. In this example, there are 62 hosts per subnet. You need to subtract 2 for the subnet address and the broadcast address, which are not valid hosts.
- *What are the valid subnets?*  $256 - \text{subnet mask} = \text{block size, or increment number}$ . An example would be  $256 - 192 = 64$ . The block size of a 192 mask is always 64. Start counting at zero in blocks of 64 until you reach the subnet mask value and these are your subnets. 0, 64, 128, 192. Easy, huh?
- *What's the broadcast address for each subnet?* Now here's the really easy part. Since we counted our subnets in the last section as 0, 64, 128, and 192, the broadcast address is always the number right before the next subnet. For example, the 0 subnet has a broadcast address of 63 because the next subnet is 64. The 64 subnet has a broadcast address of 127 because the next subnet is 128. And so on. And remember, the broadcast address of the last subnet is always 255.
- *What are the valid hosts?* Valid hosts are the numbers between the subnets, omitting the all 0s and all 1s. For example, if 64 is the subnet number and 127 is the broadcast address, then 65–126 is the valid host range—it's *always* the numbers between the subnet address and the broadcast address.

I know this can truly seem confusing. But it really isn't as hard as it seems to be at first—just hang in there! Why not try a few and see for yourself?

## Subnetting Practice Examples: Class C Addresses

Here's your opportunity to practice subnetting Class C addresses using the method I just described. Exciting, isn't it! We're going to start with the first Class C subnet mask and work through every subnet that we can using a Class C address. When we're done, I'll show you how easy this is with Class A and B networks too!

### Practice Example #1C: 255.255.255.128 (/25)

Since 128 is 10000000 in binary, there is only 1 bit for subnetting and 7 bits for hosts. We're going to subnet the Class C network address 192.168.10.0.

192.168.10.0 = Network address

255.255.255.128 = Subnet mask

Now, let's answer the big five:

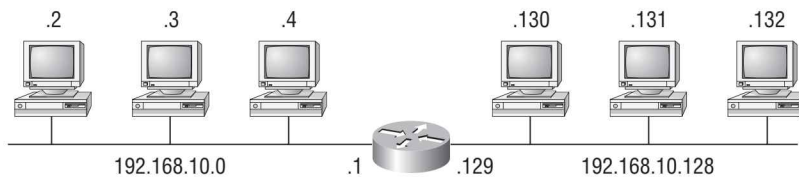
- *How many subnets?* Since 128 is 1 bit on (10000000), the answer would be  $2^1 = 2$ .
- *How many hosts per subnet?* We have 7 host bits off (10000000), so the equation would be  $2^7 - 2 = 126$  hosts.

- *What are the valid subnets?*  $256 - 128 = 128$ . Remember, we'll start at zero and count in our block size, so our subnets are 0, 128.
- *What's the broadcast address for each subnet?* The number right before the value of the next subnet is all host bits turned on and equals the broadcast address. For the zero subnet, the next subnet is 128, so the broadcast of the 0 subnet is 127.
- *What are the valid hosts?* These are the numbers between the subnet and broadcast address. The easiest way to find the hosts is to write out the subnet address and the broadcast address. This way, the valid hosts are obvious. The following table shows the 0 and 128 subnets, the valid host ranges of each, and the broadcast address of both subnets:

<b>Subnet</b>	0	128
<b>First host</b>	1	129
<b>Last host</b>	126	254
<b>Broadcast</b>	127	255

Before moving on to the next example, take a look at Figure 3.1. Okay, looking at a Class C /25, it's pretty clear there are two subnets. But so what—why is this significant? Well actually, it's not, but that's not the right question. What you really want to know is what you would do with this information!

**FIGURE 3.1** Implementing a Class C /25 logical network



```
Router#show ip route
[output cut]
C 192.168.10.0 is directly connected to Ethernet 0.
C 192.168.10.128 is directly connected to Ethernet 1.
```

I know this isn't exactly everyone's favorite pastime, but it's really important, so just hang in there; we're going to talk about subnetting—period. You need to know that the key to understanding subnetting is to understand the very reason you need to do it. And I'm going to demonstrate this by going through the process of building a physical network—and let's add a router. (We now have an internetwork, as I truly hope you already know!) All right, because we added that router, in order for the hosts on our internetwork to communicate, they must now have a logical network addressing scheme. We could use IPX or IPv6, but IPv4 is still the most popular, and it also just happens to be what we're studying at the moment, so that's what we're going with. Okay—now take a look back to Figure 3.1. There are two physical networks, so we're going to implement a logical addressing scheme that allows for two logical networks. As always, it's a really good idea to

look ahead and consider likely growth scenarios—both short and long term, but for this example, a /25 will do the trick.

### Practice Example #2C: 255.255.255.192 (/26)

In this second example, we're going to subnet the network address 192.168.10.0 using the subnet mask 255.255.255.192.

192.168.10.0 = Network address

255.255.255.192 = Subnet mask

Now, let's answer the big five:

- *How many subnets?* Since 192 is 2 bits on (11000000), the answer would be  $2^2 = 4$  subnets.
- *How many hosts per subnet?* We have 6 host bits off (11000000), so the equation would be  $2^6 - 2 = 62$  hosts.
- *What are the valid subnets?*  $256 - 192 = 64$ . Remember, we start at zero and count in our block size, so our subnets are 0, 64, 128, and 192.
- *What's the broadcast address for each subnet?* The number right before the value of the next subnet is all host bits turned on and equals the broadcast address. For the zero subnet, the next subnet is 64, so the broadcast address for the zero subnet is 63.
- *What are the valid hosts?* These are the numbers between the subnet and broadcast address. The easiest way to find the hosts is to write out the subnet address and the broadcast address. This way, the valid hosts are obvious. The following table shows the 0, 64, 128, and 192 subnets, the valid host ranges of each, and the broadcast address of each subnet:

The subnets (do this first)	0	64	128	192
Our first host (perform host addressing last)	1	65	129	193
Our last host	62	126	190	254
The broadcast address (do this second)	63	127	191	255

Okay, again, before getting into the next example, you can see that we can now subnet a /26. And what are you going to do with this fascinating information? Implement it! We'll use Figure 3.2 to practice a /26 network implementation.

The /26 mask provides four subnetworks, and we need a subnet for each router interface. With this mask, in this example, we actually have room to add another router interface.

### Practice Example #3C: 255.255.255.224 (/27)

This time, we'll subnet the network address 192.168.10.0 and subnet mask 255.255.255.224.

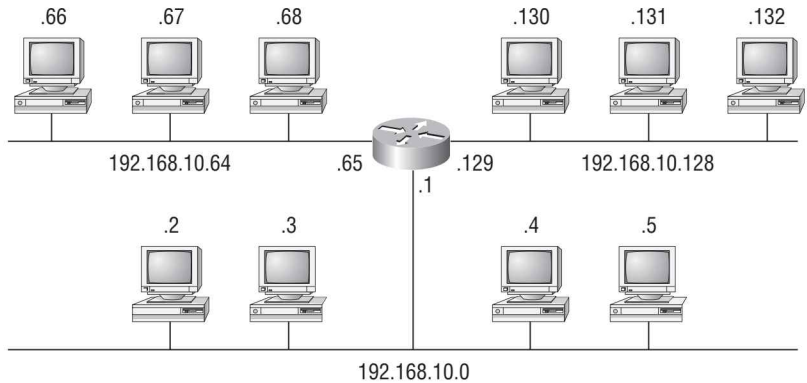
192.168.10.0 = Network address

255.255.255.224 = Subnet mask

- *How many subnets?* 224 is 11100000, so our equation would be  $2^3 = 8$ .
- *How many hosts?*  $2^5 - 2 = 30$ .
- *What are the valid subnets?*  $256 - 224 = 32$ . We just start at zero and count to the subnet mask value in blocks (increments) of 32: 0, 32, 64, 96, 128, 160, 192, and 224.

- What's the broadcast address for each subnet (always the number right before the next subnet)?
- What are the valid hosts (the numbers between the subnet number and the broadcast address)?

**FIGURE 3.2** Implementing a Class C /26 logical network



```
Router#show ip route
[output cut]
C 192.168.10.0 is directly connected to Ethernet 0
C 192.168.10.64 is directly connected to Ethernet 1
C 192.168.10.128 is directly connected to Ethernet 2
```

To answer the last two questions, first just write out the subnets, then write out the broadcast addresses—the number right before the next subnet. Last, fill in the host addresses. The following table gives you all the subnets for the 255.255.255.224 Class C subnet mask:

The subnet address	0	32	64	96	128	160	192	224
The first valid host	1	33	65	97	129	161	193	225
The last valid host	30	62	94	126	158	190	222	254
The broadcast address	31	63	95	127	159	191	223	255

**Practice Example #4C: 255.255.255.240 (/28)**

Let's practice on another one:

192.168.10.0 = Network address  
 255.255.255.240 = Subnet mask

- Subnets? 240 is 11110000 in binary.  $2^4 = 16$ .
- Hosts? 4 host bits, or  $2^4 - 2 = 14$ .
- Valid subnets?  $256 - 240 = 16$ . Start at 0:  $0 + 16 = 16$ .  $16 + 16 = 32$ .  $32 + 16 = 48$ .  $48 + 16 = 64$ .  $64 + 16 = 80$ .  $80 + 16 = 96$ .  $96 + 16 = 112$ .  $112 + 16 = 128$ .  $128 + 16 = 144$ .  $144 + 16 = 160$ .  $160 + 16 = 176$ .  $176 + 16 = 192$ .  $192 + 16 = 208$ .  $208 + 16 = 224$ .  $224 + 16 = 240$ .

- *Broadcast address for each subnet?*
- *Valid hosts?*

To answer the last two questions, check out the following table. It gives you the subnets, valid hosts, and broadcast addresses for each subnet. First, find the address of each subnet using the block size (increment). Second, find the broadcast address of each subnet increment (it's always the number right before the next valid subnet), then just fill in the host addresses. The following table shows the available subnets, hosts, and broadcast addresses provided from a Class C 255.255.255.240 mask:

<b>Subnet</b>	0	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240
<b>First host</b>	1	17	33	49	65	81	97	113	129	145	161	177	193	209	225	241
<b>Last host</b>	14	30	46	62	78	94	110	126	142	158	174	190	206	222	238	254
<b>Broadcast</b>	15	31	47	63	79	95	111	127	143	159	175	191	207	223	239	255



Cisco has figured out that most people cannot count in 16s and therefore have a hard time finding valid subnets, hosts, and broadcast addresses with the Class C 255.255.255.240 mask. You'd be wise to study this mask.

### Practice Example #5C: 255.255.255.248 (/29)

Let's keep practicing:

192.168.10.0 = Network address

255.255.255.248 = Subnet mask

- *Subnets?* 248 in binary = 11111000.  $2^5 = 32$ .
- *Hosts?*  $2^3 - 2 = 6$ .
- *Valid subnets?*  $256 - 248 = 0, 8, 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120, 128, 136, 144, 152, 160, 168, 176, 184, 192, 200, 208, 216, 224, 232, 240, \text{ and } 248$ .
- *Broadcast address for each subnet?*
- *Valid hosts?*

Take a look at the following table. It shows some of the subnets (first four and last four only), valid hosts, and broadcast addresses for the Class C 255.255.255.248 mask:

<b>Subnet</b>	0	8	16	24	...	224	232	240	248
<b>First host</b>	1	9	17	25	...	225	233	241	249
<b>Last host</b>	6	14	22	30	...	230	238	246	254
<b>Broadcast</b>	7	15	23	31	...	231	239	247	255

**Practice Example #6C: 255.255.255.252 (/30)**

Just one more:

192.168.10.0 = Network address

255.255.255.252 = Subnet mask

- *Subnets?* 64.
- *Hosts?* 2.
- *Valid subnets?* 0, 4, 8, 12, etc., all the way to 252.
- *Broadcast address for each subnet (always the number right before the next subnet)?*
- *Valid hosts (the numbers between the subnet number and the broadcast address)?*

The following table shows you the subnet, valid host, and broadcast address of the first four and last four subnets in the 255.255.255.252 Class C subnet:

<b>Subnet</b>	0	4	8	12	...	240	244	248	252
<b>First host</b>	1	5	9	13	...	241	245	249	253
<b>Last host</b>	2	6	10	14	...	242	246	250	254
<b>Broadcast</b>	3	7	11	15	...	243	247	251	255

**Real World Scenario****Should We Really Use This Mask That Provides Only Two Hosts?**

You are the network administrator for Acme Corporation in San Francisco, with dozens of WAN links connecting to your corporate office. Right now your network is a classful network, which means that the same subnet mask is on each host and router interface. You've read about classless routing where you can have different size masks but don't know what to use on your point-to-point WAN links. Is the 255.255.255.252 (/30) a helpful mask in this situation?

Yes, this is a very helpful mask in wide area networks.

If you use the 255.255.255.0 mask, then each network would have 254 hosts, but you only use 2 addresses with a WAN link! That is a waste of 252 hosts per subnet. If you use the 255.255.255.252 mask, then each subnet has only 2 hosts and you don't waste precious addresses. This is a really important subject, one that we'll address in a lot more detail in the section on VLSM network design later in this chapter.

**Subnetting in Your Head: Class C Addresses**

It really is possible to subnet in your head. Even if you don't believe me, I'll show you how. And it's not all that hard either—take the following example:

192.168.10.33 = Node address

255.255.255.224 = Subnet mask

First, determine the subnet and broadcast address of the above IP address. You can do this by answering question 3 of the big five questions:  $256 - 224 = 32$ . 0, 32, 64. The address of 33 falls between the two subnets of 32 and 64 and must be part of the 192.168.10.32 subnet. The next subnet is 64, so the broadcast address of the 32 subnet is 63. (Remember that the broadcast address of a subnet is always the number right before the next subnet.) The valid host range is 33–62 (the numbers between the subnet and broadcast address). This is too easy!

Okay, let's try another one. We'll subnet another Class C address:

192.168.10.33 = Node address

255.255.255.240 = Subnet mask

What subnet and broadcast address is the above IP address a member of?  $256 - 240 = 16$ . 0, 16, 32, 48. Bingo—the host address is between the 32 and 48 subnets. The subnet is 192.168.10.32, and the broadcast address is 47 (the next subnet is 48). The valid host range is 33–46 (the numbers between the subnet number and the broadcast address).

Okay, we need to do more, just to make sure you have this down.

You have a node address of 192.168.10.174 with a mask of 255.255.255.240. What is the valid host range?

The mask is 240, so we'd do a  $256 - 240 = 16$ . This is our block size. Just keep adding 16 until we pass the host address of 174, starting at zero, of course: 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176. The host address of 174 is between 160 and 176, so the subnet is 160. The broadcast address is 175; the valid host range is 161–174. That was a tough one.

One more—just for fun. This is the easiest one of all Class C subnetting:

192.168.10.17 = Node address

255.255.255.252 = Subnet mask

What subnet and broadcast address is the above IP address a member of?  $256 - 252 = 0$  (always start at zero unless told otherwise), 4, 8, 12, 16, 20, etc. You've got it! The host address is between the 16 and 20 subnets. The subnet is 192.168.10.16, and the broadcast address is 19. The valid host range is 17–18.

Now that you're all over Class C subnetting, let's move on to Class B subnetting. But before we do, let's have a quick review.

## What Do We Know?

Okay—here's where you can really apply what you've learned so far, and begin committing it all to memory. This is a very cool section that I've been using in my classes for years. It will really help you nail down subnetting!

When you see a subnet mask or slash notation (CIDR), you should know the following:

/25 What do we know about a /25?

- 128 mask
- 1 bits on and 7 bits off (10000000)
- Block size of 128
- 2 subnets, each with 126 hosts

/26 What do we know about a /26?

- 192 mask
- 2 bits on and 6 bits off (11000000)
- Block size of 64
- 4 subnets, each with 62 hosts

/27 What do we know about a /27?

- 224 mask
- 3 bits on and 5 bits off (11100000)
- Block size of 32
- 8 subnets, each with 30 hosts

/28 What do we know about a /28?

- 240 mask
- 4 bits on and 4 bits off
- Block size of 16
- 16 subnets, each with 14 hosts

/29 What do we know about a /29?

- 248 mask
- 5 bits on and 3 bits off
- Block size of 8
- 32 subnets, each with 6 hosts

/30 What do we know about a /30?

- 252 mask
- 6 bits on and 2 bits off
- Block size of 4
- 64 subnets, each with 2 hosts

Regardless of whether you have a Class A, Class B, or Class C address, the /30 mask will provide you with only two hosts, ever. This mask is suited almost exclusively—as well as suggested by Cisco—for use on point-to-point links.

If you can memorize this “What Do We Know?” section, you’ll be much better off in your day-to-day job and in your studies. Try saying it out loud, which helps you memorize things—yes, your significant other and/or coworkers will think you’ve lost it, but they probably already do if you are in the networking field. And if you’re not yet in the networking field but are studying all this to break into it, you might as well have people start thinking you’re an odd bird now since they will eventually anyway.

It's also helpful to write these on some type of flashcards and have people test your skill. You'd be amazed at how fast you can get subnetting down if you memorize block sizes as well as this "What Do We Know?" section.

## Subnetting Class B Addresses

Before we dive into this, let's look at all the possible Class B subnet masks first. Notice that we have a lot more possible subnet masks than we do with a Class C network address:

255.255.0.0	(/16)		
255.255.128.0	(/17)	255.255.255.0	(/24)
255.255.192.0	(/18)	255.255.255.128	(/25)
255.255.224.0	(/19)	255.255.255.192	(/26)
255.255.240.0	(/20)	255.255.255.224	(/27)
255.255.248.0	(/21)	255.255.255.240	(/28)
255.255.252.0	(/22)	255.255.255.248	(/29)
255.255.254.0	(/23)	255.255.255.252	(/30)

We know the Class B network address has 16 bits available for host addressing. This means we can use up to 14 bits for subnetting (because we have to leave at least 2 bits for host addressing). Using a /16 means you are not subnetting with class B, but it is a mask you can use.



By the way, do you notice anything interesting about that list of subnet values—a pattern, maybe? Ah ha! That's exactly why I had you memorize the binary-to-decimal numbers at the beginning of this section. Since subnet mask bits start on the left and move to the right and bits can't be skipped, the numbers are always the same regardless of the class of address. Memorize this pattern.

The process of subnetting a Class B network is pretty much the same as it is for a Class C, except that you just have more host bits and you start in the third octet.

Use the same subnet numbers for the third octet with Class B that you used for the fourth octet with Class C, but add a zero to the network portion and a 255 to the broadcast section in the fourth octet. The following table shows you an example host range of two subnets used in a Class B 240 (/20) subnet mask:

First subnet	16.0	32.0
Second subnet	31.255	47.255

Just add the valid hosts between the numbers, and you're set!



The preceding example is true only until you get up to /24. After that, it's numerically exactly like Class C.

## Subnetting Practice Examples: Class B Addresses

This section will give you an opportunity to practice subnetting Class B addresses. Again, I have to mention that this is the same as subnetting with Class C, except we start in the third octet—with the exact same numbers!

### Practice Example #1B: 255.255.128.0 (/17)

172.16.0.0 = Network address

255.255.128.0 = Subnet mask

- *Subnets?*  $2^1 = 2$  (same as Class C).
- *Hosts?*  $2^{15} - 2 = 32,766$  (7 bits in the third octet, and 8 in the fourth).
- *Valid subnets?*  $256 - 128 = 128$ . 0, 128. Remember that subnetting is performed in the third octet, so the subnet numbers are really 0.0 and 128.0, as shown in the next table. These are the exact numbers we used with Class C; we use them in the third octet and add a 0 in the fourth octet for the network address.
- *Broadcast address for each subnet?*
- *Valid hosts?*

The following table shows the two subnets available, the valid host range, and the broadcast address of each:

<b>Subnet</b>	0.0	128.0
<b>First host</b>	0.1	128.1
<b>Last host</b>	127.254	255.254
<b>Broadcast</b>	127.255	255.255

Okay, notice that we just added the fourth octet's lowest and highest values and came up with the answers. And again, it's done exactly the same way as for a Class C subnet. We just use the same numbers in the third octet and added 0 and 255 in the fourth octet—pretty simple huh! I really can't say this enough: It's just not hard; the numbers never change; we just use them in different octets!

### Practice Example #2B: 255.255.192.0 (/18)

172.16.0.0 = Network address

255.255.192.0 = Subnet mask

- *Subnets?*  $2^2 = 4$ .
- *Hosts?*  $2^{14} - 2 = 16,382$  (6 bits in the third octet, and 8 in the fourth).
- *Valid subnets?*  $256 - 192 = 64$ . 0, 64, 128, 192. Remember that the subnetting is performed in the third octet, so the subnet numbers are really 0.0, 64.0, 128.0, and 192.0, as shown in the next table.
- *Broadcast address for each subnet?*
- *Valid hosts?*

The following table shows the four subnets available, the valid host range, and the broadcast address of each:

Subnet	0.0	64.0	128.0	192.0
First host	0.1	64.1	128.1	192.1
Last host	63.254	127.254	191.254	255.254
Broadcast	63.255	127.255	191.255	255.255

Again, it's pretty much the same as it is for a Class C subnet—we just added 0 and 255 in the fourth octet for each subnet in the third octet.

### Practice Example #3B: 255.255.240.0 (/20)

172.16.0.0 = Network address

255.255.240.0 = Subnet mask

- *Subnets?*  $2^4 = 16$ .
- *Hosts?*  $2^{12} - 2 = 4094$ .
- *Valid subnets?*  $256 - 240 = 0, 16, 32, 48, \text{etc.}, \text{up to } 240$ . Notice that these are the same numbers as a Class C 240 mask – we just put them in the third octet and add a 0 and 255 in the fourth octet.
- *Broadcast address for each subnet?*
- *Valid hosts?*

The following table shows the first four subnets, valid hosts, and broadcast addresses in a Class B 255.255.240.0 mask:

Subnet	0.0	16.0	32.0	48.0
First host	0.1	16.1	32.1	48.1
Last host	15.254	31.254	47.254	63.254
Broadcast	15.255	31.255	47.255	63.255

### Practice Example #4B: 255.255.254.0 (/23)

172.16.0.0 = Network address

255.255.254.0 = Subnet mask

- *Subnets?*  $2^7 = 128$ .
- *Hosts?*  $2^9 - 2 = 510$ .
- *Valid subnets?*  $256 - 254 = 0, 2, 4, 6, 8, \text{etc.}, \text{up to } 254$ .
- *Broadcast address for each subnet?*
- *Valid hosts?*

The following table shows the first five subnets, valid hosts, and broadcast addresses in a Class B 255.255.254.0 mask:

Subnet	0.0	2.0	4.0	6.0	8.0
First host	0.1	2.1	4.1	6.1	8.1
Last host	1.254	3.254	5.254	7.254	9.254
Broadcast	1.255	3.255	5.255	7.255	9.255

### Practice Example #5B: 255.255.255.0 (/24)

Contrary to popular belief, 255.255.255.0 used with a Class B network address is not called a Class B network with a Class C subnet mask. It's amazing how many people see this mask used in a Class B network and think it's a Class C subnet mask. This is a Class B subnet mask with 8 bits of subnetting—it's considerably different from a Class C mask. Subnetting this address is fairly simple:

172.16.0.0 = Network address

255.255.255.0 = Subnet mask

- *Subnets?*  $2^8 = 256$ .
- *Hosts?*  $2^8 - 2 = 254$ .
- *Valid subnets?* 256 - 255 = 1. 0, 1, 2, 3, etc., all the way to 255.
- *Broadcast address for each subnet?*
- *Valid hosts?*

The following table shows the first four and last two subnets, the valid hosts, and the broadcast addresses in a Class B 255.255.255.0 mask:

Subnet	0.0	1.0	2.0	3.0	...	254.0	255.0
First host	0.1	1.1	2.1	3.1	...	254.1	255.1
Last host	0.254	1.254	2.254	3.254	...	254.254	255.254
Broadcast	0.255	1.255	2.255	3.255	...	254.255	255.255

### Practice Example #6B: 255.255.255.128 (/25)

This is one of the hardest subnet masks you can play with. And worse, it actually is a really good subnet to use in production because it creates over 500 subnets with 126 hosts for each subnet—a nice mixture. So, don't skip over it!

172.16.0.0 = Network address

255.255.255.128 = Subnet mask

- *Subnets?*  $2^9 = 512$ .
- *Hosts?*  $2^7 - 2 = 126$ .
- *Valid subnets?* Okay, now for the tricky part.  $256 - 255 = 1$ . 0, 1, 2, 3, etc. for the third octet. But you can't forget the one subnet bit used in the fourth octet. Remember when I showed you how to figure one subnet bit with a Class C mask? You figure this the same way. (Now you know why I showed you the 1-bit subnet mask in the Class C section—to make this part easier.) You actually get two subnets for each third octet value, hence the 512 subnets. For example, if the third octet is showing subnet 3, the two subnets would actually be 3.0 and 3.128.
- *Broadcast address for each subnet?*
- *Valid hosts?*

The following table shows how you can create subnets, valid hosts, and broadcast addresses using the Class B 255.255.255.128 subnet mask (the first eight subnets are shown, and then the last two subnets):

<b>Subnet</b>	0.0	0.128	1.0	1.128	2.0	2.128	3.0	3.128	...	255.0	255.128
<b>First host</b>	0.1	0.129	1.1	1.129	2.1	2.129	3.1	3.129	...	255.1	255.129
<b>Last host</b>	0.126	0.254	1.126	1.254	2.126	2.254	3.126	3.254	...	255.126	255.254
<b>Broad- cast</b>	0.127	0.255	1.127	1.255	2.127	2.255	3.127	3.255	...	255.127	255.255

**Practice Example #7B: 255.255.255.192 (/26)**

Now, this is where Class B subnetting gets easy. Since the third octet has a 255 in the mask section, whatever number is listed in the third octet is a subnet number. However, now that we have a subnet number in the fourth octet, we can subnet this octet just as we did with Class C subnetting. Let's try it out:

172.16.0.0 = Network address  
 255.255.255.192 = Subnet mask

- *Subnets?*  $2^{10} = 1024$ .
- *Hosts?*  $2^6 - 2 = 62$ .
- *Valid subnets?*  $256 - 192 = 64$ . The subnets are shown in the following table. Do these numbers look familiar?
- *Broadcast address for each subnet?*
- *Valid hosts?*

The following table shows the first eight subnet ranges, valid hosts, and broadcast addresses:

<b>Subnet</b>	0.0	0.64	0.128	0.192	1.0	1.64	1.128	1.192
<b>First host</b>	0.1	0.65	0.129	0.193	1.1	1.65	1.129	1.193
<b>Last host</b>	0.62	0.126	0.190	0.254	1.62	1.126	1.190	1.254
<b>Broadcast</b>	0.63	0.127	0.191	0.255	1.63	1.127	1.191	1.255

Notice that for each subnet value in the third octet, you get subnets 0, 64, 128, and 192 in the fourth octet.

### Practice Example #8B: 255.255.255.224 (/27)

This is done the same way as the preceding subnet mask, except that we just have more subnets and fewer hosts per subnet available.

172.16.0.0 = Network address

255.255.255.224 = Subnet mask

- *Subnets?*  $2^{11} = 2048$ .
- *Hosts?*  $2^5 - 2 = 30$ .
- *Valid subnets?*  $256 - 224 = 32$ . 0, 32, 64, 96, 128, 160, 192, 224.
- *Broadcast address for each subnet?*
- *Valid hosts?*

The following table shows the first eight subnets:

<b>Subnet</b>	0.0	0.32	0.64	0.96	0.128	0.160	0.192	0.224
<b>First host</b>	0.1	0.33	0.65	0.97	0.129	0.161	0.193	0.225
<b>Last host</b>	0.30	0.62	0.94	0.126	0.158	0.190	0.222	0.254
<b>Broadcast</b>	0.31	0.63	0.95	0.127	0.159	0.191	0.223	0.255

This next table shows the last eight subnets:

<b>Subnet</b>	255.0	255.32	255.64	255.96	255.128	255.160	255.192	255.224
<b>First host</b>	255.1	255.33	255.65	255.97	255.129	255.161	255.193	255.225
<b>Last host</b>	255.30	255.62	255.94	255.126	255.158	255.190	255.222	255.254
<b>Broadcast</b>	255.31	255.63	255.95	255.127	255.159	255.191	255.223	255.255

## Subnetting in Your Head: Class B Addresses

Are you nuts? Subnet Class B addresses in our heads? It's actually easier than writing it out—I'm not kidding! Let me show you how:

*Question:* What subnet and broadcast address is the IP address 172.16.10.33 255.255.255.224 (/27) a member of?

*Answer:* The interesting octet is the fourth octet.  $256 - 224 = 32$ .  $32 + 32 = 64$ . Bingo: 33 is between 32 and 64. However, remember that the third octet is considered part of the subnet, so the answer would be the 10.32 subnet. The broadcast is 10.63, since 10.64 is the next subnet. That was a pretty easy one.

*Question:* What subnet and broadcast address is the IP address 172.16.66.10 255.255.192.0 (/18) a member of?

*Answer:* The interesting octet is the third octet instead of the fourth octet.  $256 - 192 = 64$ . 0, 64, 128. The subnet is 172.16.64.0. The broadcast must be 172.16.127.255 since 128.0 is the next subnet.

*Question:* What subnet and broadcast address is the IP address 172.16.50.10 255.255.224.0 (/19) a member of?

*Answer:*  $256 - 224 = 0, 32, 64$  (remember, we always start counting at zero (0)). The subnet is 172.16.32.0, and the broadcast must be 172.16.63.255 since 64.0 is the next subnet.

*Question:* What subnet and broadcast address is the IP address 172.16.46.255 255.255.240.0 (/20) a member of?

*Answer:*  $256 - 240 = 16$ . The third octet is interesting to us. 0, 16, 32, 48. This subnet address must be in the 172.16.32.0 subnet, and the broadcast must be 172.16.47.255 since 48.0 is the next subnet. So, yes, 172.16.46.255 is a valid host.

*Question:* What subnet and broadcast address is the IP address 172.16.45.14 255.255.255.252 (/30) a member of?

*Answer:* Where is the interesting octet?  $256 - 252 = 0, 4, 8, 12, 16$  (in the fourth octet). The subnet is 172.16.45.12, with a broadcast of 172.16.45.15 because the next subnet is 172.16.45.16.

*Question:* What is the subnet and broadcast address of the host 172.16.88.255/20?

*Answer:* What is a /20? If you can't answer this, you can't answer this question, can you? A /20 is 255.255.240.0, which gives us a block size of 16 in the third octet, and since no subnet bits are on in the fourth octet, the answer is always 0 and 255 in the fourth octet. 0, 16, 32, 48, 64, 80, 96...bingo. 88 is between 80 and 96, so the subnet is 80.0 and the broadcast address is 95.255.

*Question:* A router receives a packet on an interface with a destination address of 172.16.46.191/26. What will the router do with this packet?

*Answer:* Discard it. Do you know why? 172.16.46.191/26 is a 255.255.255.192 mask, which gives us a block size of 64. Our subnets are then 0, 64, 128, 192. 191 is the broadcast address of the 128 subnet, so a router, by default, will discard any broadcast packets.

## Subnetting Class A Addresses

Class A subnetting is not performed any differently than Classes B and C, but there are 24 bits to play with instead of the 16 in a Class B address and the 8 in a Class C address.

Let's start by listing all the Class A masks:

255.0.0.0 (/8)	
255.128.0.0 (/9)	255.255.240.0 (/20)
255.192.0.0 (/10)	255.255.248.0 (/21)
255.224.0.0 (/11)	255.255.252.0 (/22)
255.240.0.0 (/12)	255.255.254.0 (/23)
255.248.0.0 (/13)	255.255.255.0 (/24)
255.252.0.0 (/14)	255.255.255.128 (/25)
255.254.0.0 (/15)	255.255.255.192 (/26)
255.255.0.0 (/16)	255.255.255.224 (/27)
255.255.128.0 (/17)	255.255.255.240 (/28)
255.255.192.0 (/18)	255.255.255.248 (/29)
255.255.224.0 (/19)	255.255.255.252 (/30)

That's it. You must leave at least 2 bits for defining hosts. And I hope you can see the pattern by now. Remember, we're going to do this the same way as a Class B or C subnet. It's just that, again, we simply have more host bits and we just use the same subnet numbers we used with Class B and C, but we start using these numbers in the second octet.

### Subnetting Practice Examples: Class A Addresses

When you look at an IP address and a subnet mask, you must be able to distinguish the bits used for subnets from the bits used for determining hosts. This is imperative. If you're still struggling with this concept, please reread the section "IP Addressing" in Chapter 2. It shows you how to determine the difference between the subnet and host bits and should help clear things up.

#### Practice Example #1A: 255.255.0.0 (/16)

Class A addresses use a default mask of 255.0.0.0, which leaves 22 bits for subnetting since you must leave 2 bits for host addressing. The 255.255.0.0 mask with a Class A address is using 8 subnet bits.

- *Subnets?*  $2^8 = 256$ .
- *Hosts?*  $2^{16} - 2 = 65,534$ .
- *Valid subnets?* What is the interesting octet?  $256 - 255 = 1$ . 0, 1, 2, 3, etc. (all in the second octet). The subnets would be 10.0.0.0, 10.1.0.0, 10.2.0.0, 10.3.0.0, etc., up to 10.255.0.0.

- *Broadcast address for each subnet?*
- *Valid hosts?*

The following table shows the first two and last two subnets, valid host range, and broadcast addresses for the private Class A 10.0.0.0 network:

<b>Subnet</b>	10.0.0.0	10.1.0.0	...	10.254.0.0	10.255.0.0
<b>First host</b>	10.0.0.1	10.1.0.1	...	10.254.0.1	10.255.0.1
<b>Last host</b>	10.0.255.254	10.1.255.254	...	10.254.255.254	10.255.255.254
<b>Broadcast</b>	10.0.255.255	10.1.255.255	...	10.254.255.255	10.255.255.255

### **Practice Example #2A: 255.255.240.0 (/20)**

255.255.240.0 gives us 12 bits of subnetting and leaves us 12 bits for host addressing.

- *Subnets?*  $2^{12} = 4096$ .
- *Hosts?*  $2^{12} - 2 = 4094$ .
- *Valid subnets?* What is your interesting octet?  $256 - 240 = 16$ . The subnets in the second octet are a block size of 1 and the subnets in the third octet are 0, 16, 32, etc.
- *Broadcast address for each subnet?*
- *Valid hosts?*

The following table shows some examples of the host ranges—the first three and the last subnets:

<b>Subnet</b>	10.0.0.0	10.0.16.0	10.0.32.0	...	10.255.240.0
<b>First host</b>	10.0.0.1	10.0.16.1	10.0.32.1	...	10.255.240.1
<b>Last host</b>	10.0.15.254	10.0.31.254	10.0.47.254	...	10.255.255.254
<b>Broadcast</b>	10.0.15.255	10.0.31.255	10.0.47.255	...	10.255.255.255

### **Practice Example #3A: 255.255.255.192 (/26)**

Let's do one more example using the second, third, and fourth octets for subnetting.

- *Subnets?*  $2^{18} = 262,144$ .
- *Hosts?*  $2^6 - 2 = 62$ .
- *Valid subnets?* In the second and third octet, the block size is 1, and in the fourth octet, the block size is 64.
- *Broadcast address for each subnet?*
- *Valid hosts?*

The following table shows the first four subnets and their valid hosts and broadcast addresses in the Class A 255.255.255.192 mask:

<b>Subnet</b>	10.0.0.0	10.0.0.64	10.0.0.128	10.0.0.192
<b>First host</b>	10.0.0.1	10.0.0.65	10.0.0.129	10.0.0.193
<b>Last host</b>	10.0.0.62	10.0.0.126	10.0.0.190	10.0.0.254
<b>Broadcast</b>	10.0.0.63	10.0.0.127	10.0.0.191	10.0.0.255

The following table shows the last four subnets and their valid hosts and broadcast addresses:

<b>Subnet</b>	10.255.255.0	10.255.255.64	10.255.255.128	10.255.255.192
<b>First host</b>	10.255.255.1	10.255.255.65	10.255.255.129	10.255.255.193
<b>Last host</b>	10.255.255.62	10.255.255.126	10.255.255.190	10.255.255.254
<b>Broadcast</b>	10.255.255.63	10.255.255.127	10.255.255.191	10.255.255.255

## Subnetting in Your Head: Class A Addresses

This sounds hard, but as with Class C and Class B, the numbers are the same; we just start in the second octet. What makes this easy? You only need to worry about the octet that has the largest block size (typically called the interesting octet; one that is something other than 0 or 255)—for example, 255.255.240.0 (/20) with a Class A network. The second octet has a block size of 1, so any number listed in that octet is a subnet. The third octet is a 240 mask, which means we have a block size of 16 in the third octet. If your host ID is 10.20.80.30, what is your subnet, broadcast address, and valid host range?

The subnet in the second octet is 20 with a block size of 1, but the third octet is in block sizes of 16, so we'll just count them out: 0, 16, 32, 48, 64, 80, 96...voilà! (By the way, you can count by 16s by now, right?) This makes our subnet 10.20.80.0, with a broadcast of 10.20.95.255 because the next subnet is 10.20.96.0. The valid host range is 10.20.80.1 through 10.20.95.254. And yes, no lie! You really can do this in your head if you just get your block sizes nailed!

Okay, let's practice on one more, just for fun!

Host IP: 10.1.3.65/23

First, you can't answer this question if you don't know what a /23, is. It's 255.255.254.0. The interesting octet here is the third one:  $256 - 254 = 2$ . Our subnets in the third octet are 0, 2, 4, 6, etc. The host in this question is in subnet 2.0, and the next subnet is 4.0, so that makes the broadcast address 3.255. And any address between 10.1.2.1 and 10.1.3.254 is considered a valid host.

# Variable Length Subnet Masks (VLSMs)

I could easily devote an entire chapter to *Variable Length Subnet Masks (VLSMs)*, but instead I'm going to show you a simple way to take one network and create many networks using subnet masks of different lengths on different types of network designs. This is called VLSM networking, and it does bring up another subject I mentioned at the beginning of this chapter: classful and classless networking.

Neither RIPv1 nor IGRP routing protocols have a field for subnet information, so the subnet information gets dropped. What this means is that if a router running RIP has a subnet mask of a certain value, it assumes that *all* interfaces within the classful address space have the same subnet mask. This is called classful routing, and RIP and IGRP are both considered classful routing protocols. (I'll be talking more about RIP and IGRP in Chapter 6, "IP Routing.") If you mix and match subnet mask lengths in a network running RIP or IGRP, that network just won't work!

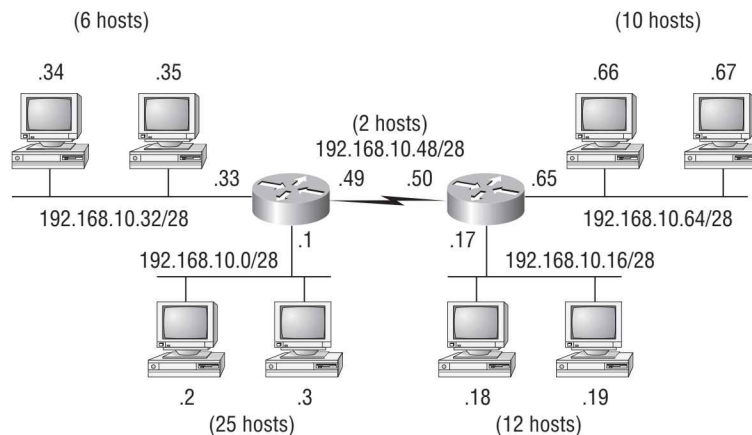
Classless routing protocols, however, do support the advertisement of subnet information. Therefore, you can use VLSM with routing protocols such as RIPv2, EIGRP, and OSPF. (EIGRP and OSPF will be discussed in Chapter 7.) The benefit of this type of network is that you save a bunch of IP address space with it.

As the name suggests, with VLSMs we can have different subnet masks for different router interfaces. Look at Figure 3.3 to see an example of why classful network designs are inefficient.

Looking at this figure, you'll notice that we have two routers, each with two LANs and connected together with a WAN serial link. In a typical classful network design (RIP or IGRP routing protocols), you could subnet a network like this:

192.168.10.0 = Network  
255.255.255.240 (/28) = Mask

**FIGURE 3.3** Typical classful network



Our subnets would be (you know this part, right?) 0, 16, 32, 48, 64, 80, etc. This allows us to assign 16 subnets to our internetwork. But how many hosts would be available on each network? Well, as you probably know by now, each subnet provides only 14 hosts. This means that each LAN has 14 valid hosts available—one LAN doesn't even have enough addresses needed for all the hosts! But the point-to-point WAN link also has 14 valid hosts. It's too bad we can't just nick some valid hosts from that WAN link and give them to our LANs!

All hosts and router interfaces have the same subnet mask—again, this is called classful routing. And if we want this network to be more efficient, we definitely need to add different masks to each router interface.

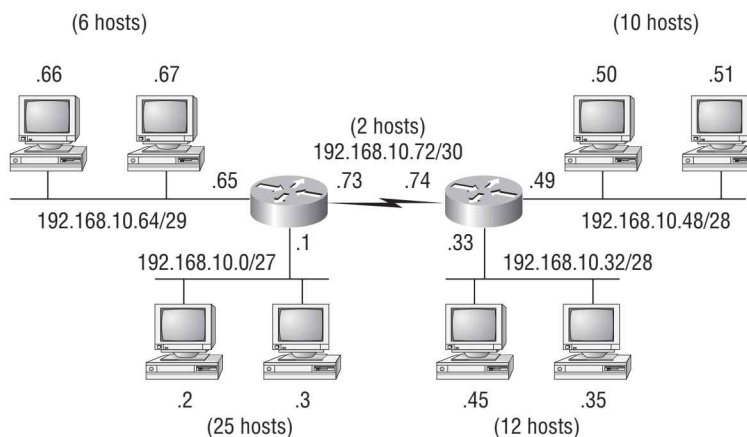
But there's still another problem—the link between the two routers will never use more than two valid hosts! This wastes valuable IP address space, and it's the big reason I'm going to talk to you about VLSM network design.

## VLSM Design

Let's take Figure 3.3 and use a classless design...which will become the new network shown in Figure 3.4. In the previous example, we wasted address space—one LAN didn't have enough addresses because every router interface and host used the same subnet mask. Not so good. What would be good is to provide only the needed number of hosts on each router interface. To do this, we use what are referred to as Variable Length Subnet Masks (VLSMs).

Now remember that we can use different size masks on each router interface. And if we use a /30 on our WAN links and a /27, /28, and /29 on our LANs, we'll get 2 hosts per WAN interface, and 30, 14, and 8 hosts per LAN interface—nice! This makes a huge difference—not only can we get just the right amount of hosts on each LAN, we still have room to add more WANs and LANs using this same network!

**FIGURE 3.4** Classless network design





Remember, in order to implement a VLSM design on your network, you need to have a routing protocol that sends subnet mask information with the route updates. This would be RIPv2, EIGRP, and OSPF. RIPv1 and IGRP will not work in classless networks and are considered classful routing protocols.



## Real World Scenario

### Why Bother with VLSM Design?

You have just been hired by a new company and need to add on to the existing network. There is no problem with starting over with a new IP address scheme. Should you use a VLSM classless network or a classful network?

Let's just say you happen to have plenty of address space because you are using the Class A 10.0.0.0 private network address in your corporate environment and can't even come close to imagining that you'd ever run out of IP addresses. Why would you want to bother with the VLSM design process?

Good question. There's a good answer too!

Because by creating contiguous blocks of addresses to specific areas of your network, you can then easily summarize your network and keep route updates with a routing protocol to a minimum. Why would anyone want to advertise hundreds of networks between buildings when you can just send one summary route between buildings and achieve the same result?

If you're confused about what summary routes are, let me explain. Summarization, also called supernetting, provides route updates in the most efficient way possible by advertising many routes in one advertisement instead of individually. This saves a ton of bandwidth and minimizes router processing. As always, you use blocks of addresses (remember that block sizes are used in all sorts of networks) to configure your summary routes and watch your network's performance hum.

But know that summarization works only if you design your network carefully. If you carelessly hand out IP subnets to any location on the network, you'll notice straight away that you no longer have any summary boundaries. And you won't get very far with creating summary routes without those, so watch your step!

## Implementing VLSM Networks

To create VLSMs quickly and efficiently, you need to understand how block sizes and charts work together to create the VLSM masks. Table 3.3 shows you the block sizes used when

creating VLSMs with Class C networks. For example, if you need 25 hosts, then you'll need a block size of 32. If you need 11 hosts, you'll use a block size of 16. Need 40 hosts? Then you'll need a block of 64. You cannot just make up block sizes—they've got to be the block sizes shown in Table 3.3. So memorize the block sizes in this table—it's easy. They're the same numbers we used with subnetting!

**TABLE 3.3** Block Sizes

Prefix	Mask	Hosts	Block Size
/25	128	126	128
/26	192	62	64
/27	224	30	32
/28	240	14	16
/29	248	6	8
/30	252	2	4

The next step is to create a VLSM table. Figure 3.5 shows you the table used in creating a VLSM network. The reason we use this table is so we don't accidentally overlap networks.

You'll find the sheet shown in Figure 3.5 very valuable because it lists every block size you can use for a network address. Notice that the block sizes are listed starting from a block size of 4 all the way to a block size of 128. If you have two networks with block sizes of 128, you'll quickly see that you can have only two networks. With a block size of 64, you can have only four networks, and so on, all the way to having 64 networks if you use only block sizes of 4. Remember that this takes into account that you are using the command `ip subnet-zero` in your network design.

Now, just fill in the chart in the lower-left corner, and then add the subnets to the worksheet and you're good to go.

So let's take what we've learned so far about our block sizes and VLSM table and create a VLSM using a Class C network address 192.168.10.0 for the network in Figure 3.6. Then fill out the VLSM table, as shown in Figure 3.7.

In Figure 3.6, we have four WAN links and four LANs connected together. We need to create a VLSM network that will allow us to save address space. Looks like we have two block sizes of 32, a block size of 16, and a block size of 8, and our WANs each have a block size of 4. Take a look and see how I filled out our VLSM chart in Figure 3.7.

**FIGURE 3.5** The VLSM table

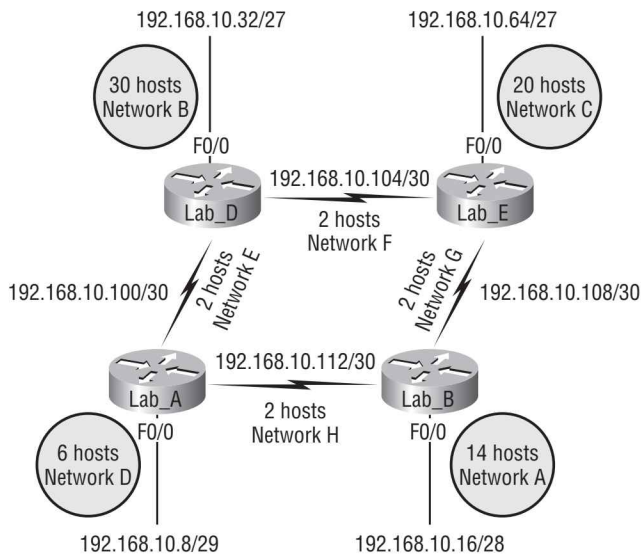
**Variable Length Subnet Masks Worksheet**

Subnet	Mask	Subnets	Hosts	Block
/26	192	4	62	64
/27	224	8	30	32
/28	240	16	14	16
/29	248	32	6	8
/30	252	64	2	4

0	_____
4	_____
8	_____
12	_____
16	_____
20	_____
24	_____
28	_____
32	_____
36	_____
40	_____
44	_____
48	_____
52	_____
56	_____
60	_____
64	_____
68	_____
72	_____
76	_____
80	_____
84	_____
88	_____
92	_____
96	_____
100	_____
104	_____
108	_____
112	_____
116	_____
120	_____
124	_____
128	_____
132	_____
136	_____
140	_____
144	_____
148	_____
152	_____
156	_____
160	_____
164	_____
168	_____
172	_____
176	_____
180	_____
184	_____
188	_____
192	_____
196	_____
200	_____
204	_____
208	_____
212	_____
216	_____
220	_____
224	_____
228	_____
232	_____
236	_____
240	_____
244	_____
248	_____
252	_____
256	_____

**Class C Network** 192.168.10.0

Network	Hosts	Block	Subnet	Mask
A				
B				
C				
D				
E				
F				
G				
H				
I				
J				
K				
L				

**FIGURE 3.6** VLSM network example 1

We still have plenty of room for growth with this VLSM network design.

We never could accomplish that with one subnet mask using classful routing. Let's do another one. Figure 3.8 shows a network with 11 networks, two block sizes of 64, one of 32, five of 16, and three of 4.

First, create your VLSM table and use your block size chart to fill in the table with the subnets you need. Figure 3.9 shows a possible solution.

Notice that we filled in this entire chart and only have room for one more block size of 4! Only with a VLSM network can you provide this type of address space savings.

Keep in mind that it doesn't matter where you start your block sizes as long as you always count from zero. For example, if you had a block size of 16, you must start at 0 and count from there—0, 16, 32, 48, etc. You can't start a block size of 16 from, say, 40 or anything other than increments of 16.

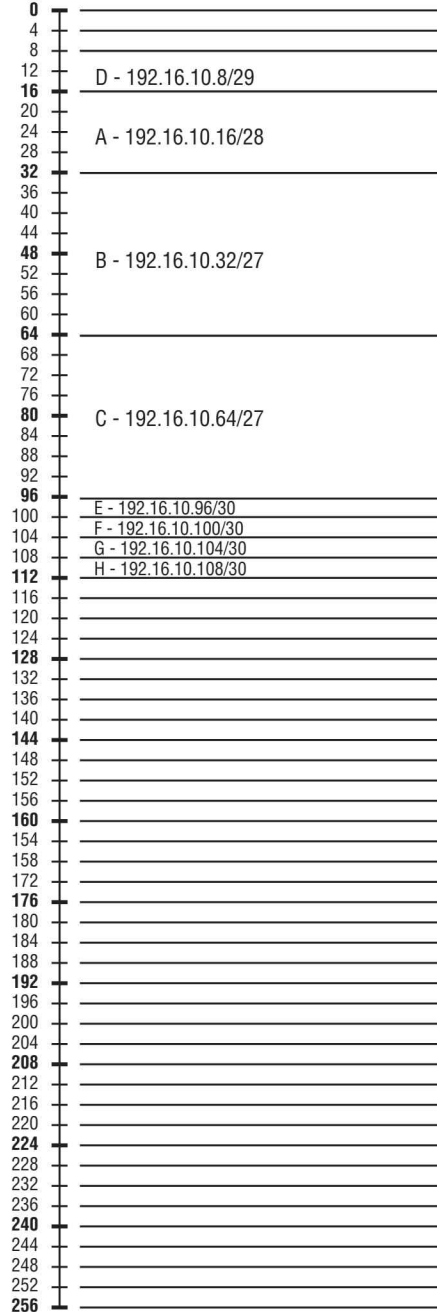
Here's another example. If you had block sizes of 32, you must start at zero like this: 0, 32, 64, 96, etc. Just remember that you don't get to start wherever you want; you must always start counting from zero. In the example in Figure 3.9, I started at 64 and 128, with my two block sizes of 64. I didn't have much choice, because my options are 0, 64, 128, and 192. However, I added the block size of 32, 16, 8, and 4 wherever I wanted just as long as they were in the correct increments of that block size.

Okay—you have three locations you need to address, and the IP network you have received is 192.168.55.0 to use as the addressing for the entire network. You'll use `ip subnet-zero` and RIPv2 as the routing protocol. (RIPv2 supports VLSM networks, RIPv1 does not—both of them will be discussed in Chapter 6.) Figure 3.10 shows the network diagram and the IP address of the RouterA S0/0 interface.

**FIGURE 3.7** A VLSM table, example one

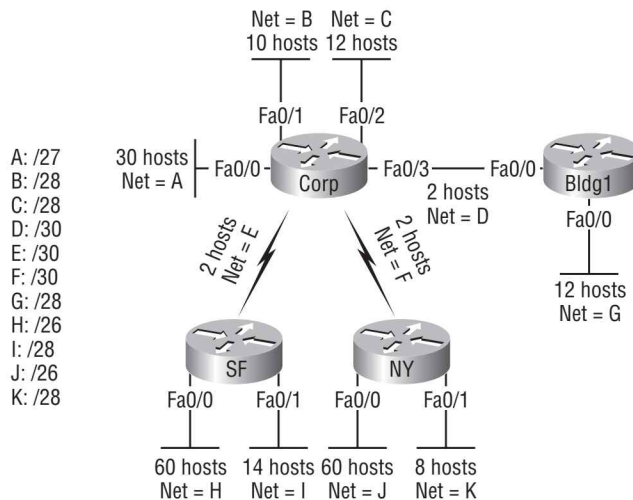
**Variable Length Subnet Masks Worksheet**

Subnet	Mask	Subnets	Hosts	Block
/26	192	4	62	64
/27	224	8	30	32
/28	240	16	14	16
/29	248	32	6	8
/30	252	64	2	4



**Class C Network** 192.16.10.0

Network	Hosts	Block	Subnet	Mask
A	12	16	/28	240
B	20	32	/27	224
C	25	32	/27	224
D	4	8	/29	248
E	2	4	/30	252
F	2	4	/30	252
G	2	4	/30	252
H	2	4	/30	252

**FIGURE 3.8** VLSM network example 2

From the list of IP addresses on the right of the figure, which IP address will be placed in each router's FastEthernet 0/0 interface and serial 0/1 of RouterB?

To answer this question, first look for clues in Figure 3.10. The first clue is that interface S0/0 on RouterA has IP address 192.168.55.2/30 assigned, which makes for an easy answer. A /30, as you know, is 255.255.255.252, which gives you a block size of 4. Your subnets are 0, 4, 8, etc. Since the known host has an IP address of 2, the only other valid host in the zero subnet is 1, so the third answer down is what you want for the s0/1 interface of RouterB.

The next clues are the listed number of hosts for each of the LANs. RouterA needs 7 hosts, a block size of 16 (/28); RouterB needs 90 hosts, a block size of 128 (/25); and RouterC needs 23 hosts, a block size of 32 (/27).

Figure 3.11 shows the answers to this question.

Once you figured out the block size needed for each LAN, this was actually a pretty simple question—all you need to do is look for the right clues and, of course, know your block sizes.

One last example of VLSM design before we move on to summarization. Figure 3.12 shows three routers, all running RIPv2. Which class C addressing scheme would you use to satisfy the needs of this network yet save as much address space as possible?

This is a really sweet network, just waiting for you to fill out the chart. There are block sizes of 64, 32, and 16 and two block sizes of 4. This should be a slam dunk for you. Take a look at my answer in Figure 3.13.

This is what I did: Starting at subnet 0, I used the block size of 64. (I didn't have to—I could have started with a block size of 4, but I usually like to start with the largest block size and move to the smallest.) Okay, then I added the block sizes of 32 and 16 and the two block sizes of 4. There's still a lot of room to add subnets to this network—very cool!

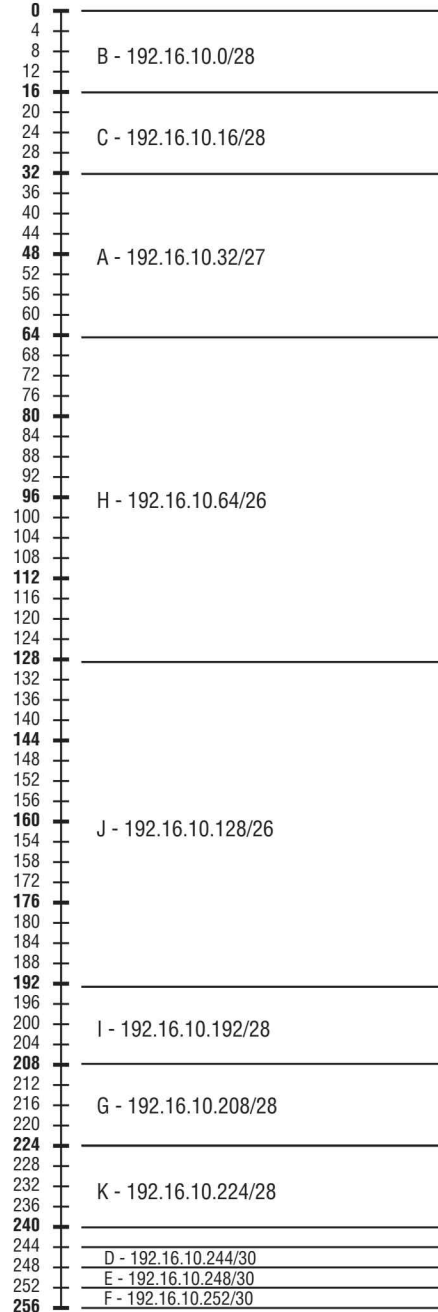
**FIGURE 3.9** VLSM table example 2

**Variable Length Subnet Masks Worksheet**

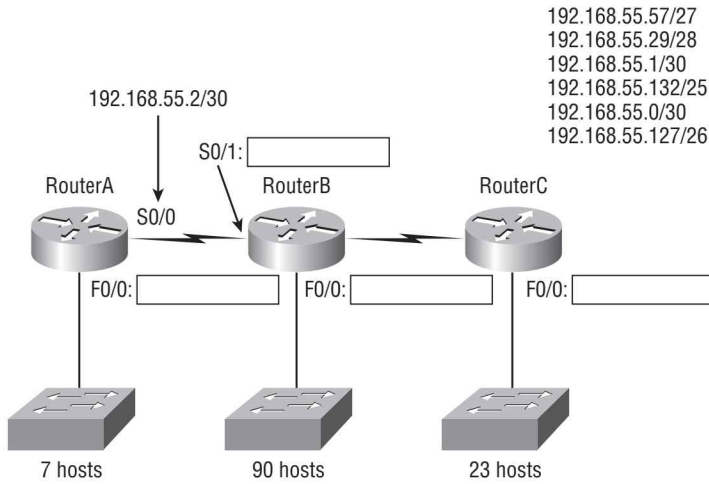
Subnet	Mask	Subnets	Hosts	Block
/26	192	4	62	64
/27	224	8	30	32
/28	240	16	14	16
/29	248	32	6	8
/30	252	64	2	4

**Class C Network** 192.168.10.0

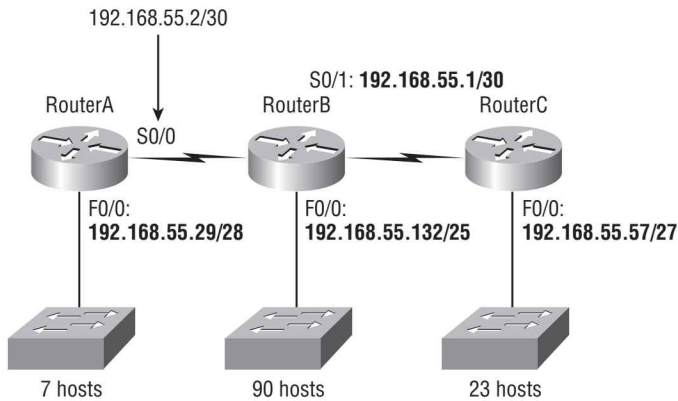
Network	Hosts	Block	Subnet	Mask
A	30	32	32	224
B	10	16	0	240
C	12	16	16	240
D	2	4	244	252
E	2	4	248	252
F	2	4	252	252
G	12	16	208	240
H	60	64	64	192
I	14	16	192	240
J	60	64	128	192
K	8	16	224	240



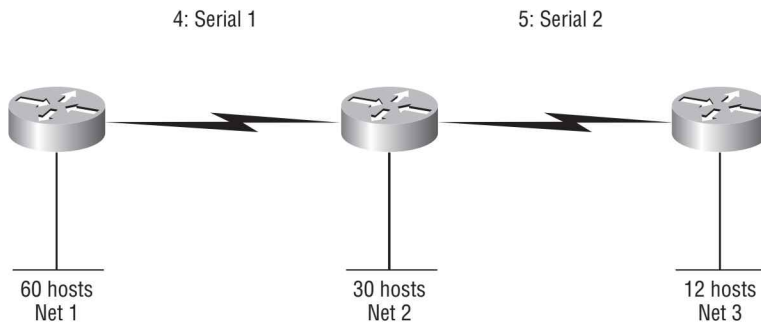
**FIGURE 3.10** VLSM design example 1



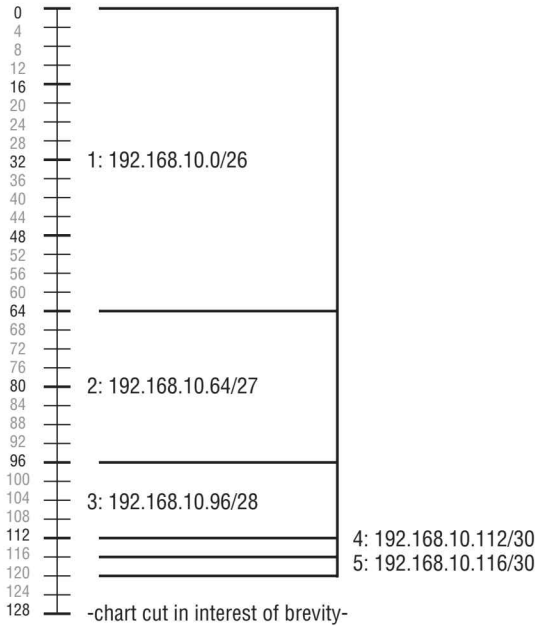
**FIGURE 3.11** Solution to VLSM design example 1



**FIGURE 3.12** VLSM design example 2



**FIGURE 3.13** Solution to VLSM design example 2

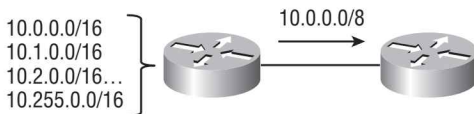


## Summarization

Summarization, also called route aggregation, allows routing protocols to advertise many networks as one address. The purpose of this is to reduce the size of routing tables on routers to save memory, which also shortens the amount of time for IP to parse the routing table and find the path to a remote network.

Figure 3.14 shows how a summary address would be used in an internetwork.

**FIGURE 3.14** Summary address used in an internetwork



Summarization is actually somewhat simple because all you really need to have down are the block sizes that we just used in learning subnetting and VLSM design. For example, if you wanted to summarize the following networks into one network advertisement, you just have to find the block size first; then you can easily find your answer:

192.168.16.0 through network 192.168.31.0

What's the block size? There are exactly 16 Class C networks, so this neatly fits into a block size of 16.

Okay, now that you know the block size, you can find the network address and mask used to summarize these networks into one advertisement. The network address used to advertise the summary address is always the first network address in the block—in this example, 192.168.16.0. To figure out a summary mask, in this same example, what mask is used to get a block size of 16? Yes, 240 is correct. This 240 would be placed in the third octet—the octet where we are summarizing. So, the mask would be 255.255.240.0.



You'll learn how to apply these summary addresses to a router in Chapter 7.

Here's another example:

Networks 172.16.32.0 through 172.16.50.0

This is not as clean as the previous example because there are two possible answers, and here's why: Since you're starting at network 32, your options for block sizes are 4, 8, 16, 32, 64, etc., and block sizes of 16 and 32 could work as this summary address.

- *Answer #1:* If you used a block size of 16, then the network address is 172.16.32.0 with a mask of 255.255.240.0 (240 provides a block of 16). However, this only summarizes from 32 to 47, which means that networks 48 through 50 would be advertised as single networks. This is probably the best answer, but that depends on your network design. Let's look at the next answer.
- *Answer #2:* If you used a block size of 32, then your summary address would still be 172.16.32.0, but the mask would be 255.255.224.0 (224 provides a block of 32). The possible problem with this answer is that it will summarize networks 32 to 63 and we only have networks 32 to 50. No worries if you're planning on adding networks 51 to 63 later into the same network, but you could have serious problems in your internetwork if somehow networks 51 to 63 were to show up and be advertised from somewhere else in your network! This is the reason why answer number one is the safest answer.

Let's take a look at another example, but let's look at it from a host's perspective.

Your summary address is 192.168.144.0/20—what's the range of host addresses that would be forwarded according to this summary? The /20 provides a summary address of 192.168.144.0 and mask of 255.255.240.0.

The third octet has a block size of 16, and starting at summary address 144, the next block of 16 is 160, so our network summary range is 144 to 159 in the third octet (again, you *must* be able to count in 16s!).

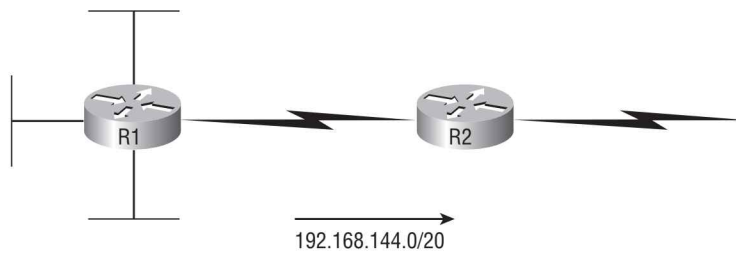
A router that has this summary address in the routing table will forward any packet with destination IP addresses of 192.168.144.1 through 192.168.159.254.

Only two more summarization examples, then we'll move on to troubleshooting.

In Figure 3.15, the Ethernet networks connected to router R1 are being summarized to R2 as 192.168.144.0/20. Which range of IP addresses will R2 forward to R1 according to this summary?

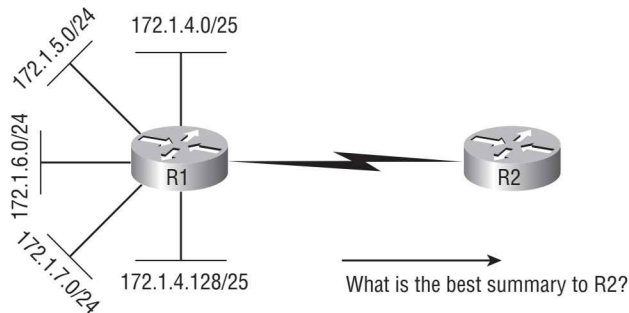
**FIGURE 3.15** Summarization example 1

The Ethernet networks connected to router R1 are being summarized to R2 as 192.168.144.0/20. Which IP addresses will R2 forward to R1 according to this summary?



No worries—this is really an easier question than it looks. The question actually has the summary address listed: 192.168.144.0/20. You already know that /20 is 255.255.240.0, which means you’ve got a block size of 16 in the third octet. Starting at 144 (this is also right there in the question), the next block size of 16 is 160, so you can’t go above 159 in the third octet. The IP addresses that will be forwarded are 192.168.144.1 through 192.168.159.255. (Yes, the broadcast address is forwarded.)

Okay, last one. In Figure 3.16, there are five networks connected to router R1. What’s the best summary address to R2?

**FIGURE 3.16** Summarization example 2

I’m going to be honest—this is a much harder question than the one in Figure 3.15. You’re going to have to look pretty hard to see the answer. The first thing to do with this is to write down all the networks and see if you can find anything in common with all six:

- 172.1.4.128/25
- 172.1.7.0/24
- 172.1.6.0/24
- 172.1.5.0/24
- 172.1.4.0/24

Do you see an octet that looks interesting to you? I do. It's the third octet. 4, 5, 6, 7, and yes, it's a block size of 4. So you can summarize 172.1.4.0 using a mask of 255.255.252.0, which means you will use a block size of 4 in the third octet. The IP addresses forwarded with this summary are 172.1.4.1 through 172.1.7.255.

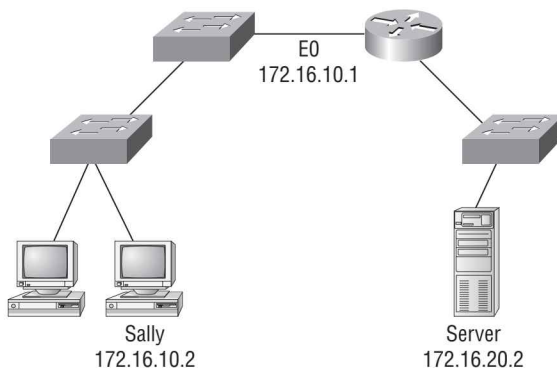
Now to summarize this summarization section: Basically, if you've nailed down your block sizes, then finding and applying summary addresses and masks is actually fairly easy. But you're going to get bogged down pretty quickly if you don't know what a /20 is or if you can't count by 16s!

## Troubleshooting IP Addressing

Troubleshooting IP addressing is obviously an important skill because running into trouble somewhere along the way is pretty much a sure thing, and it's going to happen to you. No—I'm not a pessimist; I'm just keeping it real. Because of this nasty fact, it will be great when you can save the day because you can both figure out (diagnose) the problem and fix it on an IP network whether you're at work or at home!

So this is where I'm going to show you the “Cisco way” of troubleshooting IP addressing. Let's use Figure 3.17 as an example of your basic IP trouble—poor Sally can't log in to the Windows server. Do you deal with this by calling the Microsoft team to tell them their server is a pile of junk and causing all your problems? Probably not such a great idea—let's first double-check our network instead.

**FIGURE 3.17** Basic IP troubleshooting



Okay let's get started by going over the troubleshooting steps that Cisco follows. They're pretty simple, but important nonetheless. Pretend you're at a customer host and they're complaining that they can't communicate to a server that just happens to be on a remote network. Here are the four troubleshooting steps Cisco recommends:

1. Open a DOS window and ping 127.0.0.1. This is the diagnostic, or loopback, address, and if you get a successful ping, your IP stack is considered to be initialized. If it fails, then you have an IP stack failure and need to reinstall TCP/IP on the host.

```
C:\>ping 127.0.0.1
Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Ping statistics for 127.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

2. From the DOS window, ping the IP address of the local host. If that's successful, your network interface card (NIC) is functioning. If it fails, there is a problem with the NIC. Success here doesn't mean that a cable is plugged into the NIC, only that the IP protocol stack on the host can communicate to the NIC (via the LAN driver).

```
C:\>ping 172.16.10.2
Pinging 172.16.10.2 with 32 bytes of data:
Reply from 172.16.10.2: bytes=32 time<1ms TTL=128
Reply from 172.16.10.2: bytes=32 time<1ms TTL=128
Reply from 172.16.10.2: bytes=32 time<1ms TTL=128
Reply from 172.16.10.2: bytes=32 time<1ms TTL=128
Ping statistics for 172.16.10.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

3. From the DOS window, ping the default gateway (router). If the ping works, it means that the NIC is plugged into the network and can communicate on the local network. If it fails, you have a local physical network problem that could be anywhere from the NIC to the router.

```
C:\>ping 172.16.10.1
Pinging 172.16.10.1 with 32 bytes of data:
Reply from 172.16.10.1: bytes=32 time<1ms TTL=128
Reply from 172.16.10.1: bytes=32 time<1ms TTL=128
Reply from 172.16.10.1: bytes=32 time<1ms TTL=128
Reply from 172.16.10.1: bytes=32 time<1ms TTL=128
```

```
Ping statistics for 172.16.10.1:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
  Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

4. If steps 1 through 3 were successful, try to ping the remote server. If that works, then you know that you have IP communication between the local host and the remote server. You also know that the remote physical network is working.

```
C:\>ping 172.16.20.2
Pinging 172.16.20.2 with 32 bytes of data:
Reply from 172.16.20.2: bytes=32 time<1ms TTL=128
Reply from 172.16.20.2: bytes=32 time<1ms TTL=128
Reply from 172.16.20.2: bytes=32 time<1ms TTL=128
Reply from 172.16.20.2: bytes=32 time<1ms TTL=128
Ping statistics for 172.16.20.2:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
  Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

If the user still can't communicate with the server after steps 1 through 4 are successful, you probably have some type of name resolution problem and need to check your Domain Name System (DNS) settings. But if the ping to the remote server fails, then you know you have some type of remote physical network problem and need to go to the server and work through steps 1 through 3 until you find the snag.

Before we move on to determining IP address problems and how to fix them, I just want to mention some basic DOS commands that you can use to help troubleshoot your network from both a PC and a Cisco router (the commands might do the same thing, but they are implemented differently).

**Packet InterNet Groper (ping)** Uses ICMP echo request and replies to test if a node IP stack is initialized and alive on the network.

**tracert** Displays the list of routers on a path to a network destination by using TTL time-outs and ICMP error messages. This command will not work from a DOS prompt.

**tracert** Same command as **tracert**, but it's a Microsoft Windows command and will not work on a Cisco router.

**arp -a** Displays IP-to-MAC-address mappings on a Windows PC.

**show ip arp** Same command as **arp -a**, but displays the ARP table on a Cisco router. Like the commands **tracert** and **tracert**, they are not interchangeable through DOS and Cisco.

**ipconfig /all** Used only from a DOS prompt, shows you the PC network configuration.

Once you've gone through all these steps and used the appropriate DOS commands, if necessary, what do you do if you find a problem? How do you go about fixing an IP address configuration error? Let's move on and discuss how to determine the IP address problems and how to fix them.

## Determining IP Address Problems

It's common for a host, router, or other network device to be configured with the wrong IP address, subnet mask, or default gateway. Because this happens way too often, I'm going to teach you how to both determine and fix IP address configuration errors.

Once you've worked through the four basic steps of troubleshooting and determined there's a problem, you obviously then need to find and fix it. It really helps to draw out the network and IP addressing scheme. If it's already done, consider yourself lucky and go buy a lottery ticket, because although it should be done, it rarely is. And if it is, it's usually outdated or inaccurate anyway. Typically it is not done, and you'll probably just have to bite the bullet and start from scratch.

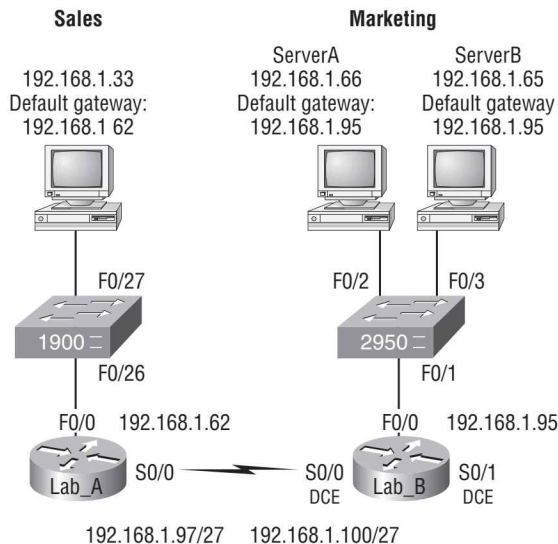


I'll show you how to draw out your network using CDP in Chapter 5, "Managing the Cisco IOS."

Once you have your network accurately drawn out, including the IP addressing scheme, you need to verify each host's IP address, mask, and default gateway address to determine the problem. (I'm assuming that you don't have a physical problem or that if you did, you've already fixed it.)

Let's check out the example illustrated in Figure 3.18. A user in the sales department calls and tells you that she can't get to ServerA in the marketing department. You ask her if she can get to ServerB in the marketing department, but she doesn't know because she doesn't have rights to log on to that server. What do you do?

**FIGURE 3.18** IP address problem 1

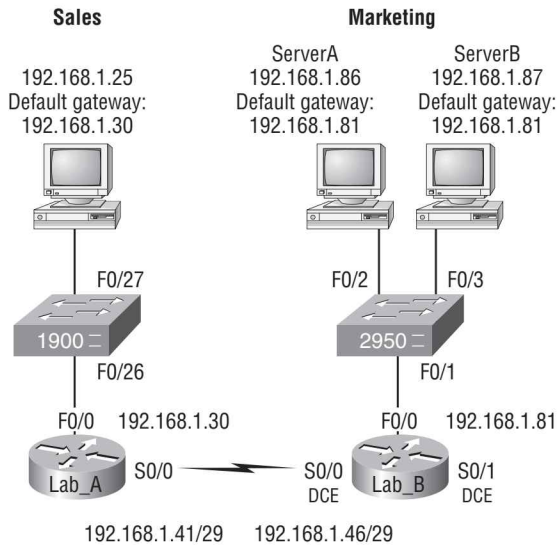


You ask the client to go through the four troubleshooting steps that you learned about in the preceding section. Steps 1 through 3 work, but step 4 fails. By looking at the figure, can you determine the problem? Look for clues in the network drawing. First, the WAN link between the Lab\_A router and the Lab\_B router shows the mask as a /27. You should already know that this mask is 255.255.255.224 and then determine that all networks are using this mask. The network address is 192.168.1.0. What are our valid subnets and hosts?  $256 - 224 = 32$ , so this makes our subnets 32, 64, 96, 128, etc. So, by looking at the figure, you can see that subnet 32 is being used by the sales department, the WAN link is using subnet 96, and the marketing department is using subnet 64.

Now you've got to determine what the valid host ranges are for each subnet. From what you learned at the beginning of this chapter, you should now be able to easily determine the subnet address, broadcast addresses, and valid host ranges. The valid hosts for the Sales LAN are 33 through 62—the broadcast address is 63 because the next subnet is 64, right? For the Marketing LAN, the valid hosts are 65 through 94 (broadcast 95), and for the WAN link, 97 through 126 (broadcast 127). By looking at the figure, you can determine that the default gateway on the Lab\_B router is incorrect. That address is the broadcast address of the 64 subnet, so there's no way it could be a valid host.

Did you get all that? Maybe we should try another one, just to make sure. Figure 3.19 shows a network problem. A user in the Sales LAN can't get to ServerB. You have the user run through the four basic troubleshooting steps and find that the host can communicate to the local network but not to the remote network. Find and define the IP addressing problem.

**FIGURE 3.19** IP address problem 2



If you use the same steps used to solve the last problem, you can see first that the WAN link again provides the subnet mask to use— /29, or 255.255.255.248. You need to determine what the valid subnets, broadcast addresses, and valid host ranges are to solve this problem.

The 248 mask is a block size of 8 ( $256 - 248 = 8$ ), so the subnets both start and increment in multiples of 8. By looking at the figure, you see that the Sales LAN is in the 24 subnet, the WAN is in the 40 subnet, and the Marketing LAN is in the 80 subnet. Can you see the problem yet? The valid host range for the Sales LAN is 25–30, and the configuration appears correct. The valid host range for the WAN link is 41–46, and this also appears correct. The valid host range for the 80 subnet is 81–86, with a broadcast address of 87 because the next subnet is 88. ServerB has been configured with the broadcast address of the subnet.

Okay, now that you can figure out misconfigured IP addresses on hosts, what do you do if a host doesn't have an IP address and you need to assign one? What you need to do is look at other hosts on the LAN and figure out the network, mask, and default gateway. Let's take a look at a couple of examples of how to find and apply valid IP addresses to hosts.

You need to assign a server and router IP addresses on a LAN. The subnet assigned on that segment is 192.168.20.24/29, and the router needs to be assigned the first usable address and the server the last valid host ID. What are the IP address, mask, and default gateway assigned to the server?

To answer this, you must know that a /29 is a 255.255.255.248 mask, which provides a block size of 8. The subnet is known as 24, the next subnet in a block of 8 is 32, so the broadcast address of the 24 subnet is 31, which makes the valid host range 25–30.

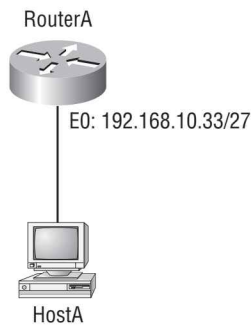
Server IP address: 192.168.20.30

Server mask: 255.255.255.248

Default gateway: 192.168.20.25 (router's IP address)

As another example, let's take a look at Figure 3.20 and solve this problem.

**FIGURE 3.20** Find the valid host #1.



Look at the router's IP address on Ethernet0. What IP address, subnet mask, and valid host range could be assigned to the host?

The IP address of the router's Ethernet0 is 192.168.10.33/27. As you already know, a /27 is a 224 mask with a block size of 32. The router's interface is in the 32 subnet. The next subnet is 64, so that makes the broadcast address of the 32 subnet 63 and the valid host range 33–62.

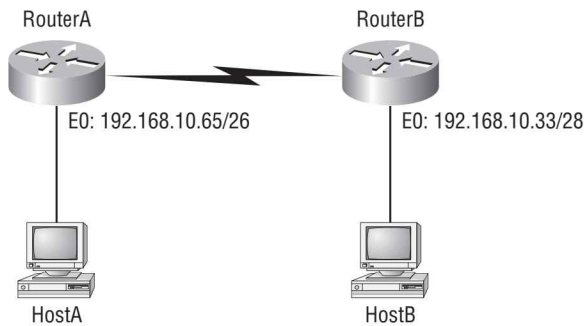
Host IP address: 192.168.10.34–62 (any address in the range except for 33, which is assigned to the router)

Mask: 255.255.255.224

Default gateway: 192.168.10.33

Figure 3.21 shows two routers with Ethernet configurations already assigned. What are the host addresses and subnet masks of hosts A and B?

**FIGURE 3.21** Find the valid host #2



RouterA has an IP address of 192.168.10.65/26 and RouterB has an IP address of 192.168.10.33/28. What are the host configurations? RouterA Ethernet0 is in the 192.168.10.64 subnet and RouterB Ethernet0 is in the 192.168.10.32 network.

Host A IP address: 192.168.10.66–126

Host A mask: 255.255.255.192

Host A default gateway: 192.168.10.65

Host B IP address: 192.168.10.34–46

Host B mask: 255.255.255.240

Host B default gateway: 192.168.10.33

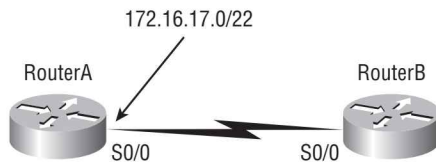
Just a couple more examples and then this chapter is history. Hang in there!

Figure 3.22 shows two routers; you need to configure the S0/0 interface on RouterA. The network assigned to the serial link is 172.16.17.0/22. What IP address can be assigned?

First, you must know that a /22 CIDR is 255.255.252.0, which makes a block size of 4 in the third octet. Since 17 is listed, the available range is 16.1 through 19.254; so, for example, the IP address S0/0 could be 172.16.18.255 since that's within the range.

Okay, last one! You have one Class C network ID and you need to provide one usable subnet per city while allowing enough usable host addresses for each city specified in Figure 3.23. What is your mask?

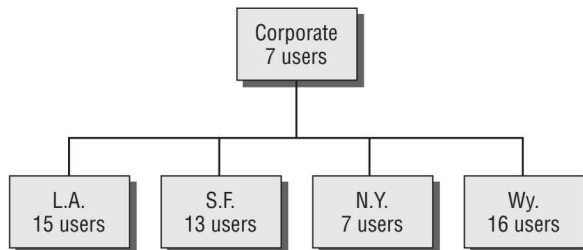
**FIGURE 3.22** Find the valid host address #3



Actually, this is probably the easiest thing you've done all day! I count 5 subnets needed and the Wyoming office needs 16 users (always look for the network that needs the most hosts). What block size is needed for the Wyoming office? 32. (Remember, you cannot use a block size of 16 because you always have to subtract 2!) What mask provides you with a block size of 32? 224. Bingo! This provides 8 subnets, each with 30 hosts.

You're done, the diva has sung, the chicken has crossed the road...whew! Okay, take a good break (but skip the shot and the beer for now), then come back and go through the written labs and review questions.

**FIGURE 3.23** Find the valid subnet mask.



## Summary

Did you read Chapters 2 and 3 and understand everything on the first pass? If so, that is fantastic—congratulations! The thing is, you probably got lost a couple of times—and as I told you, that's what usually happens, so don't stress. Don't feel bad if you have to read each chapter more than once, or even 10 times, before you're truly good to go.

This chapter provided you with an important understanding of IP subnetting. After reading this chapter, you should be able to subnet IP addresses in your head. You should also know how to design and implement simple VLSM networks.

You should also understand the Cisco troubleshooting methods. You must remember the four steps that Cisco recommends you take when trying to narrow down exactly where a network/IP addressing problem is and then know how to proceed systematically in order to fix it. In addition, you should be able to find valid IP addresses and subnet masks by looking at a network diagram.

## Exam Essentials

**Remember the steps to subnet in your head.** Understand how IP addressing and subnetting work. First, determine your block size by using the 256-subnet mask math. Then count your subnets and determine the broadcast address of each subnet—it is always the number right before the next subnet. Your valid hosts are the numbers between the subnet address and the broadcast address.

**Understand the various block sizes.** This is an important part of understanding IP addressing and subnetting. The valid block sizes are always 4, 8, 16, 32, 64, 128, etc. You can determine your block size by using the 256-subnet mask math.

**Remember the four diagnostic steps.** The four simple steps that Cisco recommends for troubleshooting are ping the loopback address, ping the NIC, ping the default gateway, and ping the remote device.

**You must be able to find and fix an IP addressing problem.** Once you go through the four troubleshooting steps that Cisco recommends, you must be able to determine the IP addressing problem by drawing out the network and finding the valid and invalid hosts addressed in your network.

**Understand the troubleshooting tools that you can use from your host and a Cisco router** ping 127.0.0.1 tests your local IP stack. `tracert` is a Windows DOS command to track the path a packet takes through an internetwork to a destination. Cisco routers use the command `traceroute`, or just `trace` for short. Don't confuse the Windows and Cisco commands. Although they produce the same output, they don't work from the same prompts. `ipconfig /all` will display your PC network configuration from a DOS prompt, and `arp -a` (again from a DOS prompt) will display IP-to-MAC-address mapping on a Windows PC.

## Written Labs 3

In this section, you'll complete the following labs to make sure you've got the information and concepts contained within them fully dialed in:

- Lab 3.1: Written Subnet Practice #1
- Lab 3.2: Written Subnet Practice #2
- Lab 3.3: Written Subnet Practice #3

*(The answers to the written labs can be found following the answers to the review questions for this chapter.)*

## Written Lab 3.1: Written Subnet Practice #1

Write the subnet, broadcast address, and valid host range for question 1 through question 6:

1. 192.168.100.25/30
2. 192.168.100.37/28
3. 192.168.100.66/27
4. 192.168.100.17/29
5. 192.168.100.99/26
6. 192.168.100.99/25
7. You have a Class B network and need 29 subnets. What is your mask?
8. What is the broadcast address of 192.168.192.10/29?
9. How many hosts are available with a Class C /29 mask?
10. What is the subnet for host ID 10.16.3.65/23?

## Written Lab 3.2: Written Subnet Practice

Given a Class B network and the net bits identified (CIDR), complete the following table to identify the subnet mask and the number of host addresses possible for each mask.

Classful Address	Subnet Mask	Number of Hosts per Subnet ( $2^x - 2$ )
/16		
/17		
/18		
/19		
/20		
/21		
/22		
/23		
/24		
/25		
/26		
/27		
/28		
/29		
/30		

## Written Lab 3.3: Written Subnet Practice

Decimal IP Address	Address Class	Number of Subnet and Host Bits	Number of Subnets ( $2^x$ )	Number of Hosts ( $2^x - 2$ )
10.25.66.154/23				
172.31.254.12/24				
192.168.20.123/28				
63.24.89.21/18				
128.1.1.254/20				
208.100.54.209/30				

*(The answers to the written labs can be found following the answers to the review questions for this chapter.)*

# Review Questions

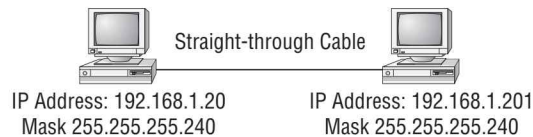


The following questions are designed to test your understanding of this chapter's material. For more information on how to get additional questions, please see this book's Introduction.

1. What is the maximum number of IP addresses that can be assigned to hosts on a local subnet that uses the 255.255.255.224 subnet mask?
  - A. 14
  - B. 15
  - C. 16
  - D. 30
  - E. 31
  - F. 62
2. You have a network that needs 29 subnets while maximizing the number of host addresses available on each subnet. How many bits must you borrow from the host field to provide the correct subnet mask?
  - A. 2
  - B. 3
  - C. 4
  - D. 5
  - E. 6
  - F. 7
3. What is the subnetwork address for a host with the IP address 200.10.5.68/28?
  - A. 200.10.5.56
  - B. 200.10.5.32
  - C. 200.10.5.64
  - D. 200.10.5.0
4. The network address of 172.16.0.0/19 provides how many subnets and hosts?
  - A. 7 subnets, 30 hosts each
  - B. 7 subnets, 2,046 hosts each
  - C. 7 subnets, 8,190 hosts each
  - D. 8 subnets, 30 hosts each
  - E. 8 subnets, 2,046 hosts each
  - F. 8 subnets, 8,190 hosts each

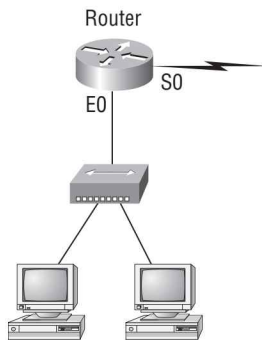
5. Which two statements describe the IP address 10.16.3.65/23? (Choose two.)
  - A. The subnet address is 10.16.3.0 255.255.254.0.
  - B. The lowest host address in the subnet is 10.16.2.1 255.255.254.0.
  - C. The last valid host address in the subnet is 10.16.2.254 255.255.254.0.
  - D. The broadcast address of the subnet is 10.16.3.255 255.255.254.0.
  - E. The network is not subnetted.
6. If a host on a network has the address 172.16.45.14/30, what is the subnetwork this host belongs to?
  - A. 172.16.45.0
  - B. 172.16.45.4
  - C. 172.16.45.8
  - D. 172.16.45.12
  - E. 172.16.45.16
7. On a VLSM network, which mask should you use on point-to-point WAN links in order to reduce the waste of IP addresses?
  - A. /27
  - B. /28
  - C. /29
  - D. /30
  - E. /31
8. What is the subnetwork number of a host with an IP address of 172.16.66.0/21?
  - A. 172.16.36.0
  - B. 172.16.48.0
  - C. 172.16.64.0
  - D. 172.16.0.0
9. You have an interface on a router with the IP address of 192.168.192.10/29. Including the router interface, how many hosts can have IP addresses on the LAN attached to the router interface?
  - A. 6
  - B. 8
  - C. 30
  - D. 62
  - E. 126

10. You need to configure a server that is on the subnet 192.168.19.24/29. The router has the first available host address. Which of the following should you assign to the server?
- A. 192.168.19.0 255.255.255.0
  - B. 192.168.19.33 255.255.255.240
  - C. 192.168.19.26 255.255.255.248
  - D. 192.168.19.31 255.255.255.248
  - E. 192.168.19.34 255.255.255.240
11. You have an interface on a router with the IP address of 192.168.192.10/29. What is the broadcast address the hosts will use on this LAN?
- A. 192.168.192.15
  - B. 192.168.192.31
  - C. 192.168.192.63
  - D. 192.168.192.127
  - E. 192.168.192.255
12. You need to subnet a network that has 5 subnets, each with at least 16 hosts. Which classful subnet mask would you use?
- A. 255.255.255.192
  - B. 255.255.255.224
  - C. 255.255.255.240
  - D. 255.255.255.248
13. A network administrator is connecting hosts A and B directly through their Ethernet interfaces, as shown in the illustration. Ping attempts between the hosts are unsuccessful. What can be done to provide connectivity between the hosts? (Choose two.)



- A. A crossover cable should be used in place of the straight-through cable.
- B. A rollover cable should be used in place of the straight-through cable.
- C. The subnet masks should be set to 255.255.255.192.
- D. A default gateway needs to be set on each host.
- E. The subnet masks should be set to 255.255.255.0.

14. If an Ethernet port on a router were assigned an IP address of 172.16.112.1/25, what would be the valid subnet address of this host?
- A. 172.16.112.0
  - B. 172.16.0.0
  - C. 172.16.96.0
  - D. 172.16.255.0
  - E. 172.16.128.0
15. Using the following illustration, what would be the IP address of E0 if you were using the eighth subnet? The network ID is 192.168.10.0/28 and you need to use the last available IP address in the range. The zero subnet should not be considered valid for this question.



- A. 192.168.10.142
  - B. 192.168.10.66
  - C. 192.168.100.254
  - D. 192.168.10.143
  - E. 192.168.10.126
16. Using the illustration from the previous question, what would be the IP address of S0 if you were using the first subnet? The network ID is 192.168.10.0/28 and you need to use the last available IP address in the range. Again, the zero subnet should not be considered valid for this question.
- A. 192.168.10.24
  - B. 192.168.10.62
  - C. 192.168.10.30
  - D. 192.168.10.127

17. Which configuration command must be in effect to allow the use of 8 subnets if the Class C subnet mask is 255.255.255.224?
- A. Router(config)#ip classless
  - B. Router(config)#ip version 6
  - C. Router(config)#no ip classful
  - D. Router(config)#ip unnumbered
  - E. Router(config)#ip subnet-zero
  - F. Router(config)#ip all-nets
18. You have a network with a subnet of 172.16.17.0/22. Which is the valid host address?
- A. 172.16.17.1 255.255.255.252
  - B. 172.16.0.1 255.255.240.0
  - C. 172.16.20.1 255.255.254.0
  - D. 172.16.16.1 255.255.255.240
  - E. 172.16.18.255 255.255.252.0
  - F. 172.16.0.1 255.255.255.0
19. Your router has the following IP address on Ethernet0: 172.16.2.1/23. Which of the following can be valid host IDs on the LAN interface attached to the router? (Choose two.)
- A. 172.16.0.5
  - B. 172.16.1.100
  - C. 172.16.1.198
  - D. 172.16.2.255
  - E. 172.16.3.0
  - F. 172.16.3.255
20. To test the IP stack on your local host, which IP address would you ping?
- A. 127.0.0.0
  - B. 1.0.0.127
  - C. 127.0.0.1
  - D. 127.0.0.255
  - E. 255.255.255.255

## Answers to Review Questions

1. D. A /27 (255.255.255.224) is 3 bits on and 5 bits off. This provides 8 subnets, each with 30 hosts. Does it matter if this mask is used with a Class A, B, or C network address? Not at all. The number of host bits would never change.
2. D. A 240 mask is 4 subnet bits and provides 16 subnets, each with 14 hosts. We need more subnets, so let's add subnet bits. One more subnet bit would be a 248 mask. This provides 5 subnet bits (32 subnets) with 3 host bits (6 hosts per subnet). This is the best answer.
3. C. This is a pretty simple question. A /28 is 255.255.255.240, which means that our block size is 16 in the fourth octet. 0, 16, 32, 48, 64, 80, etc. The host is in the 64 subnet.
4. F. A CIDR address of /19 is 255.255.224.0. This is a Class B address, so that is only 3 subnet bits, but it provides 13 host bits, or 8 subnets, each with 8,190 hosts.
5. B, D. The mask 255.255.254.0 (/23) used with a Class A address means that there are 15 subnet bits and 9 host bits. The block size in the third octet is 2 (256 – 254). So this makes the subnets in the interesting octet 0, 2, 4, 6, etc., all the way to 254. The host 10.16.3.65 is in the 2.0 subnet. The next subnet is 4.0, so the broadcast address for the 2.0 subnet is 3.255. The valid host addresses are 2.1 through 3.254.
6. D. A /30, regardless of the class of address, has a 252 in the fourth octet. This means we have a block size of 4 and our subnets are 0, 4, 8, 12, 16, etc. Address 14 is obviously in the 12 subnet.
7. D. A point-to-point link uses only two hosts. A /30, or 255.255.255.252, mask provides two hosts per subnet.
8. C. A /21 is 255.255.248.0, which means we have a block size of 8 in the third octet, so we just count by 8 until we reach 66. The subnet in this question is 64.0. The next subnet is 72.0, so the broadcast address of the 64 subnet is 71.255.
9. A. A /29 (255.255.255.248), regardless of the class of address, has only 3 host bits. Six hosts is the maximum number of hosts on this LAN, including the router interface.
10. C. A /29 is 255.255.255.248, which is a block size of 8 in the fourth octet. The subnets are 0, 8, 16, 24, 32, 40, etc. 192.168.19.24 is the 24 subnet, and since 32 is the next subnet, the broadcast address for the 24 subnet is 31. 192.168.19.26 is the only correct answer.
11. A. A /29 (255.255.255.248) has a block size of 8 in the fourth octet. This means the subnets are 0, 8, 16, 24, etc. 10 is in the 8 subnet. The next subnet is 16, so 15 is the broadcast address.
12. B. You need 5 subnets, each with at least 16 hosts. The mask 255.255.255.240 provides 16 subnets with 14 hosts—this will not work. The mask 255.255.255.224 provides 8 subnets, each with 30 hosts. This is the best answer.
13. A, E. First, if you have two hosts directly connected, as shown in the graphic, then you need a crossover cable. A straight-through cable won't work. Second, the hosts have different masks, which puts them in different subnets. The easy solution is just to set both masks to 255.255.255.0 (/24).

14. A. A /25 mask is 255.255.255.128. Used with a Class B network, the third and fourth octets are used for subnetting with a total of 9 subnet bits, 8 bits in the third octet and 1 bit in the fourth octet. Since there is only 1 bit in the fourth octet, the bit is either off or on—which is a value of 0 or 128. The host in the question is in the 0 subnet, which has a broadcast address of 127 since 128 is the next subnet.
15. A. A /28 is a 255.255.255.240 mask. Let's count to the ninth subnet (we need to find the broadcast address of the eighth subnet, so we need to count to the ninth subnet). Starting at 16 (remember, the question stated that we will not use subnet zero, so we start at 16, not 0), 16, 32, 48, 64, 80, 96, 112, 128, 144. The eighth subnet is 128 and the next subnet is 144, so our broadcast address of the 128 subnet is 143. This makes the host range 129–142. 142 is the last valid host.
16. C. A /28 is a 255.255.255.240 mask. The first subnet is 16 (remember that the question stated not to use subnet zero) and the next subnet is 32, so our broadcast address is 31. This makes our host range 17–30. 30 is the last valid host.
17. E. A Class C subnet mask of 255.255.255.224 is 3 bits on and 5 bits off (11100000) and provides 8 subnets, each with 30 hosts. However, if the command `ip subnet-zero` is not used, then only 6 subnets would be available for use.
18. E. A Class B network ID with a /22 mask is 255.255.252.0, with a block size of 4 in the third octet. The network address in the question is in subnet 172.16.16.0 with a broadcast address of 172.16.19.255. Only option E even has the correct subnet mask listed, and 172.16.18.255 is a valid host.
19. D, E. The router's IP address on the E0 interface is 172.16.2.1/23, which is 255.255.254.0. This makes the third octet a block size of 2. The router's interface is in the 2.0 subnet, and the broadcast address is 3.255 because the next subnet is 4.0. The valid host range is 2.1 through 3.254. The router is using the first valid host address in the range.
20. C. To test the local stack on your host, ping the loopback interface of 127.0.0.1.

## Answers to Written Lab 3.1

1. 192.168.100.25/30. A /30 is 255.255.255.252. The valid subnet is 192.168.100.24, broadcast is 192.168.100.27, and valid hosts are 192.168.100.25 and 26.
2. 192.168.100.37/28. A /28 is 255.255.255.240. The fourth octet is a block size of 16. Just count by 16s until you pass 37. 0, 16, 32, 48. The host is in the 32 subnet, with a broadcast address of 47. Valid hosts 33–46.
3. 192.168.100.66/27. A /27 is 255.255.255.224. The fourth octet is a block size of 32. Count by 32s until you pass the host address of 66. 0, 32, 64. The host is in the 32 subnet, broadcast address of 63. Valid host range of 33–62.
4. 192.168.100.17/29. A /29 is 255.255.255.248. The fourth octet is a block size of 8. 0, 8, 16, 24. The host is in the 16 subnet, broadcast of 23. Valid hosts 17–22.
5. 192.168.100.99/26. A /26 is 255.255.255.192. The fourth octet has a block size of 64. 0, 64, 128. The host is in the 64 subnet, broadcast of 127. Valid hosts 65–126.
6. 192.168.100.99/25. A /25 is 255.255.255.128. The fourth octet is a block size of 128. 0, 128. The host is in the 0 subnet, broadcast of 127. Valid hosts 1–126.
7. A default Class B is 255.255.0.0. A Class B 255.255.255.0 mask is 256 subnets, each with 254 hosts. We need fewer subnets. If we used 255.255.240.0, this provides 16 subnets. Let's add one more subnet bit. 255.255.248.0. This is 5 bits of subnetting, which provides 32 subnets. This is our best answer, a /21.
8. A /29 is 255.255.255.248. This is a block size of 8 in the fourth octet. 0, 8, 16. The host is in the 8 subnet, broadcast is 15.
9. A /29 is 255.255.255.248, which is 5 subnet bits and 3 host bits. This is only 6 hosts per subnet.
10. A /23 is 255.255.254.0. The third octet is a block size of 2. 0, 2, 4. The subnet is in the 16.2.0 subnet; the broadcast address is 16.3.255.

## Answers to Written Lab 3.2

Classful Address	Subnet Mask	Number of Hosts per Subnet ( $2^h - 2$ )
/16	255.255.0.0	65,534
/17	255.255.128.0	32,766
/18	255.255.192.0	16,382
/19	255.255.224.0	8,190
/20	255.255.240.0	4,094
/21	255.255.248.0	2,046
/22	255.255.252.0	1,022
/23	255.255.254.0	510
/24	255.255.255.0	254
/25	255.255.255.128	126
/26	255.255.255.192	62
/27	255.255.255.224	30
/28	255.255.255.240	14
/29	255.255.255.248	6
/30	255.255.255.252	2

## Answers to Written Lab 3.3

Decimal IP Address	Address Class	Number of Subnet and Host Bits	Number of Subnets ( $2^x$ )	Number of Hosts ( $2^x - 2$ )
10.25.66.154/23	A	15/9	32768	510
172.31.254.12/24	B	8/8	256	254
192.168.20.123/28	C	4/4	16	14
63.24.89.21/18	A	10/14	1,024	16,384
128.1.1.254/20	B	4/12	16	4094
208.100.54.209/30	C	6/2	64	2