Subnetting IPv4 networks

This chapter covers

- What subnetting is and why it's necessary
- How to borrow bits from the host portion of a network to expand the network portion and create subnets
- How to identify the five attributes of a subnet
- How to divide a network into subnets of equal and variable sizes

In chapter 7, we covered IPv4 address classes, focusing on classes A, B, and C—the three classes of addresses which can be assigned to hosts. Each class is defined by the first bit(s) of the address, and the prefix length of addresses in each range is also defined:

- Class A addresses begin with 0b0 and use a /8 prefix length.
- Class B addresses begin with 0b10 and use a /16 prefix length.
- Class C addresses begin with 0b110 and use a /24 prefix length.

This addressing architecture, called *classful addressing*, was defined in the original *Internal Protocol* standard in 1981 (RFC 791). However, with the rapid growth of the internet, classful addressing soon proved to be too rigid, resulting in inefficient use of addresses; the pool of available IPv4 addresses was drying up. Subnetting, which involves dividing a larger network up into smaller networks, is one answer to this problem and is a fundamental skill for network engineers. Subnetting is the second half of CCNA exam topic 1.6: Configure and verify IPv4 addressing and subnetting.

Before we get started, I want to emphasize that subnetting is a skill, and it requires practice to become proficient. Just reading this chapter alone won't make you good at subnetting; you need to spend some time actually doing it. However, if you read through this chapter carefully and do the recommended practice, you'll be a confident "subnetter" in no time.

11.1 What is subnetting?

The problem with classful addressing is that it doesn't allow us to create networks of appropriate sizes. The smallest network size – a class C network – contains 254 usable addresses (2^8, minus 2 for the network and broadcast addresses). That is far more addresses than necessary for a home network or many small offices. Assigning a class C network to a small office with only a few dozen devices would result in over two hundred IP addresses left unused.

However, a class C network is too small for most enterprise networks, meaning that a class B network would be required. A class B network contains 65,534 (2^{16}-2) usable addresses, far more than even a very large network requires, resulting in thousands of wasted addresses. And a class A network contains 16,777,214 usable addresses, a ludicrous number of addresses for a single network. These classful rules were designed for simplicity, not efficiency, but with the Internet's exploding popularity, a better solution was needed.

To support the fast-growing Internet and use the available IPv4 address space more efficiently, a new system was introduced in 1993: *Classless Inter-Domain Routing* (*CIDR*, pronounced like "cider"). CIDR throws out the rules of classful addressing, and replaces them with a more flexible system; with CIDR, prefix lengths don't have to be /8, /16, or /24. Instead, the boundary between the network portion and host portion of an IP address can be in the middle of an octet, resulting in prefix lengths like /23, /26, /28, etc.

Note

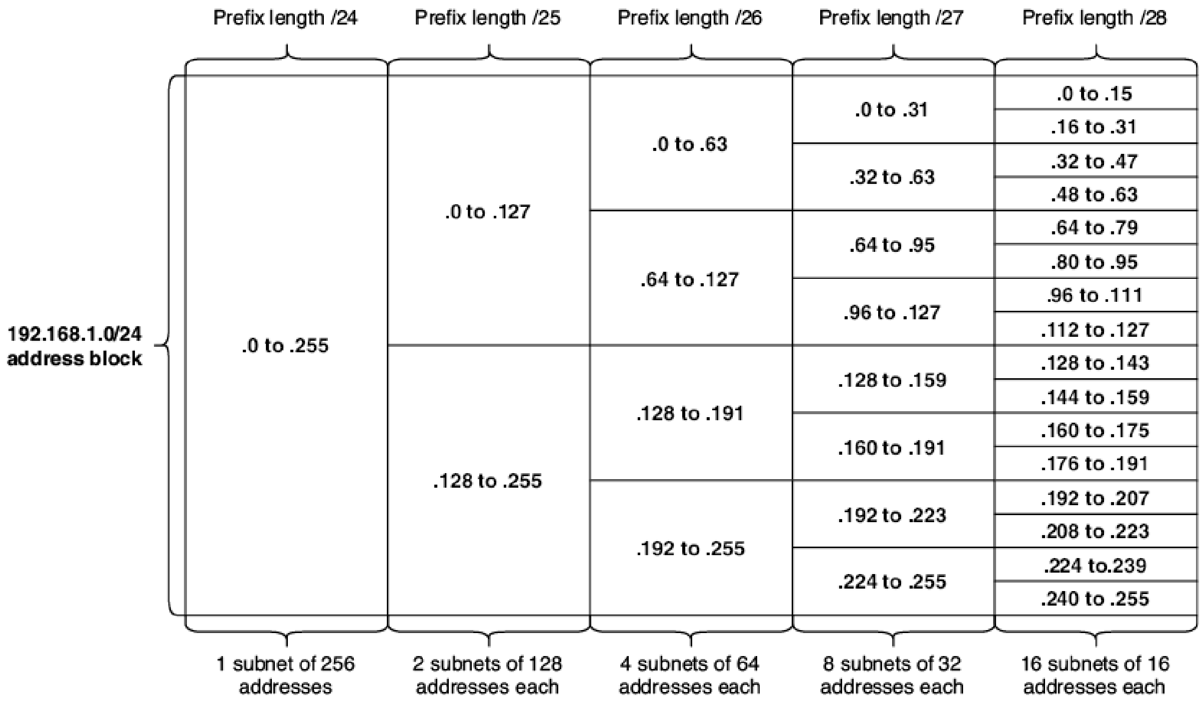
The method of notating an address's prefix length with */X* is also known as *CIDR notation*, because it was introduced with CIDR. Before CIDR notation, the prefix length was always indicated with a netmask, such as 255.255.255.0. Another term for netmask is *subnet mask*; I will use the latter term in this chapter because we are focusing on subnetting, but the terms are interchangeable.

With CIDR, an enterprise can be assigned an address block that can be divided into networks of appropriate size, called *subnets* (*subdivided networks*). The process of dividing an address block into subnets is called *subnetting*.

Note: An *address block* is a range of IP addresses. It can be used to refer to a network before it has been subnetted. For example, 192.168.1.0/24 is an address block (including IP addresses 192.168.1.0 through 192.168.1.255), which can be divided into multiple smaller networks (subnets).

Figure 11.1 demonstrates how a /24 address block can be divided into subnets. The 192.168.1.0/24 address range allows for a single subnet with a /24 prefix length, including all addresses from 192.168.1.0 through 192.168.1.255. Dividing the /24 address block in half gives two /25 subnets, each containing 128 addresses. Or, it can be divided into four /26 subnets, each containing 64 addresses. For each bit you extend the prefix length, the number of possible subnets doubles, but the number of addresses in each subnet halves.

Figure 11.1 The 192.168.1.0/24 address block (network) divided into smaller subnets. With a /24 prefix length, it is one subnet of 256 addresses. Using a /25 prefix length allows the block to be divided into two subnets of 128 addresses each. /26 allows for 4 subnets of 64 addresses each. /27 allows for 8 subnets of 32 addresses each. /28 allows for 16 subnets of 16 addresses each.



Note: Figure 11.1 only shows prefix lengths of up to /28, but longer prefix lengths follow the same pattern: increasing the length of the prefix length by one bit doubles the number of possible subnets, but halves the number of addresses contained in each subnet.

11.2 FLSM subnetting

Subnetting is the process of dividing an address block into smaller subnets. That process can be done in a couple of different ways: *Fixed-Length Subnet Masking* (FLSM) divides the block into subnets of equal sizes. On the other hand, *Variable-Length Subnet Masking* (VLSM) divides the block into subnets of varying sizes depending on how many addresses are actually needed in the subnet.

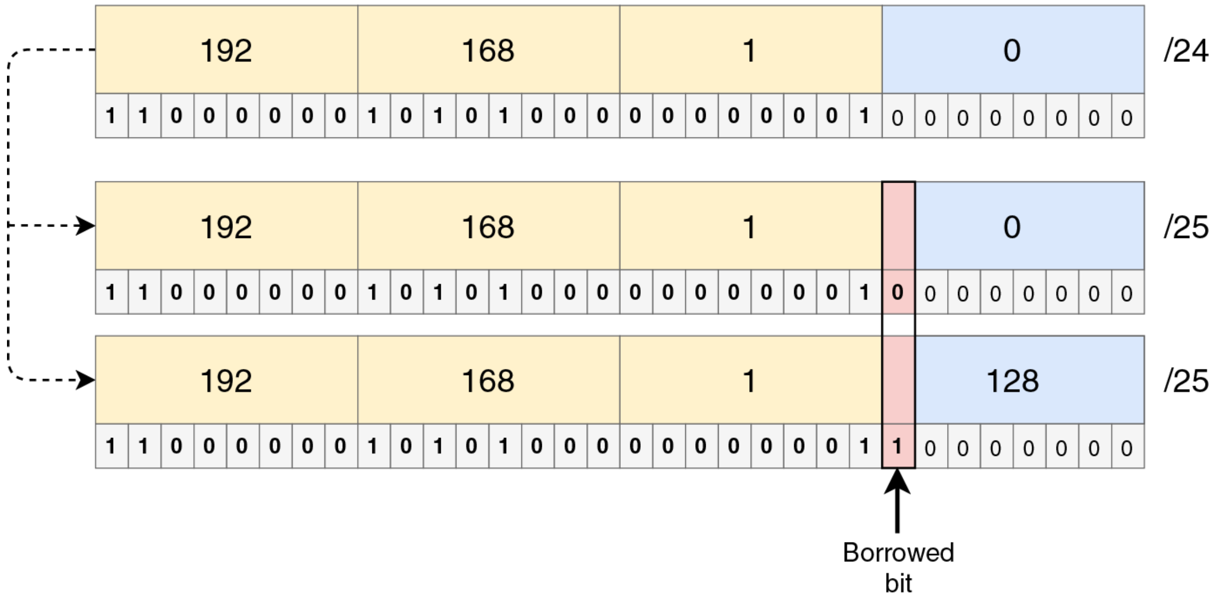
In the real world, VLSM is what you'll be using; it allows you to more efficiently use a block of addresses since you can make each subnet only as large as it needs to be – this wastes fewer addresses. However, FLSM serves as a useful stepping stone when learning how to subnet, and you should know it for the CCNA exam, so in this section we will focus on FLSM.

11.2.1 Subnetting /24 address blocks

First we will look at how to subnet address blocks with a prefix length of /24 or greater. The reason for this is that it allows us to focus only on the final octet of the address, simplifying the process a bit.

The network portion of an address block cannot be changed; if you are given the 192.168.1.0/24 address block, you can't assign 192.168.*2*.1 (or any other IP address not included in the 192.168.1.0/24 range) to a host. However, the host portion is fair game – you can use the last 8 bits to make various IP addresses to assign to hosts. This is the key to subnetting; to make subnets, you "borrow" bits from the host portion and add them to the network portion. You are then free to change the binary value of those borrowed bits between "0" and "1" to make different subnets. Figure 11.2 demonstrates how one bit can be borrowed from the host portion of 192.168.1.0/24 to make two different subnets: 192.168.1.0/25 and 192.168.1.128/25.

Figure 11.2 Borrowing 1 bit from the host portion of 192.168.1.0/24 allows us to create two subnets: 192.168.1.0/25 and 192.168.1.128/25. The borrowed bit was part of the host portion of the original address block, but it is part of the network portion of each subnet (as indicated by the /25 prefix length).

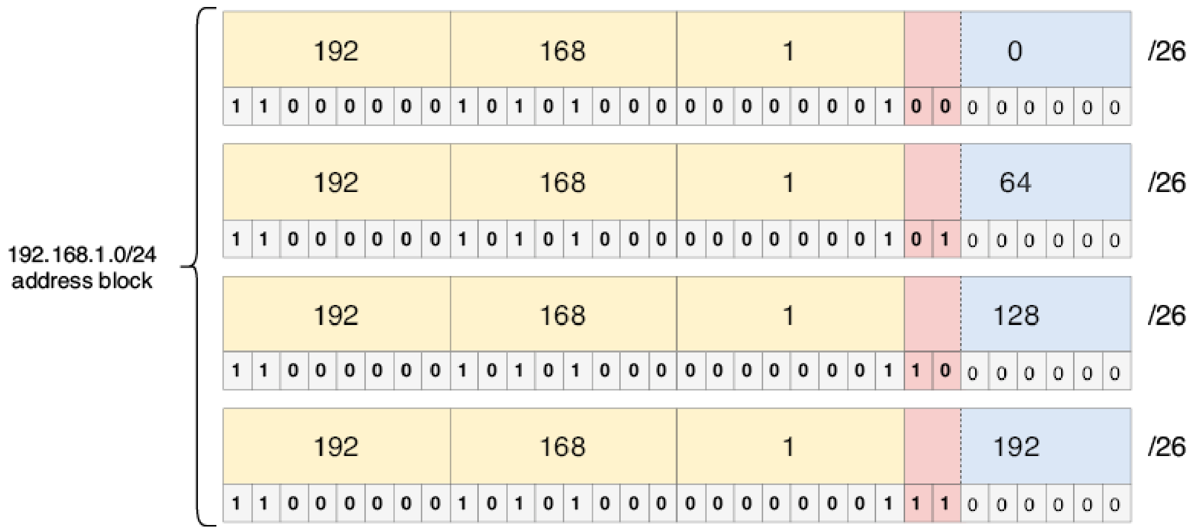


Note: In previous examples, the network address always ended in ".0". However, when subnetting, because the boundary between the network portion and host portion can lie in the middle of an octet, the network address does not necessarily end in ".0". 192.168.1.0 is the network address of the 192.168.1.0/25 subnet, and 192.168.1.128 is the network address of the 192.168.1.128/25 subnet.

With the borrowed bit set to "0", we get the first subnet: 192.168.1.0/25. If we change the borrowed bit to "1", we get the second subnet: 192.168.1.128/25. That's how subnets are made: by changing the binary value of the borrowed bit(s).

Borrowing a single bit from the host portion allows us to make two subnets, so how many subnets can we make if we borrow two bits? We covered this in section 11.1: each bit added to the prefix length (each bit borrowed from the host portion) doubles the number of subnets. The formula is 2^x, where x is the number of borrowed bits, and therefore borrowing 2 bits allows us to make 4 (2^2) subnets. Figure 11.3 shows the four subnets that can be made by borrowing two bits from the host portion of the 192.168.1.0/24 block.

Figure 11.3 Borrowing 2 bits from 192.168.1.0/24 allows for four subnets: 192.168.1.0/26, 192.168.1.64/26, 192.168.1.128/26, and 192.168.1.192/26.



Note: **In the first subnet, the borrowed bits are "00". How can you know what the next subnet is? Just count up in binary: the number after "00" is "01", then "10", and then "11". As we covered in chapter 7, counting in binary is the same process as counting in decimal, except there are only two digits to work with: 0 and 1**.

Calculating the five attributes of a /24+ subnet

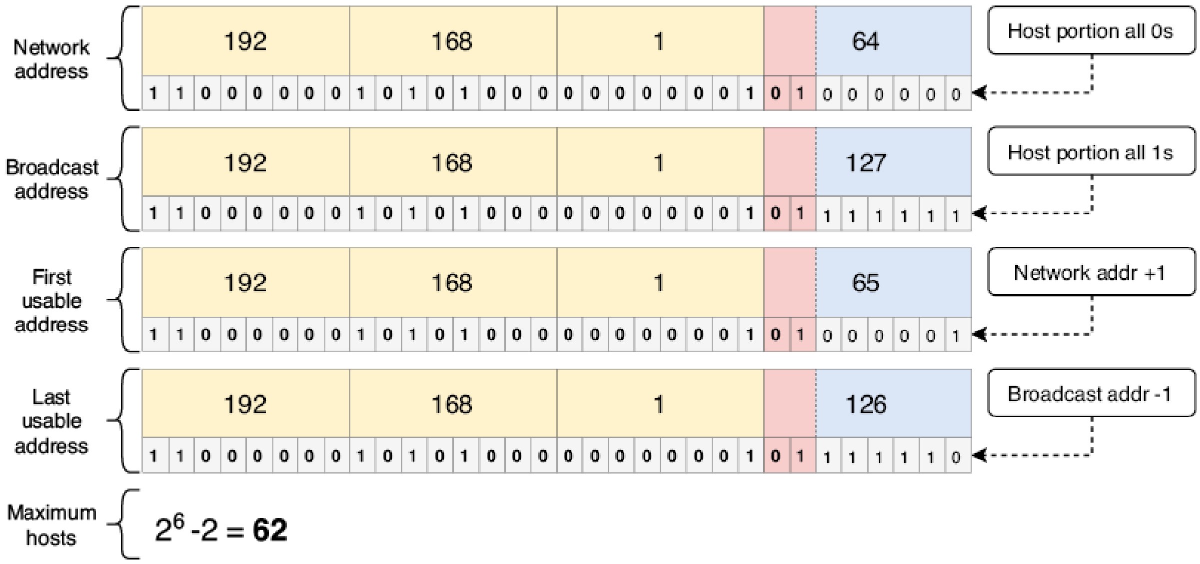
In chapter 7, we covered five attributes of an IPv4 network: network address, broadcast address, first usable address, last usable address, and maximum number of hosts. Those same attributes apply when dividing networks into subnets. Here's a quick review of each attribute:

- Network address: The first address of a subnet, with a host portion of all 0s.
- Broadcast address: The last address of a subnet, with a host portion of all 1s.
- First usable address: The first address in the subnet that can be assigned to a host. It can be calculated by adding 1 to the network address (changing the last bit to 1).
- Last usable address: The last address in the subnet that can be assigned to a host. It can be calculated by subtracting 1 from the broadcast address (changing the last bit to 0).
- Maximum number of hosts: The number of IP addresses available to assign to hosts. The formula is 2^y-2, where *y* is the number of bits in the host portion. 2 is subtracted because the network and broadcast addresses cannot be assigned to hosts.

Figure 11.4 shows the five attributes of one of the subnets from figure 11.3: the 192.168.1.64/26 subnet. The calculations are the same as we covered in chapter 7 – just keep in mind that the borrowed bits are now part of the network portion. Because the prefix length in this example is /26, only the last six bits are the host portion.

Figure 11.4 The five attributes of the 192.168.1.64/26 subnet. The first 26 bits are the network portion, and the final 6 bits are the host portion. The network address is 192.168.1.64 (host portion all 0s). The broadcast address is 192.168.1.127 (host portion all 1s). The first usable address is 192.168.1.65 (network address + 1). The last usable

address is 192.168.1.126 (broadcast address – 1). The maximum number of hosts in the subnet is 62 (26 – 2).



A common subnetting problem is something like this: "PC1 has IP address 172.16.20.27/28. What is the network address of the subnet it belongs to?". To solve a question like this, you can follow the these three steps:

1. Write the address in binary: **10101100.00010000.00010100.0001**1011
2. Change the host portion to all 0s: **10101100.00010000.00010100.0001**0000
3. Convert back to dotted decimal: 172.16.20.16. That's the answer!

Exam Tip

You should be able to identify any of the five attributes of a particular subnet, not just the network address. Such problems could appear on the CCNA exam as standalone questions, or as part of more complex questions.

/24+ subnet masks

Ideally, we would be able to just write prefix lengths in CIDR notation, without having to worry about subnet masks. However, because you have to use subnet masks when configuring IP addresses and static routes in Cisco IOS, they are necessary to learn.

To review, a subnet mask is a series of 32 bits that indicates which bits in an IP address are part of the network portion, and which are part of the host portion. A bit set to "1" in the subnet mask means that the bit in the same position of the IP address is part of the network portion, and a bit set to "0" in the subnet mask means that the bit in the same position of the IP address is part of the host portion. And because an IP address consists of the network portion followed by the host portion, a subnet mask is a series of 1s followed by a series of 0s (unless it is all 0s, as in /0, or all 1s, as in /32).

Mgno hfnx adnlegi jurw /8, /16, bnz /24 prefix length a, subnet mask a tkz lsimep: 255.0.0.0, 255. 255.0.0, tx 255. 255. 255.0, lipctveeyesr. Mkbn igsnu AJOA, verewho, yxr badyunro ebetnwe network nyc host portion zan ofj nj rxy ildedm le cn tceto, ihwch sslertu jn hoert esisbolp subnet mask c. Xsxuf 11.1 sitsl prefix length z mktl /24 er /32 bns herti uivanetqle subnet mask z wentrit jn binary bns tdetdo maelidc. Aeq uolsdh imzfiailrae ylseorfu urjw eshet subnet mask c; eyg'ff qnov rx wvno qrom vbnw configuring Xkazj routers. Vte reeerfnec, table 11.1 fcsx sstil oqr maximum number of hosts jn s setnbu lk sksg jvcc.

Table 11.1 /24+ prefix lengths and subnet masks [(view table figure)]

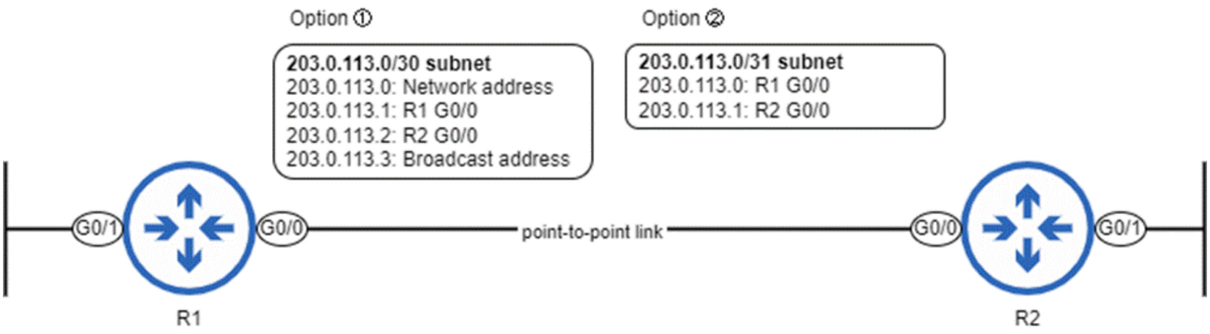
Prefix length	**Subnet mask (binary)**	**Subnet mask (decimal)**	**Maximum number of hosts (2^y-2)**
/24	11111111.11111111.11111111.00000000	255.255.255.0	254
/25	11111111.11111111.11111111.10000000	255.255.255.128	126
/26	11111111.11111111.11111111.11000000	255.255.255.192	62
/27	11111111.11111111.11111111.11100000	255.255.255.224	30
/28	11111111.11111111.11111111.11110000	255.255.255.240	14
/29	11111111.11111111.11111111.11111000	255.255.255.248	6
/30	11111111.11111111.11111111.11111100	255.255.255.252	2
/31	11111111.11111111.11111111.11111110	255.255.255.254	2 (see the following)
/32	11111111.11111111.11111111.11111111	255.255.255.255	1 (see the following)

Prefix lengths of /31 and /32 are special cases when it comes to calculating the maximum number of hosts in a subnet. A /31 prefix length, for example, leaves a single host bit. If we use the formula 2^y-2 to calculate the maximum number of hosts in the subnet, the result is 0; a single host bit only allows for two addresses, and those are taken by the network and broadcast addresses, resulting in no usable addresses. For this reason, /31 prefix lengths were unused for a long time.

However, for the purpose of further preserving the IPv4 address space, an exception to the normal rules was made for /31 prefix lengths: they can be used for *point-to-point* links – connections between two routers, which only require two IP addresses. In this case, the subnet does not have a network address or broadcast address. Before this exception was made, point-to-point links used /30 prefix lengths, leaving two host bits, and therefore two usable addresses (2^2-2=2). This works fine, and /30 prefix lengths are still commonly used for point-to-point links today, but /31 prefix lengths are more efficient; they only consume two IP addresses, rather than four.

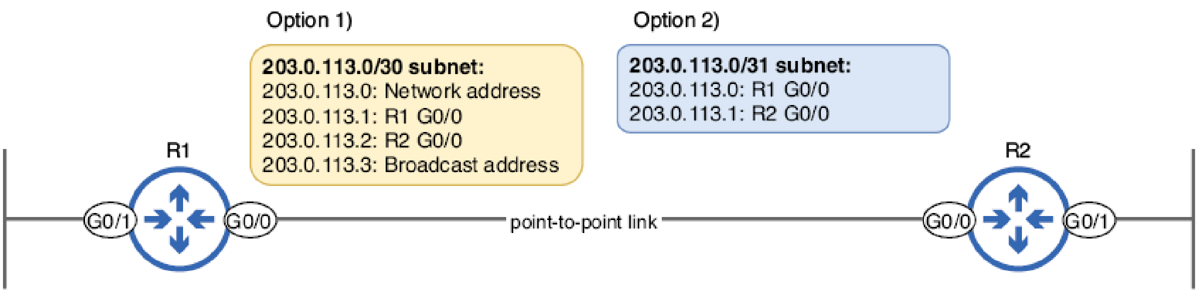
Note

Figure 11.5 A point-to-point link connecting R1 to R2. Traditionally, a /30 subnet would be used for a connection like this, as in option 1). In modern networks, a /31 subnet, as in option 2), is also valid (and more efficient).



In the following example, I configure R1 G0/0's IP address using a /31 prefix length (subnet mask 255.255.255.254). The router then displays a message, warning that /31 prefix lengths should be used cautiously.

Figure 11.5 A point-to-point link connecting R1 to R2. Traditionally, a /30 subnet would be used for a connection like this, as in option 1. In modern networks, a /31 subnet, as in option 2, is also valid (and more efficient).



R1(config-if)# ip address 203.0.113.0 255.255.255.254

% Warning: use /31 mask on non point-to-point interface cautiously

Note

Subnetting an address block with a prefix length shorter than /24 can seem intimidating at first; you can no longer focus only on the final octet of the address. However, let me assure you that the process of subnetting does not change at all:

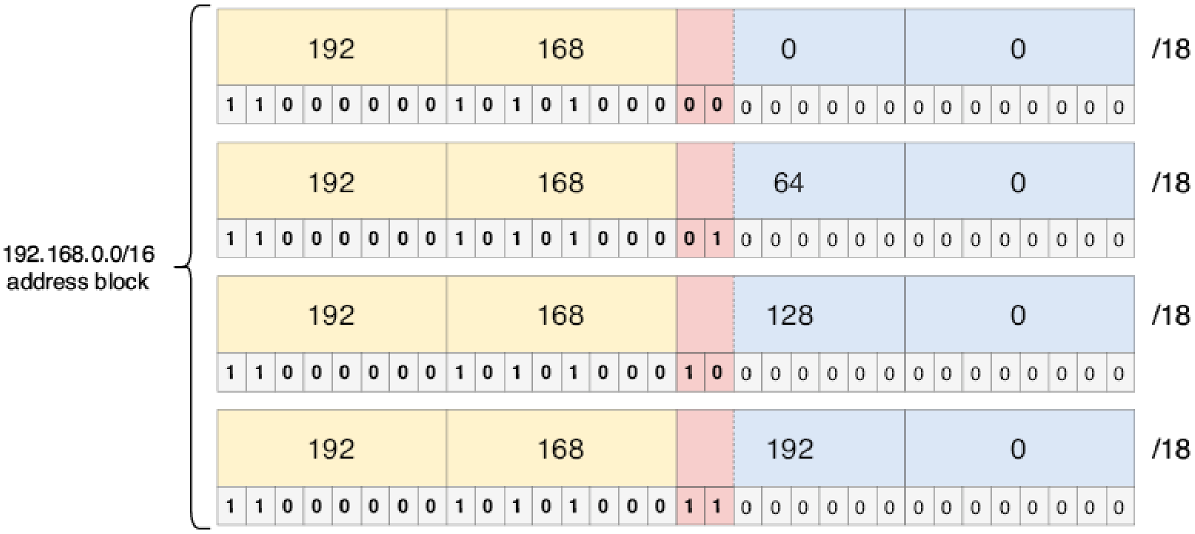
The maximum number of hosts per subnet is still 2^y-2, where *y* is the number of host bits.

- The subnet's network address is still the address with a host portion of all 0s.
- I think you get the idea! The only difference is that converting between decimal and binary takes a little more care. Because there's no need to introduce any new concepts, in this section I'll demonstrate how the previous concepts apply to address blocks with a /16+ prefix length. Table 11.2 summarizes some ways a /16 address block can be subnetted. As before, each borrowed bit doubles the amount of subnets that can be made, but halves the total number of addresses per subnet (table 11.2 displays the maximum number of hosts per subnet, rather than the total number of addresses).

Table 11.2 Subnetting a /16 address block [(view table figure)]

Prefix length	Subnet mask (decimal)	Borrowed bits	Number of subnets	Maximum number of hosts per subnet (2^y-2)
/16	255.255.0.0	0	1	65,534
/17	255.255.128.0	1	2	32,766
/18	255.255.192.0	2	4	16,382
/19	255.255.224.0	3	8	8190
/20	255.255.240.0	4	16	4094
/21	255.255.248.0	5	32	2046
/22	255.255.252.0	6	64	1022
/23	255.255.254.0	7	128	510
/24	255.255.255.0	8	256	254
/25	255.255.255.128	9	512	126

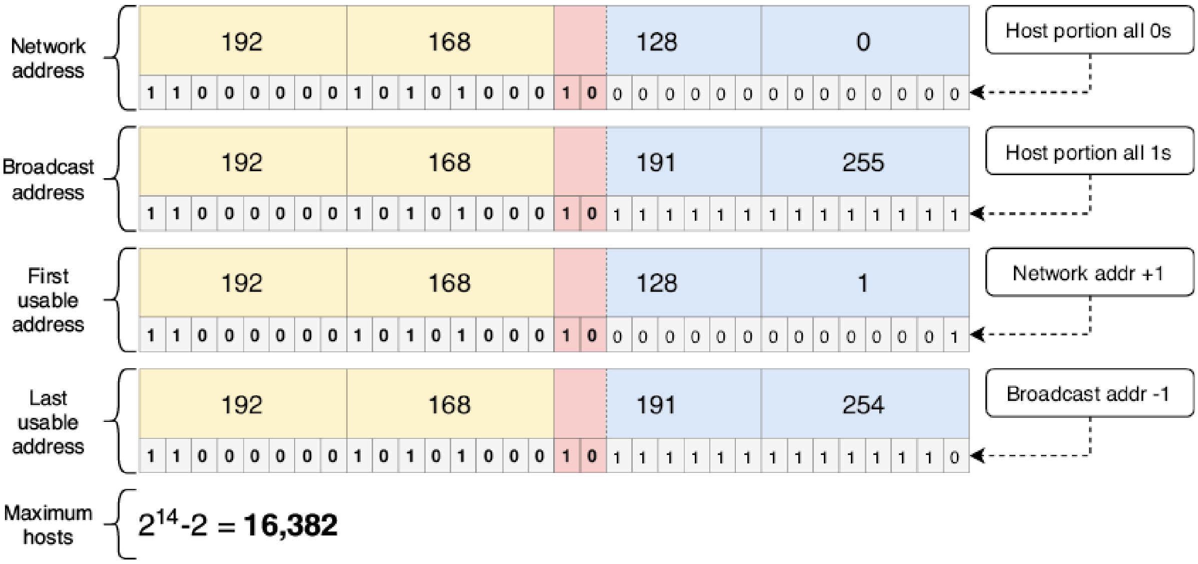
Figure 11.6 Borrowing two bits from 192.168.0.0/16 allows for four subnets: 192.168.0.0/18, 192.168.64.0/18, 192.168.128.0/18, and 192.168.192.0/18.



Note: Calculating the five attributes is the same process as before. Figure 11.7 takes one of the subnets from figure 11.6 (192.168.128.0/18), and shows the five attributes of that subnet.

Figure 11.7 The five attributes of the 192.168.128.0/18 subnet. The first 18 bits are the network portion, and the final 14 bits are the host portion. The network address is 192.168.128.0 (host portion all 0s). The broadcast address is 192.168.191.255 (host portion all 1s). The first usable address is 192.168.128.1 (network address + 1). The last

usable address is 192.168.191.254 (broadcast address – 1). The maximum number of hosts in the subnet is 16,382 (214 – 2).

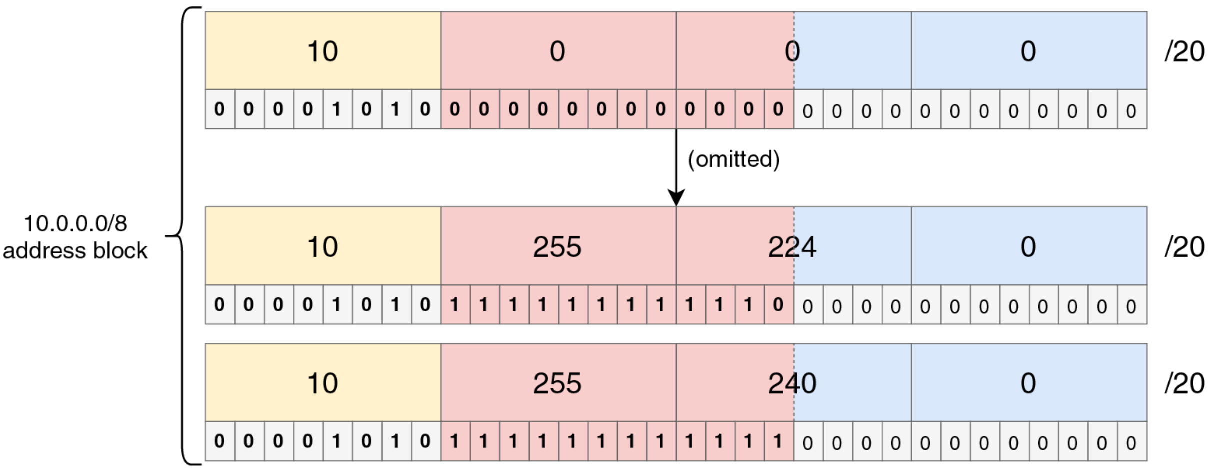


Note

Subnetting a /8 address block is, once again, the same process we have seen up to this point. However, a /8 address block means there are 24 host bits – lots of bits to either borrow and make lots of subnets, or use to make a few very large subnets. Table 11.3 summarizes some ways that a /8 address block can be subnetted.

Note

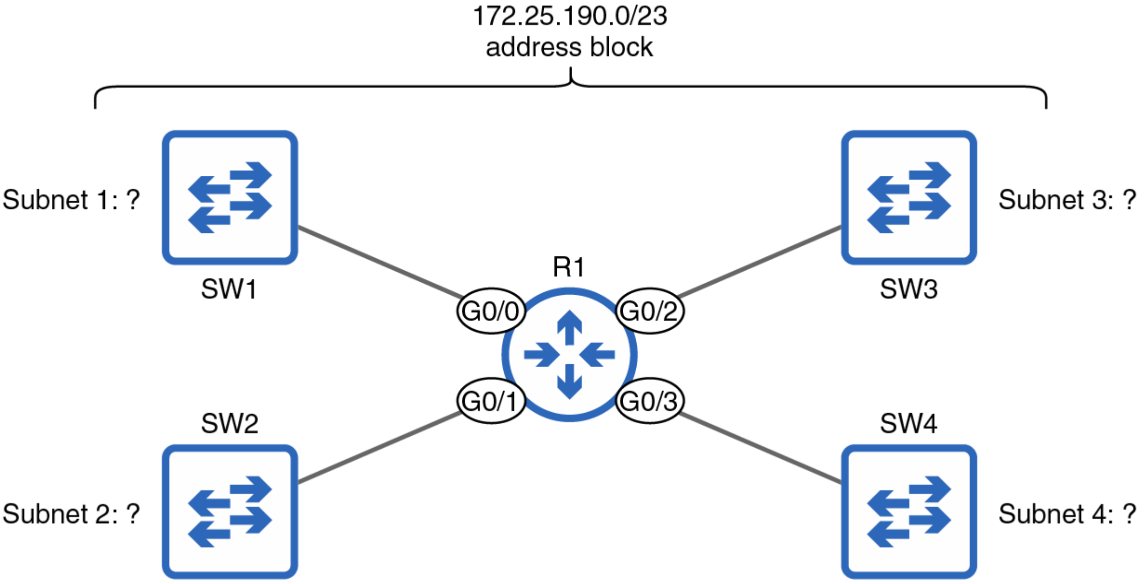
Figure 11.8 Borrowing 12 bits from 10.0.0.0/8 allows for 4,096 subnets. 10.0.0.0/20 is the first subnet, and 10.255.224.0/20 and 10.255.240.0/20 are the final two.



Note: As mentioned at the beginning of this chapter, subnetting is a skill that takes practice to become proficient. At the end of this chapter I will give some recommendations for free websites where you can find subnetting practice questions. Before that, let's go through a couple scenarios that resemble what you might find on those websites (and on the CCNA exam itself).

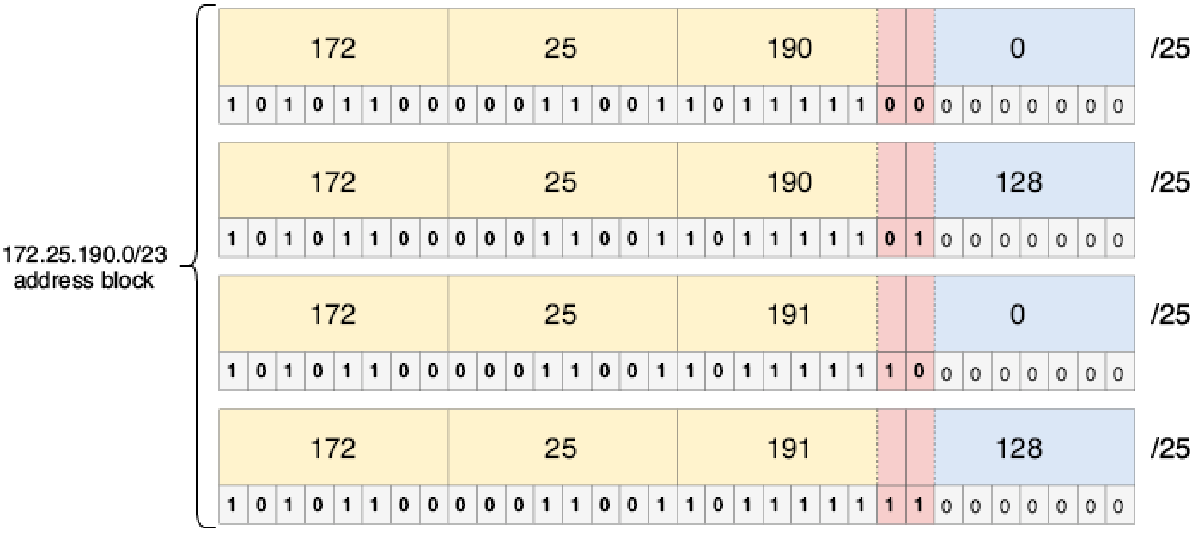
11.2.4 FLSM scenarios

Figure 11.9 An FSLM scenario in which you must subnet the 172.25.190.0/23 address block into four subnets of equal size, identify the number of host addresses per subnet, and configure the first usable address of each subnet on R1's interfaces



1. Subnet the 172.25.190.0/23 address block into four subnets, and identify each subnet.
2. How many host addresses are available in each subnet?
3. Configure the first usable address of each subnet on R1's interfaces.

Figure 11.10 The subnets that can be created by subnetting 172.25.190.0/23 into four equal parts: 172.25.190.0/25, 172.25.190.128/25, 172.25.191.0/25, and 172.25.191.128/25. Borrowing 2 bits from the host portion of the /23 address block results in four /25 subnets.



We have now solved the first part of the scenario: identifying the four subnets. The second part asks how many host addresses are available in each subnet. To solve this, just use the same formula as always: 2^y-2 (y being the number of host bits). The original address block was /23, meaning there were nine host bits. However, after borrowing two host bits to make subnets, seven host bits remain. Therefore, there are 126 (2^7-2) host addresses available in each subnet.

R1(config)# interface g0/0 R1(config-if)# ip address 172.25.190.1 255.255.255.128
R1(config-if)# interface g0/1 R1(config-if)# ip address 172.25.190.129 255.255.255.128
R1(config-if)# interface g0/2 R1(config-if)# ip address 172.25.191.1 255.255.255.128
R1(config-if)# interface g0/3 R1(config-if)# ip address 172.25.191.129 255.255.255.128

We have now solved the scenario! Let's walk through one more scenario, this time text only: *You have been given the 10.224.0.0/11 address block. You must create 2000 subnets which will be assigned to various offices and departments within a large company. What prefix length must you use to create a sufficient number of subnets? How many host addresses are in each subnet?*

R1(config)# interface g0/0

R1(config-if)# ip address 172.25.190.1 255.255.255.128

R1(config-if)# interface g0/1

R1(config-if)# ip address 172.25.190.129 255.255.255.128

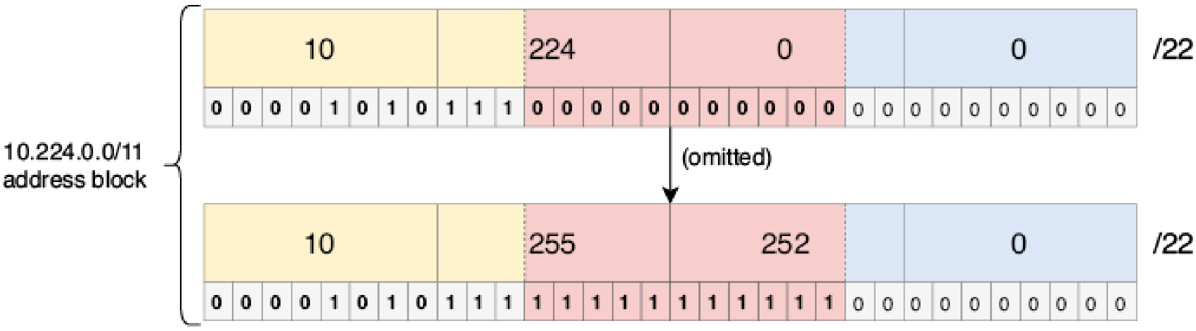
R1(config-if)# interface g0/2

R1(config-if)# ip address 172.25.191.1 255.255.255.128

R1(config-if)# interface g0/3

R1(config-if)# ip address 172.25.191.129 255.255.255.128

Figure 11.11 Borrowing 11 bits from the host portion of the 10.224.0.0/11 address block allows for 2,048 subnets. The first subnet is 10.224.0.0/22 (all borrowed bits set to 0), and the last (2048th) subnet is 10.255.252.0/22 (all borrowed bits set to 1).



A /22 prefix length means that 10 host bits remain, so now we can calculate the number of host addresses in each subnet: 1022 (2^{10}-2). And now we have solved the scenario!

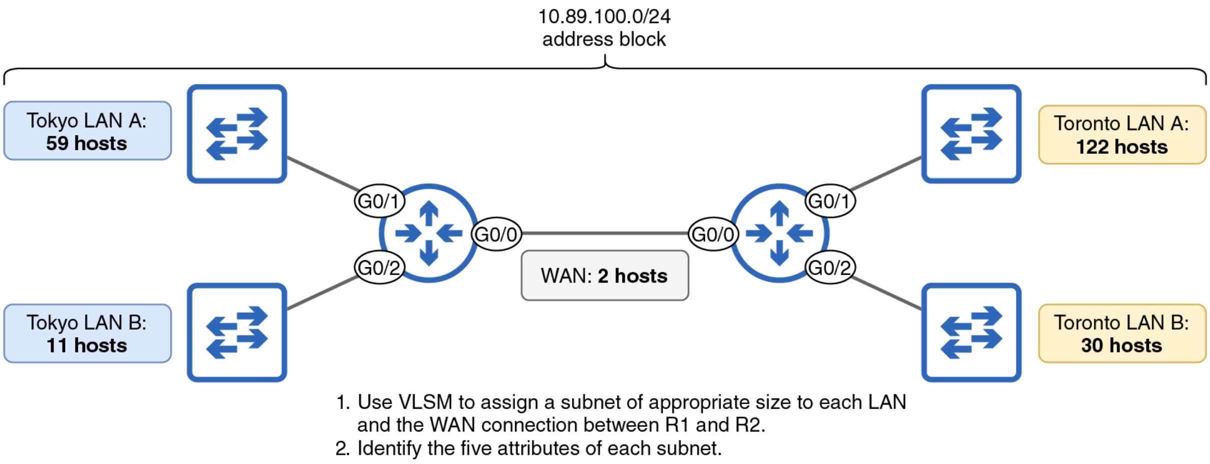
Because the number of subnets increases by a power of 2 for each borrowed bit, you often won't be able to create exactly the number of subnets needed; you'll probably end up with some extra subnets, which is not a bad thing – they can be used to accommodate network expansions in the future. Likewise, you often won't be able to create subnets of exactly the size you need; you'll usually have some extra addresses in each subnet.

Note: VLSM (Variable-Length Subnet Masking) allows us to subnet an address block even more efficiently than FLSM, by creating subnets of varying sizes. Although FLSM is a helpful introduction to subnetting, when actually subnetting networks in the real world, chances are you'll be doing VLSM. Figure 11.12 shows the scenario I will use to demonstrate VLSM.

11.3 VLSM subnetting

A /24 address block includes 254 host addresses, and the total number of host addresses required in figure 11.12's scenario is 226, so the address space is sufficient. Using FLSM to create subnets of equal size would result in some subnets having too few addresses (Toronto LAN A requires 122 host addresses), and some subnets having too many addresses (the WAN connection only requires 2 host addresses, for R1 and R2). However, if we use VLSM, we can make some subnets smaller and some larger, allowing us to efficiently use the available address block. The high-level VLSM process is as follows:

Figure 11.12 A VLSM scenario in which you must assign subnets from the 10.89.100.0/24 address block to each LAN and the WAN connection and identify the five attributes of each subnet

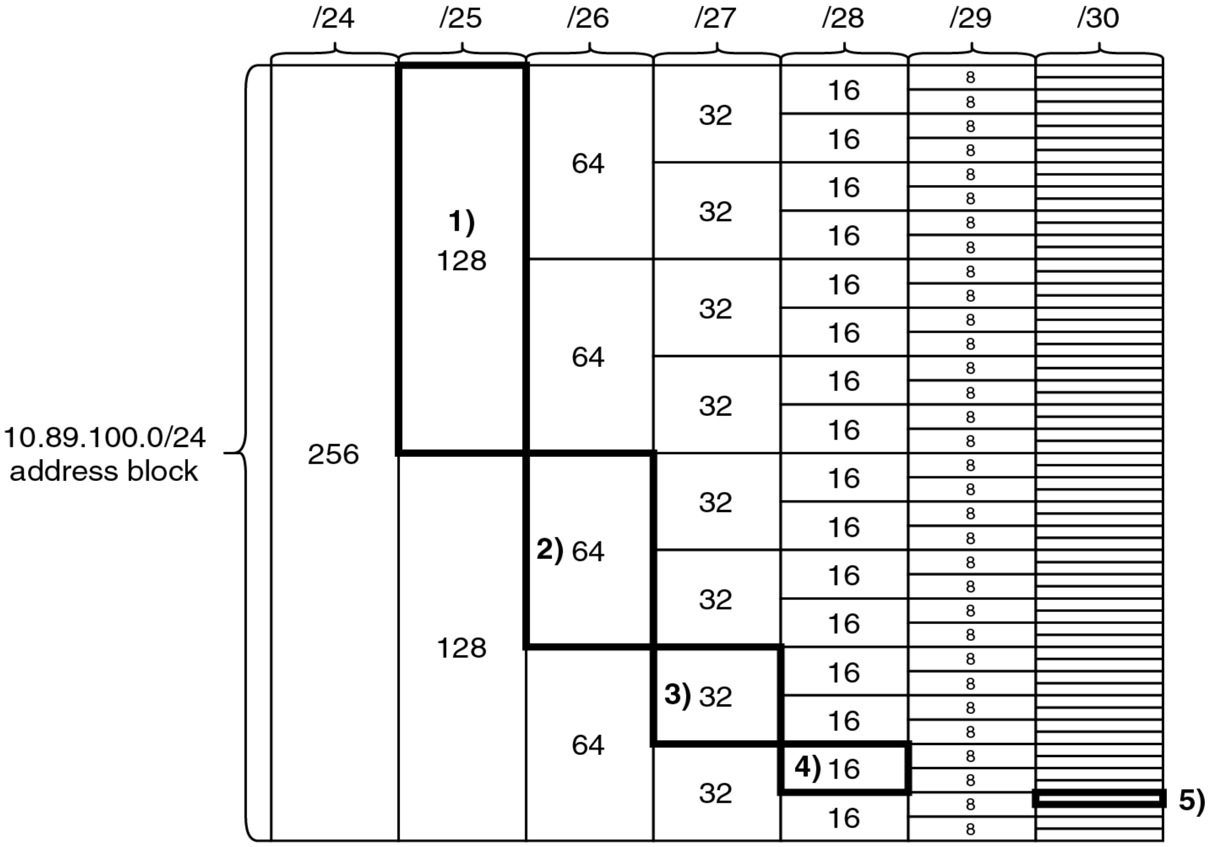


Assign the second-largest subnet after it.

1. Repeat the process until all subnets have been assigned.
2. The five subnets in figure 11.12, in order from largest to smallest, are: Toronto LAN A (122 hosts), Tokyo LAN A (59 hosts), Toronto LAN B (30 hosts), Tokyo LAN B (11 hosts), and the WAN connection (2 hosts) – so let's start by assigning Toronto LAN A.

Note: Router IP addresses are included in the "host" counts; any device with an IP address can be considered a host. The term end host is usually used to refer to PCs, servers, etc. to distinguish them from network infrastructure devices like routers.

Figure 11.13 The 10.89.100.0/24 address block, divided into five subnets of varying sizes. (1) Toronto LAN A is 10.89.100.0/25, containing 128 addresses. (2) Tokyo LAN A is 10.89.100.128/26, containing 64 addresses. (3) Toronto LAN B is 10.89.100.192/27, containing 32 addresses. (4) Tokyo LAN B is 10.89.100.224/28, containing 16 addresses. (5) The WAN connection is 10.89.100.240/30, containing 4 addresses. After subnetting, 12 addresses remain unused: 10.89.100.244 through 10.89.100.255.



1. **Toronto LAN A:** 10.89.100.0/25 (.0 to .127)
2. **Tokyo LAN A:** 10.89.100.128/26 (.128 to .191)
3. **Toronto LAN B:** 10.89.100.192/27 (.192 to .223)
4. **Tokyo LAN B:** 10.89.100.224/28 (.224 to .239)
5. **WAN:** 10.89.100.240/30 (.240 to .243)

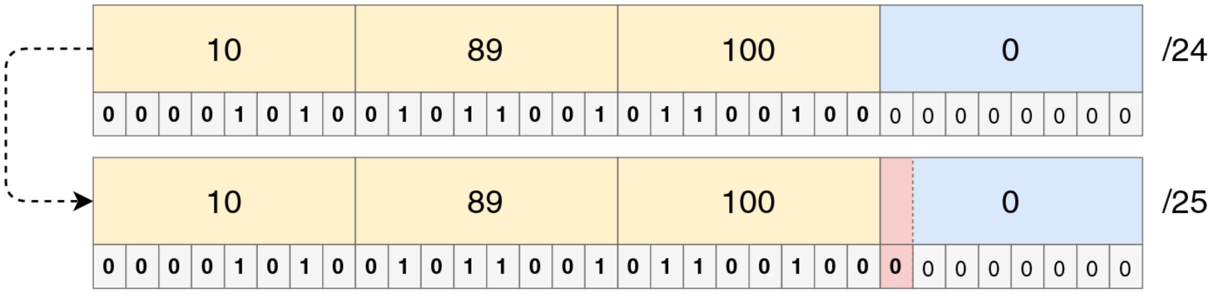
Remaining: 10.89.100.244 - 10.89.100.255

Note: Toronto LAN A requires 122 host addresses, so the question is: what is the minimum number of host bits required to provide at least 122 host addresses? Referring to the formula we use to calculate the maximum number of host bits in a subnet (2^y-2), what is the minimum *y* value that would serve our purpose? The answer is 7, because 2^7-2=126, only a few more addresses than we need. 6 host bits would give us only 62 host addresses (2^6-2) – not enough for the 122 hosts in Toronto LAN A. To leave seven host bits, we have to borrow one bit from the /24 address block. Figure 11.14 shows the resulting subnet when we borrow a single host bit from the 10.89.100.0/24 address block: 10.89.100.0/25 – Toronto LAN A's subnet! Figure 11.14 also lists the five attributes of the subnet.

11.3.1 Assigning Toronto LAN A's subnet

By assigning the 10.89.100.0/25 subnet to Toronto LAN A, we have already used half of the available address space: from 10.89.100.0 through 10.89.100.127. The entire 10.89.100.0/24 address block includes 256 addresses, and a single /25 subnet takes up 128 of those addresses (keep in mind that only 126 of those 128 addresses can be assigned to hosts). We will assign the rest of the subnets from the remaining range of addresses: 10.89.100.128 through 10.89.100.255.

Figure 11.14 Toronto LAN A's subnet. Borrowing 1 bit from the 10.89.100.0/24 address block results in the 10.89.100.0/25 subnet, supporting up to 126 host addresses—enough for the 122 hosts in the LAN.

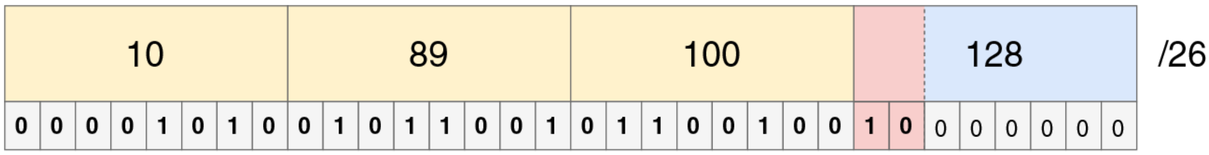


- **Network address:** 10.89.100.0
- **Broadcast address:** 10.89.100.127
- **First usable address:** 10.89.100.1
- **Last usable address:** 10.89.100.126
- **Maximum number of hosts:** 126

After assigning Toronto LAN A's subnet and identifying its five attributes, we can easily identify one more piece of information: Tokyo LAN A's network address. The last address (not the last *usable* address) in Toronto LAN A is 10.89.100.127 – the broadcast address. Without knowing any other information about Tokyo LAN A, we can identify its first IP address (the first address immediately after Toronto LAN A's broadcast address), which is 10.89.100.128. This is Tokyo LAN A's network address, because the first IP address of a subnet is the network address.

11.3.2 Assigning Tokyo LAN A's subnet

Figure 11.15 Tokyo LAN A's subnet. The network address is 10.89.100.128—the first address after Toronto LAN A. A /26 prefix length is used to allow for up to 62 host addresses—enough for the 59 hosts in the LAN.



- **Network address:** 10.89.100.128
- **Broadcast address:** 10.89.100.191
- **First usable address:** 10.89.100.129
- **Last usable address:** 10.89.100.190
- **Maximum number of hosts:** 62

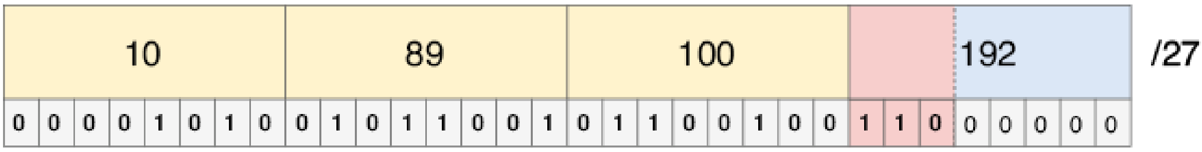
We have now assigned three quarters of the 10.89.100.0/24 address block: a /25 subnet (one half), and a /26 subnet (one quarter). The remaining range of addresses is 10.89.100.192 through 10.89.100.255, and we will assign the remaining three subnets from that range.

The IP address immediately after Tokyo LAN A's final address (its broadcast address) is 10.89.100.192, and that is Toronto LAN B's network address. What about its prefix length? Toronto LAN B requires IP addresses for at least 30 hosts, meaning 5 host bits are required, which gives exactly 30 (2^5-2) host addresses. Therefore, Toronto LAN B's subnet should use a /27 prefix length, resulting in subnet 10.89.100.192/27, Figure 11.16 shows the subnet and its five attributes.

11.3.3 Assigning Toronto LAN B's subnet

Note

Figure 11.16 Toronto LAN B's subnet. The network address is 10.89.100.192—the first address after Tokyo LAN A. A /27 prefix length is used to allow for up to 30 host addresses—exactly the amount needed.



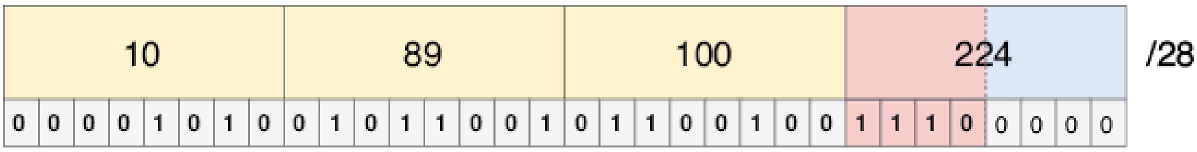
- **Network address:** 10.89.100.192
- **Broadcast address:** 10.89.100.223
- **First usable address:** 10.89.100.193
- **Last usable address:** 10.89.100.222
- **Maximum number of hosts:** 30

11.3.4 Assigning Tokyo LAN B's subnet

We can use the same process to assign Tokyo LAN B's subnet. Its network address is the first address after the previous LAN's (Toronto LAN B's) broadcast address, so Tokyo LAN B's network address is 10.89.100.224. Tokyo LAN B is a small LAN, requiring only 11 host addresses. Therefore only 4 host bits are required, allowing for up to 14 (2^4-2) host addresses, so Tokyo LAN B's subnet is 10.89.100.224/28. Figure 11.17 shows the subnet and its five attributes.

Only one 16th of the 10.89.100.0/24 address block remains – addresses 10.89.100.240 through 10.89.100.255. Fortunately, that is more than enough addresses for the final subnet: the WAN connection between R1 and R2.

Figure 11.17 Tokyo LAN B's subnet. The network address is 10.89.100.224—the first address after Toronto LAN B. A /28 prefix length is used to allow for up to 14 host addresses—sufficient for the 11 hosts in the LAN.



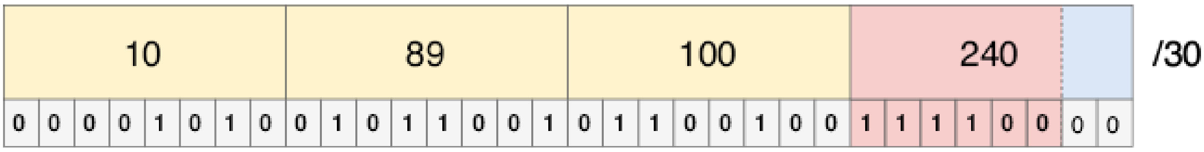
- **Network address:** 10.89.100.224
- **Broadcast address:** 10.89.100.239
- **First usable address:** 10.89.100.225
- **Last usable address:** 10.89.100.238
- **Maximum number of hosts:** 14

The WAN connection between R1 and R2 is a point-to-point connection, requiring only two host addresses. The network address is 10.89.100.240 (the first address after Tokyo LAN B's broadcast address), but what should the prefix length be? As we covered in section 11.2.1, there are two options: a /30 prefix length (two usable addresses, four addresses in total), or a /31 prefix length (two usable addresses, without network and broadcast addresses). Either prefix length is a valid choice, but for this example I'll use a /30 prefix length, so the WAN connection’s subnet is 10.89.100.240/30. Figure 11.18 shows the subnet and its five attributes.

11.3.5 Assigning the WAN connection’s subnet

Note

Figure 11.18 The WAN connection’s subnet. The network address is 10.89.100.240—the first address after Tokyo LAN B. A /30 prefix length is used to allow for two host addresses—one for R1, and one for R2.



- **Network address:** 10.89.100.240
- **Broadcast address:** 10.89.100.243
- **First usable address:** 10.89.100.241
- **Last usable address:** 10.89.100.242
- **Maximum number of hosts:** 2

11.4 Additional subnetting practice

To become proficient at subnetting, you need to practice. Fortunately, there are some free websites which generate subnetting problems you can solve. One example is https://www.subnetting.net/, but you can find others with a quick google search. I recommend spending a bit of time each day practicing subnetting for at least a week or two, or until you feel confident solving questions on the practice sites.

Exam scenarios

1. (multiple choice, multiple answers)

Some instructors teach shortcuts that can help you solve subnetting scenarios without having to think about the underlying binary; one famous example is called the “magic number” method. I do not agree with these methods because I think they only serve as a crutch, helping you to solve subnetting problems without understanding how subnetting actually works. It may seem cumbersome to always be thinking about binary, but with practice it will become effortless, and you’ll become better not just at subnetting, but at all of the other necessary skills that require proficiency with binary (I listed some in chapter 7).

A. /25 = 255.255.255.192

B. /15 = 255.252.0.0

C. /29 = 255.255.255.248

D. /27 = 255.255.255.240

E. /18 = 255.255.192.0

Each additional borrowed bit doubles the number of subnets that can be made: 1 borrowed bit = 2 subnets, 2 borrowed bits = 4 subnets, 3 borrowed bits = 8 subnets, etc. However, each additional borrowed bit halves the number of addresses in each subnet, because there are fewer bits in the host portion.

2. (drag and drop)

For point-to-point links (connections between two routers), either a /30 or /31 prefix length can be used. /30 consumes four addresses (network address, broadcast address, and two host addresses), whereas /31 consumes only two addresses (two host addresses, without a network or broadcast address).

(A) 10.23.24.128/25	10.23.24.65 – 10.23.24.79
(B) 10.23.24.128/27	10.23.24.65 – 10.23.24.78
(C) 10.23.24.64/26	10.23.24.65 – 10.23.24.126
(D) 10.23.24.64/28	10.23.24.129 – 10.23.24.254
	10.23.24.129 – 10.23.24.190
	10.23.24.129 – 10.23.24.158

The network address of the next subnet is the address immediately after the broadcast address of the current subnet.

Summary

- *Classless Inter-Domain Routing* (CIDR) replaced classful addressing, allowing prefix lengths outside of the traditional /8, /16, and /24.
- With CIDR, an address block can be divided into smaller networks called *subnets*. This process is called *subnetting*.
- *Fixed-Length Subnet Masking* (FLSM) subnetting divides an address block into subnets of equal size.
- *Variable-Length Subnet Masking* (VLSM) subnetting divides an address block into subnets of varying size.
- To subnet an address block, you “borrow” bits from the host portion of the address block and add them to the network portion. Whereas the network portion of the original address block cannot be changed, the borrowed bits can be changed to make different subnets.
- Each additional borrowed bit doubles the number of subnets that can be made: 1 borrowed bit = 2 subnets, 2 borrowed bits = 4 subnets, 3 borrowed bits = 8 subnets, etc. However, each additional borrowed bit halves the number of addresses in each subnet because there are fewer bits in the host portion.

- The five attributes of an IPv4 network are calculated in the same manner for subnets: the network address is the first address of a subnet (host portion of all 0s), the broadcast address is the last address of a subnet (host portion of all 1s), the first usable address is the first address after the network address, the last usable address is the last address before the broadcast address, and the maximum number of hosts is $2^y - 2$, where *y* is the number of host bits.
- For point-to-point links (connections between two routers), either a /30 or a /31 prefix length can be used. /30 consumes four addresses (network address, broadcast address, and two host addresses), whereas /31 consumes only two addresses (two host addresses, without a network or broadcast address).
- To subnet an address block using VLSM, assign the largest subnet at the start of the address block, assign the second-largest subnet after it, and repeat the process until all subnets have been assigned.
- The network address of the next subnet is the address immediately after the broadcast address of the current subnet.
- In a real-world situation, you should leave some room in each subnet for future growth. When doing subnetting scenarios for practice (or for the CCNA exam), be as efficient as possible (leave as few unused addresses as possible).