Chapter 19

Network Layer: Logical Addressing

19-1 IPv4 ADDRESSES

An IPv4 address is a 32-bit address that uniquely and universally defines the connection of a device (for example, a computer or a router) to the Internet.

Topics discussed in this section:

Address Space
Notations
Classful Addressing
Classless Addressing
Network Address Translation (NAT)

Note

An IPv4 address is 32 bits long.

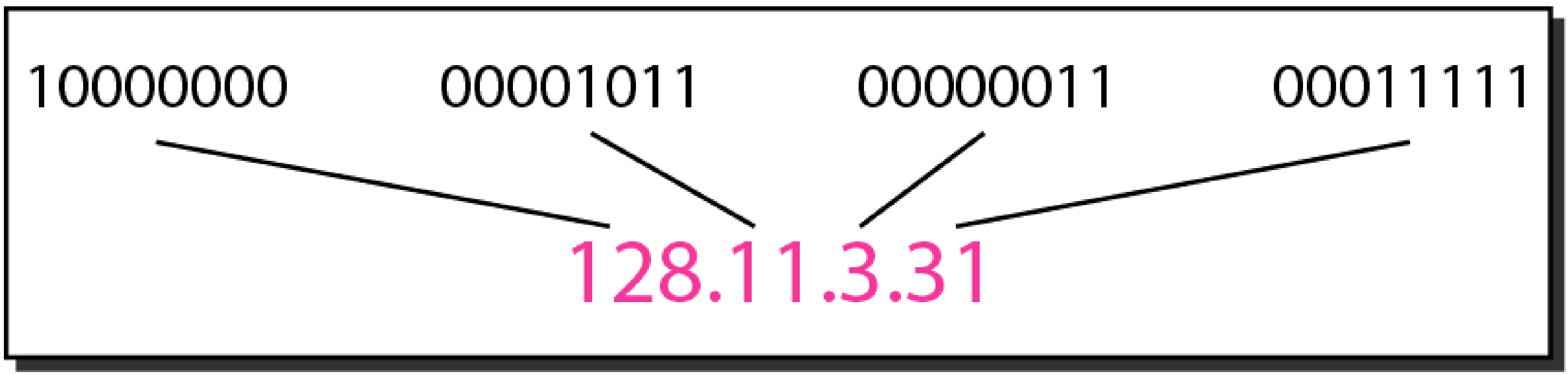
Note

The IPv4 addresses are unique and universal.

Note

The address space of IPv4 is 2^{32} or 4,294,967,296.

Figure 19.1 *Dotted-decimal notation and binary notation for an IPv4 address*



Note

Numbering systems are reviewed in Appendix B.

Example 19.1

Change the following IPv4 addresses from binary notation to dotted-decimal notation.

Solution

We replace each group of 8 bits with its equivalent decimal number (see Appendix B) and add dots for separation.

Example 19.2

Change the following IPv4 addresses from dotted-decimal notation to binary notation.

Solution

We replace each decimal number with its binary equivalent (see Appendix B).

Example 19.3

Find the error, if any, in the following IPv4 addresses.

The picture can't be displayed.

Solution

a. ***There must be no leading zero (045).***

b. ***There can be no more than four numbers.***

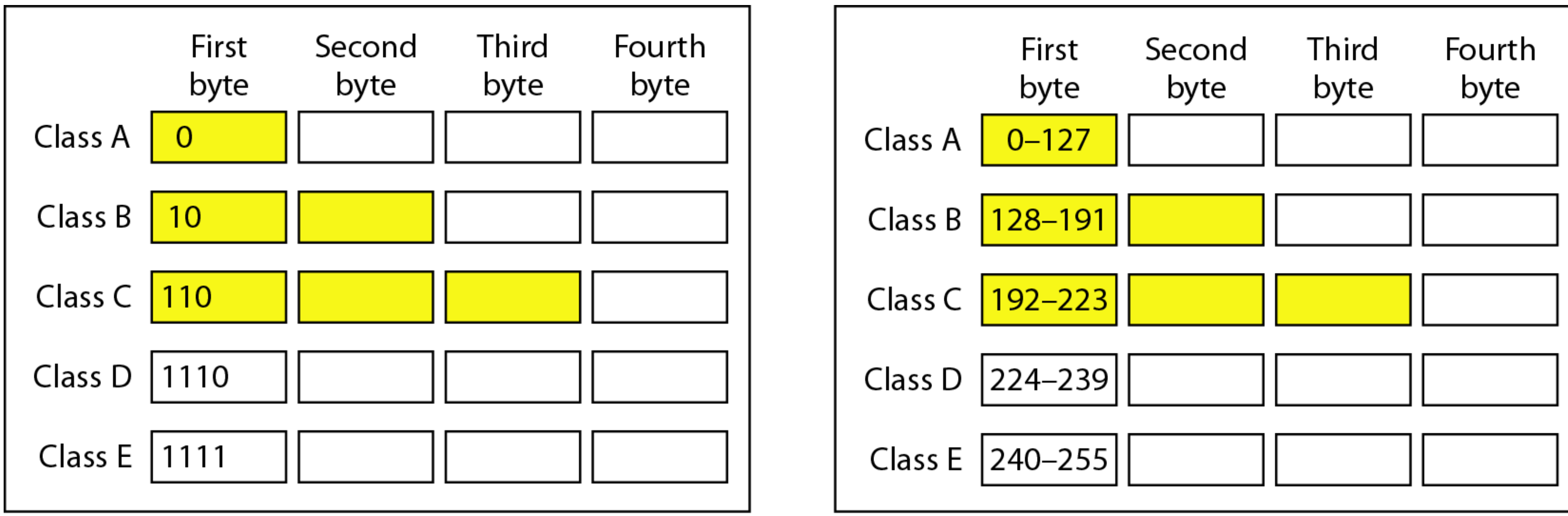
c. ***Each number needs to be less than or equal to 255.***

d. ***A mixture of binary notation and dotted-decimal notation is not allowed.***

Note

In classful addressing, the address space is divided into five classes: A, B, C, D, and E.

Figure 19.2 *Finding the classes in binary and dotted-decimal notation*



a. Binary notation

b. Dotted-decimal notation

Example 19.4

Find the class of each address.

a. <u>**0**</u>0000001 00001011 00001011 11101111

b. <u>**110**</u>00001 10000011 00011011 11111111

c. <u>**14**</u>.23.120.8

d. <u>**252**</u>.5.15.111

Solution

a. ***The first bit is 0. This is a class A address.***

b. ***The first 2 bits are 1; the third bit is 0. This is a class C address.***

c. ***The first byte is 14; the class is A.***

d. ***The first byte is 252; the class is E.***

Table 19.1 *Number of blocks and block size in classful IPv4 addressing*

Note

In classful addressing, a large part of the available addresses were wasted.

Table 19.2 *Default masks for classful addressing*

The picture can't be displayed.

Note

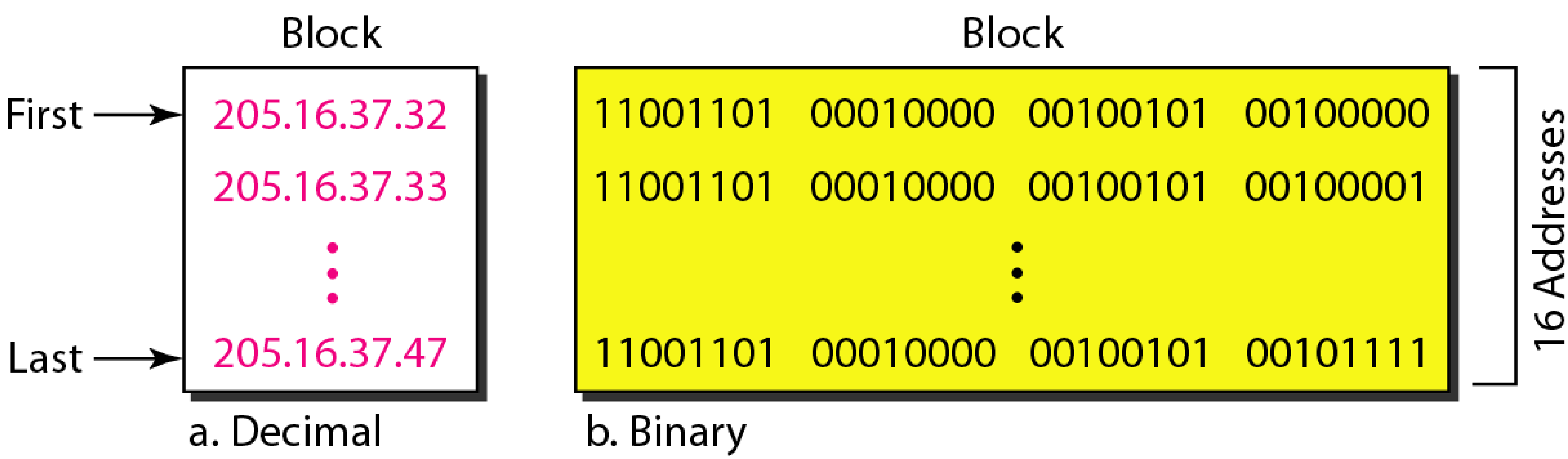
Classful addressing, which is almost obsolete, is replaced with classless addressing.

Example 19.5

Figure 19.3 shows a block of addresses, in both binary and dotted-decimal notation, granted to a small business that needs 16 addresses.

We can see that the restrictions are applied to this block. The addresses are contiguous. The number of addresses is a power of 2 ($16 = 2^4$), and the first address is divisible by 16. The first address, when converted to a decimal number, is 3,440,387,360, which when divided by 16 results in 215,024,210.

Figure 19.3 *A block of 16 addresses granted to a small organization*



Note

In IPv4 addressing, a block of addresses can be defined as

x.y.z.t /*n*

in which x.y.z.t defines one of the addresses and the /*n* defines the mask.

Note

The first address in the block can be found by setting the rightmost $32 - n$ bits to 0s.

Example 19.6

A block of addresses is granted to a small organization. We know that one of the addresses is 205.16.37.39/28. What is the first address in the block?

Solution

The binary representation of the given address is

11001101 00010000 00100101 00100111

If we set 32−28 rightmost bits to 0, we get

11001101 00010000 00100101 00100000

or

205.16.37.32.

This is actually the block shown in Figure 19.3.

Note

The last address in the block can be found by setting the rightmost 32 − n bits to 1s.

Example 19.7

Find the last address for the block in Example 19.6.

Solution

The binary representation of the given address is

11001101 00010000 00100101 00100111

If we set 32 − 28 rightmost bits to 1, we get

11001101 00010000 00100101 00101111

or

205.16.37.47

This is actually the block shown in Figure 19.3.

Note

The number of addresses in the block can be found by using the formula 2^{32-n}.

Example 19.8

Find the number of addresses in Example 19.6.

Solution

The value of n is 28, which means that number of addresses is 2 $^{32-28}$ or 16.

Example 19.9

Another way to find the first address, the last address, and the number of addresses is to represent the mask as a 32-bit binary (or 8-digit hexadecimal) number. This is particularly useful when we are writing a program to find these pieces of information. In Example 19.5 the /28 can be represented as

11111111 11111111 11111111 11110000

(twenty-eight 1s and four 0s).

Find

a. The first address

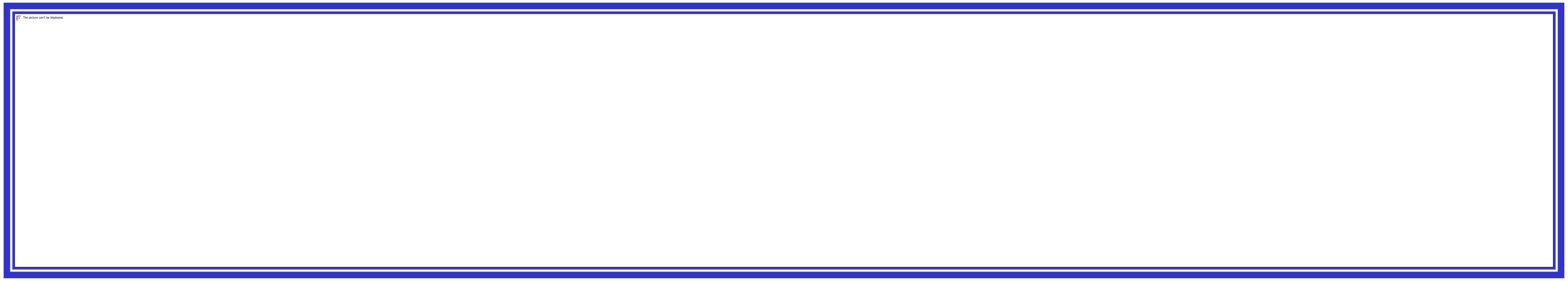
b. The last address

c. The number of addresses.

Example 19.9 (continued)

Solution

a. ***The first address can be found by ANDing the given addresses with the mask. ANDing here is done bit by bit. The result of ANDing 2 bits is 1 if both bits are 1s; the result is 0 otherwise.***



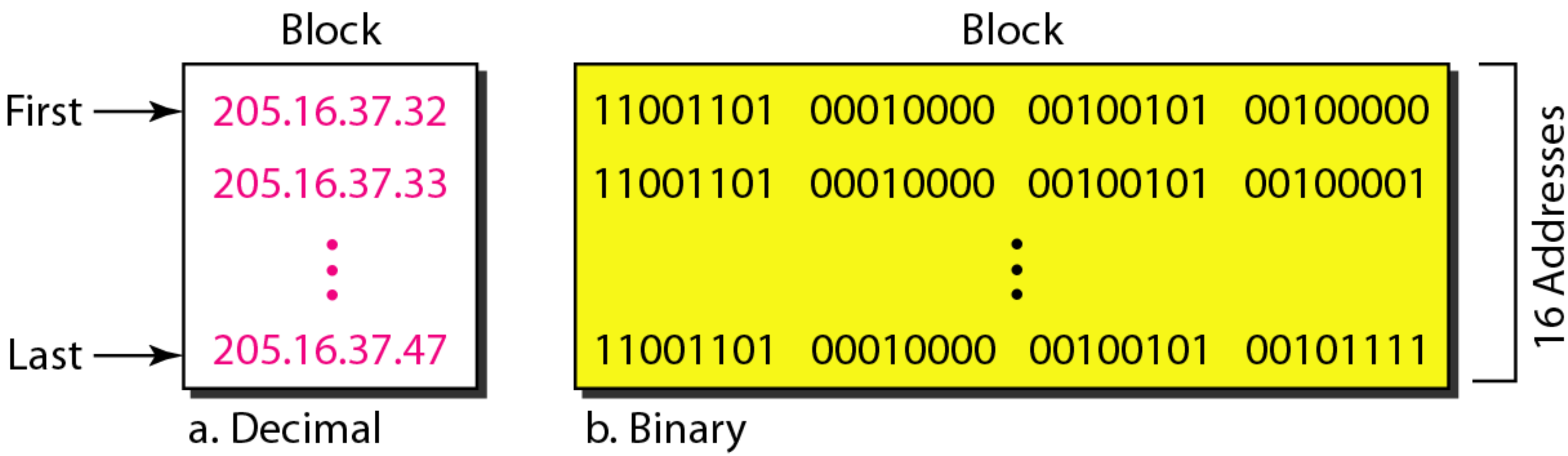
b. ***The last address can be found by ORing the given addresses with the complement of the mask. ORing here is done bit by bit. The result of ORing 2 bits is 0 if both bits are 0s; the result is 1 otherwise. The complement of a number is found by changing each 1 to 0 and each 0 to 1.***

Example 19.9 (continued)

c. *The number of addresses can be found by complementing the mask, interpreting it as a decimal number, and adding 1 to it.*

Mask complement:	000000000 00000000 00000000 00001111
Number of addresses:	15 + 1 = 16

Figure 19.4 *A network configuration for the block 205.16.37.32/28*



Note

The first address in a block is normally not assigned to any device; it is used as the network address that represents the organization to the rest of the world.

Figure 19.5 *Two levels of hierarchy in an IPv4 address*

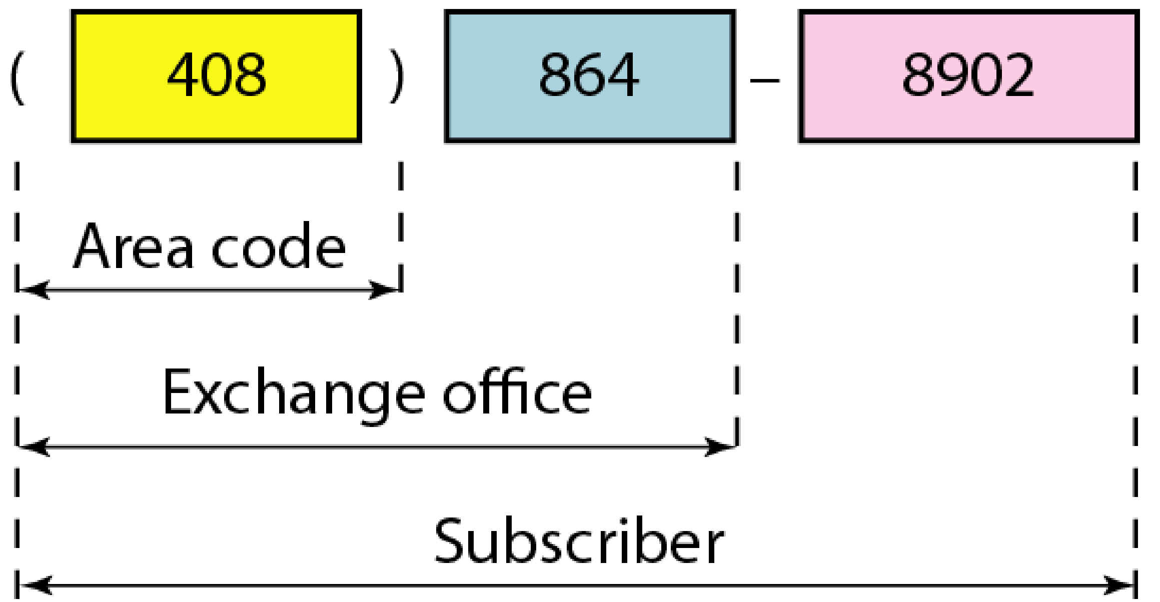
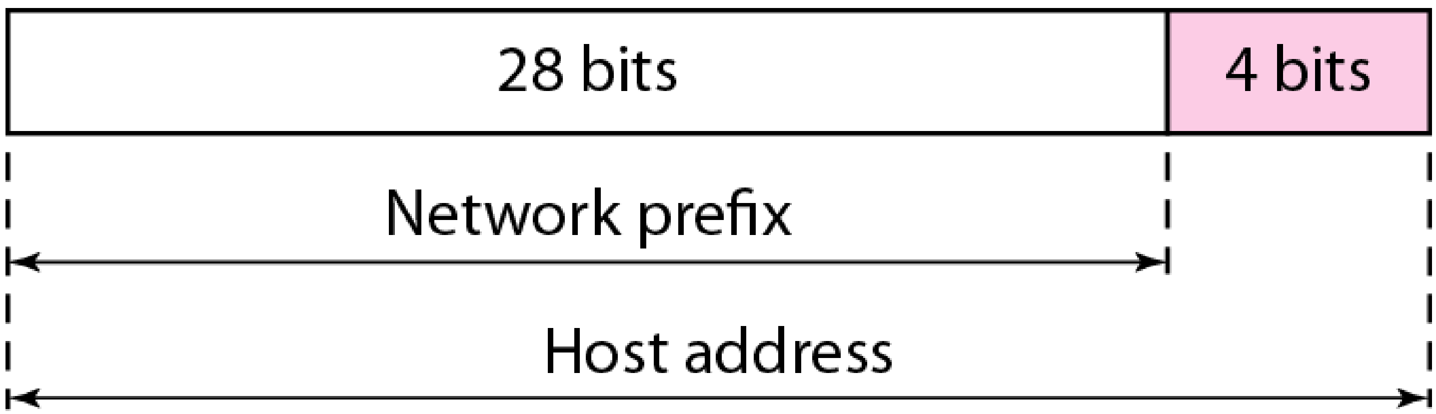


Figure 19.6 *A frame in a character-oriented protocol*



Note

Each address in the block can be considered as a two-level hierarchical structure: the leftmost *n* bits (prefix) define the network; the rightmost 32 − n bits define the host.

Figure 19.7 *Configuration and addresses in a subnetted network*

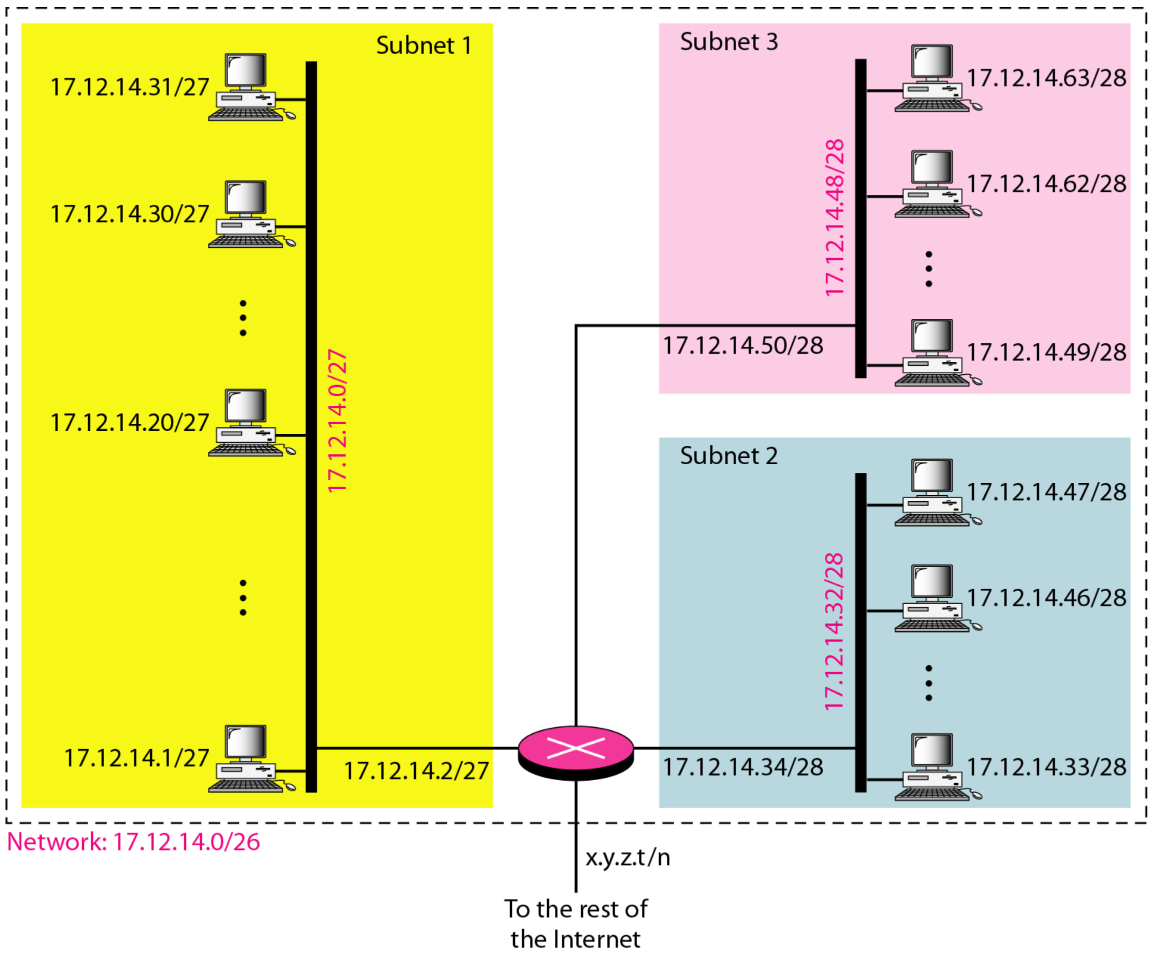
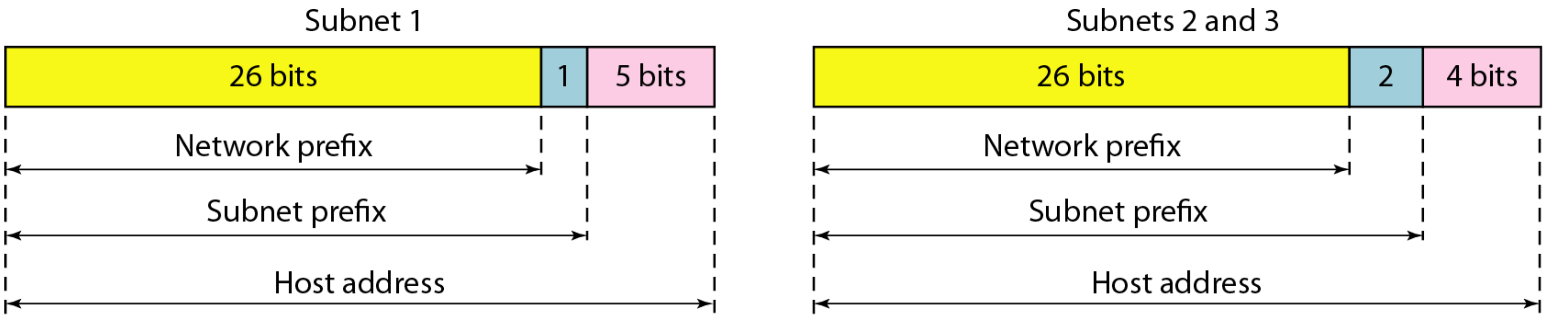


Figure 19.8 *Three-level hierarchy in an IPv4 address*



Example 19.10

An ISP is granted a block of addresses starting with 190.100.0.0/16 (65,536 addresses). The ISP needs to distribute these addresses to three groups of customers as follows:

a. ***The first group has 64 customers; each needs 256 addresses.***

b. ***The second group has 128 customers; each needs 128 addresses.***

c. ***The third group has 128 customers; each needs 64 addresses.***

Design the subblocks and find out how many addresses are still available after these allocations.

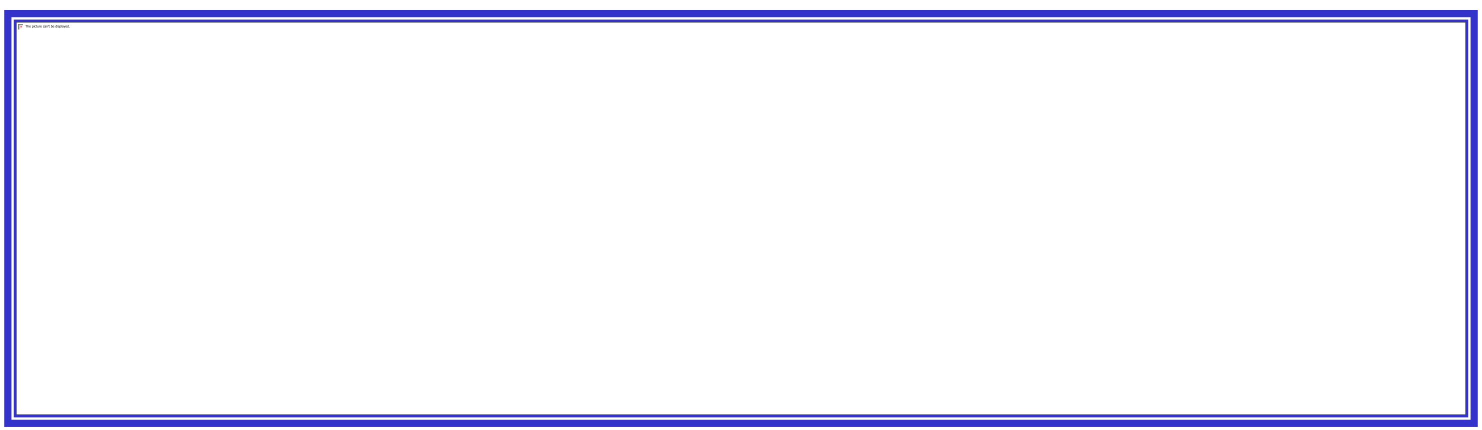
Example 19.10 (continued)

Solution

Figure 19.9 shows the situation.

Group 1

For this group, each customer needs 256 addresses. This means that 8 (log2 256) bits are needed to define each host. The prefix length is then 32 − 8 = 24. The addresses are



Example 19.10 (continued)

Group 2

For this group, each customer needs 128 addresses. This means that 7 (log2 128) bits are needed to define each host. The prefix length is then 32 − 7 = 25. The addresses are

Example 19.10 (continued)

Group 3

For this group, each customer needs 64 addresses. This means that 6 ($\log_2 64$) bits are needed to each host. The prefix length is then 32 − 6 = 26. The addresses are

Number of granted addresses to the ISP: 65,536
Number of allocated addresses by the ISP: 40,960
Number of available addresses: 24,576

Figure 19.9 *An example of address allocation and distribution by an ISP*

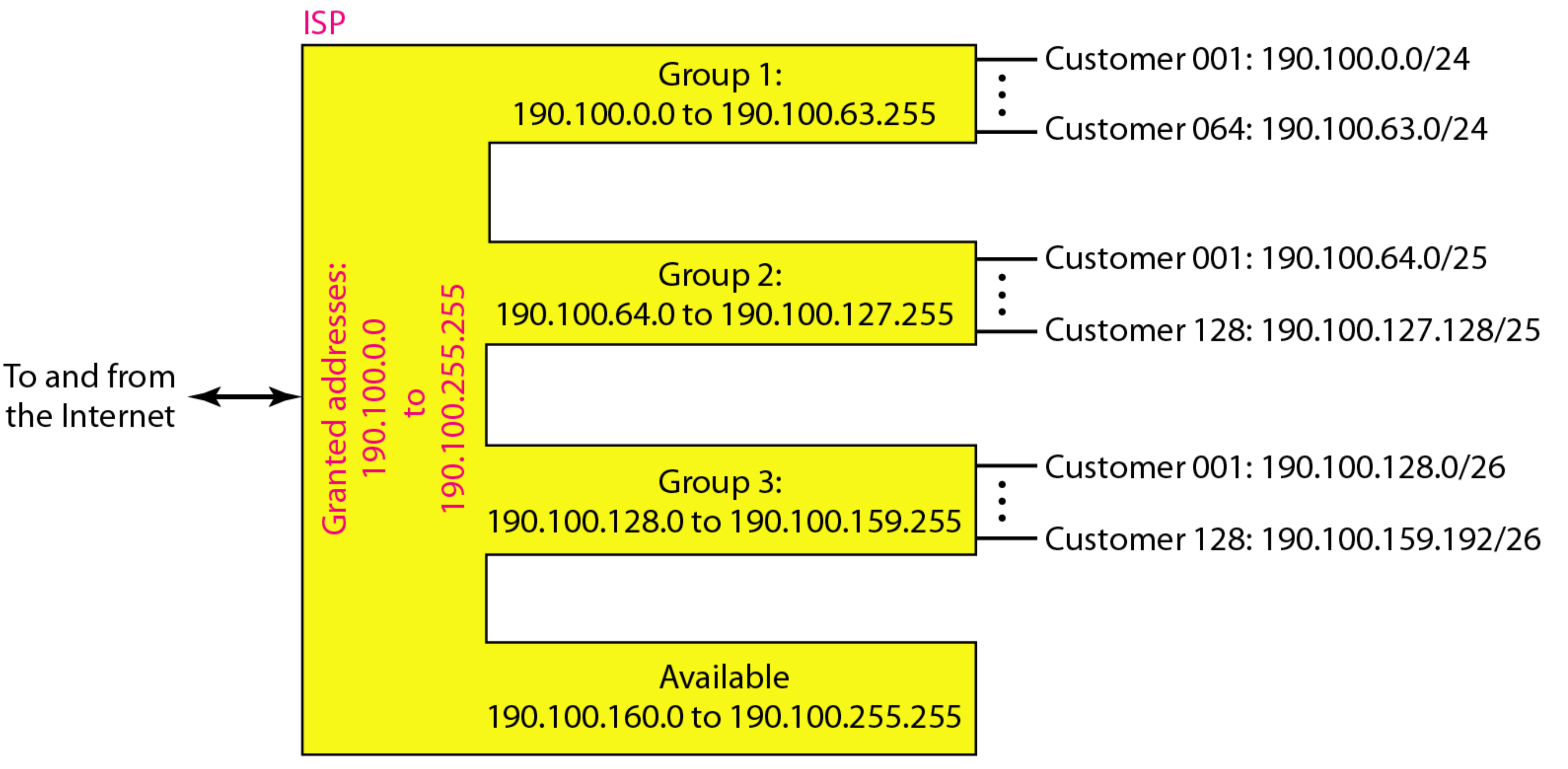


Table 19.3 ***Addresses for private networks***

Range			*Total*
10.0.0.0	to	10.255.255.255	2^{24}
172.16.0.0	to	172.31.255.255	2^{20}
192.168.0.0	to	192.168.255.255	2^{16}

Figure 19.10 *A NAT implementation*

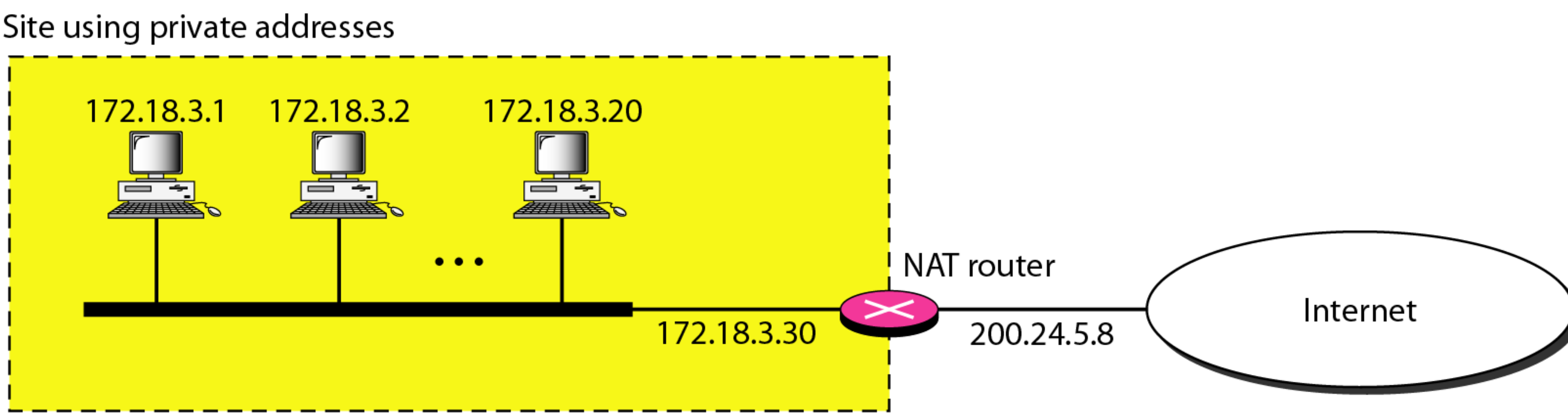


Figure 19.11 *Addresses in a NAT*

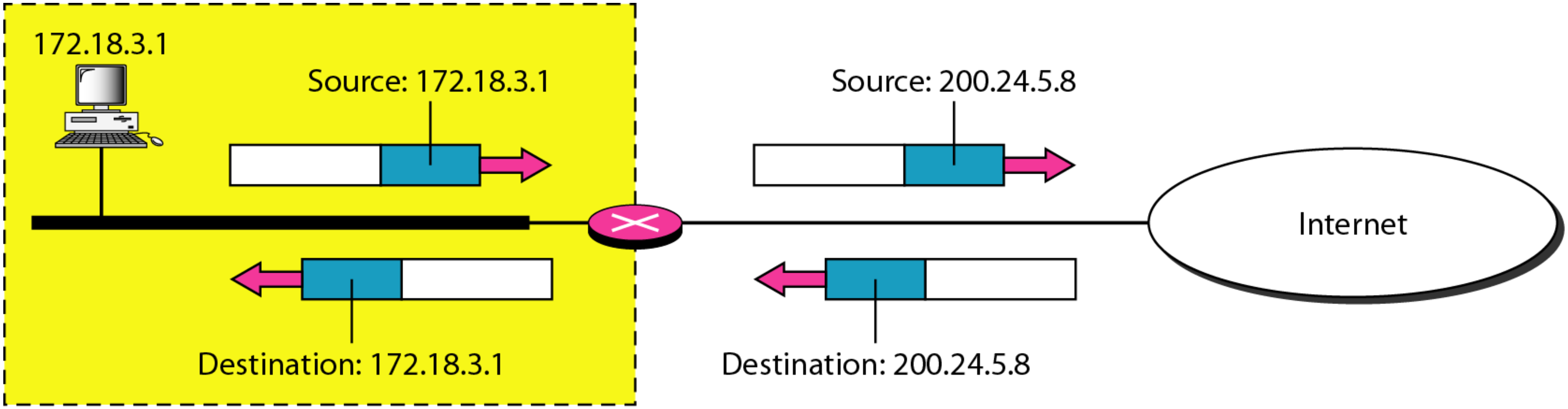


Figure 19.12 *NAT address translation*

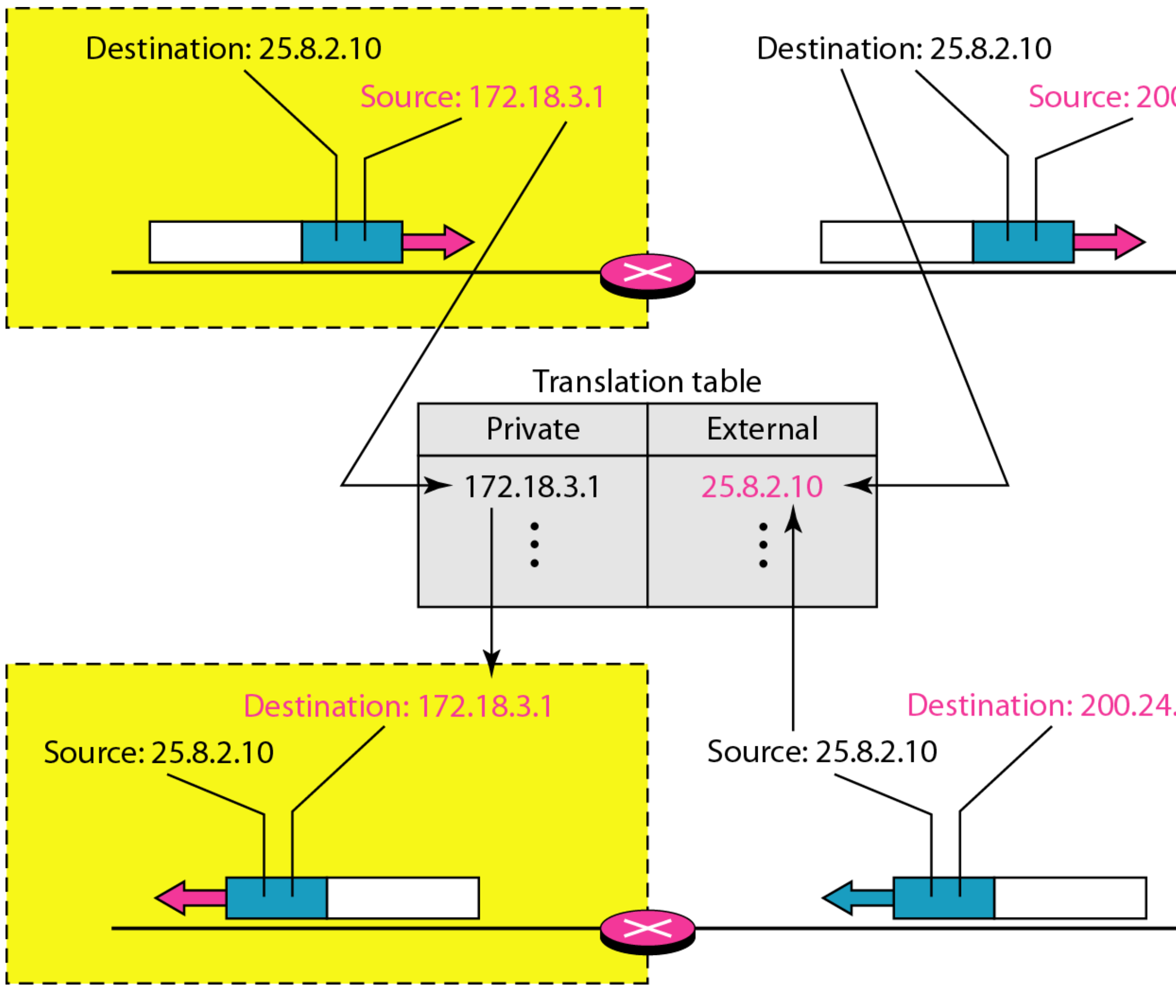
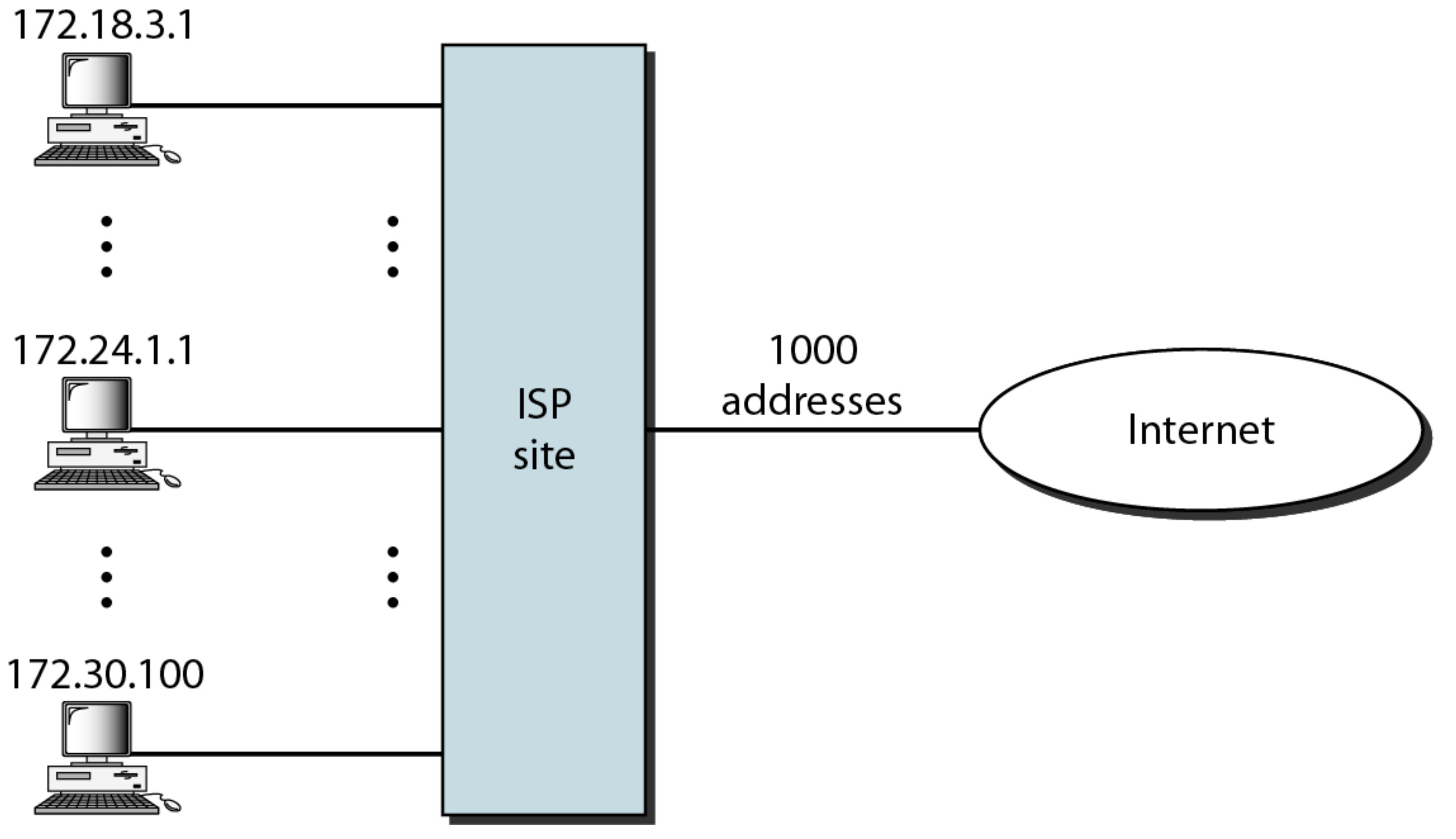


Table 19.4 *Five-column translation table*

The picture can't be displayed.

Figure 19.13 *An ISP and NAT*



19-2 IPv6 ADDRESSES

Despite all short-term solutions, address depletion is still a long-term problem for the Internet. This and other problems in the IP protocol itself have been the motivation for IPv6.

Topics discussed in this section:

Structure
Address Space

Note

An IPv6 address is 128 bits long.

Figure 19.14 *IPv6 address in binary and hexadecimal colon notation*

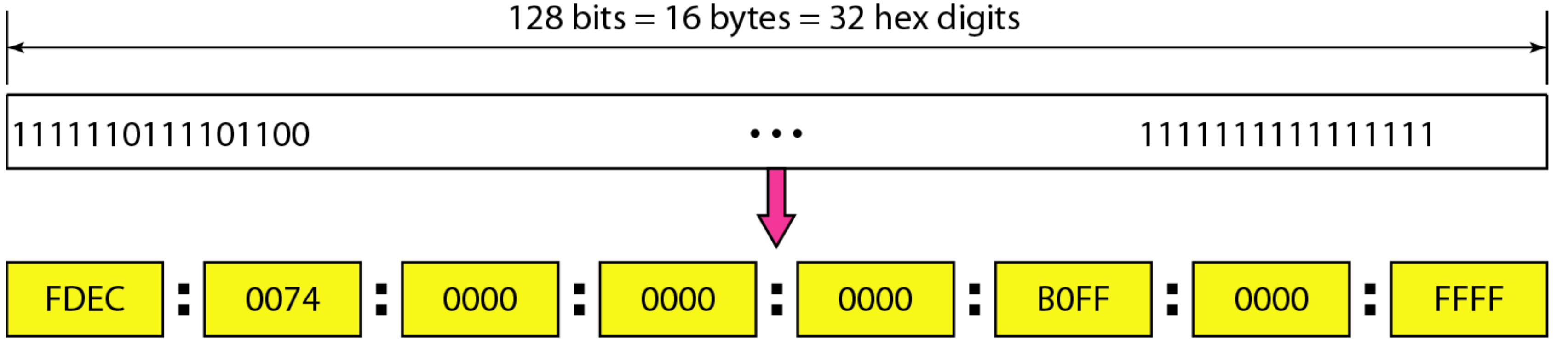
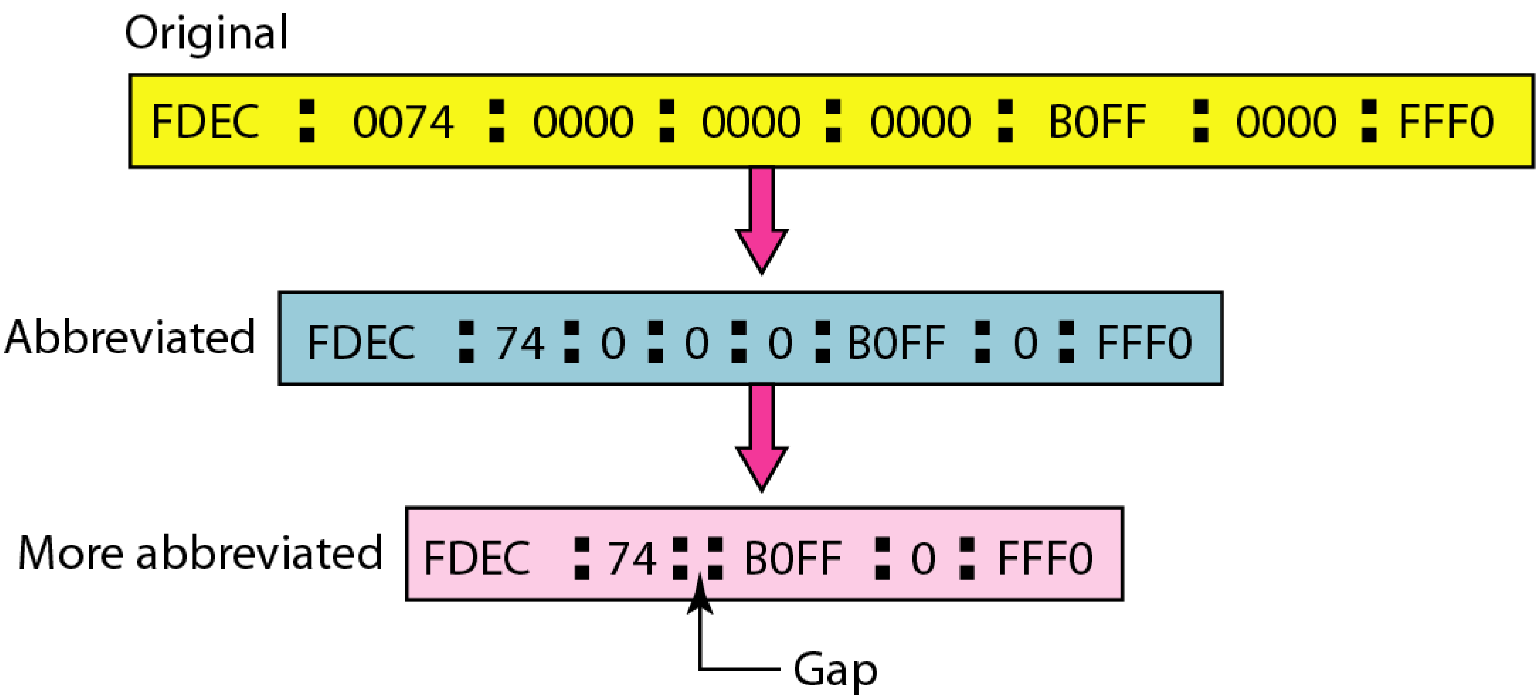


Figure 19.15 *Abbreviated IPv6 addresses*



Example 19.11

Expand the address 0:15::1:12:1213 to its original.

Solution

We first need to align the left side of the double colon to the left of the original pattern and the right side of the double colon to the right of the original pattern to find how many 0s we need to replace the double colon.

This means that the original address is.

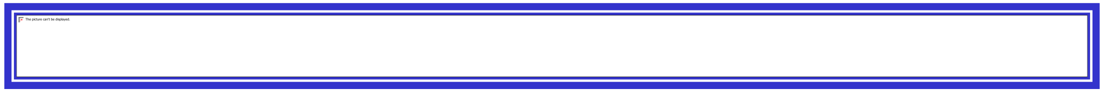


Table 19.5 *Type prefixes for IPv6 addresses*

The picture can't be displayed.

Table 19.5 *Type prefixes for IPv6 addresses (continued)*

Figure 19.16 *Prefixes for provider-based unicast address*

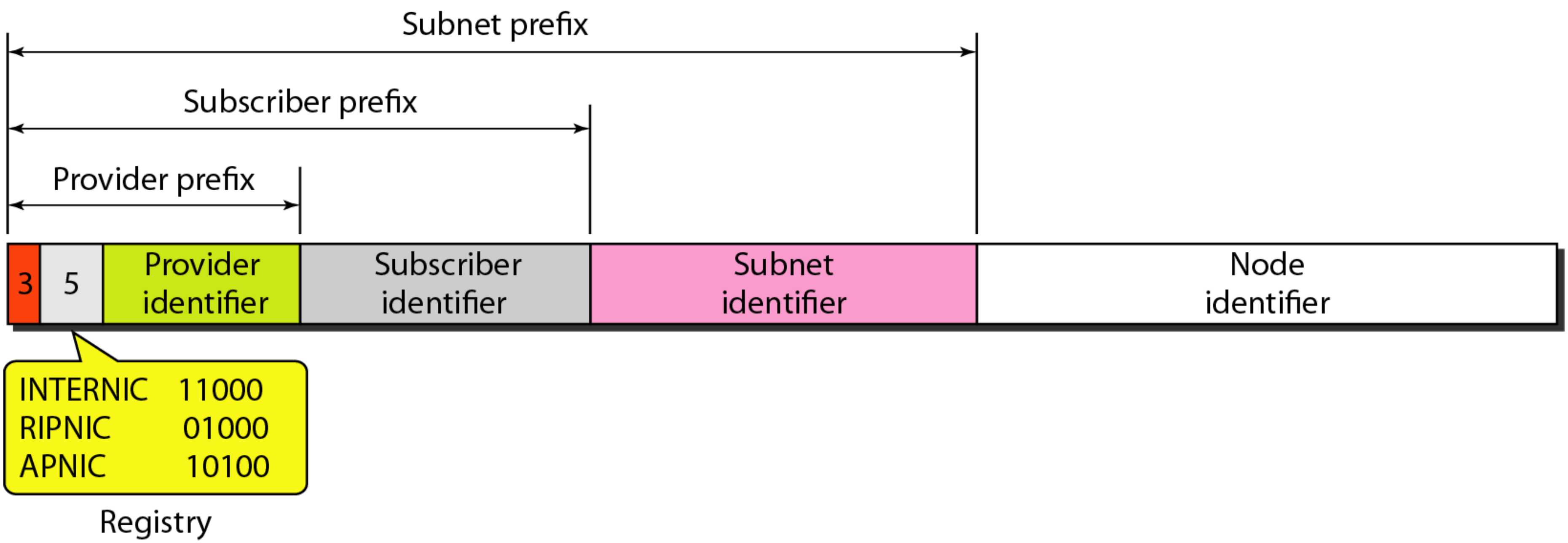


Figure 19.17 *Multicast address in IPv6*

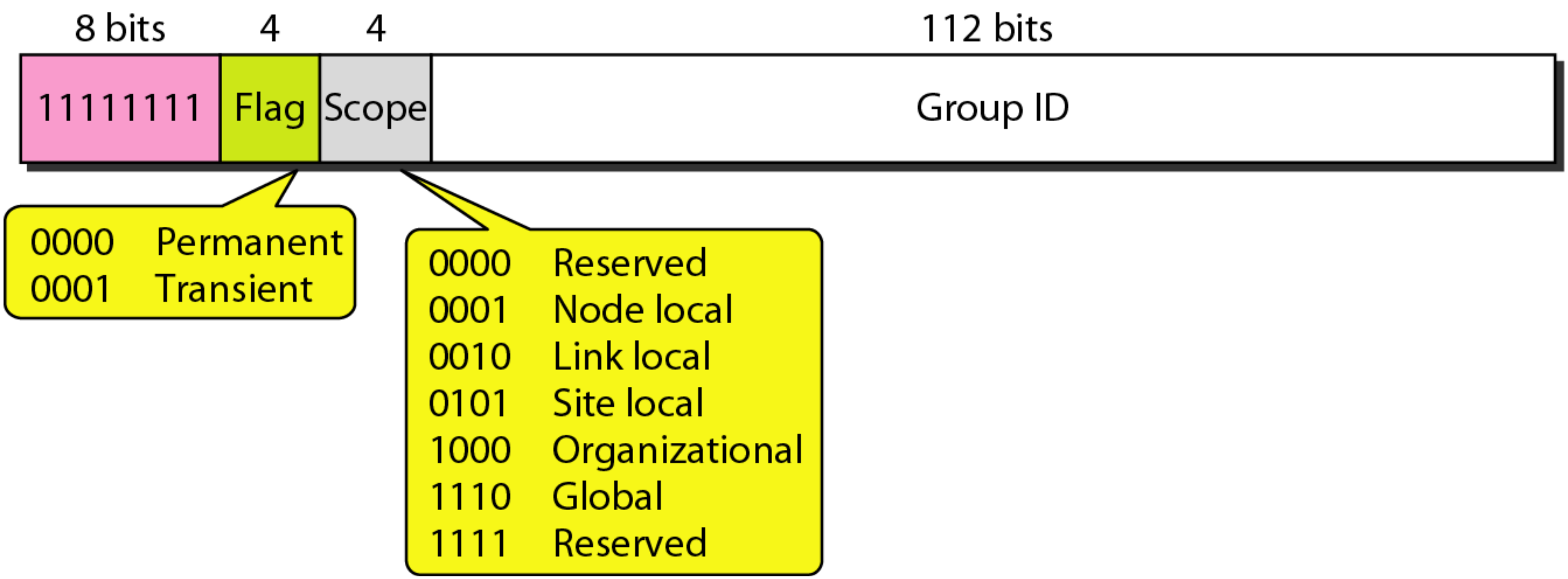


Figure 19.18 *Reserved addresses in IPv6*

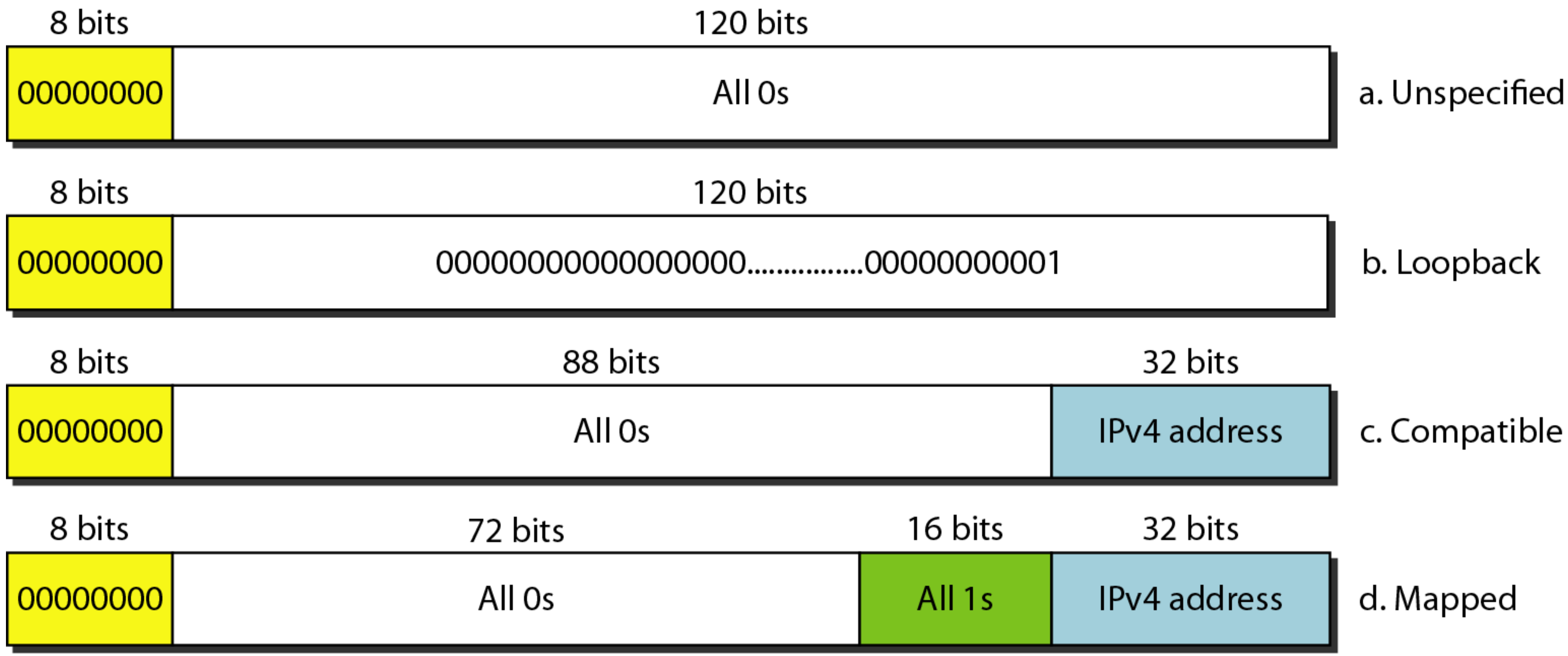


Figure 19.19 *Local addresses in IPv6*