Lecture 9

Network Layer: Logic Addressing

Section 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
How to assigning addresses
Network Address Translation (NAT)

Note

An IPv4 address is 32 bits long.

Note

The IPv4 addresses are unique and universal.

They are unique in the sense that each address defines one, and only one, connection to the Internet. Two devices on the Internet can never have the same address at the same time. By using some strategies, an address may be assigned to a device for a time period and then taken away and assigned to another device.

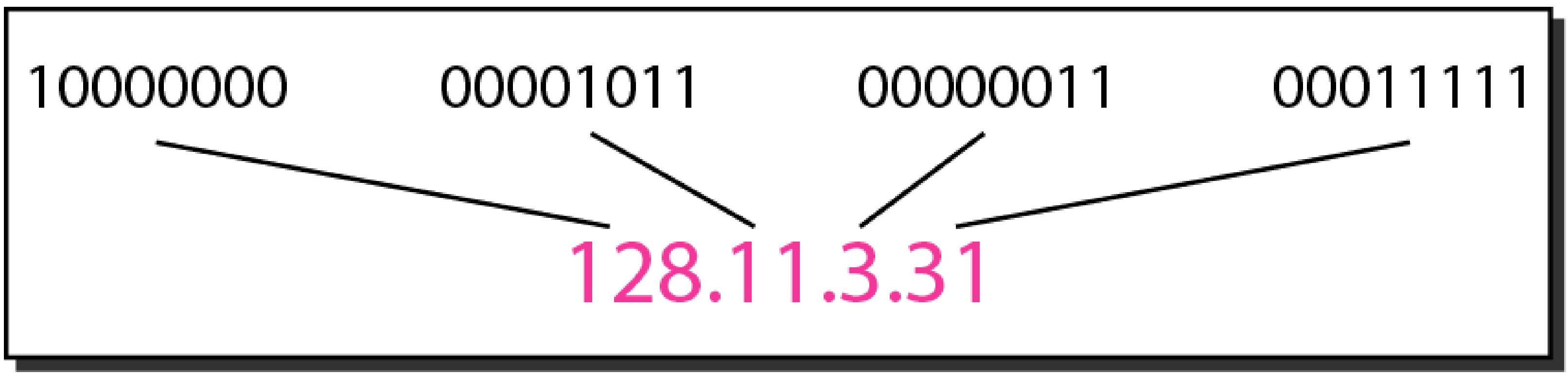
If a device operating at the network layer has m connections to the Internet, it needs to have m addresses. A router is such a device.

They IPv4 addresses are universal in the sense that the addressing system must be accepted by any host that wants to be connected to the Internet.

Note

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

Binary notation and dotted-decimal notation for an IPv4 address



Dotted-decimal notation is compact and easier to read, in which each number is a value ranging from 0 to 255

Example 1

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

a. 10000001 00001011 00001011 11101111

b. 11000001 10000011 00011011 11111111

Solution

We replace each group of 8 bits with its equivalent decimal number and add dots for separation.

a. 129.11.11.239

b. 193.131.27.255

Example 2

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

a. 111.56.45.78

b. 221.34.7.82

Solution

We replace each decimal number with its binary equivalent.

a. 01101111 00111000 00101101 01001110

b. 11011101 00100010 00000111 01010010

Example 3

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

a. 111.56.045.78

b. 221.34.7.8.20

c. 75.45.301.14

d. 11100010.23.14.67

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.

Mask

Classless Inter-Domain Routing

Binary	*Dotted-Decimal*	*CIDR*
11111111 00000000 00000000 00000000	255.0.0.0	/8
11111111 11111111 00000000 00000000	255.255.0.0	/16
11111111 11111111 11111111 00000000	255.255.255.0	/24

For each organization, we assign a number of addresses, in which the first few bits of each address are common, represented by a block of bit 1's in the above table. The remaining bits can change, represented by a block of bit 0's.

The mask (also called default mask) is defined as a 32-bit number made of contiguous 1s followed by contiguous 0s.

The mask for the first organization has eight 1s, which means the first 8 bits define the netid, and the next 24 bits define the hostid.

The last column of the above table shows the mask in the form "/n", where n is the number of common bits.

Rules for assigning addresses:

1. The addresses in a block must be contiguous, one after another.

2. The number of addresses in a block must be a power of 2 (1, 2, 4, 8, …).
If the number of bits in netid is n, then the number of bits in hostid is 32-n.
So the number of addresses in the block is 2^{32-n}

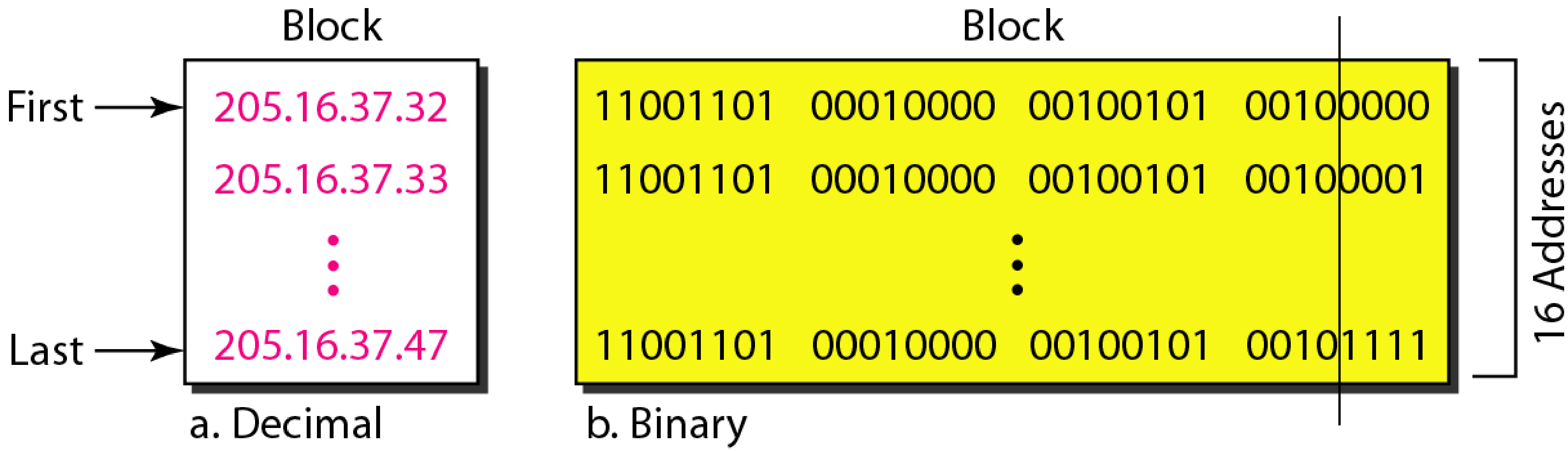
3. The first address: the hostid bits should be all zeros.

Example 5 (for blocks)

Figure on next slide 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 bits in netid is 28. The number of bits in hostid is 4. The number of addresses is 2^4.
The first address: the last four bits (in hostid) are all zeros.

A block of 16 addresses granted to a small organization



16 addresses: the last four bits change from 0000 to 1111

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.

n bits in netid; 32-n bits in hostid. Total address: 2^{32-n}

→ 32-n = $\log_2$ (number of addresses in the block).

For the block of addresses, the rightmost 32-n bits varies from all 0s to all 1s.
The first address in the block can be found by setting the rightmost 32 − n bits to 0s.
The last address in the block can be found by setting the rightmost 32 − n bits to 1s.

For the example in the preceding slide, the block of addresses can be defined as 205.16.37.32/28

Example 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? The last address? The number of addresses?

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.

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

11001101 00010000 00100101 00101111

or

205.16.37.47

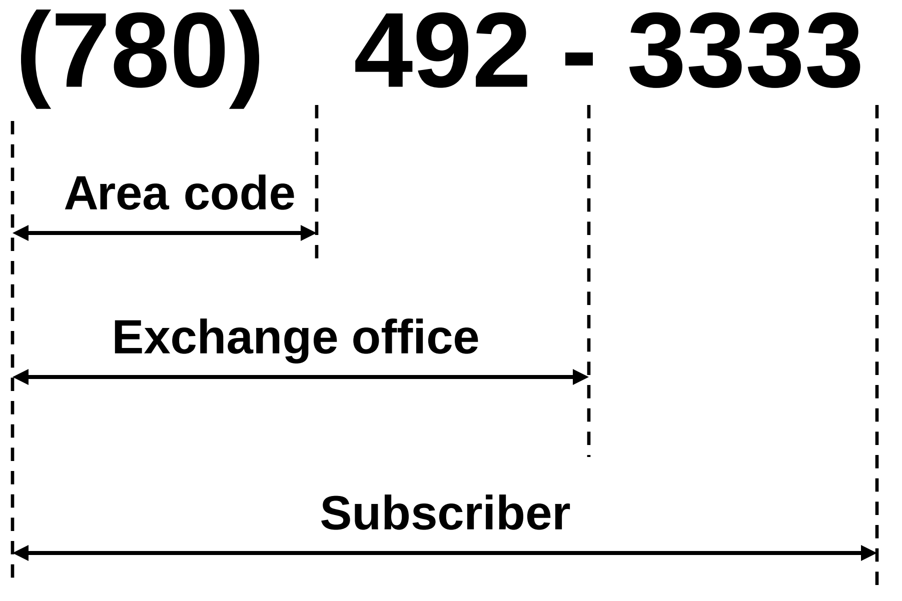
The value of n is 28, which means that number of addresses is 2^{32-28} or 16.

This is actually the block shown on slide 13.

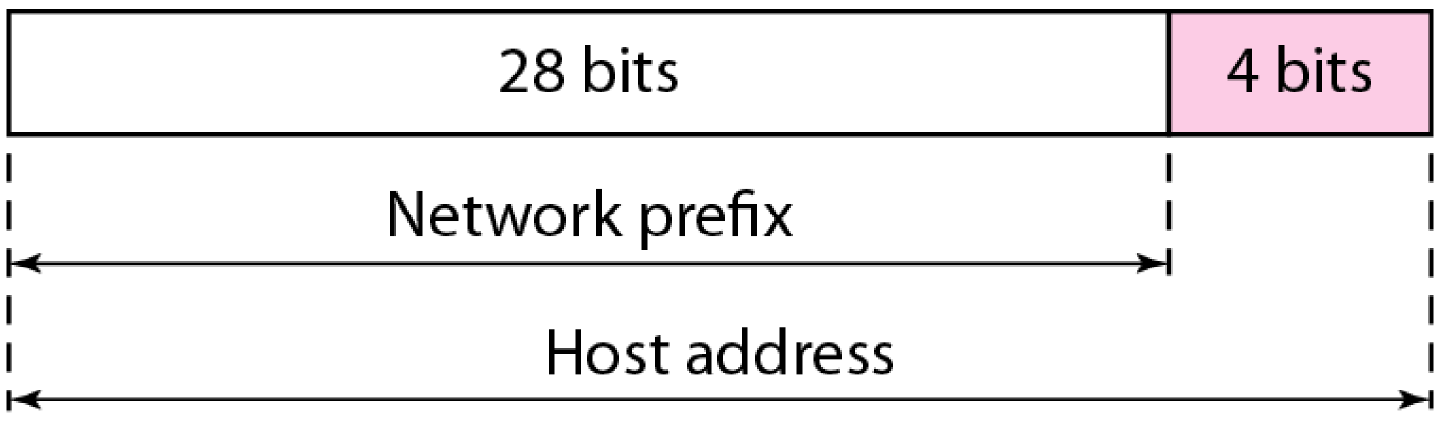
Note

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.

Hierarchy in a telephone network in north America



Two levels of hierarchy in an IPv4 address



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.

Three-Levels of Hierarchy: Subnetting

An organization that is granted a large block of addresses may want to create clusters of networks (called subnets) and divide the addresses between the different subnets. The rest of the world still sees the organization as one entity; however, internally there are several subnets. All messages are sent to the router address that connects the organization to the rest of the Internet; the router routes the message to the appropriate subnets. The organization, however, needs to create small subblocks of addresses, each assigned to specific subnets. The organization has its own mask; each subnet must also have its own.

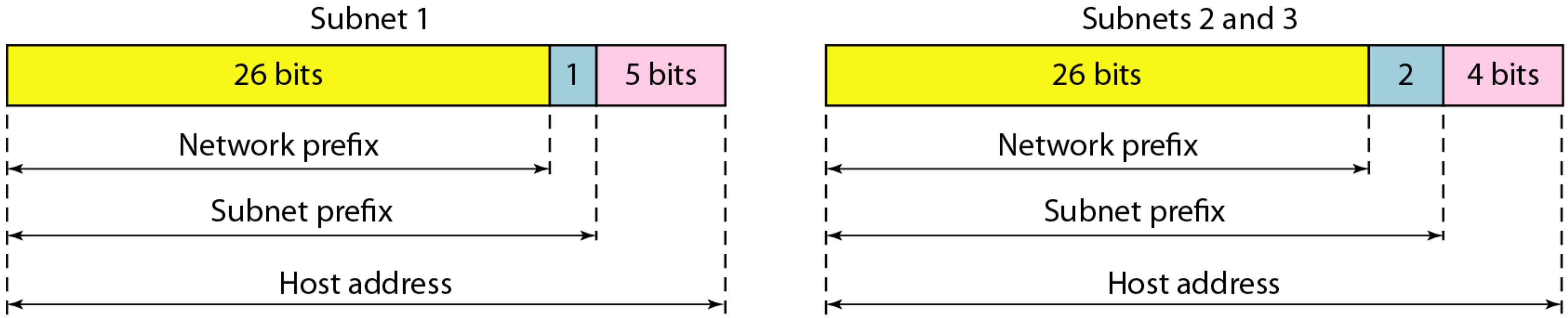
As an example, suppose an organization is given the block 17.12.14.0/26, which contains 2^{32-26}=64 addresses. The organization has three offices and needs to divide the addresses into three subblocks of 32, 16, and 16 addresses.
We can find the new masks as follows:

- the first subnet has 32 addresses. Assume its mask is n1.

  $2^{32-n1}=32$ → Then 32-n1 = $\log_2 32$, and we get n1=27.
- the second subnet has 16 addresses. Assume its mask is n2.

  $2^{32-n2}=16$ → Then 32-n2 = $\log_2 16$, and we get n2=28.
- the third subnet has 16 addresses. Assume its mask is n3.

  $2^{32-n3}=16$ → Then 32-n3 = $\log_2 16$, and we get n3=28.



17 12 14

00010001 00001100 00001110 00000000 first addr. of subnet 1

⋮

17.12.14.0/27

00010001 00001100 00001110 00011111 last addr. of subnet 1

00010001 00001100 00001110 00100000 first addr. of subnet 2

⋮

17.12.14.32/28

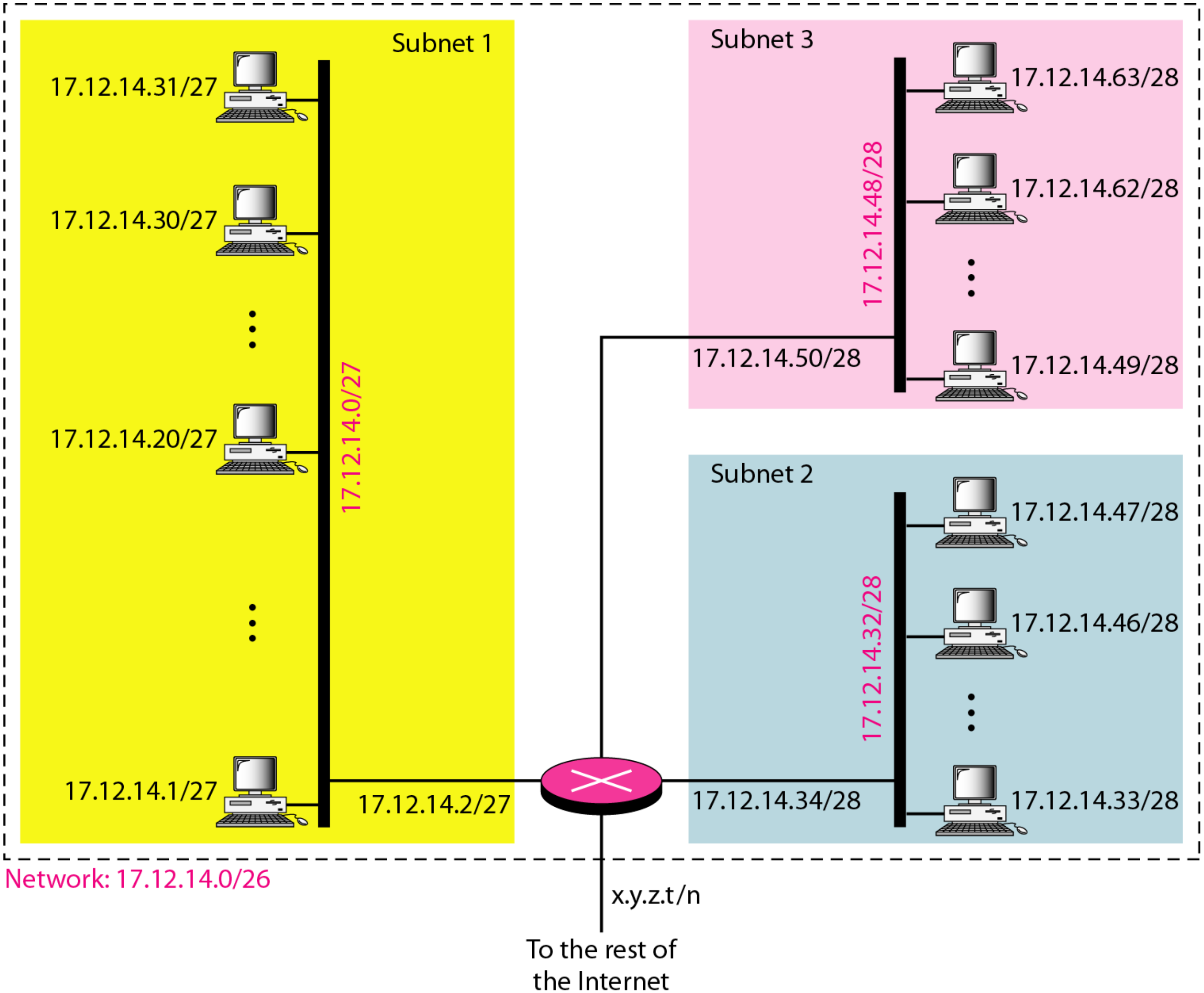
00010001 00001100 00001110 00101111 last addr. of subnet 2

00010001 00001100 00001110 00110000 first addr. of subnet 3

17.12.14.48/28

00010001 00001100 00001110 00111111 last addr. of subnet 3

Configuration and addresses in a subnetted network



Example 10

An ISP is granted a block of addresses starting with 190.100.0.0/16 (2^{32-16}=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 10 (continued)

Solution

Group 1

For this group, each customer needs 256 addresses. This means that 8 ($\log_2 256$) bits are needed to define each host. The prefix length is then 32 − 8 = 24. The addresses are

1st Customer:	*190.100.0.0/24*	*190.100.0.255/24*
2nd Customer:	*190.100.1.0/24*	*190.100.1.255/24*
. . .		
64th Customer:	*190.100.63.0/24*	*190.100.63.255/24*
Total = 64 × 256 = 16,384		

Example 10 (continued)

Group 2

For this group, each customer needs 128 addresses. This means that 7 ($\log_2 128$) bits are needed to define each host. The prefix length is then 32 − 7 = 25. The addresses are

1st Customer:	*190.100.64.0/25*	*190.100.64.127/25*
2nd Customer:	*190.100.64.128/25*	*190.100.64.255/25*
. . .		
128th Customer:	*190.100.127.128/25*	*190.100.127.255/25*
Total = 128 × 128 = 16,384		

Example 10 (continued)

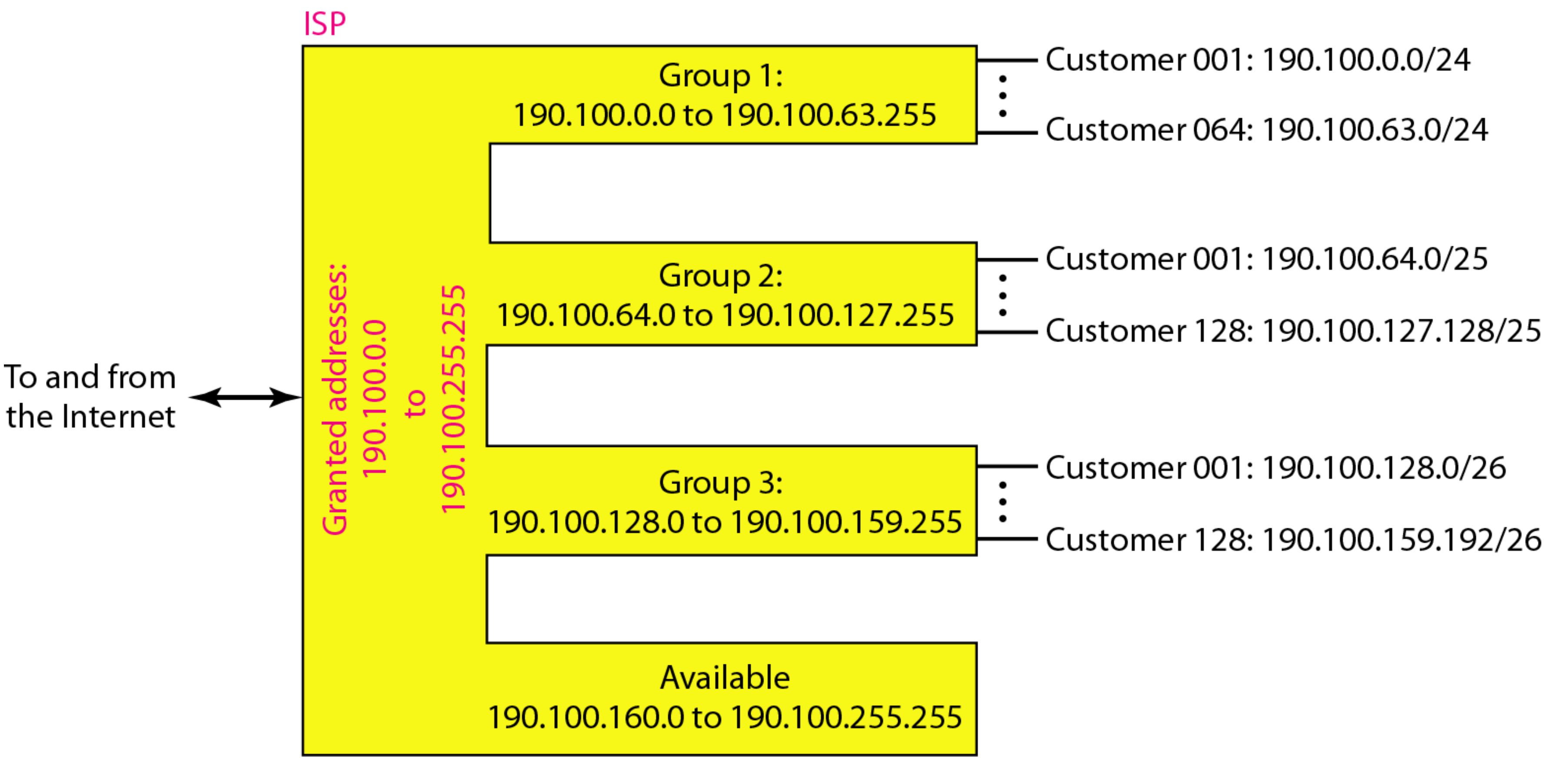
Group 3

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

1st Customer:	190.100.128.0/26	190.100.128.63/26
2nd Customer:	190.100.128.64/26	190.100.128.127/26
. . .		
128th Customer:	190.100.159.192/26	190.100.159.255/26
Total = 128 × 64 = 8192		

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

An example of address allocation and distribution by an ISP



Network Address Translation (NAT)

When a home user or a small business user is assigned an IP address, but the user has several hosts and needs an IP address for each host. For example, in your home, you may have ADSL or cable modem, and you have several computers to connect.

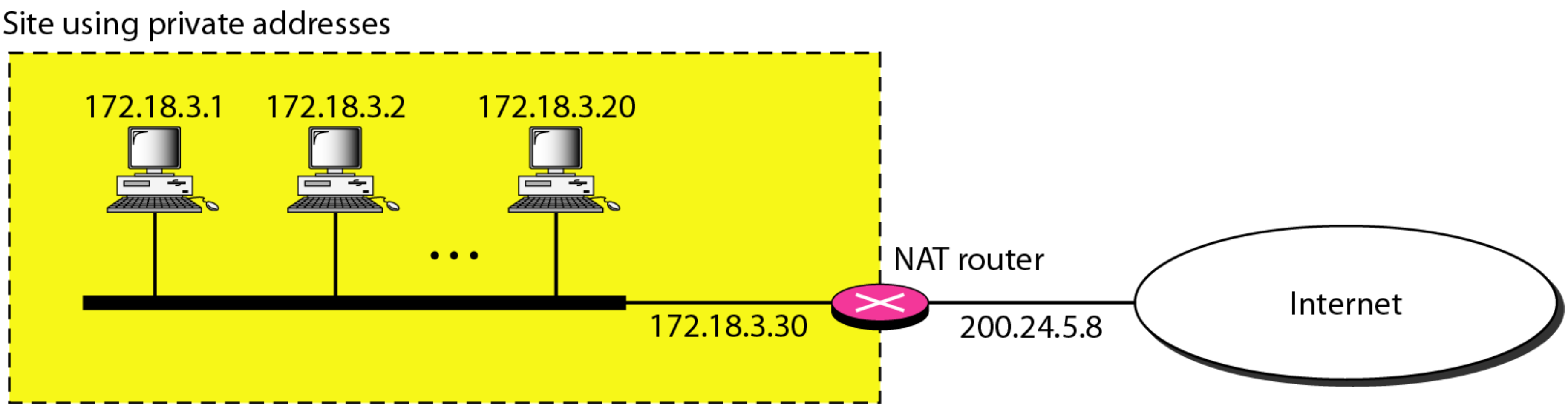
Solution: NAT, which enables a user to have a large set of private addresses internally and one universal address, or a small set of universal addresses, externally.

The Internet authorities have reserved three sets of addresses as private addresses. Any organization can use an address among these addresses without permission. They are unique inside the organization, but not unique globally. No router will forward a packet that has one of these addresses as the destination address.

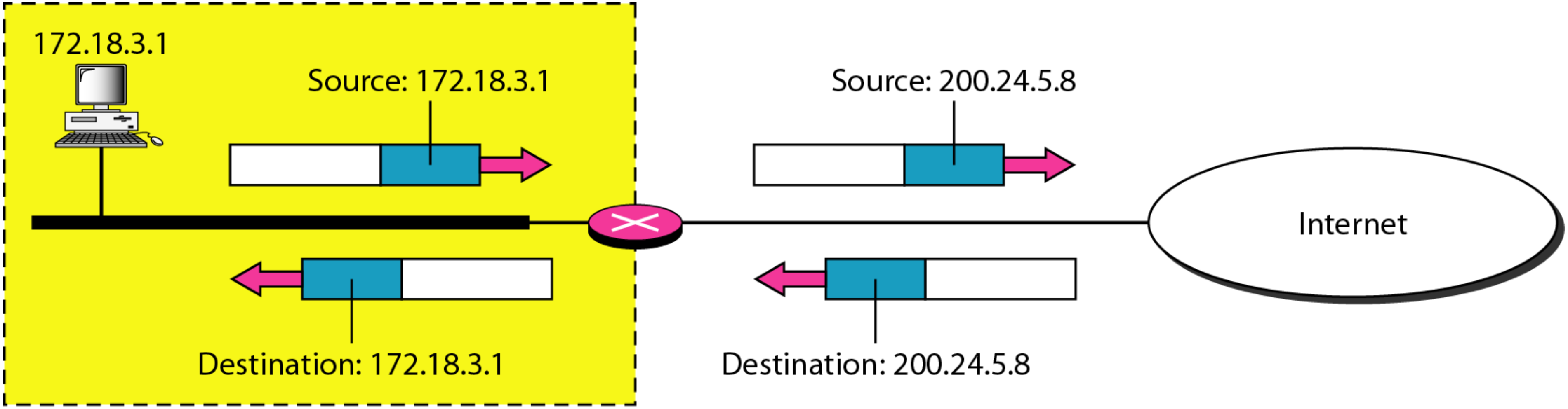
Addresses for private networks

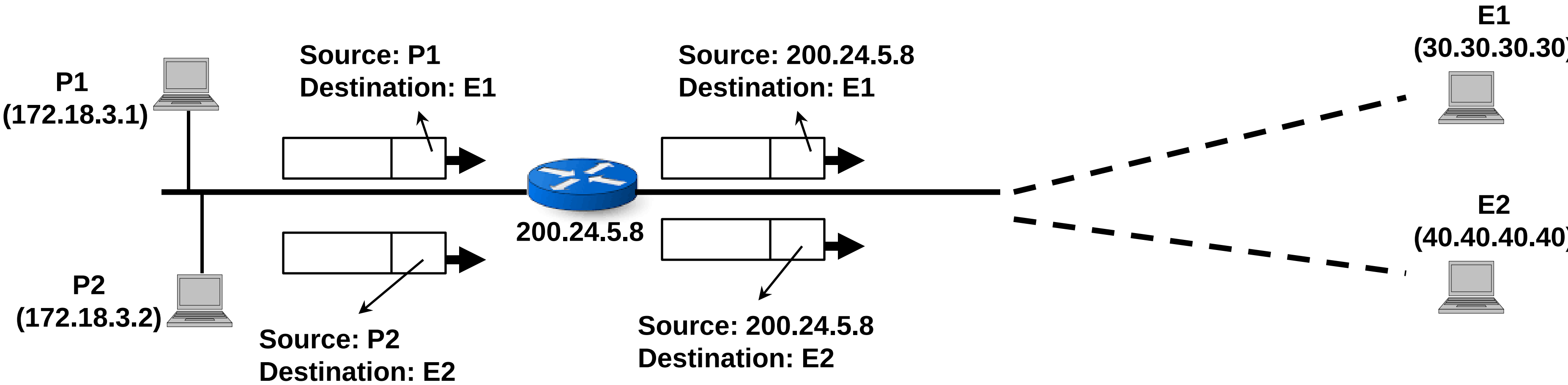
Range			*Total*
10.0.0.0	to	10.255.255.255	2^{24}
172.16.0.0	to	172.31.255.255	2^{20}
192.168.0.0	to	192.168.255.255	2^{16}

A NAT implementation



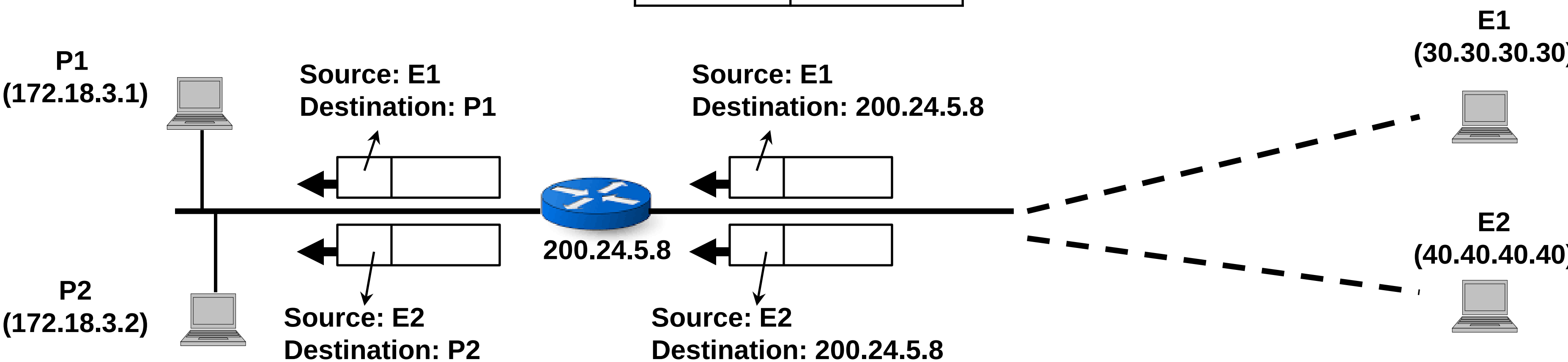
Addresses in a NAT



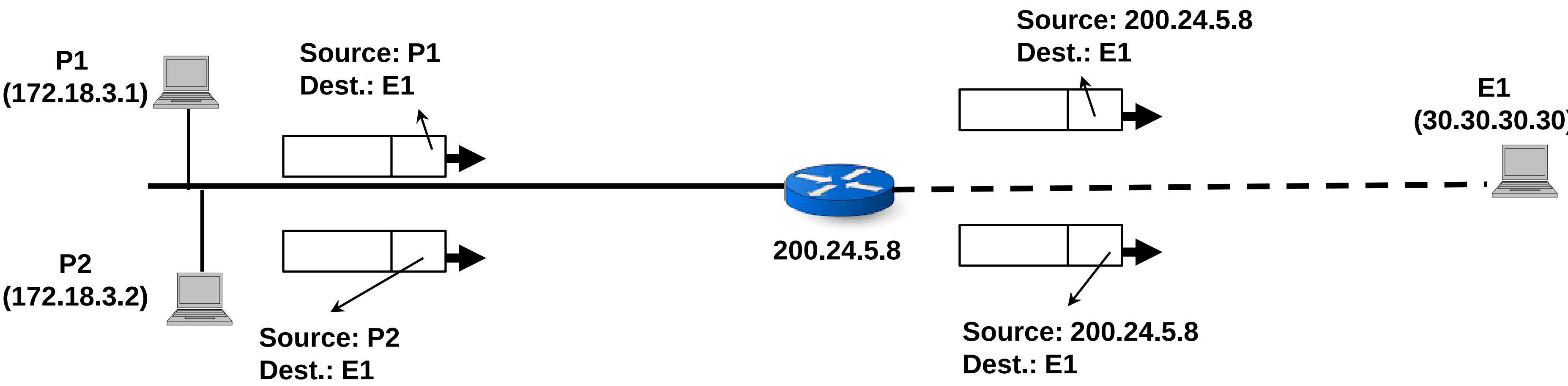


Translation Table

Private	External
P1	E1
P2	E2

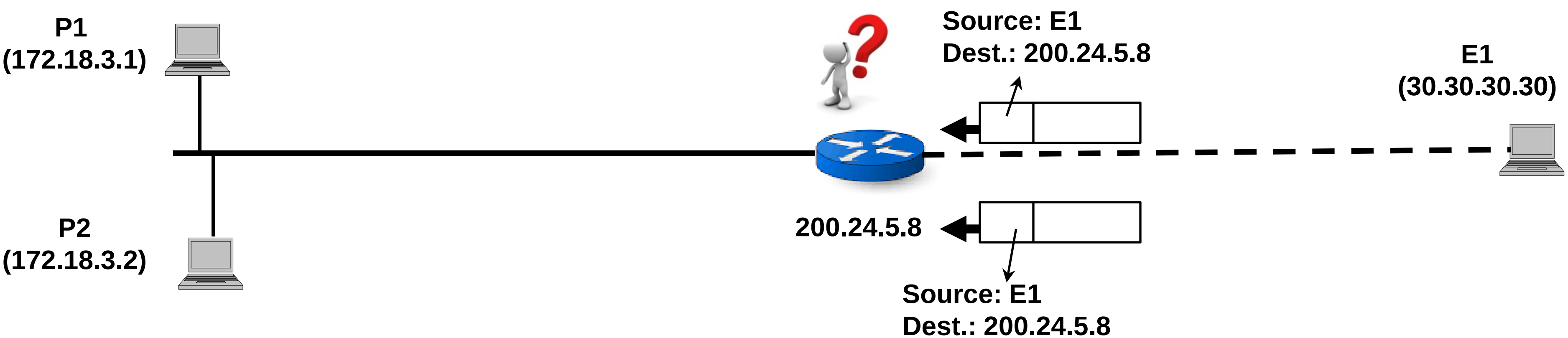


If two private hosts access the same external host….

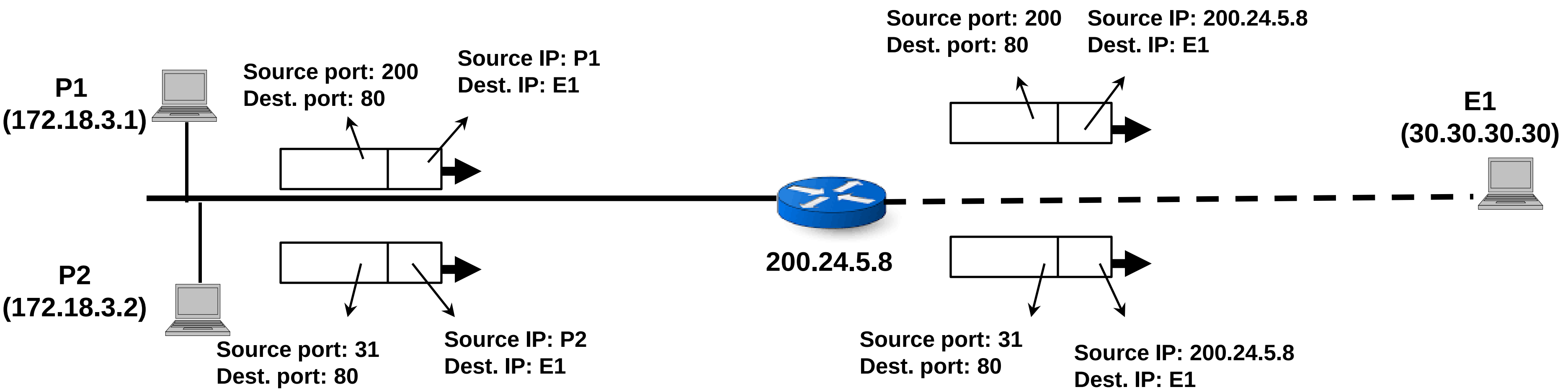


Translation Table

Private	External
P1	E1
P2	E1

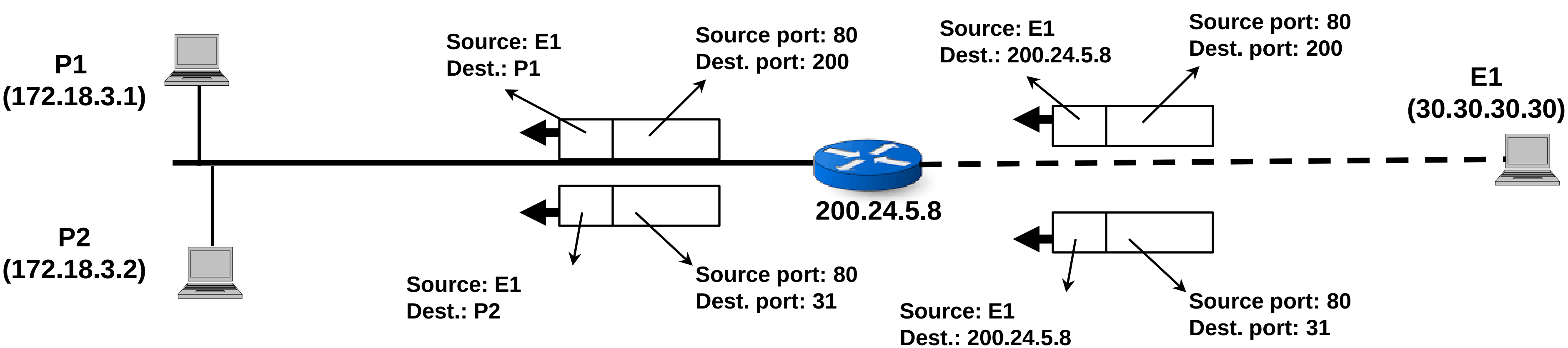


Solution: includes port addresses and transport protocol → Five-column translation table
So the router should check Layer 4 header

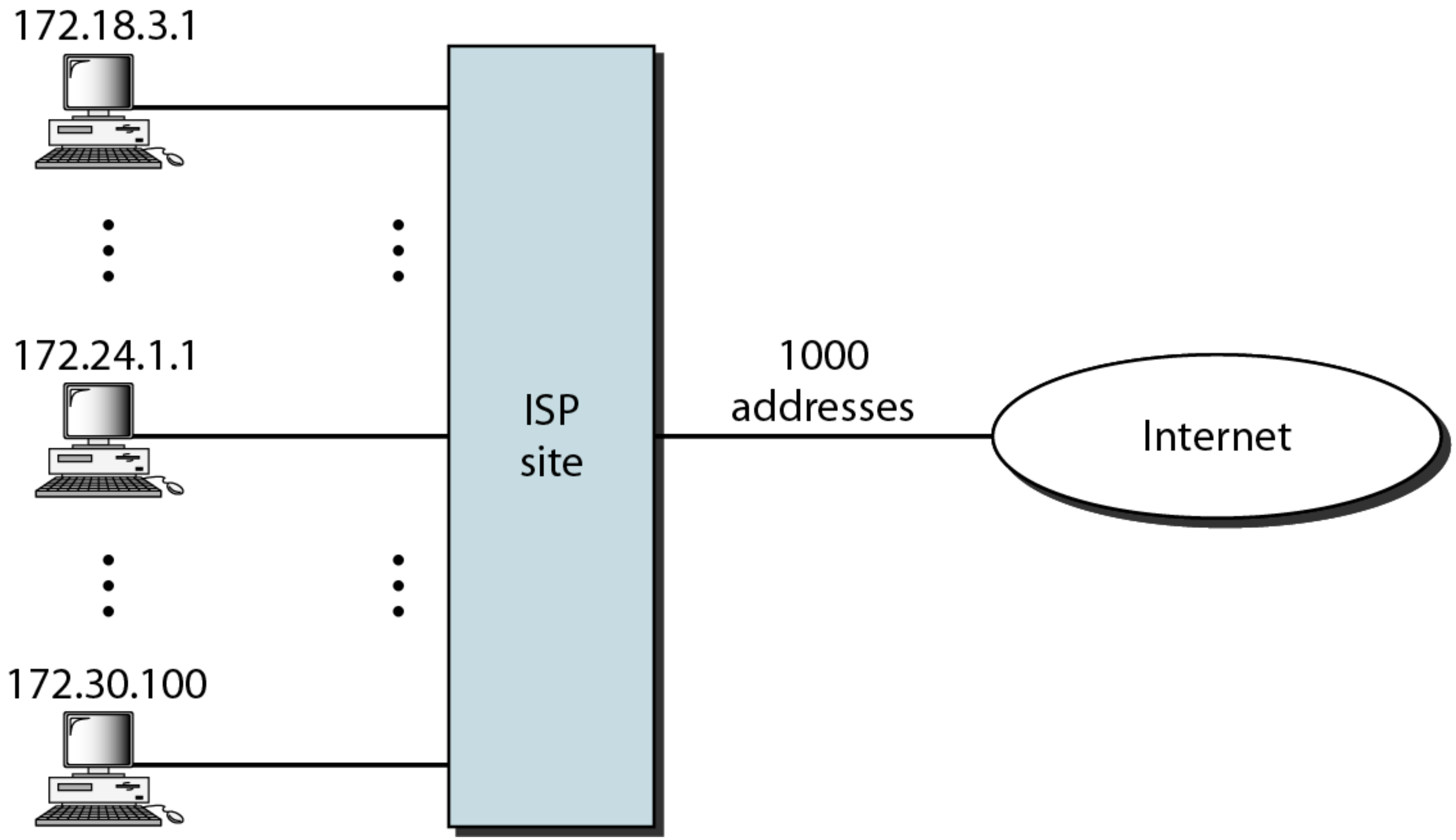


Translation Table

Private IP	Private port	External IP	External port	Transport protocol
P1	200	E1	80	TCP
P2	31	E1	80	TCP



An ISP and NAT



The ISP is granted 1000 universal addresses, but has 100,000 customers. Each customer is assigned a private network address.

Section 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.

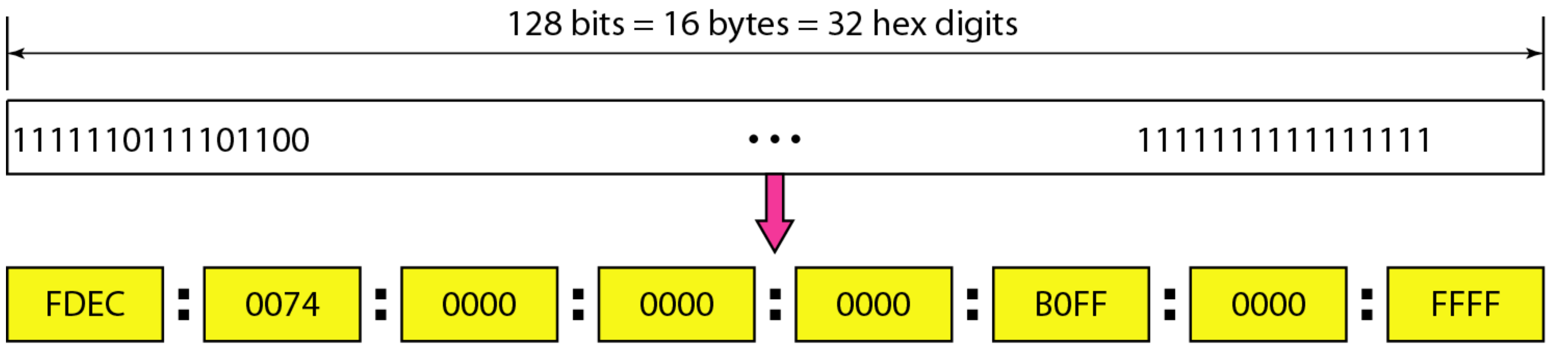
Topics discussed in this section:

Structure
Address Space

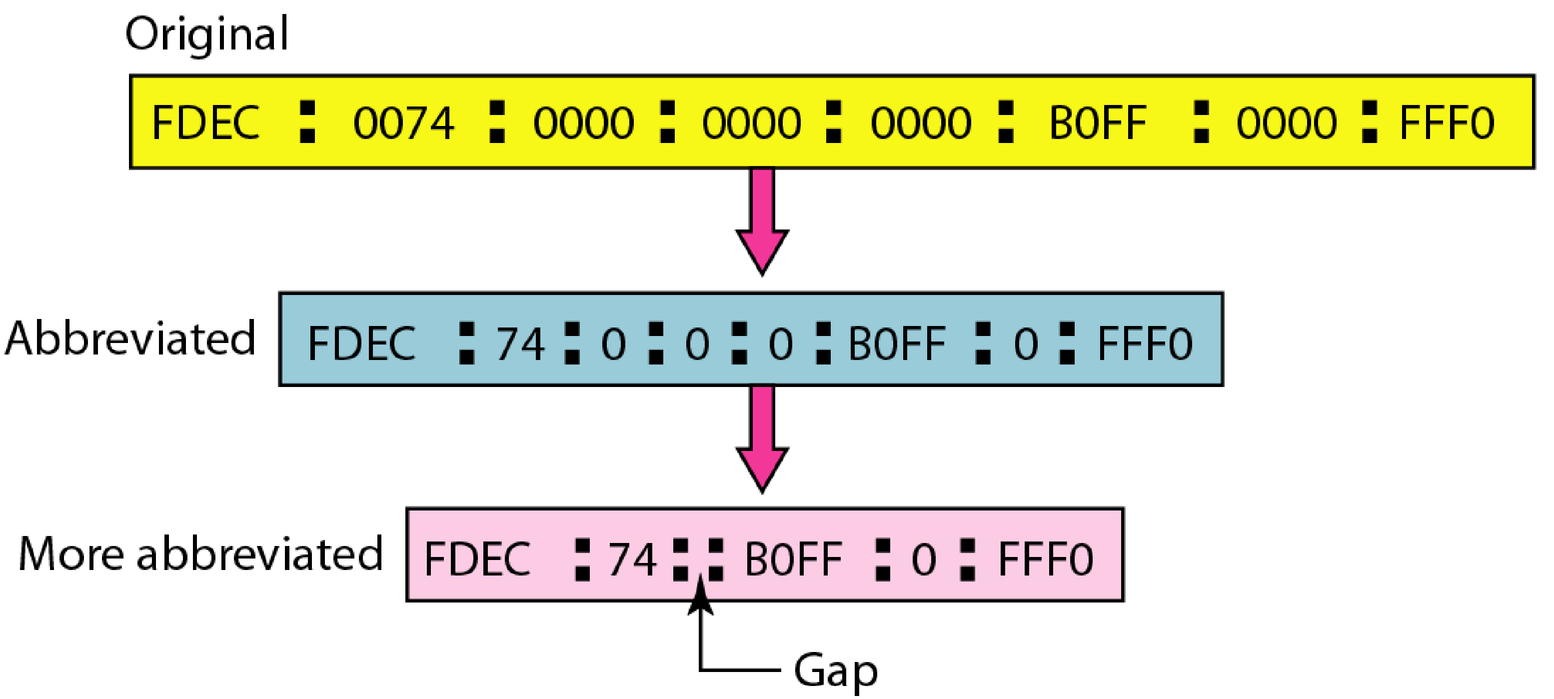
Note

An IPv6 address is 128 bits long.

IPv6 address in binary and hexadecimal colon notation



Abbreviated IPv6 addresses



Example 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.

```
xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx
   0:  15:                 :   1:  12:1213
```

This means that the original address is.

```
0000:0015:0000:0000:0000:0001:0012:1213
```

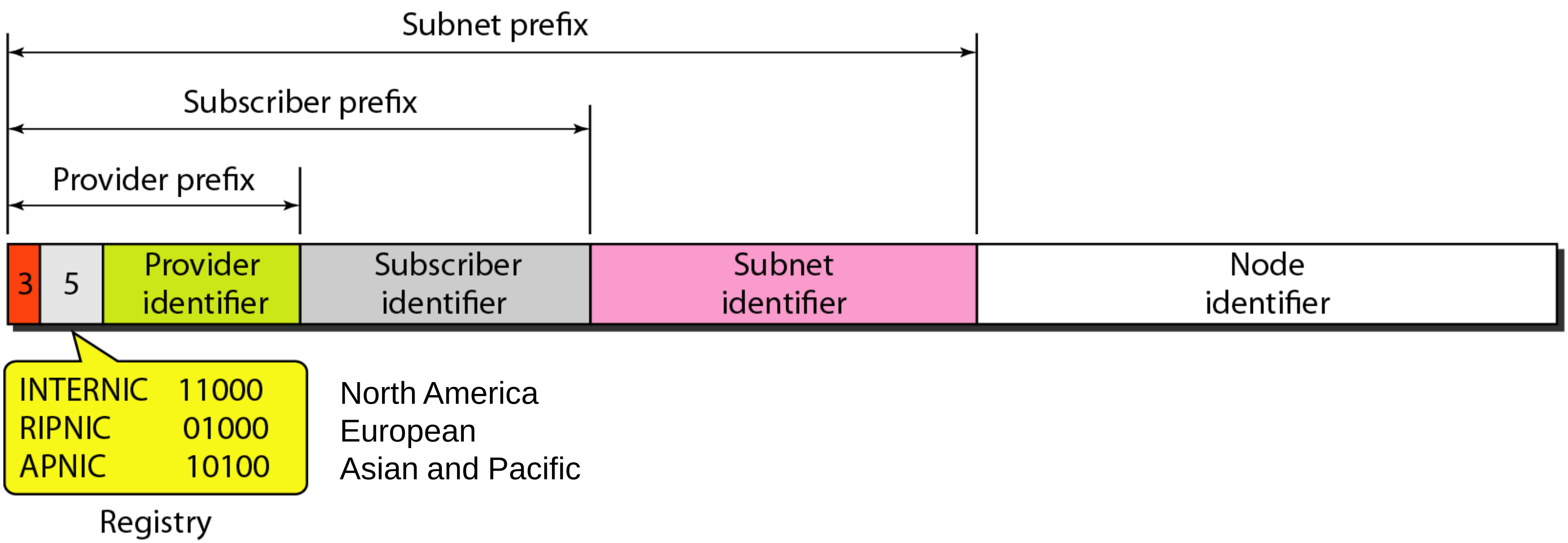
Type prefixes for IPv6 addresses

Type Prefix	*Type*	*Fraction*
0000 0000	Reserved	1/256
0000 0001	Unassigned	1/256
0000 001	ISO network addresses	1/128
0000 010	IPX (Novell) network addresses	1/128
0000 011	Unassigned	1/128
0000 1	Unassigned	1/32
0001	Reserved	1/16
001	Reserved	1/8
010	**Provider-based unicast addresses**	**1/8**

Type prefixes for IPv6 addresses (continued)

Type Prefix	*Type*	*Fraction*
011	Unassigned	1/8
100	Geographic-based unicast addresses	1/8
101	Unassigned	1/8
110	Unassigned	1/8
1110	Unassigned	1/16
1111 0	Unassigned	1/32
1111 10	Unassigned	1/64
1111 110	Unassigned	1/128
1111 1110 0	Unassigned	1/512
1111 1110 10	Link local addresses	1/1024
1111 1110 11	Site local addresses	1/1024
1111 1111	Multicast addresses	1/256

Prefixes for provider-based unicast address



North America
European
Asian and Pacific