CHAPTER 18

PRACTICE SET

Questions

Q18-1. The transport layer communication is between two ports; the network layer communication is between two hosts. This means that each layer has a different source/destination address pair; each layer needs a different header to accommodate these pair of addresses. In addition, there are other pieces of information that need to be separately added to the corresponding header.

Q18-3. Forwarding is delivery to the next node. A router uses its forwarding table to send a packet out of one of its interfaces and to make it to reach to the next node. In other words, forwarding is the decision a router makes to send a packet out of one of its interfaces. Routing, on the other hand, is an end-to-end delivery resulting in a path from the source to the destination for each packet. This means a routing process is a series of forwarding processes. To enable each router to perform its forwarding duty, routing protocols need to be running all of the time to provide updated information for forwarding tables. Although forwarding is something we can see in the foreground, in the background, routing provides help to the routers to do forwarding.

Q18-5. The number of virtual circuits is $2^8 = 256$.

Q18-7. None of these services are implemented for the IP protocol in order to make it simple.

Q18-9. The throughput is the smallest transmission rate, or 140 Kbps. The bottleneck is now the link between the source host and R1.

Q18-11. Yes. We can find the prefix length using only the block size. The prefix length is directly related to the block size as shown below:

$$n = 32 - \log_2 N$$

Problems

P18-1. The size of the address in each case is the base to the power of the number of digits:

a. The size of the address space is 2^{16} = 65,536.

b. The size of the address space is 16^{6} = 16,777,216.

c. The size of the address space is 8^{4} = 4096.

P18-3. We change each 8-bit section to the corresponding decimal value and insert dots between the bytes.

a. 94.176.117.21

b. 137.142.208.49

c. 87.132.55.15

P18-5. The class can be defined by checking the first few bits (see figure 4.31). We need to stop checking if we find a 0 bit or four bits have already been checked.

a. Since the first bit is 0, the Class is A.

b. Since the first four bits are 1110, the class is D.

c. Since the first three bits are 110, the class is C.

P18-7. We can use the formula $N = 2^{32-n}$

a. $N = 2^{32-0} = 4{,}294{,}967{,}296$

b. $N = 2^{32-14} = 262{,}144$

c. $N = 2^{32-32} = 1$

P18-9. We can first write the prefix in binary and then change each 8-bit chunk to decimal:

a. 00000000 00000000 00000000 00000000 **mask:** 0.0.0.0

b. 11111111 11111100 00000000 00000000 **mask:** 255.252.0.0

c. 11111111 11111111 11111111 11111100 **mask:** 255.252.255.252

P18-11. We first write each potential mask in binary notation and then check if it has a contiguous number of 1s from the left followed by 0s.

a. 11111111 11100001 00000000 00000000 Not a mask

b. 11111111 11000000 00000000 00000000 A mask

c. 11111111 11111111 11111111 00000110 Not a mask

P18-13. We write the address in binary and then keep only the leftmost *n* bits.

a. 10101010 00101000 00001011

b. 01101110 00101000 111100

c. 01000110 00001110 00

P18-15. Both NAT and DHCP can be used for this purpose. DHCP dynamically assigns one of the assigned addresses when a host needs to access the Internet; NAT permanently assigns a set of private addresses to the host, but maps the private address to the global address when a host needs to use the Internet.

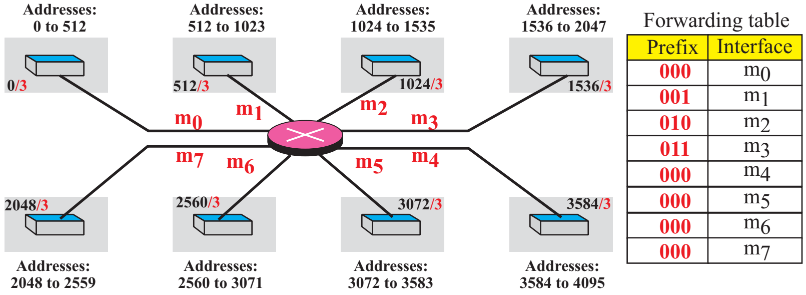
P18-17. The total number of addresses is 2^{12} = 4096. This means that there are 512 addresses for each network. We can divide the whole address space into eight blocks (block 0 to block7), each of 512 addresses. The addresses in each block are allocated as (0 to 511), (512 to 1023), (1024 to 1535), (1536 to 2047), …, (3584 to 4095). It can be checked that each block is allocated according to the two restrictions needed for the proper operation of CIDR. First, the number of addresses in each block is a power of 2. Second, the first address is divisible by the number of addresses as shown below:

Block 0: 0 / 512 = 0 **Block 1**: 512 / 512 = 1 **Block 2**: 1024 / 512 = 2 …

The prefix length for each group is $n_i = 12 - \log_2 512 = 3$. We can then write the ranges in binary to find the prefix for each block.

Block	Range	Range in binary	***n***	Prefix
0	0 to 511	000000000000 to 000111111111	3	000
1	512 to 1023	001000000000 to 001111111111	3	001
2	1024 to 1535	010000000000 to 010111111111	3	010
3	1536 to 2047	011000000000 to 011111111111	3	011
4	2048 to 2559	100000000000 to 100111111111	3	100
5	2560 to 3071	101000000000 to 101111111111	3	101
6	3072 to 3583	110000000000 to 110111111111	3	110
7	3584 to 4095	111000000000 to 111110000000	3	111

The following figure shows the outline and the forwarding table. Note that each interface can use one of the addresses in the corresponding block. The addresses are written in decimal (not dotted-decimal) because of the address space size.



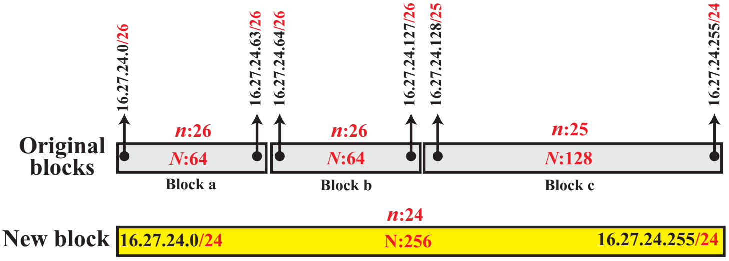
Forwarding table

Prefix	Interface
000	m_0
001	m_1
010	m_2
011	m_3
000	m_4
000	m_5
000	m_6
000	m_7

P18-19. One way to do this is to first find the size of each block. We can then add the size to the first address in the block to find the last address. Next, we can put the blocks together to find whether they can be combined into a larger block.

Block	Size	First address		Last address
a	$N = 2^{32-26} = 64$	16.27.24.0/26	→	16.27.24.63/26
b	$N = 2^{32-26} = 64$	16.27.24.64/26	→	16.27.24.127/26
c	$N = 2^{32-25} = 128$	16.27.24.128/25	→	16.27.24.255/26

Since the blocks are contiguous, we can combine the three blocks into a larger one. The new block has 256 addresses and $n = 32 - \log_2 256 = 24$.



P18-21.

a. The number of addresses in the ISP block is $N = 2^{32-20} = 4096$. We can add 4095 (which is $N - 1$) to the first address to find the last one (note that the addition can be done in base 256, as described in Appendix B. In base 256, 4095 is (15.255). We have

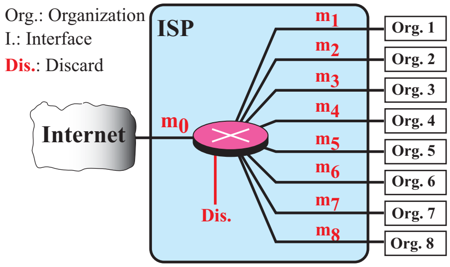
First address: 16.12.64.0/**20** Last address: 16.12.79.255/**20**

The prefix length for each organization is $n_i = 32 - \log_2 256 = 24$. We assume that the addresses are allocated from the beginning of the ISP block with each organization consuming 256 addresses. The following shows how addresses are allocated. Note that the prefix for each block is 24 bits.

Block	First address		Last address	*n*
1	16.12.64.0/**24**	→	16.12.64.255/**24**	**24**
2	16.12.65.0/**24**	→	16.12.65.255/**24**	**24**
3	16.12.66.0/**24**	→	16.12.66.255/**24**	**24**
4	16.12.67.0/**24**	→	16.12.67.255/**24**	**24**
5	16.12.68.0/**24**	→	16.12.68.255/**24**	**24**
6	16.12.69.0/**24**	→	16.12.69.255/**24**	**24**
7	16.12.70.0/**24**	→	16.12.70.255/**24**	**24**
8	16.12.71.0/**24**	→	16.12.71.255/**24**	**24**
Unassigned	16.12.72.0/**21**	→	16.12.79.255/**21**	**21**

The unallocated addresses, which can be reserved for the future use of the ISP, are 16.12.72.0/**21** to 16.12.79.255/**21**, for a total of 2048 addresses.

b. The simplified outline is given below. Note that packets having destination addresses with the last prefix in the figure are discarded until these addresses are assigned.



Forwarding table

Prefix	Interface
Default	m_0
0001000 00001100 01000000	m_1
0001000 00001100 01000001	m_2
0001000 00001100 01000010	m_3
0001000 00001100 01000011	m_4
0001000 00001100 01000100	m_5
0001000 00001100 01000101	m_6
0001000 00001100 01000110	m_7
0001000 00001100 01000111	m_8
0001000 00001100 01001	**Dis.**

P18-23. The total number of addresses in the organization is $N = 2^{32-16} = 65{,}536$.

a. Each subnet can have N_{sub} = 65,536 /1024 = 64 addresses.

b. The subnet prefix for each subnet is $n_{sub} = 32 - \log_2 N_{sub} = 32 - 6 = 26$.

c. Now we can calculate the first and the last address in the first subnet. The first address is the beginning address of the block; the last address is the first address plus 63.

First address: 130.56.0.0/**26** Last address: 130.56.0.63/**26**

d. To find the first address in subnet 1024, we need to add 65,472 (1023 × 64) in base 256 (0.0.255.192) to the first address in subnet 1. The last address can then be found by adding 63 to the first.

First address: 130. 56.255.192/**26** Last address: 130. 56.255.255/**26**

P18-25. The packet is sent to router R1 and eventually to organization 1 as shown below:

a. Router R2 applies the mask /26 to the address (or it extracts the leftmost 26 bits) resulting in the network address/mask of 140.24.7.0/26, which does not match with the first entry in the forwarding table.

b. Router R2 applies the mask /24 to the address (or it extracts the leftmost 24 bits) resulting in the network address/mask of 140.24.7.0/24, which matches with the second entry in the forwarding table. The packet is sent out from interface m0 to router R1.

c. Router R1 applies the mask /26 to the address (or it extracts the leftmost 26 bits) resulting in the network address/mask of 140.24.7.0/26, which matches with the first entry in the forwarding table. The packet is sent out from interface m0 to organization 1.