EX: What is the subnetwork address if the destination address is 200.45.34.56 and the subnet mask is 255.255.240.0?

11001000 00101101 00100010 00111000

11111111 11111111 11110000 00000000

11001000 00101101 00100000 00000000

The subnetwork address is 200.45.32.0.

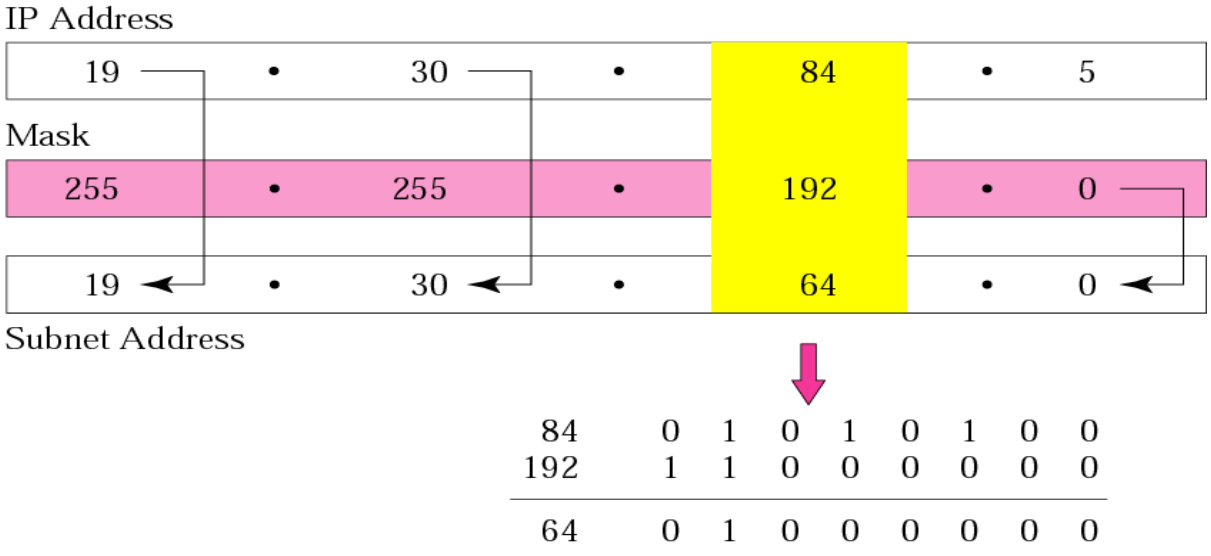
Short-Cut Method

** If the byte in the mask is 255, copy the byte in the address.

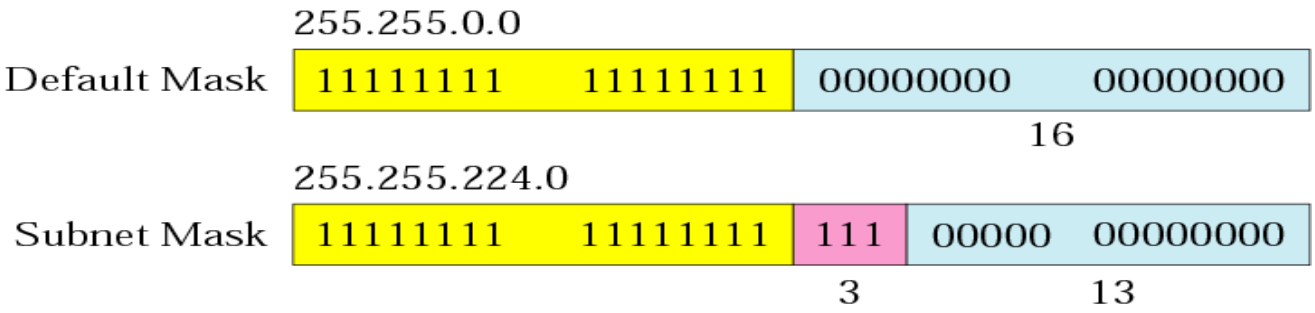
** If the byte in the mask is 0, replace the byte in the address with 0.

** If the byte in the mask is neither 255 nor 0, we write the mask and the address in binary and apply the AND operation.

EX: What is the subnetwork address if the destination address is 19.30.80.5 and the mask is 255.255.192.0?

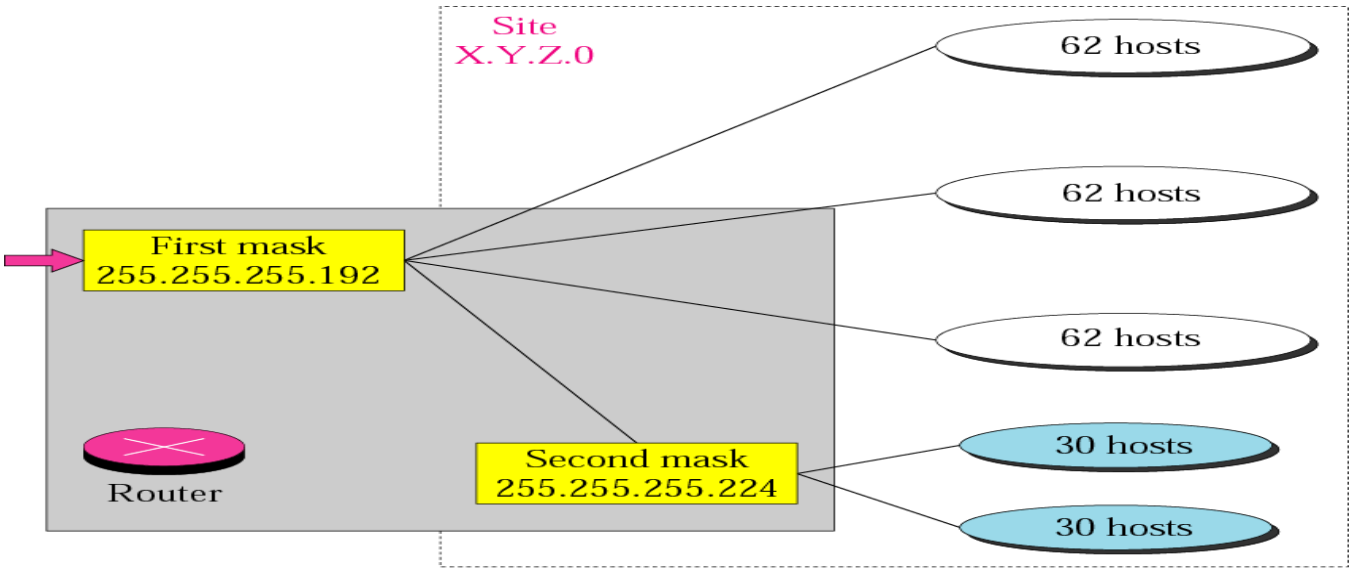


Comparison of a default mask and a subnet mask:



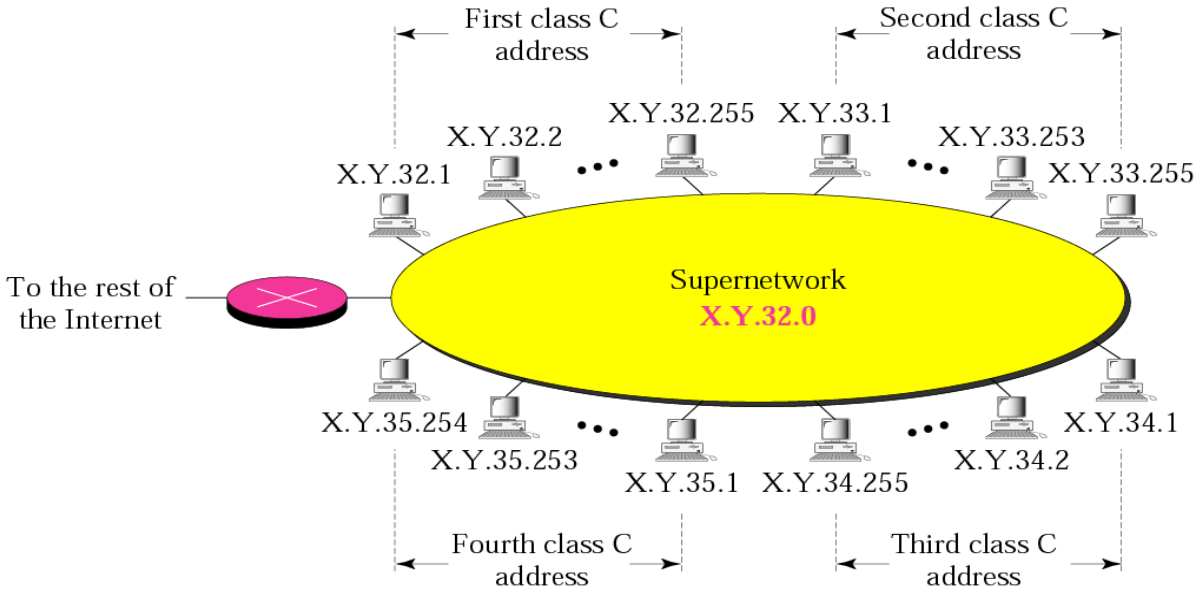
The number of subnets must be a power of 2.

Variable-length subnetting:



SUPERNETTING:

A supernetwork:



Rules:

** The number of blocks must be a power of 2 (1, 2, 4, 8, 16, . . .).

** The blocks must be contiguous in the address space (no gaps between the blocks).

** The third byte of the first address in the superblock must be evenly divisible by the number of blocks. In other words, if the number of blocks is *N*, the third byte must be divisible by *N*.

EX: A company needs 600 addresses. Which of the following set of class C blocks can be used to form a supernet for this company?

198.47.32.0 198.47.33.0 198.47.34.0

198.47.32.0 198.47.42.0 198.47.52.0 198.47.62.0

198.47.31.0 198.47.32.0 198.47.33.0 198.47.52.0

198.47.32.0 198.47.33.0 198.47.34.0 198.47.35.0

1: No, there are only three blocks.

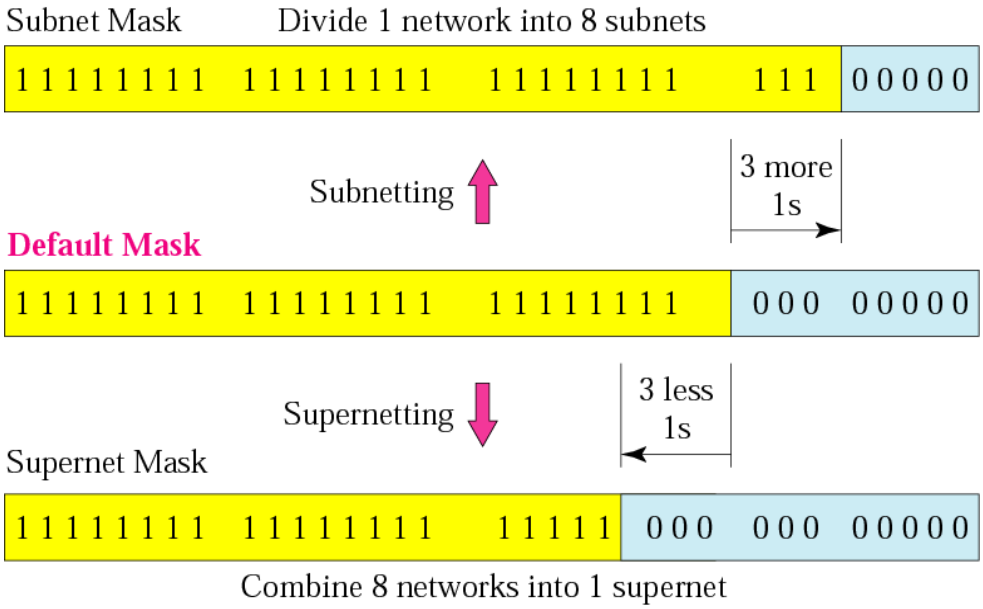
2: No, the blocks are not contiguous.

3: No, 31 in the first block is not divisible by 4.

4: Yes, all three requirements are fulfilled.

In subnetting, we need the first address of the subnet and the subnet mask to define the range of addresses. In supernetting, we need the first address of the supernet and the supernet mask to define the range of addresses.

Comparison of subnet, default, and supernet masks:



CLASSLESS ADDRESSING:

Variable-length blocks

Number of Addresses in a Block:

There is only one condition on the number of addresses in a block; it must be a power of 2 (2, 4, 8, . . .). A household may be given a block of 2 addresses. A small business may be given 16 addresses. A large organization may be given 1024 addresses.

Beginning Address:

The beginning address must be evenly divisible by the number of addresses. For example, if a block contains 4 addresses, the beginning address must be divisible by 4. If the block has less than 256 addresses, we need to check only the rightmost byte. If it has less than 65,536 addresses, we need to check only the two rightmost bytes, and so on.

EX: Which of the following can be the beginning address of a block that contains 16 addresses?

205.16.37.32
190.16.42.44
17.17.33.80
123.45.24.52

The address 205.16.37.32 is eligible because 32 is divisible by 16. The address 17.17.33.80 is eligible because 80 is divisible by 16.

Slash notation:

A.B.C.D/*n*

EX: A small organization is given a block with the beginning address and the prefix length **205.16.37.24/29** (in slash notation). What is the range of the block?

The beginning address is 205.16.37.24. To find the last address we keep the first 29 bits and change the last 3 bits to 1s.

Beginning:**11001111 00010000 00100101 00011**000

Ending : **11001111 00010000 00100101 00011**111

There are only 8 addresses in this block.

We can find the range of addresses previous Example by another method. We can argue that the length of the suffix is 32 - 29 or 3. So there are $2^3 = 8$ addresses in this block. If the

first address is 205.16.37.24, the last address is 205.16.37.31 (24 + 7 = 31).

A block in classes A, B, and C can easily be represented in slash notation as A.B.C.D/ *n* where *n* is either 8 (class A), 16 (class B), or 24 (class C).

Layer 4: Transport layer:

The TCP/IP model has two main protocols in transport layer:

a) TCP (Transmission Control Protocol):

1- TCP is connection oriented protocol.

2- TCP was designed to provide a reliable end to end byte stream over unreliable internetwork.

3- TCP was designed to be robust in the face of many kinds of failures.

1- TCP entity accepts user data streams (messages) from local process, breaks them into pieces not exceeding 64k bytes (usually about 1500 bytes) and sends each piece as a separate IP datagram. When IP datagrams that containing TCP data arrive the destination machine, they are given to the TCP entity, which reconstruct the original byte streams. The IP layer gives no guarantee that datagrams will be delivered properly, so it is up to TCP to time out and retransmit them as need be.

2- Datagrams that arrive may well do so in the wrong order; it is also up to TCP to reassemble them into messages in the proper sequence.

3- The sending and receiving entities exchange data in the form of segments. A segment consists of a fixed 20 byte header (plus an optional part) followed by zero or more data bytes. The TCP decides how big segments should be. Segments without any data are legal and are commonly used for acknowledgments and control messages.

b) UDP (User Data Protocol):

UDP is connectionless transport protocol that provides a way for sending a raw IP datagrams without establish a connection. Many client server applications have one request

and one response use UDP rather than go to trouble of establishing and releasing a connection.

Layer 5: Application layer:

This layer contains all the higher level protocols, such as: TELNET (Terminal Network: enables the establishment of a connection to a remote system in such way that the local terminal appears to be a terminal at the remote system), FTP (File Transfer), SMTP (Electronic Mail), and DNS (Domain Name System).