

Quick  
and Dirty

# Subnetting by PrepLogic®

**We can't emphasize enough the importance of being able to calculate subnets on the fly when working with Windows servers and clients, particularly for designing your Active Directory site structure (also something you will get hammered on in MS exam 70-642: MCTS: Windows Server Network Infrastructure, Configuring).**

If you know the principles and theory behind subnetting and can convert binary numbers to decimal and decimal to binary, you may find this very short summary of how to perform subnetting useful.

For people who work with IP addressing and routing as part of their jobs, subnetting (breaking a larger IP network into separated and smaller networks) or supernetting (combining two or more networks or subnets together) becomes second nature. For them, it is a fairly easy thing to calculate subnets. A lot of people who are new to networking or relatively inexperienced with TCP/IP often and wrongly think that performing subnetting is a mysterious and difficult task. I remember the frustration I felt many years ago when, as an inexperienced network admin, I could not get anyone to give me a good explanation of subnetting. Some of the more experienced network administrators, in fact, refused to explain it to me. I don't know if they were protecting their jobs or what, but it did seem to me at the time that they were going out of their way to create a false aura of mystery and difficulty concerning subnetting.

The truth is this: calculating subnets is fairly easy and doesn't require that you memorize anything except a few general principles and know how to work with binary numbers. The purpose of this short explanation is to provide you with the

minimum amount of information you have to carry around in your head in order to calculate subnet (or supernet) masks in any situation, an exam, your job, etc. In this post, we will address the more complicated subnetting known as extended subnetting. Instead of using the basic classful subnet masks (255.0.0.0, 255.255.0.0 and 255.255.255.0), we will explore the use of extended subnet masks.

## SOME RULES TO REMEMBER

**1** The network portion of the IP address described by the extended subnet mask cannot be expressed as all 0s or all 1s. (The validity of this rule depends on a number of factors such as the type of hardware and the routing protocol in use.) So, for example, let's assume we have extended a Class B network by borrowing two bits from the host portion of the address thus: 172.16.0.0/18. The subnet mask is 255.255.192.0 or 11111111.11111111.11000000.00000000. Notice that we have used the first 18 of the 32 bits for the subnet mask. This is where the /18 notation originates.

Any combination of 0s and 1s in the 3rd octet could possibly comprise a network ID. However, following our rule of not allowing all 1s or all 0s in the positions covered by the extended subnet mask, we get the following valid network IDs in the 3rd octet:

```
00000000 = 0 Not a valid network ID (all 0s)
01000000 = 64 Valid network ID (172.16.64.0)
10000000 = 128 Valid network ID (172.16.128.0)
11000000 = 192 Not a valid network ID (all 1s)
```

**2** The host portion of the IP address cannot be expressed as all 0s or all 1s. All 0s in the host portion of the address indicates the Network ID. All 1s in the host portion of the address indicates the address for the broadcast on the subnet. Therefore, using our example above, here are the valid host IDs for our two valid subnets [172.16.64.0/18](#) and [172.16.128.0/18](#):

[172.16.64.1/18](#) - [172.16.127.254/18](#)  
[172.16.128.1/18](#) - [172.16.191.254/18](#)

To understand the rule more clearly, convert the host addresses to binary:

64 1 127 254  
x.y. 01000000.00000001 -x.y. 01111111.11111110

128 1 191 254  
x.y. 10000000.00000001 -x.y. 10111111.11111110

**3** To calculate the number of hosts or the number of networks for a given subnet mask, use the following formula:

$2^{(\text{number of bits used})} - 2 = \text{number of networks/hosts}$  (The '^' means 'raised to the power of'.)

So, given the fact that we have borrowed 2 bits to extend our network, we have  $2^2 - 2 = 2$  networks (remember, we are excluding the all 1 and all 0 networks).

We have 14 bits left to describe the hosts on each network; therefore we have  $2^{14} - 2 = 16,382$  hosts per network (remember we are excluding host addresses with all 0s or all 1s).

**4** The value of the lowest order bit in the subnet mask tells you two things: a) it tells you the first network ID for the subnets created by the extended subnet mask, b) it tells you the value that you can add to one network ID to arrive at the next possible network ID.

Let's take our example extended subnet mask of [255.255.192.0](#). The 3rd octet expressed in binary with the decimal equivalents is:

128 64 32 16 8 4 2 1  
1 1 0 0 0 0 0 0

The value of the lowest order bit is 64. Therefore, our first network ID is [172.16.64.0/18](#). The next and last network ID is [172.16.128.0/18](#) ( $64 + 64 = 128$ )

*To help you calculate subnet masks, you might want to memorize the table on the right side of this page.*

That's it. The first thing I do when I go into an exam situation where I know I am going to be asked about subnetting is to draw out the subnetting table on a blank piece of paper.

For the Microsoft 70-642: MCTS: Windows Server Network Infrastructure, Configuring, you will find that you need to be able to calculate subnets quickly and accurately. I have found this quick method to be helpful, but particularly helpful on this exam.

## SUBNETTING TABLE

To use the table, find the value for the subnet mask. Let's say you are subnetting a Class B address with the subnet mask 255.255.248.0. Find 248 on the chart and follow the lines to determine the low order bit for the subnet. The value for the low order bit for 248 is 8. That tells you two things: the starting network ID and the difference between one network ID and the next: 8. Therefore, given a class B address of 172.16.0.0/21 (subnet mask of 255.255.248.0), the possible network IDs and host ranges are

### Network IDs Host Ranges

[172.16.8.0/21](#) [172.16.8.1](#) -  
[172.16.15.254](#)  
[172.16.16.0/21](#) [172.16.16.1](#) -  
[172.16.23.254](#)  
[172.16.24.0/21](#) [172.16.24.1](#) -  
[172.16.31.254](#)  
.  
.  
.  
[172.16.240.0/21](#) [172.16.240.1](#) -  
[172.16.247.254](#)

To determine the number of networks created by the subnet mask, count the number of bits used for the subnet mask. For a subnet mask of 248, the number is 5 bits. Therefore, the number of networks is  $2^5 - 2 = 30$ . To determine the number of hosts, count the remaining bits. For a subnetted Class B address with a subnet mask of [255.255.248.0](#), we have  $3 + 8 = 11$  bits remaining. Therefore, we can have  $2^{11} - 2 = 2046$  hosts per network.

So, to sum up, memorize the Subnetting Table above and the following rules:

- 1. You can't have all 0s or all 1s in the host or network portion of the address (unless the hardware and software support it).**
- 2. To determine the number of networks or hosts, use the formula  $2^{(\text{no. of bits used})} - 2$ .**
- 3. The low order bit in the subnet mask determines the starting network ID and the difference between possible network IDs.**