4.1.7 Subnetting Facts

Be aware of the following IPv4 subnetting concepts.

Concept	Description
Subnetting	*Subnetting* is the process of dividing a large network into smaller networks. When you subnet a network, each network segment, or *subnet*, has a different network address, or *subnet address*.
Supernetting	Supernetting is the process of combining two or more networks. When you create a supernet, you decrease the number of masked bits in the subnet mask. This reduces the number of available subnets, but increases the number of hosts on each subnet.
Classless Addressing	Using custom subnet masks is often called *classless* addressing because the subnet mask cannot be inferred simply from the class of a given IP address. • Using classless addresses is made possible by Classless Inter-Domain Routing (CIDR). • The *CIDR notation* is a syntax for identifying the subnet mask. • The format for the CIDR notation is the slash (/) symbol and the decimal number identifying the number of ones in the subnet mask.
ANDing	ANDing refers to performing a logical AND comparison. Use ANDing to determine the network address of a classless IP address. When determining network addresses, you convert the IP address and the subnet mask to their binary equivalents and compare each bit. The result is 1 when both numbers being compared is 1.

As you work with subnetting operations, use the following tables to quickly find the information you need. By memorizing these tables, you will be able to quickly reproduce the values necessary for identifying the binary and decimal values you use most.

The following table lists the exponent values for powers of 2.

Exponent	2^1	2^2	2^3	2^4	2^5	2^6	2^7	2^8	2^{10}	2^{16}
Exponent value	2	4	8	16	32	64	128	256	1024	65,536

Memorize the shaded values. To find smaller or larger values, divide or multiply the exponent value by 2. For example, to get the decimal value of 2^{11}, multiply 2^{10} by 2 (which equals 2048). To find the value of 2^{12}, use 2^{10} x 2 x 2 = 4096.

The following table lists the common binary and decimal values used in subnet masks:

Subnet Mask Value	Decimal Equivalent
`00000000`	0
`10000000`	128
`11000000`	192
`11100000`	224
`11110000`	240
`11111000`	248
`11111100`	252
`11111110`	254
`11111111`	255

Use the following table as a shortcut guide to subnetting.

Look for patterns in the table so you can easily reproduce the table at any time.

Masked Bits	Mask Value	Number of Subnets*	Number of Hosts per Subnet	
			Approximate	Actual (2^n - 2)**
/20	255.255.240.0	16	4000	4094
/21	255.255.248.0	32	2000	2046
/22	255.255.252.0	64	1000	1022
/23	255.255.254.0	128	500	510
/24	**255.255.255.0**	**1 or 256**	**256 or 250**	**254**
/25	255.255.255.128	2	128 or 125	126
/26	255.255.255.192	4	64 or 60	62
/27	255.255.255.224	8	32 or 30	30
/28	255.255.255.240	16	16 or 15	14
/29	255.255.255.248	32	8	6
/30	255.255.255.252	64	4	2

*The number of subnets is the number of subnets you get by subnetting a default network address, either class B or class C in this table. For example, if you subnet a class B network using a /24 mask, you would have 256 subnets.
**To identify the actual number of hosts per subnet, use the formula 2^n - 2 where n is the number of unmasked bits in the subnet mask. Remember to subtract 2 for the addresses that are not assigned to hosts. The first address in the range is the subnet address and cannot be assigned to hosts. The last address in the range is the broadcast address and cannot be assigned to hosts.

To discover if workstations are on the same subnetwork, perform the following calculation:

1. Calculate the binary value of the subnet mask and determine which octet is affected by the subnet mask. For example, a /26 subnet mask affects the last octet, as shown below:

   XXXXXXXX.XXXXXXXX.XXXXXXXX.XX000000 = /26

2. For the affected octet, determine how many subnets are available within the subnet mask and calculate the decimal value for each subnet. For example, a /26 subnet mask has four subnets available, as shown below:

   .00000000 = .0
   .01000000 = .64
   .10000000 = .128
   .11000000 = .192

3. For the affected octet, remove the first IP address (network address) and last IP address (broadcast address) in the subnet(s) as possible host addresses. For example, with a /26 subnet mask, the possible IP addresses for the first subnet are:

   .00**000000** = .0 (Network address for first possible subnet. This address is *not* valid for a workstation IP address.)
   .00**000001** = .1
   .00 **000010** = .2
   ...
   .00**111110** = .62
   .00**111111** = .63 (Broadcast address. This address is *not* valid for a workstation IP address.)
   . **01**000000 = .64 (Network address for the next possible subnet.)

4. Determine if all of the assigned IP addresses fall within the same subnet. For example, the first possible subnet for a /26 subnet mask could have IP addresses in the .1 - .62 range:

   .00**000001** = .1 (valid address)
   .00 **000010** = .2 (valid address)
   ... (valid addresses)
   .00**111101** = .61 (valid address)
   .00 **111110** = .62 (valid address)