

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

## TCP/IP Network Classification & Addressing

There are five different network classifications for IP networks: **A - B - C - D - E**. Classes D and E are for testing and experimentation so they can be discounted for actual use. All classes are distinguished by their addressing conventions. All addresses consist of four (4), eight (8) bit header units, or **octets**. The number of network allotted for each classification is dependent upon the number of octets allotted per header. Total number of networks in each class are determined by number combinations. They shift upward as the network class decreases (see diagram below).

### CLASS A

Bits (1)

= (32) Bits

0	(7) Network Address	(24) Local (Host) Address
1 to 126	126	16,777,124

### CLASS B

Bits (2)

= (32) Bits

10	(14) Network Address	(16) Local (Host) Address
128 to 191	16,384	65,534

### CLASS C

Bits (3)

= (32) Bits

110	(21) Network Address	(8) Local (Host) Address
192 to 223	2,097,152	256

### CLASS D

Bits (4)

= (32) Bits

1110	(28) Network Address	(0) Local (Host) Address
224 to 239		

### CLASS E

Bits (5)

= (32) Bits

11110	(27) Network Address	(0) Local (Host) Address
240 to 247		

Each TCP/IP host is identified by a logical *IP address*. The IP address is a network layer address and has no dependence on the Data-Link layer address (such as a MAC address of a network adapter). A

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

unique IP address is required for each host and network component that communicates using TCP/IP. The IP address identifies a system's location on the network in the same way a street address identifies a house on a city block. Just as a street address must identify a unique residence, an IP address must be globally unique and have a uniform format.

Each IP address includes a network ID and a host ID:

- The *network ID* (also known as a *network address*) identifies the systems that are located on the same physical network bounded by IP routers. All systems on the same physical network must have the same network ID. The network ID must be unique to the internetwork.
- The *host ID* (also known as a *host address*) identifies a workstation, server, router, or other TCP/IP host within a network. The address for each host must be unique to the network ID.

---

**Note** Network ID refers to any IP network ID, whether it is class-based, a subnet, or a supernet.

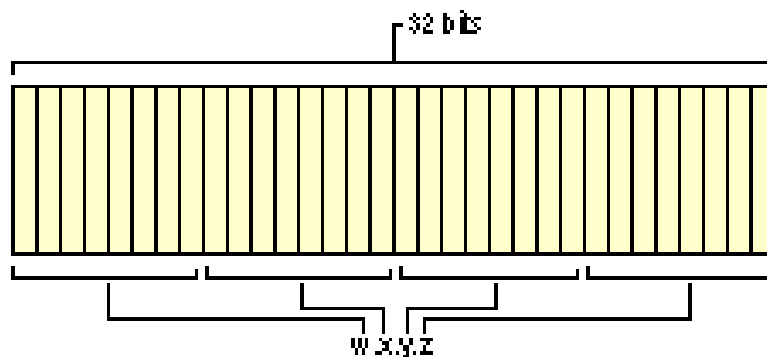
---

An IP address consists of 32 bits. Rather than working with 32 bits at a time, it is a common practice to segment the 32 bits of an IP address into four 8-bit fields called *octets*. Each octet is converted to a decimal number (the Base 10 numbering system) in the range 0-255 and separated by a period (a dot). This format is called dotted decimal notation. Table 1 provides an example of an IP address in binary and dotted decimal formats.

**Table 1: An IP Address in Binary and Dotted Decimal Formats**

Binary Format	Dotted Decimal Notation
11000000 10101000 00000011 00011000	192.168.3.24

The notation *w.x.y.z* is used when referring to a generalized IP address, and is shown in the figure below:



## Address Classes

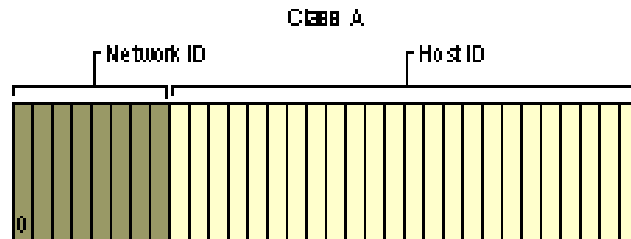
Originally five *address classes* were defined to accommodate networks of varying sizes. The class of address defines which bits are used for the network ID and which bits are used for the host ID. It also defines the possible number of networks and the number of hosts per network.

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

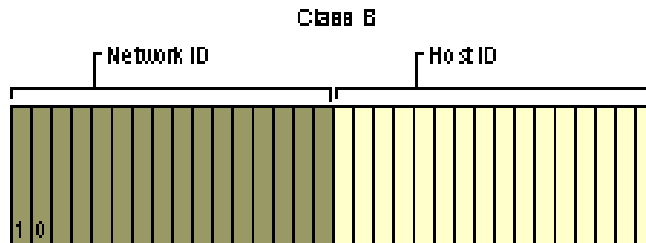
## Class A

*Class A* addresses are assigned to networks with a very large number of hosts. The high-order bit in a class A address is always set to zero. The next seven bits (completing the first octet) complete the network ID. The remaining 24 bits (the last three octets) represent the host ID. This allows for 126 networks and 16,777,214 hosts per network.



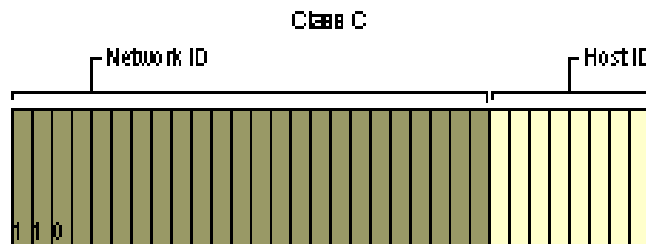
## Class B

*Class B* addresses are assigned to medium-sized to large-sized networks. The two high-order bits in a class B address are always set to binary 1 0. The next 14 bits (completing the first two octets) complete the network ID. The remaining 16 bits (last two octets) represent the host ID. This allows for 16,384 networks and 65,534 hosts per network. Figure 1.5 illustrates the structure of class B addresses.



## Class C

*Class C* addresses are used for small networks. The three high-order bits in a class C address are always set to binary 1 1 0. The next 21 bits (completing the first three octets) complete the network ID. The remaining 8 bits (last octet) represent the host ID. This allows for 2,097,152 networks and 254 hosts per network. Figure 1.6 illustrates the structure of class C addresses.



## Class D

*Class D* addresses are reserved for IP multicast addresses. The four high-order bits in a class D address are always set to binary 1 1 1 0. The remaining bits are for the address that interested hosts recognize. Microsoft supports class D addresses for applications to multicast data to multicast-capable hosts on an internetwork.

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

## Class E

Class E is an experimental address that is reserved for future use. The high-order bits in a class E address are set to 1111.

Table 2 is a summary of address classes A, B, and C that can be used for host IP addresses.

**Table 2: IP Address Class Summary**

Class	Value for w	Network ID Portion	Host ID Portion	Available Networks	Hosts per Network
A	1–126	w	x.y.z	126	16,777,214
B	128–191	w.x	y.z	16,384	65,534
C	192–223	w.x.y	z	2,097,152	254

The class A address 127.x.y.z is reserved for loopback testing and interprocess communication on the local computer.

## Network ID Guidelines

The network ID identifies the TCP/IP hosts that are located on the same physical network. All hosts on the same physical network must be assigned the same network ID to communicate with each other. Follow these guidelines when assigning a network ID:

- The network ID must be unique to the IP internetwork. If you plan on having a direct routed connection to the public Internet, the network ID must be unique to the Internet. If you do not plan on connecting to the public Internet, the local network ID must be unique to your private internetwork.
- The network ID cannot begin with the number 127. The number 127 in a class A address is reserved for internal loopback functions.
- All bits within the network ID cannot be set to 1. All 1's in the network ID are reserved for use as an IP broadcast address.
- All bits within the network ID cannot be set to 0. All 0's in the network ID are used to denote a specific host on the local network and are not routed.

Table 3 lists the valid ranges of network IDs based on the IP address classes. To denote IP network IDs, the host bits are all set to 0.

**Table 3: Class Ranges of Network IDs**

Address Class	First Network ID	Last Network ID
Class A	1.0.0.0	126.0.0.0
Class B	128.0.0.0	191.255.0.0
Class C	192.0.0.0	223.255.255.0

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

## Host ID Guidelines

The host ID identifies a TCP/IP host within a network. The combination of IP network ID and IP host ID is an IP address. Follow these guidelines when assigning a host ID:

- The host ID must be unique to the network ID.
- All bits within the host ID cannot be set to 1 because this host ID is reserved as a broadcast address to send a packet to all hosts on a network.
- All bits in the host ID cannot be set to 0 because this host ID is reserved to denote the IP network ID.

Table 4 lists the valid ranges of host IDs based on the IP address classes.

**Table 4: Class Ranges of Host IDs**

Address Class	First Host ID	Last Host ID
Class A	w.0.0.1	w.255.255.254
Class B	w.x.0.1	w.x.255.254
Class C	w.x.y.1	w.x.y.254

## Subnetting And The Subnet Mask

The Internet Address Classes accommodate three scales of IP internetworks, where the 32-bits of the IP address are apportioned between network IDs and host IDs depending on how many networks and hosts per network are needed. However, consider the class A network ID, which has the possibility of over 16 million hosts on the same network. All the hosts on the same physical network bounded by IP routers share the same broadcast traffic; they are in the same broadcast domain. It is not practical to have 16 million nodes in the same broadcast domain. The result is that most of the 16 million host addresses are unassignable and are wasted. Even a class B network with 65 thousand hosts is impractical.

In an effort to create smaller broadcast domains and to better utilize the bits in the host ID, an IP network can be subdivided into smaller networks, each bounded by an IP router and assigned a new subnetted network ID, which is a subset of the original class-based network ID. This creates *subnets*, subdivisions of an IP network each with their own unique subnetted network ID. Subnetted network IDs are created by using bits from the host ID portion of the original class-based network ID.

A **subnet** is a physical segment in a TCP/IP environment that uses IP addresses derived from a single network ID. Typically, an organization acquires one network ID from the InterNIC. Dividing the network into subnets requires that each segment use a different network ID, or subnet ID. A unique subnet ID is created for each segment by **partitioning the bits in the host ID** into two parts. One part is used to identify the segment as a unique network, and the other part is used to identify the hosts. This is referred to as **subnetting** or **subnetworking**. Subnetting is not necessary if the network is private, or if network segmentation is not necessary and there are ample available IP addresses.

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

There are several benefits to subnetting. Organizations use subnetting to apply one network across multiple physical segments. With subnetting you can:

- Mix different technologies, such as Ethernet and Token Ring.
- Overcome limitations of current technologies, such as exceeding the maximum number of hosts per segment.
- Reduce network congestion by redirecting traffic and reducing broadcasts.

With the advent of subnetting, one can no longer rely on the definition of the IP address classes to determine the network ID in the IP address. A new value is needed to define which part of the IP address is the network ID and which part is the host ID regardless of whether class-based or subnetted network IDs are being used.

RFC 950 defines the use of a *subnet mask* (also referred to as an address mask) as a 32-bit value that is used to distinguish the network ID from the host ID in an arbitrary IP address. The bits of the subnet mask are defined as follows:

- All bits that correspond to the network ID are set to 1.
- All bits that correspond to the host ID are set to 0.

Each host on a TCP/IP network requires a subnet mask even on a single segment network. Either a default subnet mask, which is used when using class-based network IDs, or a custom subnet mask, which is used when subnetting or supernetting, is configured on each TCP/IP node.  
dotted Decimal Representation of Subnet Masks

Subnet masks are frequently expressed in dotted decimal notation. After the bits are set for the network ID and host ID portion, the resulting 32-bit number is converted to dotted decimal notation. Note that even though expressed in dotted decimal notation, a subnet mask is not an IP address.

A default subnet mask is based on the IP address classes and is used on TCP/IP networks that are not divided into subnets. Table 4 lists the default subnet masks using the dotted decimal notation for the subnet mask.

**Table 4: Default Subnet Masks (Dotted Decimal Notation)**

Address Class	Bits for Subnet Mask	Subnet Mask
Class A	11111111 00000000 00000000 00000000	255.0.0.0
Class B	11111111 11111111 00000000 00000000	255.255.0.0
Class C	11111111 11111111 11111111 00000000	255.255.255.0

Custom subnet masks are those that differ from these default subnet masks when you are doing subnetting or supernetting. For example, 138.96.58.0 is an 8-bit subnetted class B network ID. Eight bits of the class-based host ID are being used to express subnetted network IDs. The subnet mask uses a total of 24 bits (255.255.255.0) to define the subnetted network ID. The subnetted network ID and its corresponding subnet mask is then expressed in dotted decimal notation as: 138.96.58.0, 255.255.255.0

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

Because the network ID bits must always be chosen in a contiguous fashion from the high order bits, a shorthand way of expressing a subnet mask is to denote the number of bits that define the network ID as a network prefix using the network prefix notation: /<# of bits>. Table 5 lists the default subnet masks using the network prefix notation for the subnet mask.

**Table 5: Default Subnet Masks (Network Prefix Notation)**

Address Class	Bits for Subnet Mask	Network Prefix
Class A	11111111 00000000 00000000 00000000	/8
Class B	11111111 11111111 00000000 00000000	/16
Class C	11111111 11111111 11111111 00000000	/24

For example, the class B network ID 138.96.0.0 with the subnet mask of 255.255.0.0 would be expressed in network prefix notation as 138.96.0.0/16.

As an example of a custom subnet mask, 138.96.58.0 is an 8-bit subnetted class B network ID. The subnet mask uses a total of 24 bits to define the subnetted network ID. The subnetted network ID and its corresponding subnet mask is then expressed in network prefix notation as 138.96.58.0/24. Network prefix notation is also known as Classless Interdomain Routing (CIDR) notation.

Because all hosts on the same network must use the same network ID, all hosts on the same network must use the same network ID as defined by the same subnet mask. For example, 138.23.0.0/16 is not the same network ID as 138.23.0.0/24. The network ID 138.23.0.0/16 implies a range of valid host IP addresses from 138.23.0.1 to 138.23.255.254. The network ID 138.23.0.0/24 implies a range of valid host IP addresses from 138.23.0.1 to 138.23.0.254.

## Subnetting

Although the conceptual notion of subnetting by utilizing host bits is straightforward, the actual mechanics of subnetting are a bit more complicated. Subnetting requires a three step procedure:

1. Determine the number of host bits to be used for the subnetting.
2. Enumerate the new subnetted network IDs.
3. Enumerate the IP addresses for each new subnetted network ID.

A subnet mask is a 32-bit address used to block or "mask" a portion of the IP address to distinguish the network ID from the host ID. This is necessary so that TCP/IP can determine whether an IP address is located on a local or remote network.

Each host on a TCP/IP network requires a subnet mask. This can either be a default subnet mask (used when a network is not divided into subnets) or a custom subnet mask (used when a network is divided into subnets). **In the default subnet mask**, all bits that correspond to the network ID are set to 1. The decimal value in each octet is then 255. All bits that correspond to the host ID are set to 0.

## Interpreting Subnets

There are two ways of specifying subnets. One notation uses the network address and mask in this format: 192.9.200.0 and 255.255.255.0. This format specifies the mask in dotted decimal notation.

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

The four bytes together represent a series of 1s and 0s that determine the amount of each IP address associated with the network and the amount associated with the hosts. In the example above, the netmask can also be expressed as the number 11111111111111111111111100000000, showing that 24 bits are used for the network address and 8 for the hosts.

Increasingly, another notation is used to represent the same information. Using the second notation, the sample network shown above would be expressed as 192.9.200.0/24. The 24 in this notation means that 24 bits of each address represent the network, leaving the remaining 8 bits for the host.

These two ways of expressing the addresses associated with a network are interchangeable and both are easy to express when the network/host boundary falls on a byte boundary. When this is not the case, the first notation is a bit trickier (for me at least), requiring some amount of binary to decimal translations. For example, stealing one bit from the host portion of the address would allow us to break a Class C address into two subnets. In the first notation, the first subnet address would be 192.9.200.0 with a netmask of 255.255.255.128. In the second, it would be 192.9.200.0/25.

As you can see, the stolen bit has a noticeable effect on the netmask. I have to pause and remember that the leftmost bit in a byte is worth 128. If we were to steal 2 bits, creating four subnets instead, the mask would be 255.255.255.192. With each additional stolen bit, the mask goes up by the next smaller power of 2.

<b>bits stolen</b>	<b>subnet mask</b>
=====	=====
1	255.255.255.128
2	255.255.255.192
3	255.255.255.224
4	255.255.255.240
5	255.255.255.248

Your subnets are almost too small to work with by the time you get down to only 3 bits remaining for hosts. With 3 bits, we only have 8 possible addresses and only six of those could be used for hosts, the other two being the network and broadcast addresses.

Given any host address and netmask, you can easily determine all of the host addresses in a particular network. For example, the address 10.20.100.200/25 would be a host on the upper subnet of the 10.20.100.0 network. The network address would be 10.20.100.128 (lowest address in the subnet), broadcast address 10.20.100.255 (network address with all 1's in the host portion), and there would be 126 host address, starting with 10.20.100.129 and going through 10.20.100.254.

The existence of the network described in the above paragraph does not indicate that the lower subnet is also defined the same way (a 128-address network). In fact, that address space could be broken into smaller chunks of 64, 32, 16 or even 8 addresses.

## ANDING - Determining the Network ID

To extract the network ID from an arbitrary IP address using an arbitrary subnet mask, IP uses a mathematical operation called a logical AND comparison. In an AND comparison, the result of two items being compared is true only when both items being compared are true; otherwise, the result is false. Applying this principle to bits, the result is 1 when both bits being compared are 1, otherwise the result is 0.

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

IP performs a logical AND comparison with the 32-bit IP address and the 32-bit subnet mask. This operation is known as a bit-wise logical AND. The result of the bit-wise logical AND of the IP address and the subnet mask is the network ID. For example, what is the network ID of the IP node 129.56.189.41 with a subnet mask of 255.255.240.0:

To obtain the result, turn both numbers into their binary equivalents and line them up. Then perform the AND operation on each bit and write down the result.

```
10000001 00111000 10111101 00101001 IP Address
11111111 11111111 11110000 00000000 Subnet Mask
10000001 00111000 10110000 00000000 Network ID
```

The result of the bit-wise logical AND of the 32 bits of the IP address and the subnet mask is the network ID 129.56.176.0.

ANDing is the internal process that IP uses to determine whether a packet is destined for a host on a local or remote network. When TCP/IP is initialized, the host's IP address is ANDed with its subnet mask. Before a packet is sent, the destination IP address is ANDed with the same subnet mask. If the results of ANDing the source IP address and destination IP address match, IP knows that the packet belongs to a host on the local network. If the results do not match, the packet is sent to the IP address of an IP router.

To AND the IP address to a subnet mask, TCP/IP compares each bit in the IP address to the corresponding bit in the subnet mask. If both bits are 1's, the resulting bit is 1. If there is any other combination, the resulting bit is 0.

## Implementing Subnetting

Before implementing subnetting, you need to determine your current requirements and plan for future requirements. Follow these three guidelines:

1. Determine the number of physical segments on your network, plus the number of segments required for the future.
2. Determine the number of required host addresses for each physical segment currently, plus the number of hosts per segments required for the future.
3. Based on the requirements, define:
  - One subnet mask for the entire network.
  - A unique subnet ID for each physical segment.
  - A range of host ID's for each subnet.

When more bits are used for the subnet mask, more subnets are available, but fewer hosts are available per subnet. For example, for a Class B network:

```
3 bits = 6 subnets = 8,000 hosts per subnet
8 bits = 254 subnets = 254 host per subnet
```

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

If more bits are used than needed, it will allow for growth in the number of subnets, but it will limit the growth in the number of hosts. If fewer bits are used than needed, it will allow for growth in the number of hosts, but it will limit the growth in the number of subnets.

## Defining A Subnet Mask

Defining a subnet is a three step process:

1. Once you have determined the number of physical segments needed in the network environment, convert this number to binary format.
2. Count the number of bits required to represent the number of physical segments in binary. For example, if you need six subnets, the binary value is 110. Representing six in binary required 3 bits.
3. Convert the required number of bits to decimal format in high order (from left to right). For example (Class B Address), if 3 bits are required, configure the first 3 bits of the host ID (unique IP address assigned by InterNIC) as the subnet ID. The decimal value for binary 11100000 is 224. **The subnet mask then is 255.255.224.0.**

### EXAMPLE:

Number of Subnets:	6			
Binary Value:	00000110 (3 bits)			
Convert to Decimal:	11111111	11111111	11100000	00000
Subnet Mask =	255.	255.	224.	0

The following three tables assist in determining the correct netmask number:

DECIMAL EQUIVALENTS FOR NETMASKING								
128	64	32	16	8	4	2	1	Decimal
1	0	0	0	0	0	0	0	128
1	1	0	0	0	0	0	0	192
1	1	1	0	0	0	0	0	224
1	1	1	1	0	0	0	0	240
1	1	1	1	1	0	0	0	248
1	1	1	1	1	1	0	0	252
1	1	1	1	1	1	1	0	254
1	1	1	1	1	1	1	1	255

SHORTENED NETMASK TABLE								
128	64	32	16	8	4	2	1	Binary
128	192	224	240	248	252	254	255	Decimal

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

<b>SUBNET MASK CONVERSION TABLE</b> (Based On One Octet Use)			
<b>Number of Subnets</b>	<b>Required Bits</b>	<b>Subnet Mask</b>	<b>Hosts per Subnet</b>
<b>Class A Networks</b>			
0	1	Invalid	Invalid
2	2	255.192.0.0	4,194,302
6	3	255.224.0.0	2,097,150
14	4	255.240.0.0	1,048,574
30	5	255.248.0.0	524,286
62	6	255.252.0.0	262,142
126	7	255.254.0.0	131,070
254	8	255.255.0.0	65,534
<b>Number of Subnets</b>	<b>Required Bits</b>	<b>Subnet Mask</b>	<b>Hosts per Subnet</b>
<b>Class B Networks</b>			
0	1	Invalid	Invalid
2	2	255.255.192.0	16,382
6	3	255.255.224.0	8,190
14	4	255.255.240.0	4,094
30	5	255.255.248.0	2,046
62	6	255.255.252.0	1,022
126	7	255.255.254.0	510
254	8	255.255.255.0	256
<b>Number of Subnets</b>	<b>Required Bits</b>	<b>Subnet Mask</b>	<b>Hosts per Subnet</b>
<b>Class C Networks</b>			
Invalid	1	Invalid	Invalid
1-2	2	255.255.255.192	62
3-6	3	255.255.225.224	30
7-14	4	255.255.255.240	14
15-30	5	255.255.255.248	6
31-62	6	255.255.255.252	2
Invalid	7	Invalid	Invalid
Invalid	8	Invalid	Invalid

## Defining Subnet IDs

You can define the subnet ID for a physical segment using the same number of host bits as used for the subnet mask. Use the following steps:

1. List the number of bits in high order used for the subnet ID. For example, if 2 bits are used for the subnet mask, the binary octet is 11000000. If 4 bits are used in the subnet mask, the binary octet is 11110000.
2. Convert the bit with the lowest value to decimal format. This is the increment value to determine each subnet. For example, if you use 2 bits, the lowest value (or increment value) is 64. If you use 4 bits, the lowest value (or increment value) is 16.

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

3. To determine the number of possible subnets, convert the number of bits to decimal format in low order, and then subtract 1. For 2 bits  $00000011 = 3 - 1 = 2$  subnets. For 4 bits  $00001111 = 15 - 1 = 14$  subnets. Another way to calculate the number of possible subnets is if you know the number of bits you need, you can raise 2 to the power of the bit, and then subtract 2. For example 2 bits  $2^2 = 4 - 2 = 2$  subnets. 4 bits  $2^4 = 16 - 2 = 14$  subnets
4. Starting with zero, increment the value (determined in step 2) for each bit combination until the next increment is 256 (but cannot use same value as subnet mask). With a lowest value of 64, the first subnet ID range would be 64 -> 127, with the second ID range being from 128 -> 191.

## Defining Host IDs For A Subnet

If you have already defined the subnet IDs, then you have already defined your host IDs for each subnet. The result of each incremental value indicates the beginning of a range of host IDs for a subnet. If you increment the value one extra time, you can determine the end of the range (one less than the subnet mask). To determine the number of hosts per subnet:

1. Calculate the number of bits available for the host ID. For example, for a Class B address that uses 16 bits for the network ID and 2 bits for the subnet ID, you have 14 bits remaining for the host ID.
2. Convert the binary host ID bits to decimal. For example, in the case of 14 host ID bits, 11111111111111 in binary is converted to 16,383 in decimal format.
3. Subtract 1. Another way to calculate this would be if you know the number of host ID bits you need, raise 2 to the power of the number of host ID bits, and then subtract 2.

## Example Of Subnetting A Class C Network

Class C networks can themselves be subnetted, although with some difficulty. Using the address 192.1.1.0, we'll show how this can be done. Let's say we want six subnetworks each having 30 hosts. Using the eight available bits to work with, we would use the three high order bits for subnet ID's (or subnetwork addresses), with the remaining five bits going to host ID's. This would allot six networks with 30 hosts each. The subnet mask would need to be 255.255.255.224. The table below shows the Class C networks available (32 is the increment number):

SUBNETWORKS FOR CLASS C NETWORK	
Binary Number	Decimal Equivalent
00100000	32 (192.1.1.32)
01000000	64 (192.1.1.64)
01100000	96 (192.1.1.96)
10000000	128 (192.1.1.128)
10100000	160 (192.1.1.160)
11000000	192 (192.1.1.192)

The table below shows what the valid host address (ID's) would be, and those that are invalid for various reasons. Remember, the **broadcast address for each subnetwork must be considered**. **To get the broadcast addresses**, take the subnet address, set all the host bit's to 1's, and add them to get the broadcast address. We use five bits for host addresses, so the decimal value of the sixth

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

bit is 32. Subtracting 1 gives 31. Thus, setting the five host bits to 1's (00011111) gives a value of 31 decimal. Adding this to the last byte of the subnet address reveals the subnetwork broadcast address.

Using the subnetwork 192.1.1.160 as an example, we end up with the following:

**Subnet Address** = **192.1.1.160**  
**Valid Host Addresses** = **192.1.1.1161-190**  
**Directed Broadcast Address** = **192.1.1.191**

ANALYSIS OF 256 VALUES OF LAST BYTE		
Last Byte Addresses	Validity	Reason
0-31	Invalid	Not part of any subnet
32	Invalid	First subnet address
33-62	Valid	Hosts on the first subnet
63	Invalid	Broadcast address of the first subnet
64	Invalid	Second subnet address
65-94	Valid	Hosts on the second subnet
95	Invalid	Broadcast address of the second subnet
96	Invalid	Third subnet address
97-126	Valid	Hosts on the third subnet
127	Invalid	Broadcast address of the third subnet
128	Invalid	Fourth subnet address
129-158	Valid	Hosts on the fourth subnet
159	Invalid	Broadcast address of the fourth subnet
160	Invalid	Fifth subnet address
161-190	Valid	Hosts on the fifth subnet
191	Invalid	Broadcast address of the fifth subnet
192	Invalid	Sixth subnet address
193-222	Valid	Hosts on the sixth subnet
223	Invalid	Broadcast address of the sixth subnet
224	Invalid	Subnet Mask (SNM)
225-255	Invalid	Above SNM

## More Examples

Let's use these two addresses for some examples: 171.68.3.3 and 171.68.2.3. If the subnet mask is 255.255.255.0, the first 24 bits are masked, so the router compares the first 3 octets of the two addresses. Since the masked bits are not the same, the router knows that these addresses belong to different subnets.

If the subnet mask is 255.255.0.0, the first 16 bits are masked, so the router compares the first 2 octets of the two addresses. Since the masked bits are the same, the router knows that these addresses belong to the same subnet.

Sometimes you need to perform a logical "AND" operation to find out what subnet your node is in. Performing an "AND" operation means that anytime you "AND" a 0 value to another 0 or a 1 value, the result is 0. Only a 1 ANDed with another 1 value will result in a 1 value. Here's how it works:

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

**0 AND 0 IS 0**

**0 AND 1 IS 0**

**1 AND 1 IS 1**

Let's compare our sample addresses (171.68.3.3 and 171.68.2.3) against the subnet mask 255.255.240.0. We need to compare the binary representation of the third octet of the mask with the binary representation of the third octets of the addresses. In order to do this, we'll perform a logical "AND" operation on the corresponding bits in each octet.

The masked bits are those that are "turned on," or 1 in the mask. Since the masked bits in both addresses are the same, the router knows that these addresses belong to the same subnet.

## Example 1: Class B

Let's use a class B address to illustrate how subnetting works. Let's say you were assigned the class B address 172.16 from the NIC. First determine how many subnets you need, and how many nodes per subnet you need to define. A typical (and easy to use) class B subnet mask would be 8 bits. Since the 3rd octet is the first "free" octet for Class B, you will start there. So, an 8 bit subnet mask would be 255.255.255.0. This means you have 254\* subnets available and 254 addresses for nodes per subnet.

\*Why are there only 254 subnets available instead of 256 (0-255)? You should not use subnet 0 or a subnet of all 1s. With an all 1s subnet mask, this is also your broadcast address. You can configure this, but it is neither proper nor recommended to make your subnet the same as your broadcast address. Subnet 0 is also not recommended. Cisco will allow the use of subnet 0 with the IP subnet zero command.

## Example 2: Class B

Now let's take this example: you have just assigned an interface the address 172.16.10.50 with a mask of 255.255.255.0. What subnet is it in? First represent the bits in binary (for class B, you start with the 3rd octet since octets 1 and 2 are fixed).

### SUBNET HOST

**00001010 00110010 (address representation - 10.50)**

**11111111 00000000 (subnet mask representation - 255.0)**

-----

**00001010 00000000 (results of logical "AND" - subnet 10) 10**

This address is in subnet 10 (172.16.10.0). Valid addresses for subnet 10 would be 172.16.10.1 through 172.16.10.254. Address 172.16.10.255 is the broadcast address for this subnet. According to the standard, any host id consisting of all 1s is reserved for broadcast.

## Example 3: Class B

Let's say you have a need for more subnets than 254. (Remember this is the maximum number of subnets in a single octet.) Sticking with our class B address, let's configure an 11bit subnet. This means we will use all 8 bits from our 3rd octet and the first three bits from the 4th octet. The subnet mask is now 255.255.255.224 (128+64+32=224). Now you need to find out what subnet the following address is in: 172.16.10.170 255.255.255.224. First, denote the address in binary representation (just octets 3 and 4 for a class B address) like this:

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

```
00001010 10101010 (address representation 10.170)
11111111 11100000 (subnet mask representation 255.224-first 11 bits subnet)
-----
00001010 10100000 (results of logical "AND") 10 160
```

So, the address here is in subnet 172.16.10.160. The valid addresses for this subnet are 172.16.10.161 through 172.16.10.190 (.191 is the broadcast address). As soon as you hit 10.192, the bits in the subnet change and you move into subnet 10.192.

## Example 4: Class B

Let's take an example where the mask is shorter than one octet. Now we want only a few subnets, but need many hosts per subnet. We'll use a 3 bit subnet mask. Now we have: 172.16.65.170 255.255.224.0 (the mask is now the first 3 bits of the 3rd octet). What subnet is this address in?

```
01000001 10101010 (address representation 65.170)
11100000 00000000 (subnet mask representation 224.0)
-----
01000000 00000000 (results of logical "AND" - subnet 64) 64
```

So the subnet here is 172.16.64.0. The range of addresses that would fall into subnet 64 would be 172.16.64.1 - 172.16.95.254 with 172.16.95.255 as the broadcast address. The next subnet would be 172.16.96.0. Class A and class C map out exactly as class B. The only differences are at which octet subnetting starts and how many octets you can use for subnetting.

## Example 5: Class C

Suppose the NIC assigned the address 192.1.10.200. You will need to use the 4th octet for your subnetting needs. Let's use a 4 bit subnet mask and map out the following address: 192.1.10.200 255.255.255.240

```
11001000 (address representation for 200)
11110000 (subnet mask representation for 240)
-----
11000000 (results of logical "AND" - 128+64=192)
```

So, address 192.1.10.200 is in subnet 192. The valid range of addresses in this subnet would be 192.1.10.192 through 192.1.10.206, with .207 as the broadcast address. The next subnet would be .208.

Keeping the same subnet mask, you can choose different addresses to be in different subnets. For instance, address 192.1.10.17 255.255.255.240 is in subnet 16 and therefore has another unique subnet address, with valid addresses in the range of 192.1.10.17 through 192.1.10.30.

## Public And Private

If your intranet is not connected to the Internet, any IP addressing can be deployed. If direct (routed) or indirect (proxy or translator) connectivity to the Internet is desired, there are two types of addresses employed on the Internet, *public addresses* and *private addresses*.

Public addresses are assigned by InterNIC and consist of class-based network IDs or blocks of CIDR-based addresses (called CIDR blocks) that are guaranteed to be globally unique to the Internet.

# IP Addressing And Subnetting Simplified

By Mark E. Donaldson

When the public addresses are assigned, routes are programmed into the routers of the Internet so that traffic to the assigned public addresses can reach their locations. Traffic to destination public addresses are reachable on the Internet.

Private intranets that have no intent on connecting to the Internet can choose any addresses they want, even public addresses that have been assigned by the InterNIC. If an organization later decides to connect to the Internet, its current address scheme might include addresses already assigned by the InterNIC to other organizations. These addresses would be duplicate or conflicting addresses and are known as *illegal addresses*. Connectivity from illegal addresses to Internet locations is not possible. As long as the private organization does not connect to the Internet, there is no problem because the two address spaces are on separate IP internetworks.

## Private Addresses

Each IP node requires an IP address that is globally unique to the IP internetwork. In the case of the Internet, each IP node on a network connected to the Internet requires an IP address that is globally unique to the Internet. For the hosts within the organization that do not require direct access to the Internet, IP addresses that do not duplicate already-assigned public addresses are required. To solve this addressing problem, there is a reserved a portion of the IP address space and named this space the *private address space*. An IP address in the private address space is never assigned as a public address. IP addresses within the private address space are known as *private addresses*. Because the public and private address spaces do not overlap, private addresses never duplicate public addresses. The private address space specified in RFC 1918 is defined by the following three address blocks:

### 10.0.0.0/8

The 10.0.0.0/8 private network is a class A network ID that allows the following range of valid IP addresses: 10.0.0.1 to 10.255.255.254. The 10.0.0.0/8 private network has 24 host bits that can be used for any subnetting scheme within the private organization.

### 172.16.0.0/12

The 172.16.0.0/12 private network can be interpreted either as a block of 16 class B network IDs or as a 20-bit assignable address space (20 host bits) that can be used for any subnetting scheme within the private organization. The 172.16.0.0/12 private network allows the following range of valid IP addresses: 172.16.0.1 to 172.31.255.254.

### 192.168.0.0/16

The 192.168.0.0/16 private network can be interpreted either as a block of 256 class C network IDs or as a 16-bit assignable address space (16 host bits) that can be used for any subnetting scheme within the private organization. The 192.168.0.0/16 private network allows the following range of valid IP addresses: 192.168.0.1 to 192.168.255.254.

Private addresses are not reachable on the Internet. Therefore, Internet traffic from a host that has a private address must either send its requests to an Application layer gateway (such as a proxy server), which has a valid public address, or have its private address translated into a valid public address by a network address translator (NAT) before it is sent on the Internet.