Global Knowledge®

Expert Reference Series of White Papers

Basics of IP Address Subnetting

Basics of IP Address Subnetting

Norbert Gregorio, Global Knowledge Instructor

Introduction

The specifications for IP version 4 were published in 1981 and remain in use today. The next version will be version 6. This document explains the basics of IP version 4 addressing.

IP is an OSI Reference Model Layer 3 protocol; its primary function is to deliver datagrams to the destination host based on the logical, network manager-assigned, host IP address.

The destination host may be on the same network as the source or on a different network. Before transmission, the source host must determine if the destination is local or remote. The IP protocol needs a facility to make this determination.

One way is to classify addresses. The classes would determine what portion of the IP address is common to all hosts on the same network. This allows a transmitting host to compare its address to the destination. If the network portion of the source IP address and the destination address are the same, the destination is local and the datagram can be delivered directly. Otherwise it must be delivered indirectly by forwarding the datagram to a router that will then forward it towards the destination.

An IP address is made up of 32 bits. While computers have an easy time dealing with a sequence of 32 binary digits, 1s and 0s, most human beings aren't quite as comfortable so addresses are divided into four groups of eight. Each group of eight is referred to as an octet and an address is usually represented as a sequence of four decimal byte values separated by dots. 10.50.60.21 is an example.

The class of an address can be determined by inspecting the first byte of the address. The following table shows the different classes.

First Byte	Class	Interpretation
0-127	A	N.h.h.h
128-191	B	N.N.h.h
192-223	C	N.N.N.h

From the table above we can determine that our example, 10.50.60.21, is a Class A address. It belongs to a device on network 10, and 50.60.21 is the unique host portion. By inspecting its own address, this device determines that it can directly contact any other host with 10 as the byte value of the first octet of its address. All other destinations must be routed through a router. Similarly, hosts with Class B addresses inspect the first two octets, Class C the first three.

While easy to understand and implement, this plan proves inefficient to deal with larger networks. Consider that using this technique, a Class A network can only be segmented based on Data Link Layer addressing. Since Data Link addressing identifies each host individually without providing a "network" portion to the address, the only devices that can be used to segment such networks are bridges or switches. These devices must maintain a bridging table with an entry for every host on the network. In a Class A network, each bridge could have up to 16,777,214 entries in its table. Moving data from network segment to network segment takes a long time since the bridge must look up the destination address in a very large table. Bridging proves insufficient in large networks, and often we can't bridge between different technologies such as LAN and WAN. In these cases, it is much preferred to use routers since they forward packets in large networks more efficiently and can function with mixed technologies.

Imagine the following scenario. A large organization has many facilities in different cities and many departments. The network administrators want an easy way to assign IP addresses that allows them to easily identify the computer, department, and city. Using Class A network address 10.0.0.0, and representing the facility in Montreal with the number 50, 60 representing the Sales department. The IP address of 10.50.60.21 references computer 21 in the sales department in Montreal.

For human beings this is great, but for IP hosts this could be a problem. Imagine that this computer wants to send a packet to computer 33 in the marketing department in New York City, 10.100.70.33. Computer 10.50.60.21 must determine if the destination is on the same LAN or accessible via a router. By examining its own IP address, the source host determines that all hosts whose addresses begin with the value 10 are on the same LAN and can be contacted without using a router. Since the facilities in Montreal and New York are linked by a WAN, and all traffic between the facilities must be forwarded by routers, the transmission will fail.

To deal with this issue, an additional method is used to determine if a packet can be delivered on the local network of forwarded to a router. This method is called **Subnet Masking**. As the name implies, this method allows a network to be divided into smaller subnets.

To use this method, an IP hosts is assigned an address, the address of a router, and a subnet mask to assist in determining when to use the router. The subnet mask consists of 32 bits. Each bit indicates if the corresponding bit in the address should be interpreted as part of the network portion or the host portion. The subnet mask is usually also represented by the byte value of each of the four octets that make up the 32 bits. Simply stated, each bit in the mask with a value of 1 indicates that the corresponding bit in the address is part of the network or subnetwork portion; a 0 value is interpreted as part of the host portion.

In our example, we want each host to interpret its addresses as meaning **network.city.department.host** or **network.subnet.subnet.host**. This means that the first 24 bits, three octets, of the address represent the

network and subnet portion of the address, and the remaining eight bits of the address indicate the individual host. The mask that indicates this to an IP host would have the first 24 bits set to 1 and the remaining eight set to 0. The byte value of these octets is 255.255.255.0.

By configuring the devices in our Class A example with this mask, the computer in Montreal, 10.50.60.21, can now determine that only IP hosts whose first three octets have a value of 10.50.60 can be contacted directly; all others can only be contacted through a router. The computer in New York has an IP address of 10.100.70.33, the first three octets do not match the source computer's and all traffic to that destination will be forwarded through a router. This time the transmission will succeed!

Let's look at another case, this one involving a Class C network. Consider a company with five departments; each department has less than twenty computers. The network administrators want each department to remain segmented to facilitate network management. A router is used between each department. Further, the network administrators want to provide Internet access to every computer. Their Internet service provider has assigned them the Class C address 220.250.20.0.

The question is how to subnet a Class C address into five subnets since only one octet is administered locally; the first three octets are the network portion provided by the ISP. Keeping in mind that both the IP address and subnet mask are interpreted one bit at a time, we realize that we can use some of the remaining bits for subnetting and others for the individual host.

The question now becomes how many bits do we use to represent the subnet? The answer involves binary to decimal conversion.

Step 1. Convert the number of desired subnets into binary. (5 decimal = 0000 0101 binary)

Step 2. Strip off all leading 0s and count the number of remaining digits. (101 -> 3 digits)

Step 3. We need 3 digits!

Remember a Class C address is interpreted as meaning N.N.N.H. This means that the first 24 bits represent the network address, the remaining eight the host. A subnet mask with the same meaning is 255.255.255.0. Indeed, this is the default mask for a Class C network. From the steps above, we have determined that we need to use three bits to represent the subnet. A subnet mask is always a series of 1 bits followed by a series of 0 bits. This means that the bits we use for subnetting are the first three bits of the last octet (1110 0000).

Step 4. Set the number of required subnet bits to 1. (1110 0000 binary = 224 decimal)

Step 5. We now have our new subnet mask -> 255.255.255.224

This means that every host, including routers, would be configured with an IP address and this mask. One last task remains; calculate the address for each host.

First another rule. Remember that an IP address is comprised of 32 bits. Some bits identify the network, some the subnet; the rest identify the host. The bits that identify the host may not all have a value of 1 or 0. All 1s is used for a local broadcast; all 0s identifies the network or subnetwork.

Now to calculate those IP addresses for our example. Our network address is 220.250.20.0, the mask we just calculated to be 255.255.255.224.

We have three bits to represent each of our subnets. We begin by listing all the possible combinations formed with three bits. The first combination is 000 followed by 001, 010, 011, 100, 101, 110 and finally 111. This means that the first station on the **second** subnet receives an IP address beginning 220.250.20, like all hosts in our network, followed by **001**0 0001, binary or 33 decimal (220.250.20.33). The last station on the **second** subnet gets configured with **001**1110 binary, 62 decimal (220.250.20.62). By continuing this logic for all the other subnets we can determine all the IP addresses for each host in our network

Subnet # 1		
First Address	220.250.20.1	1101 1100.1111 1010.0001 0100.***000***0 0001
Last Address	220.250.20.30	1101 1100.1111 1010.0001 0100.***000***1 1110

Subnet # 2		
First Address	220.250.20.33	1101 1100.1111 1010.0001 0100.***001***0 0001
Last Address	220.250.20.62	1101 1100.1111 1010.0001 0100.***001***1 1110

Subnet # 3		
First Address	220.250.20.65	1101 1100.1111 1010.0001 0100.***010***0 0001
Last Address	220.250.20.94	1101 1100.1111 1010.0001 0100.***010***1 1110

Subnet # 4		
First Address	220.250.20.97	1101 1100.1111 1010.0001 0100.***011***0 0001
Last Address	220.250.20.126	1101 1100.1111 1010.0001 0100.***011***1 1110

Subnet # 5		
First Address	220.250.20.129	1101 1100.1111 1010.0001 0100.***100***0 0001
Last Address	220.250.20.158	1101 1100.1111 1010.0001 0100.***100***1 1110

Subnet # 6		
First Address	220.250.20.161	1101 1100.1111 1010.0001 0100.***101***0 0001
Last Address	220.250.20.190	1101 1100.1111 1010.0001 0100.***101***1 1110

Subnet # 7		
First Address	220.250.20.193	1101 1100.1111 1010.0001 0100.***1100*** **0001**
Last Address	220.250.20.222	1101 1100.1111 1010.0001 0100.***1101*** **1110**

Subnet # 8		
First Address	220.250.20.225	1101 1100.1111 1010.0001 0100.***1110*** **0001**
Last Address	220.250.20.254	1101 1100.1111 1010.0001 0100.***1111*** **111**

Notice that while we only need four subnets in our example, we end up with a subnet allowing eight. Such is the nature of subnetting; we can't get exactly what we want. The remaining subnets are available for future growth.

Learn More

Learn more about how you can improve productivity, enhance efficiency, and sharpen your competitive edge through training.

CCNAX – CCNA Boot Camp v1.1

TCP/IP Networking

Understanding Networking Fundamentals

Visit **www.globalknowledge.com** or call **1-800-COURSES (1-800-268-7737)** to speak with a Global Knowledge training advisor.

About the Author

Norbert Gregorio has been a technical instructor for the last fifteen years. He has delivered Novell-, Microsoft-, and Cisco-certified curricula in both of Canada's official languages, English and French. Having delivered certified courses for many networking platforms over the years has allowed the him to develop a variety of strategies for explaining IP subnetting. The technique described above has proven very successful over the years.