

## CCNP 1: Advanced IP Addressing Management

- ?? By [Cisco Systems, Inc.](#)
- ?? Sample Chapter is provided courtesy of [Cisco Press](#).
- ?? Date: Aug 27, 2004.

## Objectives

Upon completing this chapter, you will be able to

- ?? Create and configure IPv4 addresses
- ?? Understand and resolve IP addressing crises
- ?? Assign a VLSM addressing scheme
- ?? Configure route summarization
- ?? Configure private addressing and NAT
- ?? Use IP unnumbered
- ?? Understand and configure DHCP and Easy IP
- ?? Know when to use helper addresses
- ?? Understand the concepts of IPv6

You can reinforce your understanding of the objectives covered in this chapter by opening the interactive media activities on the CD accompanying this book and performing the lab activities collected in the *Cisco Networking Academy Program CCNP 1: Advanced Routing Lab Companion*. Throughout this chapter, you will see references to these activities by title and by icon. They look like this:

Interactive Media Activity

Lab Activity

A scalable network requires an addressing scheme that allows for growth. However, several unanticipated consequences can result from unmanaged network growth. As new nodes and networks are added to the enterprise, existing addresses might need to be reassigned. Excessively large routing tables might slow down older routers, and the supply of available addresses might simply run out. You can avoid these unpleasant consequences with careful planning and deployment of a scalable network addressing system.

Network designers can choose from among many different network protocols and addressing schemes. However, with the emergence of the Internet and its nonproprietary protocol, Transmission Control Protocol/Internet Protocol (TCP/IP), this has meant that virtually every enterprise must implement an IP addressing scheme. In addition to TCP/IP, several proprietary network protocols and addressing schemes have been used. Companies such as Apple and Novell have recently migrated their network software to TCP/IP and away from their proprietary protocols. Presently, many organizations choose to run TCP/IP as the only routed protocol on the network. The bottom line is that administrators must find ways to scale their networks by using IP addressing.

Unfortunately, the architects of TCP/IP could not have predicted that their protocol would eventually sustain a global network of information, commerce, and entertainment. Twenty years ago, IP version 4 (IPv4) offered an addressing strategy that, although scalable for a time, resulted in an inefficient allocation of addresses. Over the past two decades, engineers have successfully modified IPv4 so that it can survive the Internet's exponential growth. Meanwhile, an even more extensible and scalable version of IP, IP version 6 (IPv6), has been defined and developed. Today, IPv6 is slowly being implemented in select networks. Eventually, IPv6 might replace IPv4 as the dominant Internet protocol.

This chapter explores the evolution and extension of IPv4, including the key scalability features that engineers have added to it over the years:

- ?? Subnetting
- ?? *Classless interdomain routing (CIDR)*
- ?? *Variable-length subnet masking (VLSM)*
- ?? Route summarization

Finally, this chapter examines advanced IP implementation techniques such as the following:

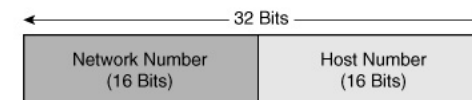
- ?? IP unnumbered
- ?? Dynamic Host Configuration Protocol
- ?? Helper addresses

## IPv4 Addressing

This section covers some of the basic concepts of IPv4 addressing, such as how the Internet's address architecture uses the binary and dotted-decimal versions of IPv4 addressing. This section also reviews the structure of IPv4 addresses, such as the various classes of IPv4 addresses. Finally, this section reviews how IPv4 addresses use subnet masks to help divide and manage the size and growth of the Internet and computer networks.

### Address Architecture of the Internet

When TCP/IP was introduced in the 1980s, it relied on a two-level addressing scheme. At the time, this scheme offered adequate scalability. The 32-bit-long IPv4 address identifies a network number and a host number, as shown in Figure 2-1.



IP Address Structure

Together, the network number and the host number uniquely identify all hosts connected by way of the Internet. It is possible that the needs of a small networked community, such as a LAN, could be satisfied with just host addresses. However, network addresses are necessary for end systems on different networks to communicate with each other. Routers use the network portion of the address to make routing decisions and to facilitate communication between hosts that belong to different networks.

Unlike routers, humans find working with strings of 32 1s and 0s tedious and clumsy. Therefore, 32-bit IP addresses are written using dotted-decimal notation. Each 32-bit address is divided into four groups of eight, called octets. Each octet is converted to decimal and then separated by decimal points, or dots. This is illustrated as follows:

?? A 32-bit IP address is a binary number:

```
1010110000011110100000000010001
```

?? This binary number can be divided into four octets:

```
10101100 00011110 10000000 00010001
```

?? Each octet (or byte) can be converted to decimal:

```
172 30 128 17
```

?? Finally, the address can be written in dotted-decimal notation:

```
172.30.128.17
```

In the dotted-decimal address 172.30.128.17, which of these four numbers represents the network portion of the address? Which numbers are the host numbers? Finding the answers to these questions is complicated by the fact that IP addresses are not really four numbers. They actually consist of 32 different numbers, or 32 bits.

In the early days of TCP/IP, a class system was used to define the network and host portions of the address. IPv4 addresses were grouped into five distinct classes. This was done according to the value of the first few bits in the first octet of the address. Although the class system can still be applied to IP addresses, networks today often ignore the rules of class in favor of a classless IP scheme.

The next few sections cover all of the following topics related to IP addressing:

- ?? The limitations of the IP address classes
- ?? The subsequent addition of the subnet mask
- ?? The addressing crisis that led to the adoption of a classless system

### Class A and B IP Addresses

In a class system, IP addresses can be grouped into one of five different classes:

- ?? A
- ?? B
- ?? C

?? D

?? E

Each of the four octets of an IP address represents either the network portion or the host portion of the address, depending on the address class. The network and host portions of the respective Class A, B, C, and D addresses are shown in Figure 2-2.

Class A	Network			Host
Octet	1	2	3	4

Class B	Network		Host	
Octet	1	2	3	4

Class C	Network			Host
Octet	1	2	3	4

Class D	Host			
Octet	1	2	3	4

### Address Structure

Only the first three classes—A, B, and C—are used to address actual hosts on IP networks. Class D addresses are used for multicasting. Class E addresses are reserved for experimentation and are not shown in Figure 2-2. The following sections explore each of the five classes of addresses.

#### Class A Addresses

If the first bit of the first octet of an IP address is a binary 0, the address is a Class A address. With that first bit being a 0, the lowest number that can be represented is 00000000, decimal 0. The highest number that can be represented is 01111111, decimal 127. Any address that starts with a value between 0 and 127 in the first octet is a Class A address. These two numbers, 0 and 127, are reserved and cannot be used as a network address.

Class A addresses were intended to accommodate very large networks, so only the first octet is used to represent the network number. This leaves three octets, or 24 bits, to represent the host portion of the address. With 24 bits total,  $2^{24}$  combinations are possible, yielding 16,777,216 possible addresses. Two of those possibilities, the lowest and highest values, are reserved for special purposes. The low value is 24 0s, and the high value is 24 1s. Therefore, each Class A address can support up to 16,777,214 unique host addresses.

Why are two host addresses reserved for special purposes? Every network requires a network number. A network number is an ID number that is used to refer to the entire range of hosts when building routing tables. The address that contains all 0s in the host portion is used as the network number and cannot be used to address an individual node. 46.0.0.0 is a Class A network number. Similarly, every network requires a broadcast address that can be used to address a message to

every host on a network. It is created when the host portion of the address has all 1s. For example, a broadcast address for network 46.0.0.0 would be 46.255.255.255.

With almost 17 million host addresses available, a Class A network actually provides too many possibilities for one company or campus. Although it is easy to imagine an enormous global network with that many nodes, the hosts in such a network could not function as members of the same logical group. Administrators require much smaller logical groupings to control broadcasts, apply policies, and troubleshoot problems. Fortunately, the subnet mask allows subnetting, which breaks a large block of addresses into smaller groups called subnetworks. All Class A networks are subnetted. If they were not, Class A networks would represent huge waste and inefficiency.

How many Class A addresses are there? Because only the first octet is used as a network number, and it contains a value between 0 and 126, 126 Class A networks exist. Each of the 126 Class A addresses has almost 17 million possible host addresses that make up about half of the entire IPv4 address space. Recall that the network address 127.0.0.1 is reserved for the local loopback address, which is why Class A addresses stop at 126.0.0.0 and Class B addresses start at 128.0.0.0. Under this system, a mere handful of organizations control half of the available Internet addresses.

### Class B Addresses

Class B addresses start with a binary 10 in the first 2 bits of the first octet. Therefore, the lowest number that can be represented with a Class B address is 10000000, decimal 128. The highest number that can be represented is 10111111, decimal 191. Any address that starts with a value in the range of 128 to 191 in the first octet is a Class B address.

Class B addresses were intended to accommodate medium -size networks. Therefore, the first two octets are used to represent the network number, which leaves two octets or 16 bits to represent the host portion of the address. With 16 bits total,  $2^{16}$  combinations are possible, yielding 65,536 Class B addresses. Recall that two of those numbers, the lowest and highest values, are reserved for special purposes. Therefore, each Class B address can support up to 65,534 hosts. Although it is significantly smaller than the networks created by Class A addresses, a logical group of more than 65,000 hosts is still unmanageable and impractical. Therefore, like Class A networks, Class B addresses are subnetted to improve efficiency.

Because the first 2 bits of a Class B address are always 10, 14 bits are left in the network portion of the address, resulting in  $2^{14}$  or 16,384 Class B networks. The first octet of a Class B address offers 64 possibilities, 128 to 191. The second octet has 256 possibilities, 0 to 255. That yields 16,384 addresses, or 25 percent of the total IP space. Nevertheless, given the popularity and importance of the Internet, these addresses have run out quickly. This essentially leaves only Class C addresses available for new growth.

### Classes of IP Addresses: C, D, and E

This section covers Class C, D, and E IP addresses.

#### Class C Addresses

A Class C address begins with binary 110. Therefore, the lowest number that can be represented is 11000000, decimal 192. The highest number that can be represented is 11011111, decimal 223. If an IPv4 address contains a number in the range of 192 to 223 in the first octet, it is a Class C address.

Class C addresses were originally intended to support small networks. The first three octets of a Class C address represent the network number. The last octet may be used for hosts. One host octet yields 256 ( $2^8$ ) possibilities. After the all-0s network number and the all-1s broadcast address are subtracted, only 254 hosts may be addressed on a Class C network. Whereas Class A and Class B networks prove impossibly large without subnetting, Class C networks can impose an overly restrictive limit on hosts.

Because the first 3 bits of a Class C address are always 110, 21 bits are left in the network portion of the address, resulting in  $2^{21}$  or 2,097,152 Class C networks. With 2,097,152 total network addresses containing a mere 254 hosts each, Class C addresses account for 12.5 percent of the Internet address space. Because Class A and B addresses are nearly exhausted, the remaining Class C addresses are all that is left to be assigned to new organizations that need IP networks. Table 2-1 summarizes the ranges and availability of the three address classes used to address Internet hosts.

**Table 2-1 IP Addresses Available to Internet Hosts**

Address Class	First Octet Range	Number of Possible Networks	Number of Hosts Per Network
Class A	0 to 126	127 (2 are reserved)	16,777,214
Class B	128 to 191	16,384	65,534
Class C	192 to 223	2,097,152	254

#### Class D Addresses

A Class D address begins with binary 1110 in the first octet. Therefore, the first octet range for a Class D address is 11100000 to 11101111, or 224 to 239. Class D addresses are not used to address individual hosts. Instead, each Class D address can be used to represent a group of hosts called a host group, or multicast group.

For example, a router configured to run Enhanced Interior Gateway Routing Protocol (EIGRP) joins a group that includes other nodes that are also running EIGRP. Members of this group still have unique IP addresses from the Class A, B, or C range, but they also listen for messages addressed to 224.0.0.10. The 224 octet designates the address as a Class D address. Therefore, a single routing update message can be sent to 224.0.0.10, and all EIGRP routers will receive it. A single message sent to several select recipients is called a multicast. Class D addresses are also called multicast addresses.

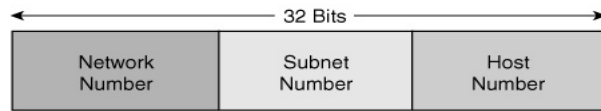
A multicast is different from a broadcast. Every device on a logical network must process a broadcast, whereas only devices configured to listen for a Class D address receive a multicast.

#### Class E Addresses

If the first octet of an IP address begins with 1111, the address is a Class E address. Therefore, the first octet range for Class E addresses is 11110000 to 11111111, or 240 to 255. Class E addresses are reserved for experimental purposes and should not be used to address hosts or multicast groups.

### Subnet Masking

Subnet masking, or subnetting, is used to break one large group into several smaller subnetworks, as shown in Figure 2-3. These subnets can then be distributed throughout an enterprise. This results in less IP address waste and better logical organization. Formalized with RFC 950 in 1985, subnetting introduced a third level of hierarchy to the IPv4 addressing structure. The number of bits available to the network, subnet, and host portions of a given address varies depending on the size of the subnet mask.



### IP Address Structure After Subnetting

A subnet mask is a 32-bit number that acts as a counterpart to the IP address. Each bit in the mask corresponds to its counterpart bit in the IP address. Logical ANDing is applied to the address and mask. If a bit in the IP address corresponds to a 1 bit in the subnet mask, the IP address bit represents a network number. If a bit in the IP address corresponds to a 0 bit in the subnet mask, the IP address bit represents a host number.

When the subnet mask is known, it overrides the address class to determine whether a bit is either a network or a host. This allows routers to recognize addresses differently than the format dictated by class. The mask can be used to tell hosts that although their addresses are Class B, the first three octets, instead of the first two, are the network number. In this case, the additional octet acts like part of the network number, but only inside the organization where the mask is configured.

The subnet mask applied to an address ultimately determines the network and host portions of an IP address. The network and host portions change when the subnet mask changes. If a 16-bit mask, 255.255.0.0, is applied to an IP address, only the first 16 bits, or two octets, of the IP address 172.24.100.45 represent the network number. Therefore, the network number for this host address is 172.24.0.0. The colored portion of the address shown in Figure 2-4 indicates the network number.

#### IP Address 172.24.100.45

<b>Dotted Decimal</b>	172	24	100	45
<b>Binary</b>	10101100	00011000	01100100	00101101

#### Subnet Mask 255.255.0.0

<b>Binary</b>	11111111	11111111	00000000	00000000
<b>Dotted Decimal</b>	255	255	0	0

### Class B Address Without Subnetting

Because the rules of class dictate that the first two octets of a Class B address are the network number, this 16-bit mask does not create subnets within the 172.24.0.0 network.

To create subnets with this Class B address, a mask must be used that identifies bits in the third or fourth octet as part of the network number.

If a 24-bit mask such as 255.255.255.0 is applied, the first 24 bits of the IP address are specified as the network number. The network number for the host in this example is 172.24.100.0. The gray portion of the address shown in Figure 2-5 indicates this.

Routers and hosts configured with this mask see all 8 bits in the third octet as part of the network number. These 8 bits are considered to be the subnet field because they represent network bits beyond the two octets prescribed by classful addressing.

Inside this network, devices configured with a 24-bit mask use the 8 bits of the third octet to determine to what subnet a host belongs. Because 8 bits remain in the host field, 254 hosts may populate each network. Just as hosts must have identical network addresses, they also must match subnet fields to communicate with each other directly. Otherwise, the services of a router must be used so that a host on one network or subnet can talk to a host on another.

<b>IP Address 172.24.100.45</b>			<b>Subnet Field</b>	
<b>Dotted Decimal</b>	172	24	100	45
<b>Binary</b>	10101100	00011000	01100100	00101101

<b>Subnet Mask 255.255.255.0</b>			<b>Subnet Field</b>	
<b>Binary</b>	11111111	11111111	11111111	00000000
<b>Dotted Decimal</b>	255	255	255	0

### Class B Address with Subnetting

A Class B network with an 8-bit subnet field creates  $2^8$ , or 256, potential subnets, each one equivalent to one Class C network. Because 8 bits remain in the host field, 254 hosts may populate each network. Two host addresses are reserved as the network number and broadcast address, respectively. By dividing a Class B network into smaller logical groups, the internetwork can be made more manageable, more efficient, and more scalable.

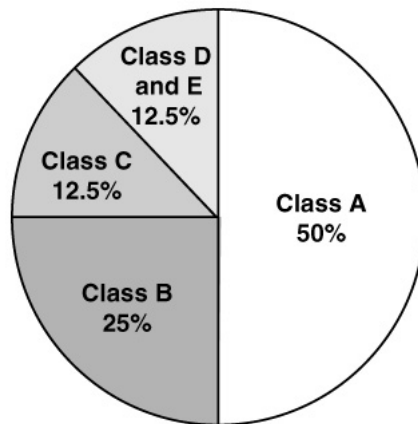
Notice that subnet masks are not sent as part of an IP packet header. This means that routers outside this network will not know what subnet mask is configured inside the network. An outside router, therefore, treats 172.24.100.45 as just one of 65,000 hosts that belong to the 172.24.0.0 network. In effect, subnetting classful IP addresses provides a logical structure that is hidden from the outside world.

## IP Addressing Crisis and Solutions

This section discusses some of the restraints involved in using IPv4 addressing. It also discusses some of the various methods and solutions that can be used to help get the most out of the depleted IPv4 address pool, such as CIDR, VLSM, route aggregation, and supernetting.

### IP Addressing Crisis

Class A and B addresses make up 75 percent of the IPv4 address space. However, a relative handful of organizations, fewer than 17,000, can be assigned a Class A or B network number. Class C network addresses are far more numerous than Class A and B addresses, although they account for only 12.5 percent of the possible 4 billion, or  $2^{32}$ , IP hosts, as illustrated in Figure 2-6.



IP Address Allocation

Unfortunately, Class C addresses are limited to 254 hosts, which will not meet the needs of larger organizations that cannot acquire a Class A or B address. Even if there were more Class A, B, and C addresses, too many network addresses would cause Internet routers to grind to a halt under the weight of enormous routing tables.

Ultimately, the classful system of IP addressing, even with subnetting, could not scale to effectively handle global demand for Internet connectivity. As early as 1992, the Internet Engineering Task Force (IETF) identified two specific concerns:

- ?? Exhaustion of the remaining, unassigned IPv4 network addresses. At the time, the Class B space was on the verge of depletion.

- ?? The rapid and substantial increase in the size of the Internet routing tables is because of the Internet's growth. As more Class C addresses came online, the resulting flood of new network information threatened the capability of Internet routers to cope effectively.

In the short term, the IETF decided that a retooled IPv4 would have to hold out long enough for engineers to design and deploy a completely new Internet Protocol. That new protocol, IPv6, solves the address crisis by using a 128-bit address space. After years of planning and development, IPv6 promises to be ready for wide-scale implementation. However, IPv6 continues, for the most part, to wait for that implementation.

One reason that IPv6 has not been rushed into service is that the short-term extensions to IPv4 have been so effective. By eliminating the rules of class, IPv4 now enjoys renewed viability.

### Classless Interdomain Routing

Routers use a form of IPv4 addressing called Classless Interdomain Routing (CIDR) that ignores class.

CIDR was introduced in 1993 by RFCs 1517, 1518, 1519, and 1520. It was deployed in 1994. CIDR dramatically improves the scalability and efficiency of IPv4 by providing the following:

- ?? Replacement of classful addressing with a more flexible and less wasteful classless scheme
- ?? Enhanced route aggregation, also known as supernetting or summarization
- ?? Supernetting, which is the combination of contiguous network addresses into a new address defined by the subnet mask

The following sections describe route aggregation, supernetting, and address allocation in more detail.

### Route Aggregation and Supernetting

CIDR allows routers to aggregate, or summarize, routing information. It does this by using a bit mask instead of an address class to determine the network portion of an address. This shrinks the size of the routing tables used by the router. In other words, just one address and mask combination can represent the routes to multiple networks.

Without CIDR and route aggregation, a router must maintain many individual entries for the routes within the same network, as opposed to one route for that particular network when using CIDR addressing.

The shaded entries in Table 2-2 identify the 16 bits that, based on the rules of class, represent the network number. Classful routers are forced to handle Class B networks using these 16 bits. Because the first 16 bits of each of these eight network numbers are unique, a classful router sees eight unique networks and must create a routing table entry for each. However, these eight networks do have common bits.

Table 2-2 Route Aggregation and Supernetting

Network Number	First Octet	Second Octet	Third Octet	Fourth Octet
10.0.0.0	10	0	0	0
10.0.0.0	10	0	0	0
10.0.0.0	10	0	0	0
10.0.0.0	10	0	0	0
10.0.0.0	10	0	0	0
10.0.0.0	10	0	0	0
10.0.0.0	10	0	0	0
10.0.0.0	10	0	0	0

172.24.0.0/16	10101100	00011000	00000000	00000000
172.25.0.0/16	10101100	00011001	00000000	00000000
172.26.0.0/16	10101100	00011010	00000000	00000000
172.27.0.0/16	10101100	00011011	00000000	00000000
172.28.0.0/16	10101100	00011100	00000000	00000000
172.29.0.0/16	10101100	00011101	00000000	00000000
172.30.0.0/16	10101100	00011110	00000000	00000000
172.31.0.0/16	10101100	00011111	00000000	00000000

Table 2-3 shows that the eight network addresses have the first 13 bits in common. A CIDR-compliant router can summarize routes to these eight networks by using a 13-bit prefix. Only these eight networks share these bits:

10101100 00011

**Table 2-3 Dotted-Decimal Notation**

Network Number	First Octet	Second Octet	Third Octet	Fourth Octet
172.24.0.0/16	10101100	00011000	00000000	00000000
172.25.0.0/16	10101100	00011001	00000000	00000000
172.26.0.0/16	10101100	00011010	00000000	00000000
172.27.0.0/16	10101100	00011011	00000000	00000000
172.28.0.0/16	10101100	00011100	00000000	00000000
172.29.0.0/16	10101100	00011101	00000000	00000000
172.30.0.0/16	10101100	00011110	00000000	00000000
172.31.0.0/16	10101100	00011111	00000000	00000000

To represent this prefix in decimal terms, the rest of the address is padded with 0s and then paired with a 13-bit subnet mask:

10101100 00011000 00000000 00000000 = 172.24.0.0  
 11111111 11111000 00000000 00000000 = 255.248.0.0

Therefore, a single address and mask define a classless prefix that summarizes routes to the eight networks, 172.24.0.0/13.

By using a prefix address to summarize routes, routing table entries can be kept more manageable. The following benefits are a result of the summarized routes:

- ?? More efficient routing
- ?? Reduced number of CPU cycles when recalculating a routing table or when sorting through the routing table entries to find a match
- ?? Reduced router memory requirements

Supernetting is the practice of using a bit mask to group multiple classful networks as a single network address. Supernetting and route aggregation are different names for the same process. However, the term supernetting is most often applied when the aggregated networks are under common administrative control. Supernetting takes bits from the network portion of the network mask, whereas subnetting takes bits from the host portion of the subnet mask. Supernetting and route aggregation are essentially the inverse of subnetting.

Recall that the Class A and Class B address space is almost exhausted, leaving large organizations little choice but to request multiple Class C network addresses from providers. If a company can acquire a block of contiguous Class C network addresses, supernetting can be used so that the addresses appear as a single large network, or supernet.

### Supernetting and Address Allocation

Consider Company XYZ, which requires addresses for 400 hosts. Under the classful addressing system, XYZ could apply to a central Internet address authority for a Class B address. If the company got the Class B address and then used it to address one logical group of 400 hosts, tens of thousands of addresses would be wasted. A second option for XYZ would be to request two Class C network numbers, yielding 508, or  $2 * 254$ , host addresses. The drawback of this approach is that XYZ would have to route between its own logical networks. Also, Internet routers would still need to maintain two routing table entries for the XYZ network, rather than just one.

Under a classless addressing system, supernetting allows XYZ to get the address space it needs without wasting addresses or increasing the size of routing tables unnecessarily. Using CIDR, XYZ asks for an address block from its Internet service provider (ISP), not a central authority, such as the Internet Assigned Numbers Authority (IANA). The ISP assesses XYZ's needs and allocates address space from its own large CIDR block of addresses. Providers assume the burden of managing address space in a classless system. With this system, Internet routers keep only one summary route, or supernet route, to the provider network. The provider keeps routes that are more specific to its customer networks. This method drastically reduces the size of Internet routing tables.

In the following example, XYZ receives two contiguous Class C addresses, 207.21.54.0 and 207.21.55.0. If you examine the shaded portions of Table 2-4, you will see that these network addresses have this common 23-bit prefix:

11001111 00010101 0011011

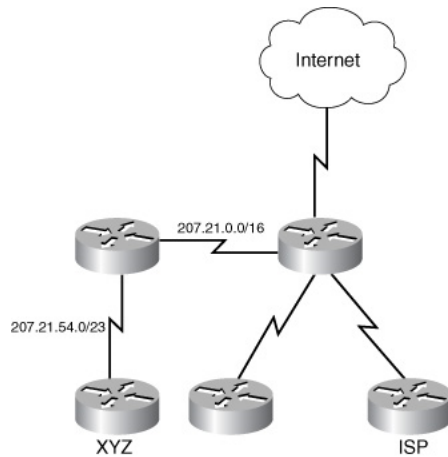
**Table 2-4 Supernetting and Address Allocation**

Network Number	First Octet	Second Octet	Third Octet	Fourth Octet
207.21.54.0	11001111	00010101	00110110	00000000
207.21.55.0	11001111	00010101	00110111	00000000

When the sample topology shown in Figure 2-7 is supernetted with a 23-bit mask, 207.21.54.0/23, the address space provides well over 400, or  $2^9$ , host addresses without the tremendous waste of a Class B address. With the ISP acting as the addressing authority for a CIDR block of addresses, the ISP's customer networks, which include XYZ, can be advertised among Internet routers as a single

supernet. The ISP manages a block of 256 Class C network addresses and advertises them to the world using a 16-bit prefix:

207.21.0.0/16



### Addressing with CIDR

When CIDR enabled ISPs to hierarchically distribute and manage blocks of contiguous addresses, IPv4 address space enjoyed the following benefits:

- ?? Efficient allocation of addresses
- ?? Reduced number of routing table entries

## VLSM

This section discusses VLSMs and how they can be used to further maximize IPv4 addressing efficiency.

### Variable-Length Subnet Masks

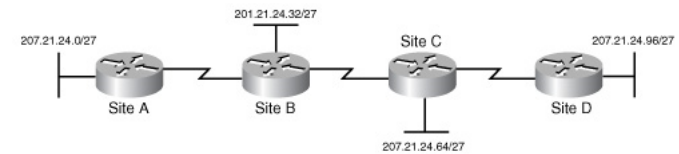
VLSM allows an organization to use more than one subnet mask within the same network address space. Implementing VLSM is often called subnetting a subnet. It can be used to maximize addressing efficiency.

Consider Table 2-5, in which the subnets are created by borrowing 3 bits from the host portion of the Class C address, 207.21.24.0.

**Table 2-5 Subnetting with One Mask**

Subnet Number	Subnet Address
Subnet 0	207.21.24.0/27
Subnet 1	207.21.24.32/27
Subnet 2	207.21.24.64/27
Subnet 3	207.21.24.96/27
Subnet 4	207.21.24.128/27
Subnet 5	207.21.24.160/27
Subnet 6	207.21.24.192/27
Subnet 7	207.21.24.224/27

If the **ip subnet-zero** command is used, this mask creates seven usable subnets of 30 hosts each. Four of these subnets can be used to address remote offices at Sites A, B, C, and D, as shown in Figure 2-8.



### Using Subnets to Address a WAN

Unfortunately, only three subnets are left for future growth, and three point-to-point WAN links between the four sites remain to be addressed. If the three remaining subnets were assigned to the WAN links, the supply of IP addresses would be completely exhausted. This addressing scheme would also waste more than a third of the available address space.

There are ways to avoid this kind of waste. Over the past 20 years, network engineers have developed three critical strategies for efficiently addressing point-to-point WAN links:

- ?? Use VLSM
- ?? Use private addressing (RFC 1918)
- ?? Use IP unnumbered

Private addresses and IP unnumbered are discussed in detail later in this chapter. This section focuses on VLSM. When VLSM is applied to an addressing problem, it breaks the address into groups or subnets of various sizes. Large subnets are created for addressing LANs, and very small subnets are created for WAN links and other special cases.

A 30-bit mask is used to create subnets with two valid host addresses. This is the exact number needed for a point-to-point connection. Figure 2-9 shows what happens if one of the three remaining subnets is subnetted again, using a 30-bit mask.

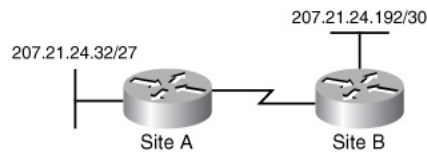
Subnet 0	207.21.24.0/27
Subnet 1	207.21.24.32/27
Subnet 2	207.21.24.64/27
Subnet 3	207.21.24.96/27
Subnet 4	207.21.24.128/27
Subnet 5	207.21.24.160/27
Subnet 6	207.21.24.192/27
Subnet 7	207.21.24.224/27

Sub-Subnet 0	207.21.24.192/30
Sub-Subnet 1	207.21.24.196/30
Sub-Subnet 2	207.21.24.200/30
Sub-Subnet 3	207.21.24.204/30
Sub-Subnet 4	207.21.24.208/30
Sub-Subnet 5	207.21.24.212/30
Sub-Subnet 6	207.21.24.216/30
Sub-Subnet 7	207.21.24.220/30

### Subnetting with VLSMs

Subnetting the 207.21.24.192/27 subnet in this way supplies another eight ranges of addresses to be used for point-to-point networks. For example, in Figure 2-10, the network 207.21.24.192/30 can be used to address the point-to-point serial link between the Site A router and the Site B router.



### Using VLSM to Address Point-to-Point Links

Example 2-1 shows the commands needed to configure the Site A router, labeled RTA, with a 27-bit mask on its Ethernet port and a 30-bit mask on its serial port.

### Example 2-1 Configuring VLSM

```
RTA(config)#interface e0
RTA(config-if)#ip address 207.21.24.33 255.255.255.224
RTA(config-if)#interface s0
RTA(config-if)#ip address 207.21.24.193 255.255.255.252
```

### Interactive Media Activity Drag and Drop: VLSM Calculation

After completing this activity, you will have a better understanding of VLSM.

In this lab, you will configure VLSM and test its functionality with two different routing protocols, RIPv1 and RIPv2. Finally, you will use IP unnumbered in place of VLSM to further conserve addresses.

### Classless and Classful Routing Protocols

For routers in a variably subnetted network to properly update each other, they must send masks in their routing updates. Without subnet information in the routing updates, routers would have nothing but the address class and their own subnet mask to go on. Only routing protocols that ignore the rules of address class and use classless prefixes work properly with VLSM. Table 2-6 lists common classful and classless routing protocols.

Table 2-6 Classful and Classless Routing Protocols

Classful Routing Protocols	Classless Routing Protocols
RIP Version 1	RIP Version 2
IGRP	EIGRP
EGP	OSPF
BGP3	IS-IS
	BGP4

**Routing Information Protocol version 1 (RIPv1)** and Interior Gateway Routing Protocol (IGRP), common interior gateway protocols, cannot support VLSM because they do not send subnet information in their updates. Upon receiving an update packet, these classful routing protocols use one of the following methods to determine an address's network prefix:

- ?? If the router receives information about a network, and if the receiving interface belongs to that same network, but on a different subnet, the router applies the subnet mask that is configured on the receiving interface.
- ?? If the router receives information about a network address that is not the same as the one configured on the receiving interface, it applies the default, subnet mask (by class).

Despite its limitations, RIP is a very popular routing protocol and is supported by virtually all IP routers. RIP's popularity stems from its simplicity and universal compatibility. However, the first version of RIP, RIPv1, suffers from several critical deficiencies:

- ?? RIPv1 does not send subnet mask information in its updates. Without subnet information, VLSM and CIDR cannot be supported.



- ?? RIPv1 broadcasts its updates, increasing network traffic.
- ?? RIPv1 does not support authentication.

In 1988, RFC 1058 prescribed the new and improved **Routing Information Protocol version 2 (RIPv2)** to address these deficiencies. RIPv2 has the following features:

- ?? RIPv2 sends subnet information and, therefore, supports VLSM and CIDR.
- ?? RIPv2 multicasts routing updates using the Class D address 224.0.0.9, providing better efficiency.
- ?? RIPv2 provides for authentication in its updates.

Because of these key features, RIPv2 should always be preferred over RIPv1, unless some legacy device on the network does not support it.

When RIP is first enabled on a Cisco router, the router listens for version 1 and 2 updates but sends only version 1. To take advantage of the RIPv2 features, turn off version 1 support, and enable version 2 updates with the following commands:

```
Router(config)#router rip
Router(config-router)#version 2
```

The straightforward RIP design ensures that it will continue to survive. A new version has already been designed to support future IPv6 networks.

## Route Summarization

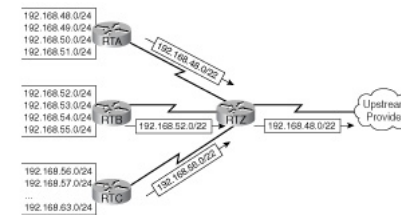
This section discusses some of the details of route summarization and how CIDR and VLSMs use it to deal with the size of routing tables.

### An Overview of Route Summarization

The use of CIDR and VLSM not only reduces address waste, but it also promotes route aggregation, or **route summarization**. Without route summarization, Internet backbone routing would likely have collapsed sometime before 1997.

Figure 2-11 shows how route summarization reduces the burden on upstream routers. This complex hierarchy of variable-sized networks and subnetworks is summarized at various points using a prefix address until the entire network is advertised as a single aggregate route of 192.168.48.0/20.

Recall that this kind of route summarization, or supernetting, is possible only if the network routers run a classless routing protocol, such as OSPF or EIGRP. Classless routing protocols carry the prefix length and subnet mask with the 32-bit address in routing updates. In Figure 2-11, the summary route that eventually reaches the provider contains a 20-bit prefix common to all the addresses in the organization. This prefix is 192.168.48.0/20, or 11000000.10101000.00110000.00000000, with a subnet mask of 11111111.11111111. 11110000.00000000. For summarization to work properly, addresses must be carefully assigned in a hierarchical fashion so that summarized addresses share the same high-order bits.

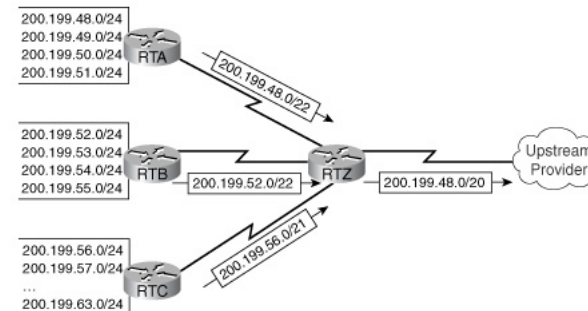


### Route Summarization

#### Route Flapping

Route flapping occurs when a router interface alternates rapidly between the up and down states. This can be caused by a number of factors, including a faulty interface or poorly terminated media.

Summarization can effectively insulate upstream routers from route-flapping problems. Consider RTC in Figure 2-12. If the RTC interface connected to the 200.199.56.0 network goes down, RTC removes that route from its table. If the routers were not configured to summarize, RTC would then send a triggered update to RTZ about the removal of the specific network, 200.199.56.0. In turn, RTZ would update the next router upstream, and so on. Every time these routers are updated with new information, their processors must go to work. It is possible, especially in the case of OSPF routing, that the processors can work hard enough to noticeably affect performance. Now, consider the impact on performance if the RTC interface to network 200.199.56.0 comes back up after only a few seconds. The routers update each other and recalculate. In addition, what happens when the RTC link goes back down seconds later? And then back up? This is called route flapping, and it can cripple a router with excessive updates and recalculations.



### Routes Summarized to 200.199.48.0/20

However, the summarization configuration prevents the RTC route flapping from affecting any other routers. RTC updates RTZ about a supernet, 200.199.56.0/21, which includes eight networks, 200.199.56.0 through 200.199.63.0. The loss of one network does not invalidate the route to the supernet. While RTC might be kept busy dealing with its own route flap, RTZ, all upstream routers are unaware of any downstream problem. Summarization effectively insulates the other routers from the problem of route flapping.

## Private Addressing and NAT

This section explains Network Address Translation (NAT) and how it can limit the waste of IP addresses by using the private addressing scheme.

### Private IP Addresses (RFC 1918)

Because TCP/IP is the dominant routed protocol in the world, most network applications and operating systems offer extensive support for it. Therefore, many designers build their networks around TCP/IP, even if they do not require Internet connectivity. Internet hosts require globally unique IP addresses. However, private hosts that are not connected to the Internet can use any valid address, as long as it is unique within the private network.

Because many private networks exist alongside public networks, just grabbing any address is strongly discouraged. RFC 1918 sets aside three blocks of IP addresses for private or internal use:

- ?? A Class A range
- ?? A Class B range
- ?? A Class C range

Addresses in one of these ranges, shown in Table 2-7, are not routed on the Internet backbone. Internet routers immediately discard private addresses.

**Table 2-7 Private Addresses in the WAN**

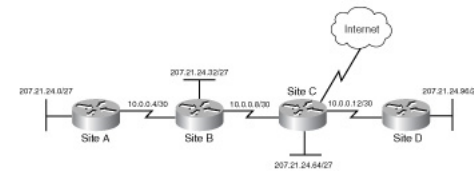
Class	RFC 1918 Internal Address Range	CIDR Prefix
A	10.0.0.0 to 10.255.255.255	10.0.0.0/8
B	172.16.0.0 to 172.31.255.255	172.16.0.0/12
C	192.168.0.0 to 192.168.255.255	192.168.0.0/16

If any of the following are being addressed, these private addresses can be used instead of globally unique addresses:

- ?? A nonpublic intranet
- ?? A test lab
- ?? A home network

Global addresses must be obtained from a provider or a registry at some expense.

RFC 1918 addresses have found a home in production networks as well. Earlier in this chapter, the advantages of using VLSM to address the point-to-point WAN links in an internetwork were discussed. Recall that with VLSM, you can further subnet one of the subnets left in the address space of a Class C network. Although this solution is better than wasting an entire 30-host subnet on each two-host WAN link, it still costs one subnet that could have been used for future growth. A less-wasteful solution is to address the WAN links using private network numbers. The WAN links shown in Figure 2-13 are addressed using subnets from the private address space, 10.0.0.0/8.



**Using Subnets to Address the WAN**

How can these routers use private addresses if LAN users at Sites A, B, C, and D expect to access the Internet? End users at these sites should have no problem, because they use globally unique addresses from the 207.21.24.0 network. The routers use their serial interfaces with private addresses merely to forward traffic and exchange routing information. Upstream providers and Internet routers see only the source and destination IP addresses in the packet. Upstream providers do not care if the packet traveled through links with private addresses at some point. In fact, many providers use RFC 1918 network numbers in the core of their network to avoid depleting their supply of globally unique addresses.

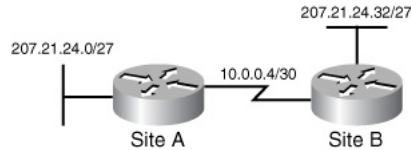
There is one trade-off when using private numbers on WAN links. The serial interfaces cannot be the original source of traffic bound for the Internet or the final destination of traffic from the Internet. Routers normally do not spend time surfing the web. Therefore, this limitation typically becomes an issue only when you're troubleshooting with Internet Control Message Protocol (ICMP), using Simple Network Management Protocol (SNMP), or connecting remotely with Telnet over the Internet. In those cases, the router can be addressed only by its globally unique LAN interfaces.

The following sections discuss implementing a private address scheme, including the pitfalls of discontinuous subnets and the advantages of NAT.

### Discontiguous Subnets

Mixing private addresses with globally unique addresses can create discontinuous subnets. Discontiguous subnets are subnets from the same major network that are separated by a completely different major network or subnet.

In Figure 2-14, Site A and Site B both have LANs that are addressed using subnets from the same major network, 207.21.24.0. They are discontinuous because the 10.0.0.4/30 network separates them. Classful routing protocols—notably, RIPv1 and IGRP—cannot support discontinuous subnets because the subnet mask is not included in routing updates. If Site A and Site B are running RIPv1, Site A receives updates about network 207.21.24.0/24 but not about 207.21.24.32/27. This is because the subnet mask is not included in the update. Because Site A has an interface directly connected to that network—in this case, e0—Site A rejects the Site B route.



**Discontiguous Subnets**

Even some classless routing protocols require additional configuration to solve the problem of discontinuous subnets. RIPv2 and EIGRP automatically summarize on classful boundaries, unless explicitly told not to. Usually, this type of summarization is desirable. However, in the case of discontinuous subnets, you must enter the following command for both RIPv2 and EIGRP to disable automatic summarization:

```
Router(config-router)#no auto-summary
```

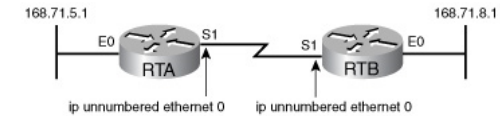
Finally, when using private addresses on a network that is connected to the Internet, packets and routing updates should be filtered. This is done to avoid leaking any RFC 1918 addresses between autonomous systems. If both the LAN and the provider use addresses from the 192.168.0.0/16 block, the routers could get confused if confronted with updates from both systems.

## Network Address Translation (NAT)

NAT, as defined by RFC 1631, is the process of swapping one address for another in the IP packet header. In practice, NAT is used to allow hosts that are privately addressed using RFC 1918 addresses to access the Internet.

A NAT-enabled device, such as a UNIX computer or a Cisco router, operates at the border of a stub domain. An example is an internetwork that has a single connection to the outside world. When a host inside the stub domain wants to transmit to a host on the outside, it forwards the packet to the NAT-enabled device. The NAT process then looks inside the IP header and, if appropriate, replaces the inside IP address with a globally unique IP address. When an outside host sends a response, as shown in Figure 2-15, the NAT does the following:

1. Receives it.
2. Checks the current table of network address translations.
3. Replaces the destination address with the original inside source.



**NAT Router**

NAT translations can occur dynamically or statically and can be used for a variety of purposes.

The most powerful feature of NAT routers is their capability to use Port Address Translation (PAT), which allows multiple inside addresses to map to the same global address. This is sometimes called a many-to-one NAT. With PAT, or address overloading, literally hundreds of privately addressed nodes can access the Internet using only one global address. The NAT router keeps track of the different conversations by mapping TCP and UDP port numbers.

## IP Unnumbered

This chapter has presented several ways to maximize the use of IP addresses in an organization. In previous sections, you learned that you can avoid wasting an entire subnet on point-to-point serial links by using VLSM, or use private addresses instead. Neither technique can be supported by classful routing protocols, such as the popular RIPv1 and IGRP. Fortunately, the Cisco IOS software offers a third option for efficiently addressing serial links—IP unnumbered.

### Using IP Unnumbered

When a serial interface is configured for IP unnumbered, it does not need its own address. This is because it borrows the IP address of another interface, usually a LAN interface or loopback interface. Example 2-2 shows how to configure an unnumbered interface. Not only does IP unnumbered avoid wasting addresses on point-to-point WAN links, but it also can be used with classful routing protocols, whereas VLSM and discontinuous subnets cannot. If the network runs RIPv1 or IGRP, IP unnumbered might be the only solution to maximize the addresses.

#### Example 2-2 Configuring an IP Unnumbered Interface

```
RTA(config)#interface e0
RTA(config-if)#ip address 168.71.5.1 255.255.255.0
RTA(config-if)#interface s1
RTA(config-if)#ip unnumbered e0
RTB(config)#interface e0
RTB(config-if)#ip address 168.71.8.1 255.255.255.0
RTB(config-if)#interface s1
RTB(config-if)#ip unnumbered e0
```

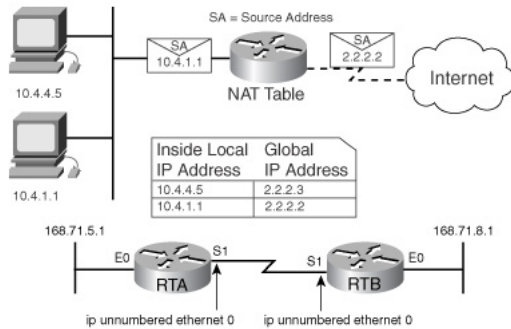
RTA e0, 168.71.5.1, and RTB e0, 168.71.8.1, can communicate using TCP/IP over this serial link, even though they do not belong to the same IP network, as shown in Figure 2-16. This is possible because the serial link is a point-to-point link, so there is no confusion about which device a packet is originating from or destined for. In this case, the command **ip unnumbered e0** is entered in serial

1 interface configuration mode on both RTA and RTB. Configuring IP unnumbered on an interface has two ground rules:

- ?? The interface is both serial and connected by way of a point-to-point link.
- ?? The same major network with the same mask is used to address the LAN interfaces that lend their IP address on both sides of the WAN link.

or

Different major networks with no subnetting are used to address the LAN interfaces on both sides of the WAN link.



### IP Unnumbered Interfaces

Certain drawbacks come with using IP unnumbered:

- ?? The use of ping cannot determine whether the interface is up, because the interface has no IP address.
- ?? A network IOS image cannot boot over an unnumbered serial interface.
- ?? IP security options cannot be supported on an unnumbered interface.

## DHCP and Easy IP

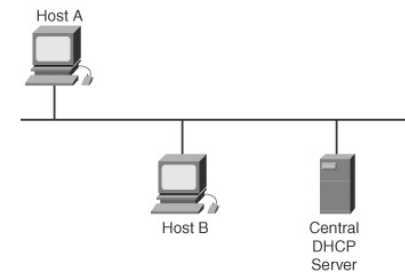
This section discusses DHCP and its operation. It covers how to configure DHCP, and use Easy IP as well.

### DHCP Overview

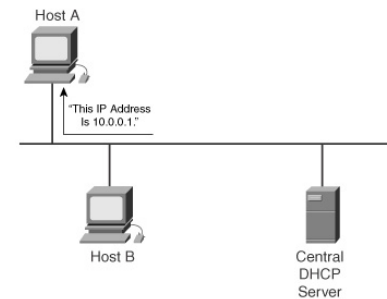
After designing a scalable IP addressing scheme for the enterprise, the next step is implementation. Routers, servers, and other key nodes usually require special attention from administrators. However, desktop clients are often automatically assigned IP configurations using Dynamic Host

Configuration Protocol (DHCP). Because desktop clients typically make up the bulk of network nodes, DHCP is good news for systems administrators. Small offices and home offices can also take advantage of DHCP by using Easy IP, a Cisco IOS software feature set that combines DHCP with NAT functions.

DHCP works by configuring servers to give out IP configuration information to clients. Clients lease the information from the server for an administratively defined period. When the lease is up, the host must ask for another address, although it is typically reassigned the same one. Figures 2-17 and 2-18 illustrate this process. In Figure 2-17, Host A issues a DHCP request for an IP address. In Figure 2-18, the DHCP server replies to the DHCP request by leasing an IP address from the configured IP address pool.



### Simple DHCP Operation: Client/Server



### Simple DHCP Operation: Reply

Administrators typically prefer to use a Microsoft 2000 server or a UNIX computer to offer DHCP services because these solutions are highly scalable and relatively easy to manage. Even so, the Cisco IOS software offers an optional, fully featured DHCP server, which leases configurations for 24 hours by default.

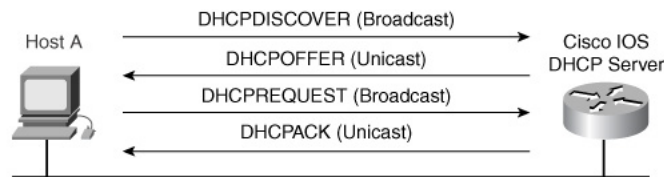
Administrators set up DHCP servers to assign addresses from predefined pools. DHCP servers can also provide other information:

- ?? Default gateway address
- ?? DNS server addresses
- ?? WINS server addresses
- ?? Domain names

Most DHCP servers also let you specifically define what client MAC addresses can be serviced and automatically assign the same number to a particular host each time.

## DHCP Operation

The DHCP client configuration process is shown in Figure 2-19.



**DHCP Operation**

The DHCP client configuration process follows these steps:

1. When a client is set up for DHCP and needs an IP configuration, typically at boot time, it tries to locate a DHCP server by sending a broadcast called a DHCPDISCOVER.
2. The server sends a DHCPOFFER unicast to the client. When the server receives the broadcast, it determines whether it can service the request from its own database. If it cannot, the server might forward the request to another DHCP server or servers, depending on its configuration. If it can service the request, the DHCP server offers the client IP configuration information in the form of a unicast DHCPOFFER. The DHCPOFFER is a proposed configuration that may include IP address, DNS server address, and lease time.
3. The client sends a DHCPREQUEST broadcast to all nodes. If the client finds the offer agreeable, it sends another broadcast. This broadcast is a DHCPREQUEST, specifically requesting those particular IP parameters. Why does the client broadcast the request instead of unicasting it to the server? A broadcast is used because the very first message, the

DHCPDISCOVER, might have reached more than one DHCP server. After all, it was a broadcast. If more than one server makes an offer, the broadcasted DHCPREQUEST lets the servers know which offer was accepted, which is usually the first offer received.

4. The server sends a DHCPACK unicast to the client. The server that receives the DHCPREQUEST makes the configuration official by sending a unicast acknowledgment, the DHCPACK. Note that it is possible but highly unlikely that the server will not send the DHCPACK, because it might have leased that information to another client in the interim. Receipt of the DHCPACK message lets the client begin using the assigned address immediately.

Depending on an organization's policies, it might be possible for an end user or administrator to statically assign a host an IP address that belongs in the DHCP server address pool. Just in case, the Cisco IOS software DHCP server always checks to make sure that an address is not in use before the server offers it to a client. The server issues ICMP echo requests (pings) to a pool address before sending the DHCPOFFER to a client. Although it can be configured, the default number of pings used to check for potential IP address conflict is two. The more pings, the longer the configuration process takes.

## Configuring the IOS DHCP Server

The DHCP server process is enabled by default on versions of the Cisco IOS software that support it. If for some reason the DHCP server process becomes disabled, you can reenable it by using the **service dhcp** global configuration command. The **no service dhcp** command disables the server.

Like NAT, DHCP servers require that the administrator define a pool of addresses. In Example 2-3, the **ip dhcp pool** command defines which addresses are assigned to hosts.

### Example 2-3 Configuring a DHCP Address Pool

```

RTA(config)#ip dhcp pool room12
RTA(dhcp-config)#network 172.16.1.0 255.255.255.0
RTA(dhcp-config)#exit
RTA(config-if)#ip dhcp excluded-address 172.16.1.1 172.16.1.10
  
```

The first command, **ip dhcp pool room12**, creates a pool named room12 and puts the router in a specialized DHCP configuration mode. In this mode, you use the **network** statement to define the range of addresses to be leased. If specific addresses are to be excluded on this network, return to global configuration mode and enter the **ip dhcp excluded-address** command.

The **ip dhcp excluded-address** command configures the router to exclude 172.16.1.1 through 172.16.1.10 when assigning addresses to clients. The **ip dhcp excluded-address** command may be used to reserve addresses that are statically assigned to key hosts.

A DHCP server can configure much more than an IP address. Other IP configuration values can be set from DHCP configuration mode, as shown in Example 2-4.

### Example 2-4 Assigning Key DHCP Information

```

RTA(config)#ip dhcp pool room12
RTA(dhcp-config)#dns-server 172.16.1.2
RTA(dhcp-config)#netbios-name-server 172.16.1.2
RTA(config-if)#default-router 172.16.1.1
  
```

IP clients will not get very far without a default gateway, which can be set by using the **default-router** command. The address of the DNS server, **dns-server**, and WINS server, **netbios-name-server**, can be configured here as well. The IOS DHCP server can configure clients with virtually any TCP/IP information.

Table 2-8 lists the key IOS DHCP server commands. These commands are entered in DHCP pool configuration mode, identified by the router(dhcp-config)# prompt.

**Table 2-8 Key DHCP Server Commands**

Command	Description
<b>network</b> <i>network-number</i> [ <i>mask</i>  / <i>prefix-length</i> ]	Specifies the subnet network number and mask of the DHCP address pool. The <i>prefix-length</i> portion specifies the number of bits that comprise the address prefix. The prefix is the alternative way of specifying the client's network mask. The <i>prefix-length</i> must be preceded by a slash (/).
<b>default-router</b> <i>address</i> [ <i>address2</i> ... <i>address8</i> ]	Specifies the IP address of the default router or default gateway for a DHCP client. One IP address is required, although up to eight addresses can be specified in one command line.
<b>dns-server</b> <i>address</i> [ <i>address2</i> ... <i>address8</i> ]	Specifies the IP address of a DNS server that is available to a DHCP client. One IP address is required, although up to eight addresses can be specified in one command line.
<b>netbios-name-server</b> <i>address</i> [ <i>address2</i> ... <i>address8</i> ]	Specifies the IP address of the NetBIOS WINS server that is available to a Microsoft DHCP client. One IP address is required, although up to eight addresses can be specified in one command line.
<b>domain-name</b> <i>domain</i>	Specifies the client's domain name.
<b>lease</b> [ <i>days</i> [ <i>hours</i> ] [ <i>minutes</i> ]   <i>infinite</i> ]	Specifies the duration of the DHCP lease. The default is a one-day lease.

Use the EXEC mode commands, shown in Table 2-9, to monitor DHCP server operation.

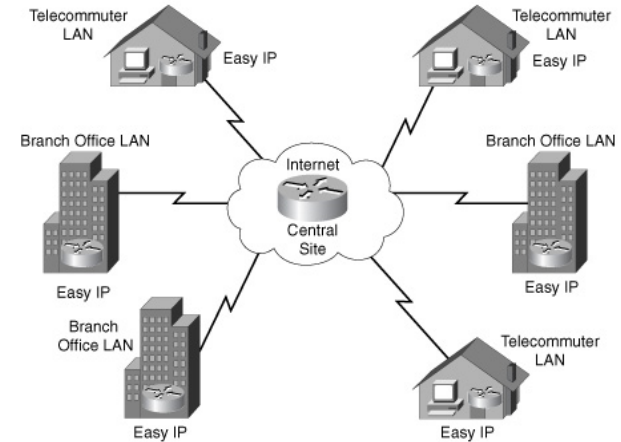
**Table 2-9 Key Commands for Monitoring DHCP Operation**

Command	Definition
<b>show ip dhcp binding</b> [ <i>address</i> ]	Displays a list of all bindings (MAC to IP address) created on a specific DHCP server.
<b>show ip dhcp conflict</b> [ <i>address</i> ]	Displays a list of all address conflicts recorded by a specific DHCP server.
<b>show ip dhcp database</b> [ <i>url</i> ]	Displays recent activity on the DHCP database. (Use this command in privileged EXEC mode.)
<b>show ip dhcp server statistics</b>	Displays count information about server statistics and messages sent and received.

## Easy IP

Easy IP is a combination suite of Cisco IOS software features that allows a router to negotiate its own IP address, as a DHCP client, and to do NAT through that negotiated address. Easy IP is typically deployed on a small office, home office (SOHO) router. It is useful in cases where a small

LAN connects to the Internet by way of a provider that dynamically assigns only one IP address for the entire remote site, as shown in Figure 2-20.



**Cisco IOS Easy IP**

A SOHO router with the Easy IP feature set uses DHCP, as a server, to automatically address local LAN clients with RFC 1918 addresses. When the router dynamically receives its WAN interface address by way of PPP, it uses NAT overload to translate between local inside addresses and its single global address. Therefore, both the LAN side and the WAN side are dynamically configured with little or no administrative intervention. In effect, Easy IP offers plug-and-play routing.

## Helper Addresses

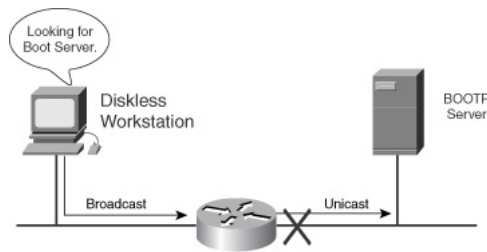
This section describes how networks and routers use helper addresses to forward broadcasts to another server or router on another network. This section describes some of the purposes of and scenarios in which to use helper addresses.

### Using Helper Addresses

DHCP is not the only critical service that uses broadcasts. Cisco routers and other devices might use broadcasts to locate TFTP servers. Some clients might need to broadcast to locate a TACACS security server. In a complex hierarchical network, clients might not reside on the same subnet as key servers. Such remote clients broadcast to locate these servers, but routers, by default, do not forward client broadcasts beyond their subnet. Some clients are unable to make a connection without services such as DHCP. For this reason, the administrator must provide DHCP and DNS servers on all subnets or use the Cisco IOS software helper address feature. Running services such as DHCP or DNS on several computers creates overhead and administrative problems, so the first

option is not very appealing. When possible, administrators use the **ip helper-address** command to relay broadcast requests for these key User Datagram Protocol (UDP) services.

By using the **ip helper-address** command, a router can be configured to accept a broadcast request for a UDP service and then forward it as a unicast to a specific IP address, as shown in Figure 2-21. Alternatively, the router can forward these requests as directed broadcasts to a specific network or subnetwork.



**Helper Addresses**

### Configuring IP Helper Addresses

To configure the helper address, identify the router interface that will receive the broadcasts for UDP services. In interface configuration mode, use the **ip helper-address** command to define the address to which UDP broadcasts for services should be forwarded.

By default, the **ip helper-address** command forwards the eight UDP services listed in Table 2-10.

**Table 2-10 Default Forward UDP Services**

Service	Port
Time	37
TACACS	49
DNS	53
BOOTP/DHCP Server	67
BOOTP/DHCP Client	68
TFTP	69
NetBIOS name service	137
NetBIOS datagram service	138

What if Company XYZ needs to forward requests for a service not on this list? The Cisco IOS software provides the global configuration command **ip forward-protocol** to allow an administrator to forward any UDP port in addition to the default eight. To forward UDP on port

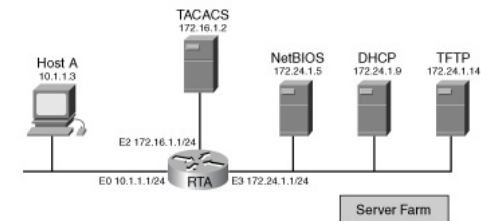
517, use the global configuration command **ip forward-protocol udp 517**. This command is used not only to add a UDP port to the default eight, but also to subtract an unwanted service from the default group. When forwarding DHCP, TFTP, and DNS without forwarding Time, TACACS, and NetBIOS, the Cisco IOS software requires that the router be configured according to the syntax shown in Example 2-5.

### Example 2-5 Forwarding UDP Services

```
RTA(config-if)#ip helper-address 192.168.1.254
RTA(config-if)#exit
RTA(config)#ip forward-protocol udp 517
RTA(config)#no ip forward-protocol udp 37
RTA(config)#no ip forward-protocol udp 49
RTA(config)#no ip forward-protocol udp 137
RTA(config)#no ip forward-protocol udp 138
```

### IP Helper Address Example

Consider the complex sample helper address configuration shown in Figure 2-22. You want Host A to automatically obtain its IP configuration from the DHCP server at 172.24.1.9. Because RTA will not forward the Host A DHCPDISCOVER broadcast, RTA must be configured to help Host A.



### IP Helper Address Example

To configure RTA e0, the interface that receives the Host A broadcasts, to relay DHCP broadcasts as a unicast to the DHCP server, use the following commands:

```
RTA(config)#interface e0
RTA(config-if)#ip helper-address 172.24.1.9
```

With this simple configuration, Host A broadcasts using any of the eight default UDP ports that are relayed to the DHCP server's IP address. However, what if Host A also needs to use the services of the NetBIOS server at 172.24.1.5? As configured, RTA forwards NetBIOS broadcasts from Host A to the DHCP server. Moreover, if Host A sends a broadcast TFTP packet, RTA also forwards this to the DHCP server at 172.24.1.9. What is needed in this example is a helper address configuration

that relays broadcasts to all servers on the segment. The following commands configure a directed broadcast to the IP subnet that is being used as a server farm:

```
RTA(config)#interface e0
RTA(config-if)#ip helper-address 172.24.1.255
```

Configuring a directed broadcast to the server segment, 172.24.1.255, is more efficient than entering the IP address of every server that could potentially respond to the Host A UDP broadcasts.

Finally, some devices on the Host A segment need to broadcast to the TACACS server, which does not reside in the server farm. Configure the RTA e0 to make it work by adding the command **ip helper-address 172.16.1.2**.

Verify the correct helper configuration with the **show ip interface** command, as shown in Example 2-6.

#### Example 2-6 Verifying IP Helper Address Configuration

```
RTA#show ip interface e0
Ethernet0 is up, line protocol is up
Internet address is 10.1.1.1/24
Broadcast address is 255.255.255.255
Address determined by setup command
MTU is 1500 bytes
Helper addresses are 172.24.1.255
                    172.16.1.2
Directed broadcast forwarding is disabled
<output omitted>
```

Notice that the RTA interface e3 in Example 2-7, which connects to the server farm, is not configured with helper addresses. However, the output shows that for this interface, directed broadcast forwarding is disabled. This means that the router does not convert the logical broadcast 172.24.1.255 into a physical broadcast with a Layer 2 address of FF-FF-FF-FF-FF-FF.

#### Example 2-7 Verifying Directed Broadcast Forwarding

```
RTA#show ip interface e3
Ethernet3 is up, line protocol is up
Internet address is 172.24.1.1/24
Broadcast address is 255.255.255.255
Address determined by setup command
MTU is 1500 bytes
Helper addresses is not set
Directed broadcast forwarding is disabled
<output omitted>
```

To allow all the nodes in the server farm to receive the broadcasts at Layer 2, configure e3 to forward directed broadcasts with the following commands:

```
RTA(config)#interface e3
RTA(config-if)#ip directed-broadcast
```

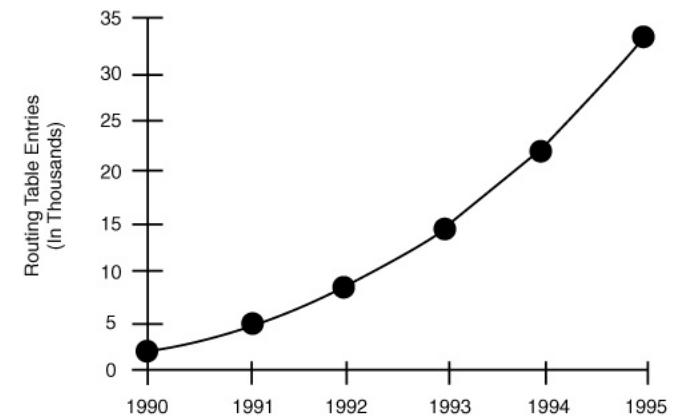
## IPv6

IPv6 is an alternative and a solution to the IPv4 address crisis. This section explains what IPv6 is and describes its address structure.

### IP Address Issues Solutions

This chapter has shown that IPv4 addressing faces two major issues:

- ?? The depletion of addresses, particularly the key medium-sized space
- ?? The pervasive growth of Internet routing tables, which is illustrated in Figure 2-23



Growth of Routing Tables

In the early 1990s, CIDR ingeniously built on the concept of the address mask and stepped forward to temporarily alleviate these overwhelming problems. CIDR's hierarchical nature dramatically improved IPv4's scalability. Once again, a hierarchical design has proven to be a scalable one.

Yet even with subnetting in 1985, variable-length subnetting in 1987, and CIDR in 1993, a hierarchical structure could not save IPv4 from one simple problem: not enough addresses exist to meet future needs. At roughly four billion possibilities, the IPv4 address space is formidable. However, it will not suffice in a future world of mobile Internet-enabled devices and IP-addressable household appliances.



Recent short-term IPv4 solutions to the address crunch have been developed. These include RFC 1918, which sets aside private addresses for unlimited internal use, and NAT, which allows thousands of hosts to access the Internet with only a handful of valid addresses.

However, the ultimate solution to the address shortage is the introduction of IPv6 and its 128-bit address. Developed to create a supply of addresses that would outlive demand, IPv6 is on course to eventually replace IPv4. IPv6's large address space will provide not only far more addresses than IPv4, but additional levels of hierarchy as well.

In 1994, the IETF proposed IPv6 in RFC 1752, and a number of working groups were formed in response. IPv6 covers issues such as the following:

- ?? Address depletion
- ?? Quality of service
- ?? Address autoconfiguration
- ?? Authentication
- ?? Security

It will not be easy for organizations deeply invested in the IPv4 scheme to migrate to a totally new architecture. As long as IPv4, with its recent extensions and CIDR-enabled hierarchy, remains viable, administrators will shy away from adopting IPv6. A new IP protocol requires new software, new hardware, and new methods of administration. It is likely that IPv4 and IPv6 will coexist, even within an autonomous system, for years.

### IPv6 Address Format

As defined in RFC 1884 and later revised in RFC 2373, IPv6 addresses are 128-bit identifiers for interfaces and sets of interfaces, not nodes. Three general types of addresses exist:

- ?? **Unicast**—An identifier for a single interface. A packet sent to a unicast address is delivered to the interface identified by that address.
- ?? **Anycast**—An identifier for a set of interfaces that typically belong to different nodes. A packet sent to an anycast address is delivered to the nearest interface in the anycast group.
- ?? **Multicast**—An identifier for a set of interfaces that typically belong to different nodes. A packet sent to a multicast address is delivered to all interfaces in the multicast group.

To write 128-bit addresses so that they are more readable to human eyes, the IPv6 architects abandoned dotted-decimal notation in favor of a hexadecimal format. Therefore, IPv6 is written as 32-hex digits, with colons separating the values of the eight 16-bit pieces of the address.

IPv6 addresses are written in hexadecimal:

**1080:0000:0000:0000:0008:0800:200C:417A**

Leading 0s in each 16-bit value can be omitted, so this address can be expressed as follows:

**1080:0:0:0:8:800:200C:417A**

Because IPv6 addresses, especially in the early implementation phase, might contain consecutive 16-bit values of 0, one such string of 0s per address can be omitted and replaced by a double colon. As a result, this address can be shortened as follows:

**1080::8:800:200C:417A**

Under current plans, IPv6 nodes that connect to the Internet will use what is called an aggregatable global unicast address. This is the familiar counterpart to the IPv4 global addresses. Like CIDR-enhanced IPv4, aggregatable global unicast addresses rely on hierarchy to keep Internet routing tables manageable. IPv6 global unicast addresses feature three levels of hierarchy:

- ?? **Public topology**—The collection of providers that offer Internet connectivity.
- ?? **Site topology**—The level local to an organization that does not provide connectivity to nodes outside itself.
- ?? **Interface identifier**—The level specific to a node's individual interface.

This three-level hierarchy is reflected by the structure of the aggregatable global unicast address (see Figure 2-24), which includes the following fields:

- ?? **Format Prefix (FP) field, 3 bits**—The 3-bit FP is used to identify the type of address—unicast, multicast, and so on. The bits 001 identify aggregatable global unicasts.
- ?? **Top-Level Aggregation Identifier (TLA ID) field, 13 bits**—The TLA ID field is used to identify the authority responsible for the address at the highest level of the routing hierarchy. Internet routers necessarily maintain routes to all TLA IDs. With 13 bits set aside, this field can represent up to 8192 TLAs.
- ?? **Reserved (Res) field, 8 bits**—IPv6 architecture defined the Res field so that the TLA or NLA IDs could be expanded as future growth warrants. Currently, this field must be set to 0.
- ?? **Next-Level Aggregation Identifier (NLA ID) field, 24 bits**—The NLA ID field is used by organizations assigned a TLA ID to create an addressing hierarchy and to identify sites.
- ?? **Site-Level Aggregation Identifier (SLA ID) field, 16 bits**—The SLA ID is used by an individual organization to create its own local addressing hierarchy and to identify subnets.
- ?? **Interface ID field, 64 bits**—The Interface ID field is used to identify individual interfaces on a link. This field is analogous to the host portion of an IPv4 address, but it is derived using the IEEE EUI-64 format. When this field is on LAN interfaces, the Interface ID adds a 16-bit field to the interface MAC address.

Number of Bits					
3	13	8	24	16	64
FP	TLA ID	Res	NLA ID	SLA ID	Interface ID
Public Topology				Site Topology	Interface Identifier

### IPv6 Address Format

In addition to the global unicast address space, IPv6 offers internal network numbers, or site local use addresses. These are analogous to RFC 1918 addresses. If a node is not normally addressed with a global unicast address or an internal site local use address, it can be addressed using a link local use address, which is specific to a network segment.

## Summary

This chapter described how all of the following can enable more efficient use of IP addresses:

- ?? Subnet masks
- ?? VLSMs
- ?? Private addressing
- ?? NAT

This chapter also showed that hierarchical addressing allows for efficient allocation of addresses and a reduced number of routing table entries. VLSMs, specifically, provide the capability to include more than one subnet mask within a network and the capability to subnet an already subnetted network address. Proper IP addressing is required to ensure the most efficient network operations. Finally, the IPv6 addressing format was presented.

## Key Terms

**anycast** @An identifier for a set of interfaces that typically belong to different nodes. A packet sent to an anycast address is delivered to the nearest, or first, interface in the anycast group.

**CIDR (classless interdomain routing)** @An IP addressing scheme that replaces the older system based on Classes A, B, and C. With CIDR, a single IP address can be used to designate many unique IP addresses.

**Format Prefix (FP) field** @The 3-bit FP identifies the type of address—unicast, multicast, and so on. The bits 001 identify aggregatable global unicast.

**interface identifier** @The level specific to a node's individual interface.

**Interface ID field** @The 64-bit Interface ID field identifies individual interfaces on a link. This field is analogous to the host portion of an IPv4 address, but it is derived using the IEEE EUI-64 format. When this field is on LAN interfaces, the Interface ID adds a 16-bit field to the interface MAC address.

**multicast** @An identifier for a set of interfaces that typically belong to different nodes. A packet sent to a multicast address is delivered to all interfaces in the multicast group.

**Next-Level Aggregation Identifier (NLA ID) field** @The 24-bit NLA ID field is used to identify ISPs. The field itself can be organized to reflect a hierarchy or a multitiered relationship among providers.

**public topology** @The collection of providers that offer Internet connectivity.

**Reserved (Res) field** @The IPv6 architecture defines the 8-bit Res field so that the TLA or NLA IDs can be expanded as future growth warrants. Currently, this field must be set to 0.

**RIP (Routing Information Protocol)** @An Interior Gateway Protocol (IGP) supplied with the FreeBSD version of UNIX. The most common IGP in the Internet. RIP uses hop count as a routing metric.

**RIPv2 (Routing Information Protocol version 2)** @Defined in RFC 1723 and supported in Cisco IOS software versions 11.1 and later. RIPv2 is not a new protocol; it is just RIPv1 with some extensions to bring it up-to-date with modern routing environments. RIPv2 has been updated to support VLSM, authentication, and multicast updates.

**route summarization** @The consolidation of advertised addresses in OSPF and IS-IS. In OSPF, this causes a single summary route to be advertised to other areas by an area border router.

**Site-Level Aggregation Identifier (SLA ID) field** @The 16-bit SLA ID is used by an individual organization to create its own local addressing hierarchy and to identify subnets.

**site topology** @The level local to an organization that does not provide connectivity to nodes outside itself.

**Top-Level Aggregation Identifier (TLA ID) field** @The 13-bit TLA ID field is used to identify the authority responsible for the address at the highest level of the routing hierarchy. Internet routers necessarily maintain routes to all TLA IDs. With 13 bits set aside, this field can represent up to 8192 TLAs.

**unicast** @An identifier for a single interface. A packet sent to a unicast address is delivered to the interface identified by that address.

**VLSM (variable-length subnet masking)** @The ability to specify a different subnet mask for the same network number on different subnets. VLSM can help optimize available address space.

## Check Your Understanding

Complete all of the review questions to test your understanding of the topics and concepts in this chapter. The answers appear in Appendix B, "Answers to the Check Your Understanding Questions."

For additional, more in-depth questions, refer to the chapter-specific study guides on the companion CD-ROM.

1. What feature lets a Cisco router act as a DHCP client?
  - A. NAT
  - B. DHCP
  - C. DNS
  - D. Easy IP
2. What is a summarization address for the networks 172.21.136.0/24 and 172.21.143.0/24?
  - A. 172.21.136.0/21
  - B. 172.21.136.0/20
  - C. 172.21.136.0/22
  - D. 172.21.128.0/21
3. The subnet 172.6.32.0/20 is again subnetted to 172.6.32.0/26. What is the result?
  - A. 1024 subnets
  - B. 64 hosts per subnet
  - C. 62 hosts per subnet
  - D. 2044 subnets
4. What routing protocol does not contain subnet mask information in its routing updates?
  - A. EIGRP

- B. OSPF
  - C. RIP
  - D. RIPv2
5. What method is used to represent a collection of IP network addresses with a single IP network address?
- A. Classful routing
  - B. Subnetting
  - C. Address translation
  - D. Route summarization
6. According to RFC 1918, which of the following is a private Internet address?
- A. 10.215.34.124
  - B. 192.32.146.23
  - C. 172.34.221.18
  - D. 119.12.73.215
7. Which of the following is a characteristic of IP unnumbered?
- A. Avoids wasted addresses on multiaccess links
  - B. Efficient addressing on Ethernet links
  - C. Uses another router interface address
  - D. May be used with classful and classless routing protocols
8. How many bits make up an IPv6 address?
- A. 32
  - B. 48
  - C. 64
  - D. 128
9. What is the purpose of IP helper addresses?
- A. To relay key UDP broadcast requests to hosts on the same subnet
  - B. To relay key UDP broadcast requests to hosts on the other subnets
  - C. To relay key TCP broadcast requests to hosts on other subnets
  - D. To relay key TCP broadcast requests to hosts on the same subnet
10. What are the three general types of IPv6 addresses?
- A. Class 1, Class 2, Class 3
  - B. Class A, Class B, Class C
  - C. Unicast, anycast, multicast
  - D. Public, site, interface