MICROSOFT TRAINING AND CERTIFICATION

Microsoft Official Curriculum

Module 8: Optimizing IP Address Allocation

Contents

Microsoft

Project Lead: Red Johnston
Instructional Designers: Meera Krishna (NIIT (USA) Inc.), Bhaskar Sengupta (NIIT (USA) Inc.)
Instructional Design Contributors: Aneetinder Chowdhry (NIIT (USA) Inc.),
Jay Johnson (The Write Stuff)**,** Sonia Pande (NIIT (USA) Inc.)
Lead Program Manager: Jim Cochran (Volt)
Program Manager: Jamie Mikami (Volt)
Technical Contributors: Rodney Miller, Gregory Weber (Volt)
Testing Leads: Sid Benavente, Keith Cotton
Testing Developer: Greg Stemp (S&T OnSite)
Simulation Developer: Wai Chan (Meridian Partners Ltd.)
Courseware Test Engineers: Jeff Clark, Jim Toland (ComputerPREP, Inc.)
Graphic Artist: Julie Stone (Independent Contractor)
Editing Manager: Lynette Skinner
Editor: Patricia Rytkonen (The Write Stuff)
Copy Editor: Kaarin Dolliver (S&T Consulting)
Online Program Manager: Debbi Conger
Online Publications Manager: Arlo Emerson (Aditi)
Online Support: Eric Brandt (S&T Consulting)
Multimedia Development: Kelly Renner (Entex)
Courseware Testing: Data Dimensions, Inc.
Production Support: Ed Casper (S&T Consulting)
Manufacturing Manager: Rick Terek (S&T OnSite)
Manufacturing Support: Laura King (S&T OnSite)
Lead Product Manager, Development Services: Bo Galford
Lead Product Manager: Gerry Lang
Group Product Manager: Robert Stewart
Simulations and interactive exercises were made with Macromedia Authorware

Instructor Notes

Presentation:
120 Minutes

Labs:
60 Minutes

This module provides students with the knowledge and skills required to assign IP addresses optimally. The module begins with a discussion of the limitations of classful IP addressing and introduces the benefits of Classless Inter-Domain Routing (CIDR). The next section describes the procedures to convert numbers from decimal to binary notation and vice versa. The students test the skills they learn in this section in the lab that follows it.

The third section of the module introduces the use of CIDR notation to represent IP addresses and explains the procedure for calculating a network ID in binary notation. This section is followed by a lab in which the students calculate the network ID when given an IP address and subnet mask and determine whether a destination computer is local or remote. The last section of the module provides an overview of the range of host IDs available and the concepts of supernetting and subnetting. In the lab that follows this section, the students determine the size of the network ID required based on the number of computers on the network.

At the end of this module, students will be able to:

- Describe the features of Classless Inter-Domain Routing (CIDR).
- Convert IP addresses from decimal format to binary format.
- Convert subnet masks to binary format to calculate the network ID and to determine if the destination computer is local or remote in relation to the source computer.
- Describe IP address allocation by using CIDR.

Materials and Preparation

This section provides you with the required materials and preparation tasks that are needed to teach this module.

Required Materials

To teach this module, you need the following materials:

- Microsoft® PowerPoint® file 2151A_08.ppt
- Module 8, "Optimizing IP Address Allocation"

Preparation Tasks

To prepare for this module, you should:

- Read all of the materials for this module.
- Read the white papers, *Introduction to TCP/IP* and *TCP/IP Implementation Details for Windows 2000*, on the Trainer Materials compact disc.
- Complete the three labs.
- Review the Delivery Tips and Key Points for each section and topic.
- Study the review questions and prepare alternative answers for discussion.
- Anticipate the questions that students may ask and prepare answers to them.

Module Strategy

Use the following strategy to present this module:

- Classless Inter-Domain Routing (CIDR)

  Discuss the limitations of classful IP addressing and explain the underlying philosophy of CIDR.

- Binary IP Addresses

  Explain the concept of binary notation and describe the procedure to convert decimal notation to binary notation and vice versa.

- Binary Subnet Masks

  Explain the advantage of using subnet mask bits. Define CIDR notation and describe the procedure to calculate the network ID in binary notation. Finally, explain how to determine whether a destination host is local or remote.

- IP Address Allocation Using CIDR

  Describe the procedure to calculate the number of hosts specified by an address block and explain the guidelines for binary IP addresses. Then discuss the concepts of supernetting and subnetting.

Customization Information

This section identifies the lab setup requirements for a module and the configuration changes that occur on student computers during the labs. This information is provided to assist you in replicating or customizing Microsoft Official Curriculum (MOC) courseware.

Important The labs in this module are also dependent on the classroom configuration that is specified in the Customization Information section at the end of the Classroom Setup Guide for course 2151A, *Microsoft Windows 2000 Network and Operating System Essentials*.

Lab Results

There are no configuration changes on student computers that affect replication or customization.

Overview

Slide Objective
To provide an overview of the module topics and objectives.

Lead-in
In this module, you will learn about the binary representation and the effective allocation of IP addresses.

- **Classless Inter-Domain Routing (CIDR)**
- **Binary IP Addresses**
- **Binary Subnet Masks**
- **IP Address Allocation Using CIDR**

The designers of the Internet did not foresee the popularity it enjoys today. Unaware of the long-term consequences of their actions, they assigned large numbers of IP addresses without considering the effect on IP address availability. As the Internet increased in size, the number of available IP addresses declined at a rapid rate.

This shortage of available IP addresses was a result of using IP address classes to organize IP addresses. Classful IP addressing proved inefficient because it allows for only three fixed sizes for networks on the Internet—one size for each of the address classes: A, B, and C. These sizes are a result of the natural division of an IP address in decimal notation.

The depleted supply of IP addresses led to the creation of a new addressing system called Classless Inter-Domain Routing (CIDR). CIDR represents IP addresses and subnet masks in binary notation to divide the traditional fixed network sizes. This makes CIDR a more efficient choice for the allocation of IP addresses than does the classful method.

At the end of this module, you will be able to:

- Describe the features of Classless Inter-Domain Routing.
- Convert IP addresses from decimal format to binary format.
- Convert subnet masks to binary format and calculate the network ID to determine local and remote hosts.
- Describe IP address allocation by using CIDR.

◆ Classless Inter-Domain Routing (CIDR)

Slide Objective
To introduce the concept of Classless Inter-Domain Routing (CIDR).

Lead-in
The classful method of categorizing IP addresses presents certain problems.

- **Limitations of Classful IP Addressing**
- **Defining CIDR**

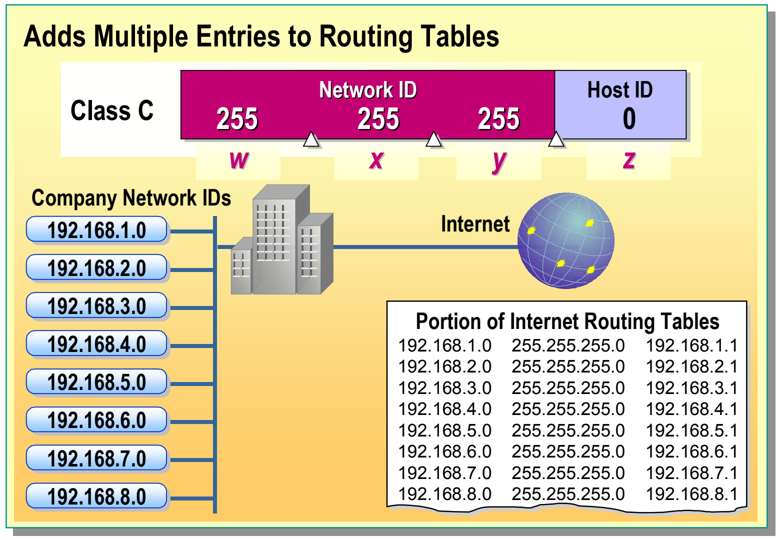
IP address classes provide a simple method for differentiating local hosts from remote hosts and for locating the route to a remote host. However, this method permits very few variations in network sizes, which has led to such problems as inappropriately assigning IP addresses to networks. To overcome these limitations, a method known as Classless Inter-Domain Routing (CIDR) was developed for breaking up networks into a larger variety of sizes.

In this section, you will learn about the limitations of classful IP addressing and the advantages of using CIDR.

Limitations of Classful IP Addressing

Slide Objective
To illustrate the limitations of classful IP addressing.

Lead-in
As their usage grew, the IP address classes could not accommodate the demand for medium-sized networks.



Delivery Tip
Use the animated slides to demonstrate the unused IP addresses of a class B network. Then show how a solution to the problem of unused addresses produced another problem: overloaded routers on the Internet. The third limitation of the classful addressing system is partially addressed by simply not wasting IP addresses. An additional solution (using NAT or a proxy server) to the problem of the third limitation is addressed in module 9, "Examining Web Services," in course 2151A, *Microsoft Windows 2000 Network and Operating System Essentials*. These solutions are summarized in the final section.

As a result of its dramatic growth, the Internet began to experience problems with scalability. These problems resulted from using address classes to allocate IP addresses to networks that needed to connect to the Internet.

Classful IP addressing involved three major limitations:

- The number of available addresses in the class B address space was near depletion.
- The Internet routing tables were nearly full.
- All available IP addresses would be eventually assigned.

Depletion of Class B Addresses

The disparate network sizes offered by the classful method caused the depletion of class B addresses. In this system, an organization with a medium-sized network of 2,000 computers belongs to the class B category and is assigned 65,534 IP addresses although it may require only 2,000. Therefore, 63,534 IP addresses are not used because of this allocation.

Filling Up of Internet Routing Tables

To overcome the problem of unused IP addresses, an organization with a medium-sized network of 2,000 computers can divide its network into eight smaller class C networks with 254 computers each. This solution results in the generation of eight routes, or paths, to the organization's eight smaller networks. Consequently, each router on the Internet needs to maintain eight routes to forward a packet to this single organization, thus increasing the amount of information in the Internet's routing tables.

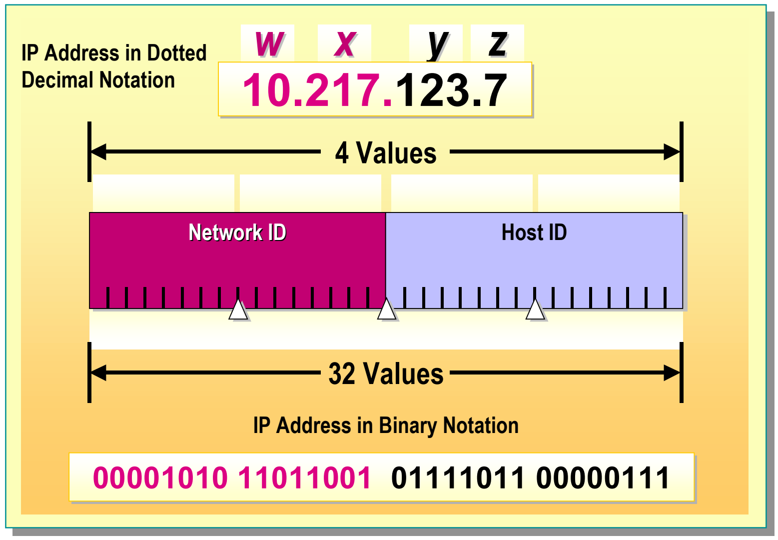
Depletion of All IP Addresses

Because of the wastefulness of the classful method and the finite number of IP addresses available, the entire pool of IP addresses would be depleted if the classful method were still in use.

Defining CIDR

Slide Objective
To illustrate the binary representation of an IP address.

Lead-in
In response to the need to optimize the available IP address space, the CIDR method was developed.



Delivery Tip
Use the slide to show how the binary representation of an IP address has 32 separate values. Compare the 32 separate values available for host and network IDs with the four values visible in the decimal format.

Classless Inter-Domain Routing (CIDR) uses binary notation, whereas the classful method used decimal notation.

Use of Binary Notation

Computers use binary notation for their internal processing because they communicate using signals that have only two states: on or off. Because the binary system also has only two values: 0 or 1, computers operate using binary notation.

Increased Choice of Network Sizes

CIDR translates all IP address and subnet masks to binary notation. CIDR divides an IP address into a set of 32 values, in place of the four values used in the classful system. This division allows for more variations in network size and optimizes the allocation of IP addresses. By using CIDR, not many IP addresses go unused because it is now possible for companies to obtain IP addresses in numbers that are much closer to what they require.

CIDR does not define a default subnet mask based on the IP address. Instead, each host is configured with a custom subnet mask, and each router is sent the IP address as part of the data packet. The router then uses a subnet mask from its routing table to determine the network ID of the computer to which the packet must be forwarded.

◆ Binary IP Addresses

Slide Objective
To introduce the concept of binary notation.

Lead-in
This section describes the basics of the binary notation system.

- **Converting to Binary Format**
- **Converting to Binary Format Using a Calculator**

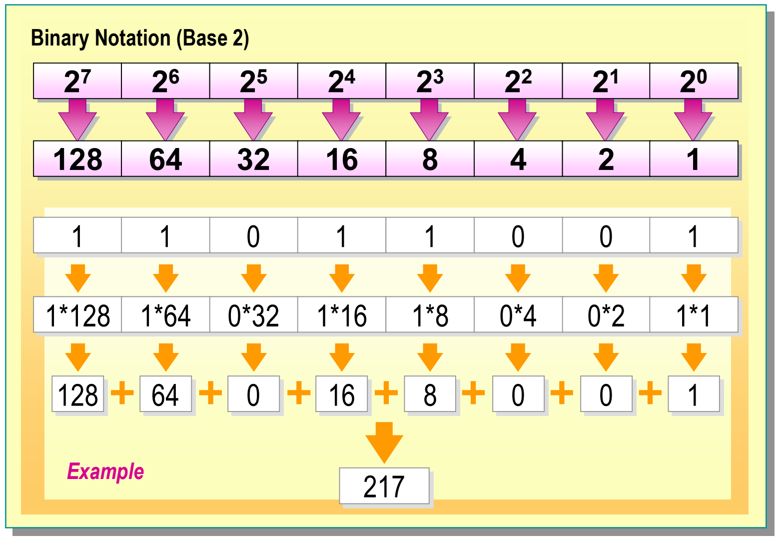
To avoid the limitations of classful IP addressing, CIDR presents IP addressing in binary notation. Binary notation consists of two values: 0 and 1, whereas decimal notation has 10 values: 0 to 9. In decimal notation, a number is expressed in terms of powers of 10, in binary notation in terms of powers of 2.

This section covers the procedure used to convert binary IP addresses to decimal notation and vice versa, both manually and by using the calculator in Microsoft® Windows® 2000.

Converting to Binary Format

Slide Objective
To illustrate how to work with the binary system.

Lead-in
Computation in the binary system is similar to that in the decimal system.



Delivery Tip
The students need to know the basis of the binary system. So explain the mathematics behind binary notation before teaching them how to use a calculator to convert between decimal and binary. Depending on the specific class, you will focus on either converting manually or on using the calculator. Use the slide to show the values of each position of a binary number. Point out to the students that an IP address consists of four distinct numbers, each of which must be converted, not just one large binary number. Then use the second and third slides as examples of how both the decimal and binary number systems work.

In the binary system, an IP address is represented as a string of 32 numbers. This string can be divided into four fields; each field is called an octet, or byte. Each octet consists of eight bits. A bit has a value of either 0 or 1. Therefore, an IP address consists of 4 bytes, for a total of 32 bits.

An example of an octet in binary notation is 11011001, and an example of an IP address in binary notation is 00001010 11011001 01111011 00000111. The decimal representations of this octet and IP address are 217 and 10.217.123.7, respectively.

Decimal Notation

In decimal, or base 10 representation, you calculate the value of a number as follows: Starting with the rightmost digit, multiply each digit by increasing powers of 10, beginning with 10^0. Then add these values to obtain the number. For example, the computation for the number 217 is:

$7*10^0 = 7*1 = 7$

$1*10^1 = 1*10 = 10$

$2*10^2 = 2*100 = 200$

$200+10+7 = 217$

Binary Notation

You use a similar procedure to calculate the decimal value of a binary representation. In this case, base 10 is replaced by base 2. Multiply each number in the representation by powers of 2 in place of powers of 10. For example, for the number 11011001, starting with the rightmost digit:

$1*2^0 = 1*1 = 1$

$0*2^1 = 0*2 = 0$

$0*2^2 = 0*4 = 0$

$1*2^3 = 1*8 = 8$

$1*2^4 = 1*16 = 16$

$0*2^5 = 0*32 = 0$

$1*2^6 = 1*64 = 64$

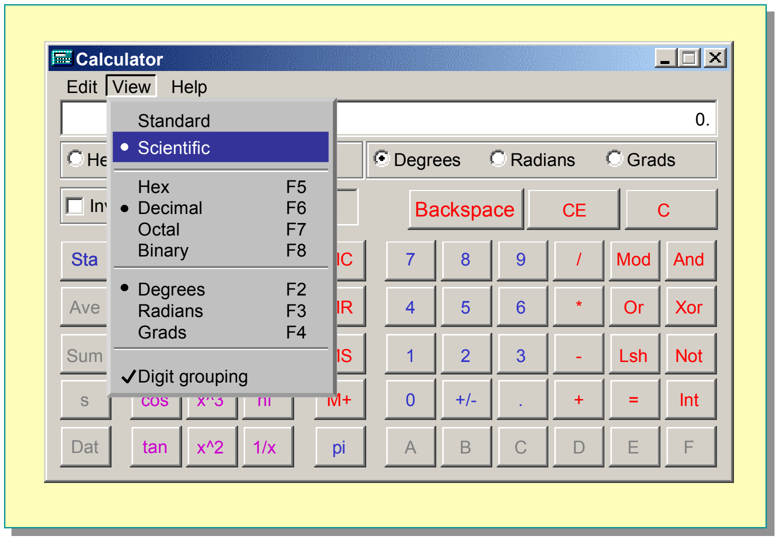
$1*2^7 = 1*128 = 128$

128+64+0+16+8+0+0+1 = 217

Converting to Binary Format Using a Calculator

Slide Objective
To display the scientific mode of the Windows Calculator.

Lead-in
Windows 2000 provides a calculator that simplifies the conversion of decimal numbers to binary and vice versa.



Delivery Tip
Demonstrate how to convert from decimal format to binary format and back by using Windows Calculator in scientific mode. Remind students that they do not need to memorize the binary system because they can use the calculator to convert the numbers.

Multiplying each number in the binary representation by powers of 2 is a tedious method for obtaining its decimal equivalent. The process is simplified if you use the calculator provided by the Windows 2000 operating system.

By default, the calculator appears in standard mode. To perform calculations that include binary values, you must switch to scientific mode.

To switch to scientific mode

- On the **View** menu, click **Scientific**.

The calculator expands to display many choices for performing mathematical functions.

Using Windows Calculator to convert from decimal to binary format

1. With the calculator set to scientific mode, click **Dec**.
2. Type a decimal value, for example **29** and then click **Bin**.

   The binary value, 11101, is displayed.

Using Windows Calculator to convert from binary to decimal format

1. With the calculator set to scientific mode, click **Bin**.
2. Type a binary value, for example **10010** and then click **Dec**.

   The decimal value, 18, is displayed.

Note Windows Calculator does not accept the leftmost zeros (0s) in the binary notation of a number. Enter the binary values beginning with the first 1 in the notation. Therefore, in the above example, in place of 00010010, you type only 10010. Similarly, the calculator does not display the leftmost zeros (0s) in an octet that begins with 0. To obtain the 8-bit octet, you must prefix the displayed number with the required number of 0s. For example, the octet value of 11101 is 00011101.

Lab A: Using Windows Calculator to Convert Decimal and Binary Numbers

Slide Objective
To introduce the lab.

Lead-in
In this lab, you will use the Windows Calculator in scientific mode to convert numbers from decimal to binary and back to decimal.

Objectives

After completing this lab, you will be able to:

- Convert decimal numbers to binary format.
- Convert binary numbers to decimal format.

Prerequisites

Before working on this lab, you must have:

- Knowledge of how to log on to Windows 2000.

Estimated time to complete this lab: 15 minutes

Exercise 1
Converting Decimal Numbers to Binary

Scenario

You are an administrator and must convert your computers' IP addresses to binary formats before you can efficiently allocate their addresses.

Goal

In this exercise, you will convert multiple IP addresses to binary format by first converting their individual decimal numbers.

Tasks	Detailed Steps
1. Log on as Administrator with a password of **password**, and then open Windows Calculator in scientific mode.	**a.** Log on to Windows 2000 as Administrator with a password of **password**. **b.** Click **Start**, point to **Programs**, point to **Accessories**, and then click **Calculator**. **c.** On the **View** menu, click **Scientific**. *Notice that the calculator expands and four option buttons appear to allow you to convert decimal numbers to binary and back again.* **d.** Click **Dec** to ensure that the calculator is in decimal mode.
2. Convert the IP address 131.107.2.200 to binary.	**a.** In the calculator window, type **131** and then click **Bin**.

What is the binary result of 131?

10000011

How many digits are in the binary number for 131?

8

Tasks	Detailed Steps
2. *(continued)*	**b.** Click **Dec**, type **107** and then click **Bin**.

(*continued*)

Tasks	Detailed Steps
❓ What is the binary result of 107? **1101011**	
❓ How many digits are in the binary number for 107? **7**	
2. *(continued)*	**c.** Click **Dec**, type **2** and then click **Bin**.
❓ What is the binary result of 2? **10**	
❓ How many digits are in the binary number for 2? **2**	
2. *(continued)*	**d.** Click **Dec**, type **200** and then click **Bin**.
❓ What is the binary result of 200? **11001000**	

(*continued*)

Tasks	Detailed Steps
	How many digits are in the binary number for 200? **8**
	All octets in a binary IP address have eight digits. For any decimal numbers that are converted to binary that have fewer than eight digits, leading zeros precede the binary numbers. What will be the binary numbers for 107 and 2 with the addition of leading zeros? **For 107, the binary number 1101011 becomes 01101011. For 2, the binary number 10 becomes 00000010.**
	An IP address is represented by four octets separated by decimal points. What is the full binary version of the 131.107.2.200 IP address? **10000011.01101011.00000010.11001000**

(*continued*)

Tasks	Detailed Steps
3. Convert the IP addresses in the following table to binary. The first one is provided as an example.	**a.** Convert the first decimal number to binary. **b.** Convert the second decimal number to binary. **c.** Convert the third decimal number to binary. **d.** Convert the fourth decimal number to binary. **e.** Add leading zeros to all binary results to complete the octet. **f.** Separate the four binary octets with decimal points. **g.** Repeat until the table on the next page is completed.

Decimal IP address	Binary portions	Binary IP address
122.131.25.64	**1111010, 10000011, 11001, 1000000**	**01111010. 10000011.00011001.01000000**
215.34.211.9		
97.49.153.122		
64.144.25.100		
176.34.68.78		
42.89.215.61		
71.73.65.166		
47.245.235.84		
156.213.67.23		
124.87.235.87		
7.23.87.2		

Exercise 2
Converting Binary Numbers to Decimal

Scenario

You are an administrator and must convert your computer's IP addresses from binary formats to decimal formats so that they can be configured in Windows 2000.

Goal

In this exercise, you will convert multiple IP addresses from binary to decimal format by first converting the individual octets to decimal format.

Tasks	Detailed Steps
1. Convert the IP address 00110101.11100101.11101010. 00001011 to decimal format.	**a.** Using Windows Calculator, make sure you are in scientific mode. **b.** Click **Bin**, type **110101** and then click **Dec**.
❓ What is the decimal result of the binary value 00110101? **53**	
1. *(continued)*	**c.** Click **Bin**, type **11100101** and then click **Dec**.
❓ What is the decimal result of the binary value 11100101? **229**	
1. *(continued)*	**d.** Click **Bin**, type **11101010** and then click **Dec**.

(*continued*)

Tasks	Detailed Steps
❓ What is the decimal result of the binary value 11101010? **234**	
1. *(continued)*	**e.** Click **Bin**, type **1011** and then click **Dec**.
❓ What is the decimal result of the binary value 00001011? **11**	
❓ What is the decimal result of the binary IP address 00110101.11100101.11101010.00001011? **53.229.234.11**	
2. Convert the IP addresses in the following table to decimal format. The first one is provided as an example.	**a.** Convert the binary numbers to decimals, one at a time. **b.** Separate the four decimal numbers with a decimal point. **c.** Repeat until the table on the next page is completed.

(*continued*)

Binary IP address	Decimal portions	Decimal IP address
01110110.00011010.10101111.01011101	**118, 26, 175, 93**	**118.26.175.93**
10101001.01010101.10101010.11011000		
00011011.11011000.10110101.01010111		
01111111.11100000.00000101.00101011		
11000100.10101100.01100001.11101111		
01110111.00111100.10111000.10101001		
10100011.11101101.10100010.10101110		
01010101.01100100.11110111.10101000		
00111100.00111010.10101000.10101111		
01010111.10111100.111011101.1010101		

Tasks	Detailed Steps
3. Close all windows and log off from Windows 2000.	**a.** Close all windows and log off from Windows 2000.

◆ Binary Subnet Masks

Slide Objective
To introduce the concept of binary subnet masks.

Lead-in
After converting the IP address, the subnet mask and network ID need to be converted to binary notation as well.

- Subnet Mask Bits
- CIDR Notation
- Calculating the Network ID
- Determining Local and Remote Hosts

To be compatible with IP addresses in binary notation, subnet mask bits also must be converted to binary notation. Following the conversion of both IP addresses and subnet masks to binary notation, an IP address can be represented in CIDR notation. CIDR notation is an addressing method that associates a subnet mask with the IP address.

Regardless of whether the IP address and subnet mask are specified in CIDR notation or in decimal notation, you can calculate the network ID of the IP address by converting all values to binary notation. After you calculate the network ID, you can determine whether a destination host is local or remote.

Subnet Mask Bits

Slide Objective
To illustrate the binary values of subnet masks expressed in decimal notation.

Lead-in
Subnet masks can be converted to binary notation by using the same procedure as for IP addresses.

Binary Representation	Decimal Representation
11111111	255
11111110	254
11111100	252
11111000	248
11110000	240
11100000	224
11000000	192
10000000	128
00000000	0

Delivery Tip
Use the animated slide to show the students an example of a subnet mask in binary format. Then show all of the possible decimal and binary values of a subnet mask. Emphasize the concept of contiguous maximums, in this case binary 1s, verses the maximum of 255 when viewed in decimal.

In binary notation, a subnet mask is represented by four octets, just as the IP address is. The following table shows the binary notation for the default subnet masks used in the classful method.

Decimal representation	Binary representation
255.0.0.0	11111111 00000000 00000000 00000000
255.255.0.0	11111111 11111111 00000000 00000000
255.255.255.0	11111111 11111111 11111111 00000000

Using the binary representation of a subnet mask, you can manipulate 32 numbers in place of the four numbers in the decimal representation. This increased capability provides a greater selection of network sizes compared with the three fixed sizes in the classful method.

Binary Versus Decimal Subnet Masks

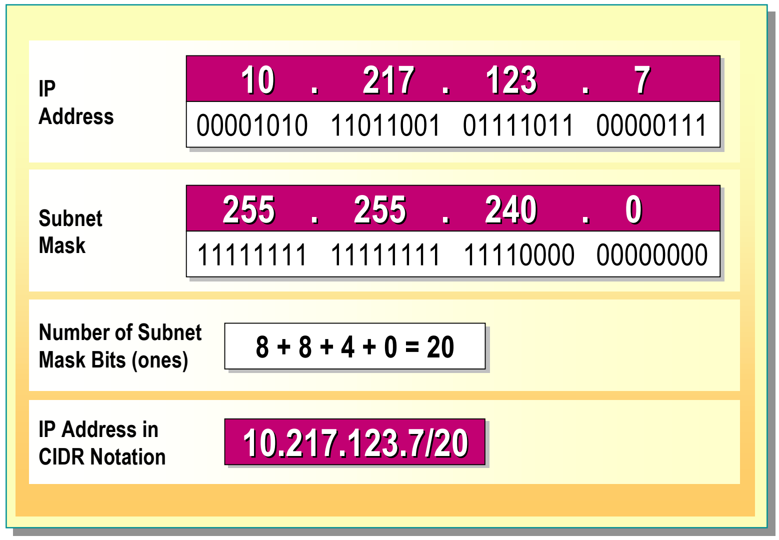
Subnet masks are always composed of contiguous maximum values followed by contiguous minimum values. In binary notation, this translates to a series of contiguous 1s followed by contiguous 0s. The contiguous 1s determine the network ID and the contiguous 0s determine the host ID. Because a binary subnet mask consists of a series of contiguous 1s followed by contiguous 0s, each octet in the subnet mask represents only a limited number of decimal values, as illustrated in the following table.

Binary representation	Decimal representation
11111111	255
11111110	254
…	…
10000000	128
00000000	0

CIDR Notation

Slide Objective
To illustrate CIDR notation.

Lead-in
CIDR notation provides a simple method for identifying the subnet mask associated with an IP address.



Delivery Tip
IP addresses are divided into groups represented by a network ID. By using CIDR, the number of groups available is greater than in the classful method. The chart on the following page illustrates the relationship between classful and CIDR addresses. Remind students again that the popularity of communicating on the Internet is the reason for this increase in the number of groups.

CIDR notation involves specifying a dotted decimal notation with a bit mask. The bit mask specifies the number of contiguous 1s in the binary notation of the subnet mask associated with the IP address. The contiguous 1s are the leftmost bits in the subnet mask.

For example, the IP address represented in CIDR notation as 10.217.123.7/20 specifies that its subnet mask has 20 contiguous 1s. Consequently, the 12 bits remaining from the original 32 must be 0s.

The IP addresses in CIDR notation are known by the number of bits from the IP address that make up the network ID and are represented as /*x*. For example, a 10-bit network ID is represented as /10.

Note CIDR notation is also known as network prefix notation.

CIDR and IP Address Classes

In CIDR notation, the IP address represented with the /20 bit mask could be a former class A, class B, or class C IP address. The routers that support CIDR do not use the first three bits of the address to determine whether the destination host on the packet is local or remote, as they did in the classful method. Instead, they rely on the bit mask information provided with the route to make a determination.

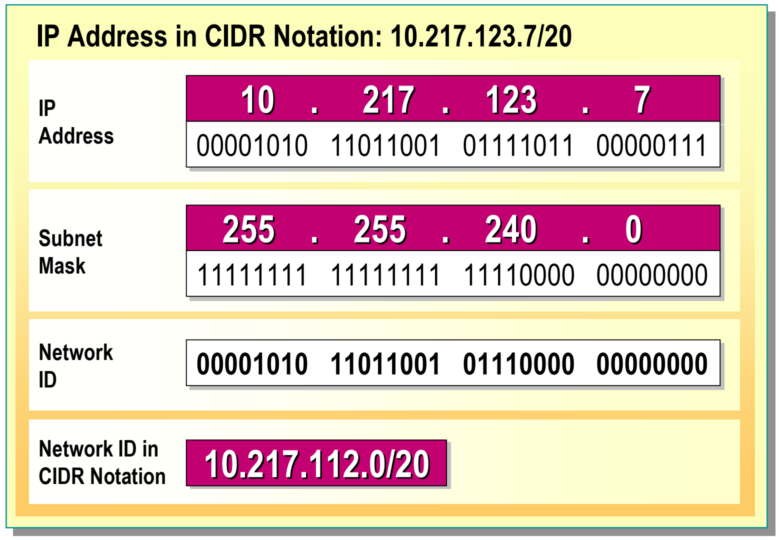
The following table lists the practical bit masks, the associated subnet masks, and the number of classful networks possible for each.

CIDR notation	Subnet mask	Number of classful networks
/8	255.0.0.0	256 class Bs
/9	255.128.0.0	128 class Bs
/10	255.192.0.0	64 class Bs
/11	255.224.0.0	32 class Bs
/12	255.240.0.0	16 class Bs
/13	255.248.0.0	8 class Bs
/14	255.252.0.0	4 class Bs
/15	255.254.0.0	2 class Bs
/16	255.255.0.0	1 class B or 256 class Cs
/17	255.255.128.0	128 class Cs
/18	255.255.192.0	64 class Cs
/19	255.255.224.0	32 class Cs
/20	255.255.240.0	16 class Cs
/21	255.255.248.0	8 class Cs
/22	255.255.252.0	4 class Cs
/23	255.255.254.0	2 class Cs
/24	255.255.255.0	1 class C
/25	255.255.255.128	½ class C
/26	255.255.255.192	1/4 class C
/27	255.255.255.224	1/8 class C
/28	255.255.255.240	1/16 class C

Calculating the Network ID

Slide Objective
To illustrate the procedure for calculating the network ID.

Lead-in
The network ID needs to be calculated in binary notation to determine whether one host is local or remote with respect to another.



Delivery Tip
Although it is possible to use Windows Calculator to AND two decimal numbers together, the students will benefit by learning the principle behind the calculation. Also remind students that this process is very similar to the process of dividing network IDs by using simple decimal subnet masks. Calculating the network ID prepares the students for identifying local and remote computers, which is described in the following section.

When you configure IP addresses in Windows 2000, you must type the IP address and subnet mask information in dotted decimal notation. Windows 2000 does not accept input in CIDR notation. However, whether the IP address and subnet mask are specified in dotted decimal notation or CIDR notation, the network ID is best calculated in binary notation.

To calculate the network ID when the IP address is specified in CIDR notation

1. Convert the IP address to binary format.
2. Use the bit mask to determine the number of bits in the IP address that make up the network ID.
3. Add 0s to the network ID to obtain its four-octet structure.

Example 1

For example, consider the IP address 10.217.123.7/20. Because this address shows that the subnet mask consists of 20 contiguous 1s, the network ID consists of the first 20 bits in the IP address followed by 0s. The following table illustrates the calculation of the network ID in binary notation.

	Binary notation
IP address	00001010 11011001 01111011 00000111
Subnet mask	11111111 11111111 11110000 00000000
Network ID	00001010 11011001 0111000 00000000

To calculate the network ID when the IP address and subnet mask are specified in decimal notation

1. Convert the IP address to binary format.
2. Convert the subnet mask to binary format.
3. Use the number of contiguous 1s in the subnet mask to determine the number of bits in the IP address that make up the network ID.

Example 2

For example, consider the IP address 10.217.123.7 and its associated subnet mask 255.248.0.0. The following table illustrates the calculation of the network ID in binary notation and the conversion of both the IP address and subnet mask to binary notation.

	Binary notation
IP address	00001010 11011001 01111011 00000111
Subnet mask	11111111 11111000 00000000 00000000
Network ID	00001010 11011000 00000000 000000000

Because the binary notation of the subnet mask consists of 13 contiguous 1s, the network ID consists of the first 13 bits in the IP address followed by 0s.

Note The process of combining the IP address with its subnet mask is known as binary ANDing.

To convert the network ID from binary notation to decimal notation

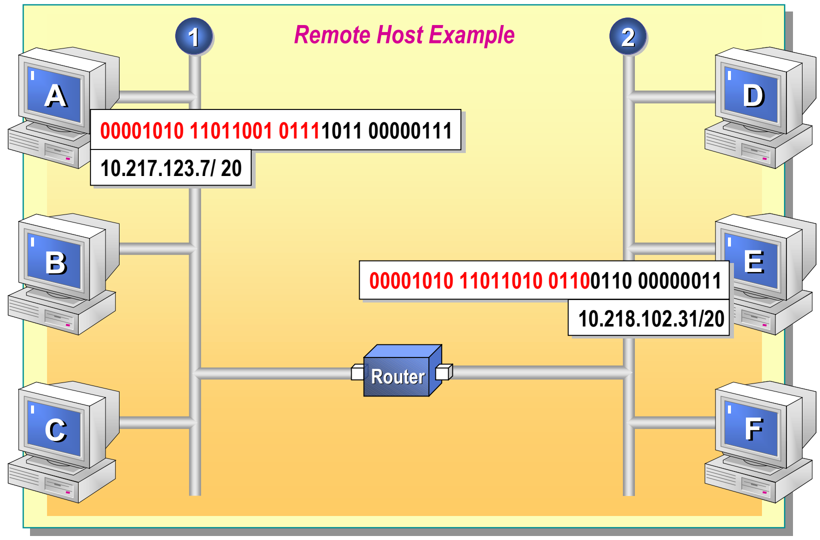
After you calculate the network ID in binary notation, you should convert it back to the dotted decimal system for users to work with it. The following table illustrates the dotted decimal notations for the network IDs calculated in the two examples above.

	Binary notation	CIDR notation
Network ID, Ex 1	00001010 11011001 0111000 00000000	10.217.56.0/20
Network ID, Ex 2	00001010 11011000 0000000 00000000	10.216.0.0/13

Determining Local and Remote Hosts

Slide Objective
To illustrate the procedure for determining whether a host is local or remote with respect to another.

Lead-in
After you convert the network ID of an IP address to binary notation, it is possible to determine if another host is local or remote with respect to it.



Delivery Tip
Remind the students that this process is identical to the one used with classful IP addresses: If the network IDs are different, then the computers are remote; if they are the same, the computers are local. Also ask the students if they remember what physical device is used to communicate if the computers are remote. The animated slide shows an example of both local and remote hosts that use the same IP addresses with different subnet masks.

After the network ID is identified, a computer can distinguish whether a destination host is local or remote by comparing its own network ID with that of the destination host. This determines whether a router is required as an intermediate host.

Local Host Example

Consider the pair of IP addresses 10.217.123.7/10 and 10.218.102.31/10 belonging to Computer A and Computer B. The following tables detail how the two network IDs are calculated to determine whether they are local or remote in relation to each other.

	Computer A
IP addresses	00001010 11011001 01111011 00000111
Subnet masks	11111111 11000000 00000000 00000000
Network ID (binary)	**00001010 11**000000 00000000 00000000
Network ID (decimal)	10.192.0.0

	Computer B
IP addresses	00001010 11011010 01100110 00000011
Subnet masks	11111111 11000000 00000000 00000000
Network ID (binary)	**00001010 11**000000 00000000 00000000
Network ID (decimal)	10.192.0.0

As the tables show, the network IDs of the two IP addresses match. Therefore, Computer A is local in relation to Computer B.

Remote Host Example

As another example, consider the pair of IP addresses 10.217.123.7/20 and 10.218.102.31/20 belonging to Computer A and Computer E. The following tables detail how the two network IDs are calculated to determine whether they are local or remote in relation to each other.

	Computer A
IP addresses	00001010 11011001 01111011 00000111
Subnet masks	11111111 11111111 11110000 00000000
Network ID (binary)	**00001010 11011001 0111**0000 00000000
Network ID (decimal)	10.217.112.0

	Computer E
IP addresses	00001010 11011010 01100110 00000011
Subnet masks	11111111 11111111 11110000 00000000
Network ID (binary)	**00001010 11011010 0110**0000 00000000
Network ID (decimal)	10.218.96.0

As these tables show, the network IDs of the two IP addresses do not match. Therefore, Computer A is remote in relation to Computer E.

Lab B: Determining Local and Remote Destinations

Slide Objective
To introduce the lab.

Lead-in
In this lab, you will use Windows Calculator to create a subnet mask when given a number of hosts and report the number of hosts for a given subnet mask. You will also evaluate a subnet mask to determine if it is valid.

Objectives

After completing this lab, you will be able to:

- Calculate a network ID given a decimal IP address and subnet mask.
- Calculate whether the destination computer is local or remote.

Lab Setup

This lab is a simulation. To complete this lab, you need the following:

- A computer running Microsoft Windows 2000, Microsoft Windows NT® version 4.0, Microsoft Windows 98, or Microsoft Windows 95.
- Microsoft Internet Explorer 5 or later.
- A minimum display resolution of 800 x 600 with 256 colors. Recommended display resolution of 800 x 600 with high color (16-bit).

► **To start the lab**

1. Log on to Windows 2000 as Administrator with a password of **password**.
2. On the desktop, double-click the **Internet Explorer** icon.
3. On the Student Materials Web page, click **Lab Simulations**.
4. Click **Determining Local and Remote Destinations**.
5. Read the introduction information, and then click the link to begin the simulation.

Estimated time to complete this lab: 30 minutes

◆ IP Address Allocation Using CIDR

Slide Objective
To introduce the process of allocating IP addresses by using CIDR.

Lead-in
CIDR streamlines the process of allocating IP addresses.

- **Available Host IDs**
- **Optimizing the Allocation of IP Addresses**

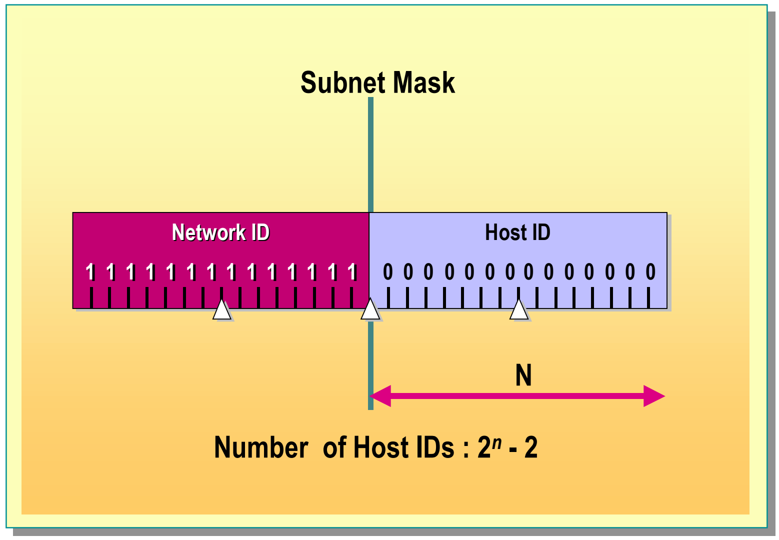
CIDR provides a simple method for calculating the number of available host IDs based on the subnet mask associated with a block of IP addresses.

CIDR optimizes the allocation of IP addresses through the use of *supernetting* and *subnetting*. Supernetting is a procedure that combines multiple addresses to form a single network ID. Subnetting divides a large network into multiple subnets.

Available Host IDs

Slide Objective
To illustrate the calculation of the number of available host IDs in an address block.

Lead-in
The number of host IDs available per IP address block can be determined by using the subnet mask associated with the IP address block.



Delivery Tip
Review the addressing guidelines described in module 7, "Examining IP Addressing," in course 2151A, *Microsoft Windows 2000 Network and Operating System Essentials*. The first two guidelines in that topic also apply to this topic. Use the examples to point out the importance of viewing the IP address and subnet mask in binary format as a means of identifying valid and invalid host IDs. Use the chart on the following page to identify which subnet mask is needed to configure a specific size of network. Ask the students what the CIDR notation and subnet mask would be for several networks of different sizes.

The number of hosts supported by a network ID is calculated by using the number of 0s in the associated subnet mask. If the number of 0s is denoted by *n*, the number of hosts is calculated as 2^n-2. The two addresses that are subtracted from the expression are the reserved IP addresses that cannot be allocated to any host.

Reserved Host IDs

The values of the bits in a host ID cannot be all 0s or 1s. The host ID with the value 0 is used to denote a network ID. The host ID consisting of all 1s is used as an IP broadcast address. The following tables show an example of a valid host ID and a reserved host ID.

Valid Host ID

	Binary notation	Decimal notation
IP address	11000000 10101000 11000001.00000000	192.168.193.0
Subnet mask	11111111 11111111 11110000 00000000	255.255.240.0
Host ID	0001 00000000	1.0

Reserved Host ID

	Binary notation	Decimal notation
IP address	11000000 10101000 11000000.00000000	192.168.192.0
Subnet mask	11111111 11111111 11110000 00000000	255.255.240.0
Host ID	0000.00000000	0.0

The following table shows the possible host values for a specific number of non-zero bits in the CIDR notation of an IP address.

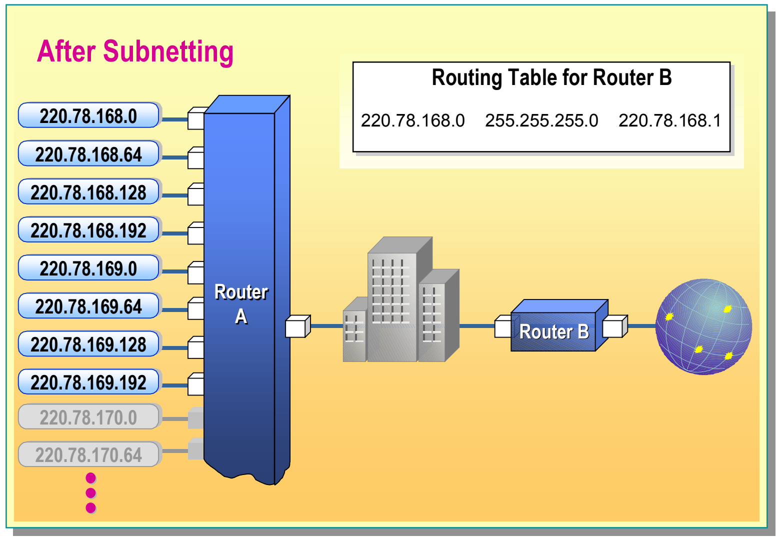
CIDR notation	Subnet mask	Number of 0s	Number of hosts (2^n-2)
w.x.y.z/1	128.0.0.0	31	2,147,483,646
w.x.y.z/2	192.0.0.0	30	1,073,741,822
w.x.y.z/3	224.0.0.0	29	536,870,910
w.x.y.z/4	240.0.0.0	28	268,435,454
w.x.y.z/5	248.0.0.0	27	134,217,726
w.x.y.z/6	252.0.0.0	26	67,108,862
w.x.y.z/7	254.0.0.0	25	33,554,430
w.x.y.z/8	255.0.0.0	24	16,777,214
w.x.y.z/9	255.128.0.0	23	8,388,606
w.x.y.z/10	255.192.0.0	22	4,194,302
w.x.y.z/11	255.224.0.0	21	2,097,150
w.x.y.z/12	255.240.0.0	20	1,048,574
w.x.y.z/13	255.248.0.0	19	524,286
w.x.y.z/14	255.252.0.0	18	262,142
w.x.y.z/15	255.254.0.0	17	131,070
w.x.y.z/16	255.255.0.0	16	65,534
w.x.y.z/17	255.255.128.0	15	32,766
w.x.y.z/18	255.255.192.0	14	16,382
w.x.y.z/19	255.255.224.0	13	8,190
w.x.y.z/20	255.255.240.0	12	4,094
w.x.y.z/21	255.255.248.0	11	2,046
w.x.y.z/22	255.255.252.0	10	1,022
w.x.y.z/23	255.255.254.0	9	510
w.x.y.z/24	255.255.255.0	8	254
w.x.y.z/25	255.255.255.128	7	126
w.x.y.z/26	255.255.255.192	6	62
w.x.y.z/27	255.255.255.224	5	30
w.x.y.z/28	255.255.255.240	4	14
w.x.y.z/29	255.255.255.248	3	6
w.x.y.z/30	255.255.255.252	2	2
w.x.y.z/31	255.255.255.254	1	Not available
w.x.y.z/32	255.255.255.255	0	Not available

Note Some individuals and small businesses are assigned the subnet mask 255.255.255.255 by their Internet provider. Although many of the network IDs in this chart are not practical, such as 2 or 2 billion, it is a useful reference for determining the subnet mask needed for subnets of particular sizes.

Optimizing the Allocation of IP Addresses

Slide Objective
To illustrate the concepts of supernetting and subnetting.

Lead-in
You can use two procedures to optimally allocate IP addresses.



Delivery Tip
This page introduces the topic of supernetting and reviews the topic of subnetting. This page does not provide the students with the skills to subnet and supernet an existing network. The slides illustrate router A inside an organization and router B on the Internet. Use the slides to illustrate how supernetting reduces the problems described at the beginning of the module, whereas subnetting enables an organization to maintain an internal structure that effectively separates their overall network.

In addition to providing a new method for differentiating between local and remote hosts, CIDR also provides solutions to the problems associated with classful IP addressing. These problems include the assignment of an incorrect number of IP addresses or the addition of multiple entries to Internet routing tables.

Adding Multiple IP Addresses

Consider an organization that must connect 800 computers to the Internet. It can do this by choosing either one class B network ID that provides 65,534 IP addresses or four class C addresses that provide 1,016 addresses. The first choice results in 64,734 unused IP addresses. With the second choice, only 216 IP addresses go unused, but each router on the Internet is required to add four routes to the organization in its routing table.

Supernetting

To avoid such inappropriate address allocations, CIDR uses the concept of *supernetting*. Supernetting is a strategy that combines multiple addresses from the classful environment into a single network ID in the classless environment. Using supernetting, CIDR combines multiple class C network IDs into a single CIDR network ID. In CIDR notation, a network ID is represented by the number of bits in a subnet mask, similar to an IP address, for example, 192.168.0.0/22.

In the example above, the organization can purchase a single 22-bit network ID (/22), which allows a maximum of 1,022 valid hosts, thereby providing the current 800-computer network with the ability to expand. This solution neither wastes numerous IP addresses nor adds multiple entries to the Internet routing tables.

Subnetting

Using supernetting, each organization is assigned a single CIDR network ID, which represents a single network. However, it is not efficient to work with a single large network because such factors as high network noise make the network inefficient.

To reduce the noise on its internal network, an organization can physically segment the network into subnets by using routers. As each subnet requires its own network ID, the single CIDR network ID obtained for the organization can be divided into smaller network IDs for each subnet. The process of dividing a network ID into smaller network IDs is known as subnetting. The smaller network IDs are also known as subnet IDs.

In the example considered earlier, after the organization is assigned a single network ID, it can choose to subdivide its network into smaller segments, depending upon its requirements. The organization does this by using an appropriate subnet mask. After segmenting the network physically, it is important to ensure that the network is divided logically into subnets and that a subnet is created for every segment in the network.

Then the organization assigns an internal network ID, the subnet ID, to each of the subnets, based on the number of computers that exist on each subnet. However, because subnetting is an internal process that is specific to a network, any router outside of the organization's network cannot view the separate subnets and their subnet IDs.

In the example, if the organization wanted to have no more than 62 computers on each subnet, it could use the subnet mask 255.255.255.192.

Lab C: Allocating IP Addresses

Slide Objective
To introduce the lab.

Lead-in
In this lab, you will identify the size of the network ID needed based on the number of computers on a network and the number of computers per subnet.

Objectives

After completing this lab, you will be able to:

- Identify the size of network ID needed based on the number of computers on a network.
- Identify the size of network ID needed based on the number of computers per subnet.

Prerequisites

Before working on this lab, you must have:

- Knowledge of classful IP addresses.
- Knowledge of converting binary numbers using a calculator.

Estimated time to complete this lab: 15 minutes

Exercise 1
Allocating IP Addresses

Scenario

You are an administrator and need to configure the network ID needed to ensure that there are enough IP addresses available for all of the computers in your network. You also need to consider using a network ID to divide your network to avoid excess broadcast traffic.

Goal

In this exercise, you will identify the network ID needed, given a specific number of hosts.

Tasks	Detailed Steps
1. Log on as Administrator with a password of **password**, and calculate the number of available IP addresses given a subnet mask.	**a.** Log on to Windows 2000 as Administrator with a password of **password**. **b.** Open Windows 2000 Calculator in scientific mode. **c.** Convert the CIDR notation to a binary subnet mask. **d.** Write down the binary subnet mask in the table below. **e.** Count the number of zeros in the subnet mask. **f.** Write down the number of zeros in the table below. **g.** In the calculator, type **2** **h.** In the calculator, click **x^y**. **i.** In the calculator, type **18** (This is the first calculation.) **j.** In the calculator, click – (the minus button). **k.** In the calculator, type **2** **l.** In the calculator, click = (the equal button). **m.** Write down the value returned from the calculator in the chart below.
Note: Repeat the above steps for each IP address in the table below. The first calculation is provided as an example.	

CIDR notation	Subnet mask	Number of 0s	Number of hosts (2*n*-2)
w.x.y.z/18	**11111111.11111111.11000000.00000000**	**18**	**16,382**
w.x.y.z/19			
w.x.y.z/20			
w.x.y.z/21			
w.x.y.z/22			
w.x.y.z/23			
w.x.y.z/24			
w.x.y.z/25			
w.x.y.z/26			
w.x.y.z/27			

(*continued*)

Tasks	Detailed Steps
❓	Using the previous table, what is the CIDR notation for a network that needs 2,000 IP addresses? How many IP addresses are available and how many will be unused? Hint: do not forget that the number of unusable IP addresses also includes the network ID and broadcast address. ***w.x.y.z/21*** **is the CIDR notation. With 2,046 IP addresses available, 2,000 IP addresses will be used, resulting in a total of 48 unused IP addresses (46 plus the two unusable IP address for a total of 48).**
❓	What IP address class is required to provide 2,000 IP addresses? How many IP addresses will be provided and how many will be unused? **A class B, which provides 65,534 IP addresses with 63,534 plus two for a total of 63,536 unused addresses.**
❓	What is the minimum number of class C address blocks that is needed to provide 2,000 IP addresses? How many IP addresses will be available for use? How many IP addresses will be unused? **There are 254 available IP addresses per class C address, which requires at least eight class C address blocks, for a total of 2,032 addresses. Needing only 2,000 addresses would give 32 unused available addresses plus 16 (8 times 2) for a total of 48 unused IP addresses.**

(*continued*)

Tasks	Detailed Steps
	Although the number of addresses unused in the class C question is the same as the number unused in the CIDR notation, in which additional way are the multiple class C addresses inefficient? **Using multiple class C addresses would also cause seven additional (eight total) routes to be added to every router on the Internet.**
	If your organization decided to reduce broadcast traffic by subnetting a 21-bit network ID (2,048 IP addresses) to allow for a maximum of 126 computers per subnet, what will be the CIDR notation and decimal subnet mask for each subnet? **Subnetting a CIDR notation of *w.x.y.z*/21 into 126 computers per subnet allows for a CDIR notation of *w.x.y.z*/25 and a decimal subnet mask of 255.255.255.128.**
2. Close all windows and log off from Windows 2000.	**a.** Close all windows and log off from Windows 2000.

Review

Slide Objective
To reinforce module objectives by reviewing key points

Lead-in
The review questions cover some of the key concepts taught in the module.

- Classless Inter-Domain Routing (CIDR)
- Binary IP Addresses
- Binary Subnet Masks
- IP Address Allocation Using CIDR

1. What are the three major problems that developed as a result of using classful IP addressing?

   The class B address space was near depletion.

   The routing tables of the Internet were nearly full.

   All IP addresses would be eventually assigned.

2. What is CIDR?

   CIDR is a new method for organizing IP addresses. In this method, all IP addresses are represented using binary notation in place of decimal notation.

3. In the binary notation system, each IP address is represented as a string of 32 numbers. This string can be broken up into four fields, each called ___ or ___.

   an octet; a byte.

4. In a subnet mask, the network ID is determined by the number of contiguous __, and the host ID is determined by the number of contiguous __.

   1s; 0s.

5. Which notation must you use to type the IP address and subnet mask information when configuring IP addresses in Windows 2000?

   Decimal notation.

6. What is supernetting?

   Supernetting is a strategy that combines multiple IP addresses from the classful environment into a single address in the classless environment.