MICROSOFT
TRAINING
AND CERTIFICATION

Microsoft® Official
Curriculum

Module 7: Configuring and Supporting TCP/IP

Contents

Microsoft®

Instructor Notes

Presentation:
120 Minutes

Lab:
45 Minutes

This module provides students with an overview of TCP/IP concepts and how to configure an IP address for Microsoft® Windows® XP Professional communicating over a network through TCP/IP. The module focuses on tasks required to establish dynamic or static IP addressing and how to perform basic troubleshooting tasks to determine if the Windows XP Professional connection to the network is fully functioning.

After completing this module, students will be able to:

- Explain the function of TCP/IP protocols.
- Explain IP addressing requirements for Windows XP Professional.
- Explain the purpose of IP subnetting.
- Determine if hosts are on a local or remote network.
- Describe the difference between classful and classless IP addressing.
- Configure Windows XP Professional to use either Dynamic Host Control Protocol (DHCP) or static IP addressing.
- Configure an alternate Transmission Control Protocol/Internet Protocol (TCP/IP) configuration for a client computer.
- Troubleshoot IP addressing problems using TCP/IP utilities.

Materials and Preparation

This section provides the materials and preparation tasks that you need to teach this module.

Required Materials

To teach this module, you need Microsoft PowerPoint® file 2272A_07.ppt.

Preparation Tasks

To prepare for this module, you should:

- Read all of the materials for this module.
- Complete the labs.
- Review the Delivery Tips and Key Points for each section and topic.
- Study the review questions and prepare alternative answers for discussions.
- Anticipate student questions about material and write out answers to those questions.
- Familiarize yourself with the build Topics in this module.

Instructor Setup for Lab

This section provides setup instructions that are required to prepare the instructor computer or classroom configuration for a lab.

Lab 7A: Configuring IP Addresses for Windows XP Professional

► **To prepare for the lab**

1. The lab in this module requires that the student computers are running Microsoft Windows XP Professional.
2. The instructor Windows 2000 Advanced Server is configured with DHCP service, with a scope of address large enough for student computers.
3. The student computers should be configured as DHCP clients.

Module Strategy

Use the following strategy to present this module:

- Introduction to TCP/IP

  The first section of the module begins with a conceptual overview to network communications. Use an analogy of sending a letter to sending data across the network. Explain that TCP/IP is a protocol suite that enables transmission of data. Review each layer of the protocol stack pointing out its function. For each layer, describe some of the more commonly used protocols. Stress the difference between UDP and TCP at the Transport layer. Then using an animated Topic, step through the TCP/IP communication process. Point out that each host must have a unique IP address for successful communication to occur.

- Examining Classful IP Addressing

  In this discussion, students identify the components of an IP address. Describe the purpose of classful addressing and how to distinguish between the various address classes. Demonstrate how to convert binary to decimal and for the purpose of understanding the different class sizes. Later, students will use their conversion skills to determine if hosts are remote or local with respect to one another. Finally, point out how IP addresses are assigned to networks, hosts, and default gateways.

- Defining Subnets

  This section provides a discussion on subnetting. Describe the reason for creating subnets in a network. Continue by explaining the concept of a subnet masks and the reasons why subnet masks are useful for dividing network IDs. Distinguish between default and custom subnets and describe how a custom subnet is created. Finally, demonstrate the procedure for determining whether two hosts are local or remote with respect to each other.

- Using Classless Inter-Domain Routing

  This section provides an overview of the purpose and function of Classless Inter-Domain Routing (CIDR). Describe how CIDR IP addresses are assigned. Point out that CIDR is the current practice for assigning IP addresses.

- Configuring IP Addresses

  In this section, configuration of dynamic and static IP addresses is discussed. Demonstrate how to configure a dynamic address, a static address, and a primary and alternate IP address.

- Troubleshooting IP Addresses

  This section provides an overview of TCP/IP troubleshooting utilities. Demonstrate and discuss using **ipconfig** and **ping** as a means of troubleshooting IP addresses.

- Lab A: Configuring IP Addresses for Windows XP Professional

  In this lab, students configure Windows XP Professional for a dynamic and static address and use the **ipconfig** command to verify TCP/IP network settings. Students then configure a primary dynamic address and an alternate address. They use the **ipconfig** and **ping** commands to verify these settings.

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 Training and Certification courseware.

Important The lab in this module is also dependent on the classroom configuration that is specified in the Customization Information section at the end of the *Classroom Setup Guide* for Course 2272A, *Implementing and Supporting Microsoft Windows XP Professional (Beta Course)*.

Lab Setup

There are no lab setup requirements that affect replication or customization.

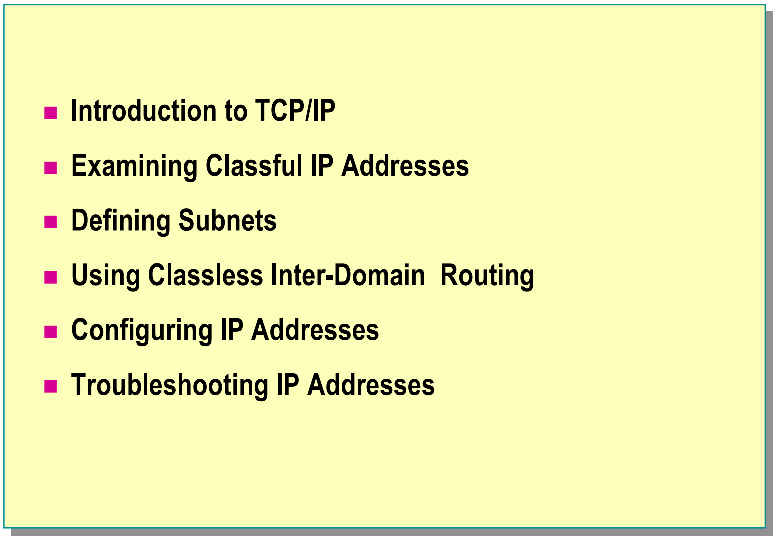
Lab Results

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

Overview

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

Lead-in
In this module, you will learn about assigning and troubleshooting TCP/IP addresses in Windows XP Professional.



- **Introduction to TCP/IP**
- **Examining Classful IP Addresses**
- **Defining Subnets**
- **Using Classless Inter-Domain Routing**
- **Configuring IP Addresses**
- **Troubleshooting IP Addresses**

Transmission Control Protocol/Internet Protocol (TCP/IP) for Microsoft® Windows® XP Professional offers a standard, routable enterprise networking protocol that is the most complete and accepted protocol available. Most network operating systems in use today offer TCP/IP support, and large networks rely on TCP/IP for much or all of their network traffic.

The various protocols in the TCP/IP stack function together to make network communication happen. The communication process involves multiple activities, including resolving user-friendly computer names to Internet Protocol (IP) addresses; determining the location of the destination computer; and packaging, addressing, and routing the data so that it reaches the destination successfully.

To effectively manage and support Windows XP Professional in a network environment, you need to understand the TCP/IP communication process, how IP addresses are assigned, and how to use TCP/IP utilities to troubleshoot communication problems.

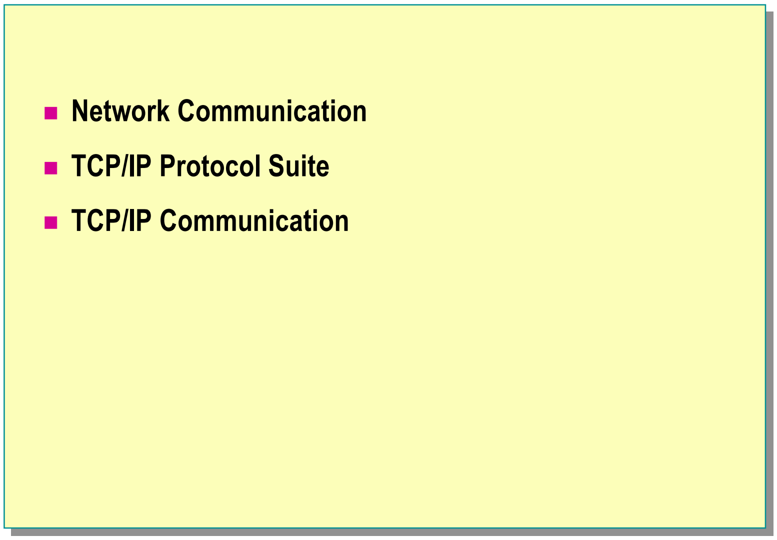
After completing this module, you will be able to:

- Explain the function of TCP/IP protocols.
- Explain IP addressing requirements for Windows XP Professional.
- Explain the purpose of IP subnetting.
- Determine if hosts are on a local or remote network.
- Describe the difference between classful and classless IP addressing.
- Configure Windows XP Professional to use either Dynamic Host Control Protocol (DHCP) or static IP addressing.
- Configure an alternate Transmission Control Protocol/Internet Protocol (TCP/IP) configuration for a client computer.
- Troubleshoot IP addressing problems using TCP/IP utilities.

# ◆ Introduction to TCP/IP

Topic Objective
To introduce basic concepts relevant to TCP/IP communication and addressing.

Lead-in
In Windows XP Professional environments, TCP/IP enables connectivity to transmit data by using unique address identifiers.



- **Network Communication**
- **TCP/IP Protocol Suite**
- **TCP/IP Communication**

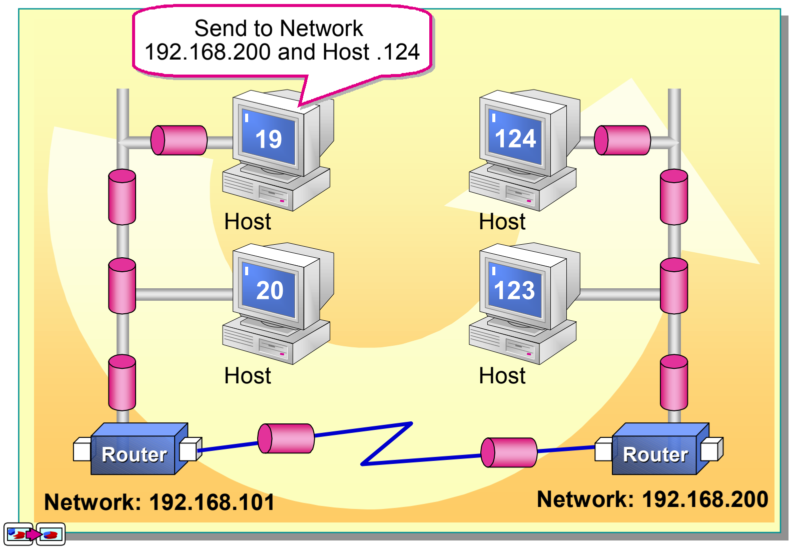
TCP/IP uses a layered communication model to transmit data. This model is comprised of a set of protocols that are layered upon each other. Each layer of protocols provides a specific function and service in transmitting data.

Understanding the overall network communication flow and the function of the protocols in the communication process can assist you with both configuration and troubleshooting activities.

Network Communication

Topic Objective
To provide an overview of network communications.

Lead-in
Successful communications on a TCP/IP network are dependent on adhering to IP addressing standards.



The process by which TCP/IP transmits data between two locations is analogous to someone creating and sending a letter from one city to someone in another city. The TCP/IP communication process is initiated by using an application on the source computer that prepares the data to be transmitted in a format that an application on the destination computer can read. This process is similar to writing a letter in a language that the recipient can understand. The data is then associated with the destination application and computer, similar to how you address a letter to a recipient and household. The address of the destination computer is then added to the data, just as the address of the recipient is specified on the letter.

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

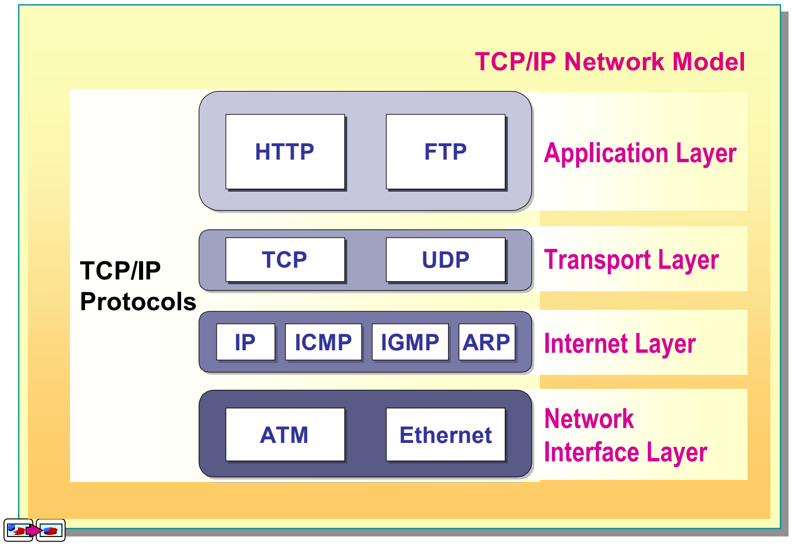
The TCP/IP communication process is initiated using an application on the source computer that prepares the data to be transmitted. The address of the destination computer is then added to the data. The data and destination address are sent over the network to the destination. The network medium used for transmitting the data is independent of the activities that are directly related to creating the data.

In a network, many applications use network resources at the same time. Furthermore, the network may contain many devices. TCP/IP employs a method for differentiating protocols from one another and uniquely identifying each device on the network. Adhering to these addressing standards is important for a fully functioning TCP/IP network.

TCP/IP Protocol Suite

Topic Objective
To illustrate the TCP/IP layers.

Lead-in
TCP/IP uses a four-layer communication model.



Delivery Tip
Explain the four layers of the TCP/IP protocol stack. Do not spend a large amount of time on the specific protocols within the stack. The information is provided as a framework for understanding TCP/IP communication and utilities. Instead, use this page to point out various layers and the function of the layers.

The tasks that are involved in using TCP/IP in the communication process are distributed between protocols that are organized into four distinct layers of the TCP/IP stack. The four layers are *application*, *transport*, *Internet*, and *network interface*. All protocols that belong to the TCP/IP protocol stack are located in these four layers of the model.

Application Layer

The application layer is the topmost layer in the TCP/IP stack. All applications and utilities are contained in this layer and use this layer to gain access to the network. The protocols in this layer are used for the formatting and exchange of user information. The following are some examples of protocols that exist within the application layer:

- Hypertext Transfer Protocol (HTTP)

  HTTP is used to establish a connection with a Web server and transmit Hypertext Markup Language (HTML) pages to the World Wide Web.

- File Transfer Protocol (FTP)

  FTP is used to transfer files over a TCP/IP network; for example, after developing the HTML pages for a Web site on a local computer, the pages are typically uploaded to a Web server by using FTP.

Transport Layer

The transport layer provides the ability to gain and guarantee communication between computers, and passes the data up to the application layer or down to the Internet layer depending on whether the data is being sent or received. The transport layer also specifies the unique identifier of the application to which data is to be delivered. The transport layer has two core protocols that control the method by which data is delivered:

- Transmission Control Protocol (TCP)

  TCP is a reliable, connection-oriented delivery service. *Connection-oriented* means that a session must be established before hosts can exchange data. Reliability is achieved by assigning a sequence number to each segment of data transmitted. If a TCP segment is broken into smaller pieces, the receiving host knows whether all pieces have been received. An acknowledgment is used to verify that the other host received the data. For each segment sent, the receiving host must return an acknowledgment (ACK) within a specified period, for the segment that is received.

- User Datagram Protocol (UDP)

  UDP provides fast delivery of data, but does not guarantee data delivery. UDP provides a connectionless service that offers unreliable, “best effort” delivery. This means that the arrival of data is not guaranteed, nor is the correct sequencing of delivered segment of data. UDP is used by applications that do not require an acknowledgment of receipt of data and that typically transmit small amounts of data at one time. Reliability becomes the responsibility of the application.

Internet Layer

The Internet layer is responsible for addressing, packaging, and routing the data that is to be transmitted. This layer contains four core protocols:

- Internet Protocol (IP)

  IP is responsible for addressing the data that is to be transmitted and transmitting the data to its destination.

- Address Resolution Protocol (ARP)

  ARP is responsible for identifying the media access control address of the network adapter on the destination computer. The *media access control* address is a unique 12-character hexadecimal number, such as B5-50-04-22-D4-65, for a physical device such as the network adapter.

- Internet Control Message Protocol (ICMP)

  ICMP is responsible for providing diagnostic functions and reporting errors that result from the unsuccessful delivery of data.

- Internet Group Management Protocol (IGMP)

  IGMP is responsible for the management of multicasting within TCP/IP. *Multicasting* is the process of sending a message simultaneously to more than one destination host on a network.

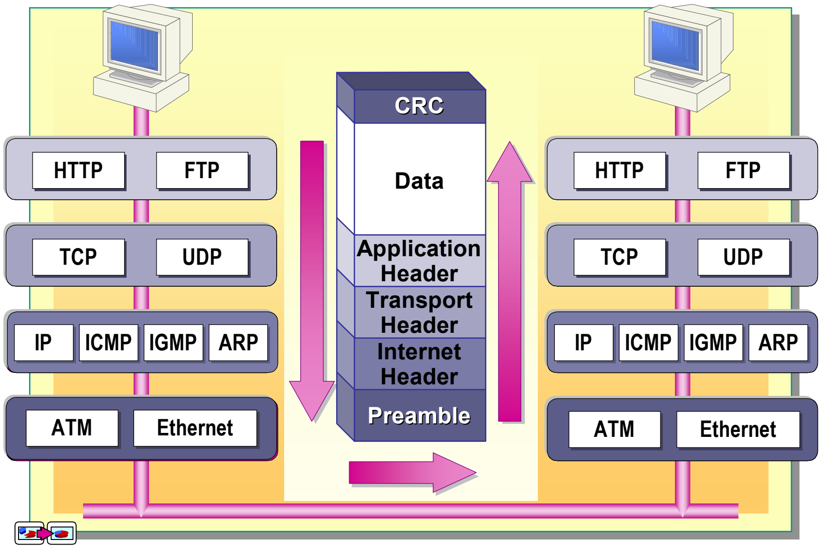
Network Interface Layer

The network interface layer is responsible for merging data on the network medium and receiving data that resides on the network medium. The network interface layer contains protocols such as Ethernet and asynchronous transfer mode (ATM), which define how data is transmitted on the network.

TCP/IP Communication

Topic Objective
To illustrate data flow within a TCP/IP network.

Lead-in
Each layer of the protocol suite encapsulates data until the data reaches its destination.



Delivery Tip
The Topic on this page illustrates the flow of data through the TCP/IP protocol stack.

TCP/IP transmits data on the network by dividing it into smaller portions called *packets*. As the data packets pass through each layer, the protocol in that layer attaches information, referred to as a header. The *header* contains instructions used to transport the data to the specified destination. When sending packets from the source, the data and header are encapsulated and treated as data by the protocol in the layer below.

When the packet is received at the destination, the corresponding layer strips off a header and treats the remaining packet as data. For example, the header added by the transport layer on the originating computer is examined, used, and removed by the transport layer at the destination computer. The packet is then passed up the protocol stack to the next appropriate protocol until it reaches the destination computer.

Application Layer

The data transmission process begins at the application layer of the TCP/IP protocol stack. An application, such as the FTP utility, initiates the process at the source computer by preparing the data in a format that the application at the destination computer recognizes. The application at the source computer controls the entire process.

Transport Layer

From the application layer, the data moves to the transport layer. This layer contains the TCP and UDP protocols. The application initiating the transmission request selects which protocol to use, for example TCP, and the checksum is added. A *checksum* is a numerical value that is used to test the data to ensure the transport was free from errors.

When TCP is used, it:

- Assigns a sequence number to each segment to be transmitted.
- Adds acknowledgement information for a connection-oriented transmission.
- Adds the TCP port number for the source and destination applications.

Internet Layer

After the transport information is added, the data packet is passed to the Internet layer of the TCP/IP protocol stack. In this layer, IP adds the following information:

- The source IP address
- The destination IP address
- The transport protocol
- A checksum value
- Time to Live (TTL) information

In addition to adding this information, the Internet layer is also responsible for resolving the destination IP addresses to a media access control address. The media access control address is added to the packet header and the packet is handed down to the network interface layer.

Network Interface Layer

At the network interface layer, two types of information are added: a preamble and a cyclical redundancy check (CRC). The *preamble* is a sequence of bytes that identifies the beginning of a packet. The *CRC* is a mathematical computation that is added to the end of the packet to verify that the packet has not been corrupted.

After the information is added to the frames at the network interface layer, they are merged onto the network.

Destination Computer

When the frames reach the destination computer, the network interface layer on this computer discards the preamble and recalculates the CRC. If this value matches the value calculated before transmission, then the destination media access control address on the frame is examined.

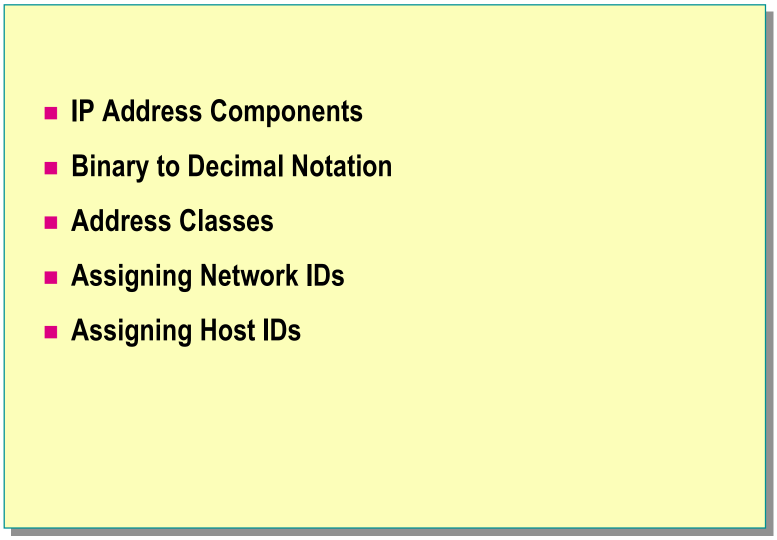
If the media access control address is a broadcast address or if the media access control address matches that of the destination computer, the frame is passed to the IP in the Internet layer above, otherwise the frame is discarded. At the IP layer, IP recalculates the checksum and compares it with the value calculated before transmission to determine if the packet arrived intact. Then IP passes the packet to the transport protocol identified in the IP header.

At the transport layer, if the packet is received by TCP, it checks the sequence number on the packet and sends an acknowledgement back to TCP at the source computer. Then, sends it onwards to the appropriate application in the application layer above. After the application receives the data, it processes it as required.

# ◆ Examining Classful IP Addressing

Topic Objective
To introduce classful IP addressing.

Lead-in
A computer needs to determine the location of another computer before it can send data to it.



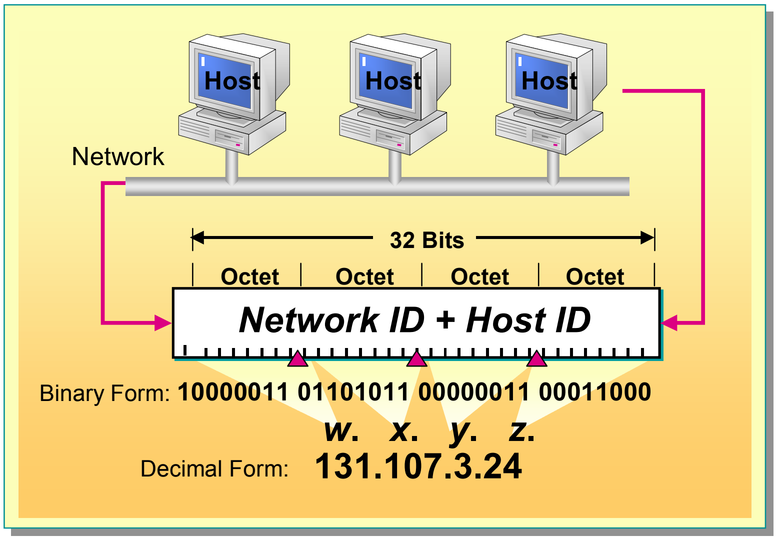
- **IP Address Components**
- **Binary to Decimal Notation**
- **Address Classes**
- **Assigning Network IDs**
- **Assigning Host IDs**

To communicate on a network, each computer must have a unique IP address and conform to a standard format. The IP address identifies a computer’s location on the network. Understanding the IP address format and standards is necessary when assigning and troubleshooting IP addresses.

IP Address Components

Topic Objective
Describe the components of an IP addresses.

Lead-in
An IP address has two parts: a network ID and a host ID.



An IP address has two parts: the network ID and the host ID.

Network ID

The first part of the IP address is the *network ID*, which identifies the network on which the computer is located. All computers on the same network must share the same network ID.

Host ID

The second part of the IP address is the *host ID*, which identifies a computer, router, or other device within a network. The host ID for each host must be unique within the network ID.

Each IP address is 32 bits long and is composed of four 8-bit fields, called octets. Octets are separated by periods.

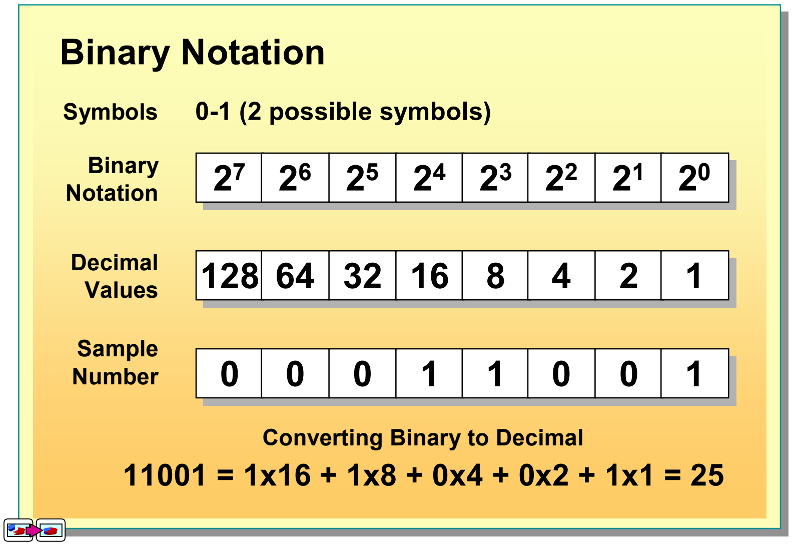
Binary notation is ideal for IP addressing because it contains only two symbols which can represent the two states of a bit: on or off. However, binary notation is awkward when communicating with others ("Your IP address is 11101001 11101111.etc."), and prone to input errors. Therefore, binary notation is converted into its decimal value. The octet represents a decimal number in the range 0–255. This format is called dotted decimal notation. The following table gives an example of an IP address in binary and dotted decimal formats.

Binary format	**Dotted decimal notation**
10000011 01101011 00000011 00011000	131.107.3.24

Binary to Decimal Notation

Topic Objective
To demonstrate the process of converting binary to decimal notation and decimal to binary notation.

Lead-in
Being able to convert binary to decimal notation assists you in understanding how IP addressing is done.



Binary Notation

Symbols 0-1 (2 possible symbols)

Binary Notation	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Decimal Values	128	64	32	16	8	4	2	1
Sample Number	0	0	0	1	1	0	0	1

Converting Binary to Decimal

11001 = 1x16 + 1x8 + 0x4 + 0x2 + 1x1 = 25

Delivery Tips
Use the graphic to first review decimal notation: number of symbols, positional system, calculation of the value based on number and position.

Then advance the Topic animation and make the comparison with binary notation. Point out that a binary system has only two symbols: 0 and 1 and because of this, binary is an ideal number system to represent electrical states: on and off.

Emphasize that each octet is converted separately.

Ask for and answer any questions. Ask students to complete the practice, and allow 10 minutes to complete the activity.

Knowing how to translate decimal to binary and binary to decimal is useful in deciphering an IP address. When using decimal notation, there are ten symbols available: 0 through 9. Each decimal number has a positional value expressed as 10^x power.

The principles of binary notation are the same as decimal, except that you are working with two symbols: 0 and 1, and each position in the number is 2^x in value. A bit that is set to 0 always has a zero value. A bit that is set to 1 can be converted to a decimal value.

The low-order bit, or the rightmost bit, represents a decimal value of one. The high-order bit, or the leftmost bit, represents a decimal value of 128. The highest decimal value of an octet is 255—that is, when all bits are set to 1.

The following table shows how the bits in one octet are converted from binary code to a decimal value.

Binary code	Bit values	Decimal value
00000000	0	0
00000001	1	1
00000011	2+1	3
00000111	4+2+1	7
00001111	8+4+2+1	15
00011111	16+8+4+2+1	31
00111111	32+16+8+4+2+1	63
01111111	64+32+16+8+4+2+1	127
11111111	128+64+32+16+8+4+2+1	255

Practice

1. Convert the following binary numbers to decimal format.

Binary value	Decimal value
10001011	**139**
10101010	**170**
10111111 11100000 00000111 10000001	**191.224.7.129**
01111111 00000000 00000000 00000001	**127.0.0.1**

2. Convert the following decimal values to binary format.

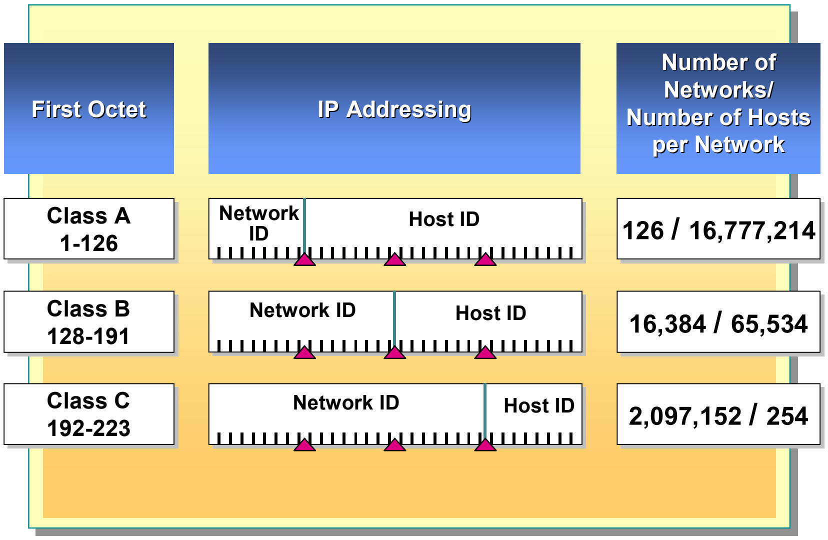
Binary value	Decimal value
250	**11111010**
19	**00010011**
109.128.255.254	**01101101 10000000 11111111 11111110**
131.107.2.89	**10000011 01101011 00000010 01011001**

Tip Use the calculator (scientific view) in the **Accessories** group to convert decimal format to binary format, and vice versa.

Address Classes

Topic Objective
To summarize the number of networks and number of hosts per network in each class, and to describe the range of network IDs in each class.

Lead-in
The 32-bit IP addressing scheme supports a total of 3,720,314,628 hosts.



Delivery Tips
Review the number of networks and the number of hosts per network in each class.

Discuss the network ID range for each address class.

Key Point
You can identify the class of address by the number in the first octet.

Because each IP address must be unique, IP addresses were initially distributed in groups or classes from a centralized organization called InterNIC. The classes were divided into several major groups based on size, in terms of the number of hosts per network. The value of the first octet determines which class the IP address belongs to.

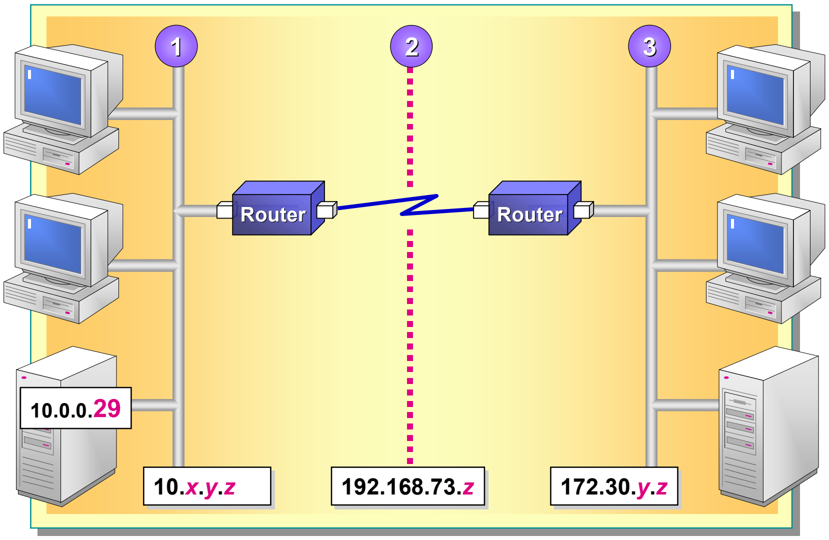
The size of your network determines the IP address class. You need to obtain enough IP addresses for each device on your network, including computers, printers, and other accessible devices. Because each IP address is made up of 32 bits, by using one octet for the network ID and the remaining three octets for the host IDs, you can have numerous hosts (16,777,214 or 2^{24} minus 2 addresses, one for a broadcast addresses and one for a default route) associated for a single network ID. The following table describes the different IP address classes.

Classes	Description
Class A	These addresses are assigned to networks with a very large number of hosts. This class allows for 126 networks, because it uses the first number for the network ID. The remaining three octets are used for the host ID, allowing for 16,777,214 hosts per network.
Class B	These addresses are assigned to networks that range from medium to large in size. This class allows for 16,382 networks, because it uses the first two octets for the network ID. The remaining two octets are used for the host ID, allowing for 65,534 hosts per network.
Class C	These addresses are used for small, local area networks (LANs). This class allows for approximately 2,097,152 networks, because it uses the first three octets for the network ID. The remaining octet is used for the host ID, allowing for 254 hosts per network.
Classes D and E	These are not allocated to hosts. Class D addresses are used for multicasting, and class E addresses are not available to users for general use. The first octet in a Class D address ranges from 224 to 239, and for a Class E address it is 240-255.

Assigning Network IDs

Topic Objective
To identify the network components that require a network ID.

Lead-in
A unique network ID is required for each network and wide area connection.



Delivery Tip
Use the graphic to show the three different network IDs. Ask which class each network belongs to.

Key Points
All hosts on a given network must have the same network ID to communicate with each other.

The network ID identifies the TCP/IP hosts that are located on the same physical network. All hosts on the same physical network must be assigned the same network ID to communicate with each other.

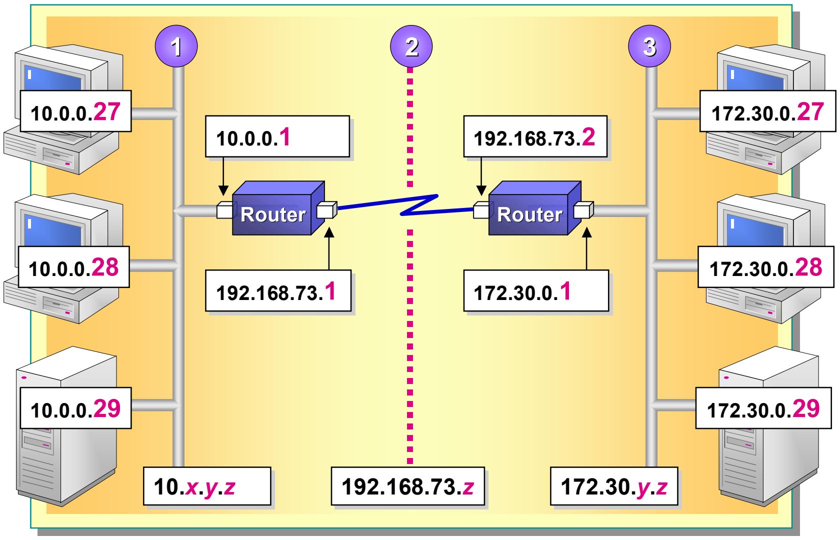
If your networks are connected by routers, a unique network ID is required for each wide area connection. For example, in the Topic above:

- Networks 1 and 3 represent two routed networks.
- Network 2 represents the wide area network (WAN) connection between the routers. Network 2 requires a network ID so that unique host IDs can be assigned to the interfaces between the two routers.

Assigning Host IDs

Topic Objective
To identify the TCP/IP hosts that require a host ID.

Lead-in
All TCP/IP hosts, including interfaces to routers, require unique host IDs.



Delivery Tip
Use the graphic to show the host IDs that are assigned to the different TCP/IP hosts.

Key Points
All TCP/IP hosts, including interfaces to routers, require unique host IDs.

The host ID of the router interface is the IP address configured as a workstation's default gateway when TCP/IP is installed.

The host ID identifies a TCP/IP host within a network and must be unique to the network ID. All TCP/IP hosts, including interfaces to routers, require unique host IDs.

The host ID of the router interface is the IP address configured as a workstation's default gateway when TCP/IP is installed. For example, for the host on network 1 with an IP address of 124.0.0.27, the IP address of the default gateway is 124.0.0.1.

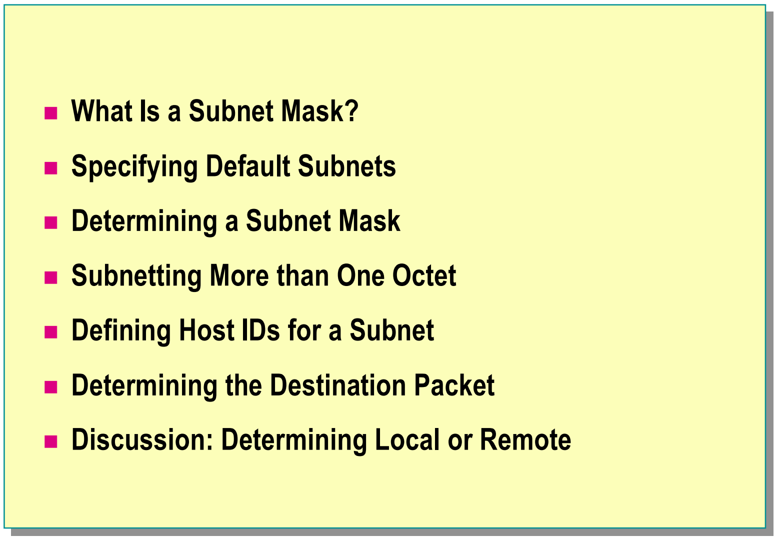
The following table lists the valid ranges of host IDs for a private network.

Address class	Beginning range	Ending range
Class A	*w*.0.0.1	*w*.255.255.254
Class B	*w.x*.0.1	*w.x*.255.254
Class C	*w.x.y*.1	*w.x.y*.254

# ◆ Defining Subnets

Topic Objective
To introduce the concept of subnetting.

Lead-in
Subnets enable a large number of computers to effectively communicate in a network.



- **What Is a Subnet Mask?**
- **Specifying Default Subnets**
- **Determining a Subnet Mask**
- **Subnetting More than One Octet**
- **Defining Host IDs for a Subnet**
- **Determining the Destination Packet**
- **Discussion: Determining Local or Remote**

Given a single network ID and the need to increase the overall size of the network, you can use devices such as routers or bridges to add new network segments. By dividing the network into segments instead of just adding additional hosts to the existing network, you increase the network's traffic efficiency. Network segments separated by routers are called *subnets*.

When you create subnets, you must break up the network ID for the hosts on the subnets. Dividing the network ID that is used to communicate on the Internet into smaller (based on the number of IP addresses identified) network IDs for a subnet is called *subnetting* a network.

Organizations use subnetting to apply one network across multiple physical segments. Thus, you can:

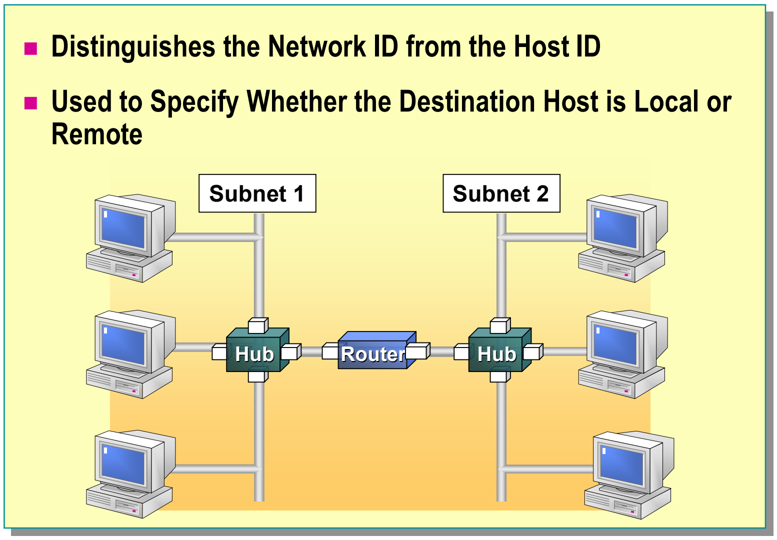
- Mix different media, such as Ethernet and token ring.
- Overcome limitations of current technologies, such as exceeding the maximum number of hosts per segment.
- Reduce network congestion by redirecting traffic and reducing broadcasts.

You can locate a host on a network by analyzing the host's network ID. Matching network IDs show which hosts are on the same subnet. If the network IDs are not the same, you know that they are on different subnets and that you need a router to establish communication between them.

What Is a Subnet Mask?

Topic Objective
To describe the function and purpose of a subnet mask.

Lead-in
A subnet mask is used to “mask” a portion of the IP address to distinguish the network ID from the host ID, so that TCP/IP can determine whether an IP address is located on a local or remote network.



- **Distinguishes the Network ID from the Host ID**
- **Used to Specify Whether the Destination Host is Local or Remote**

Key Points
Each host on a TCP/IP network requires a subnet mask.

A default subnet mask is used when a network is not divided into subnets.

A custom subnet mask is used when a network is divided into subnets.

Key Point
Point out that in creating a subnet, each subnet receives its own network ID. Also stress the important function of a router as the boundary between subnets.

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

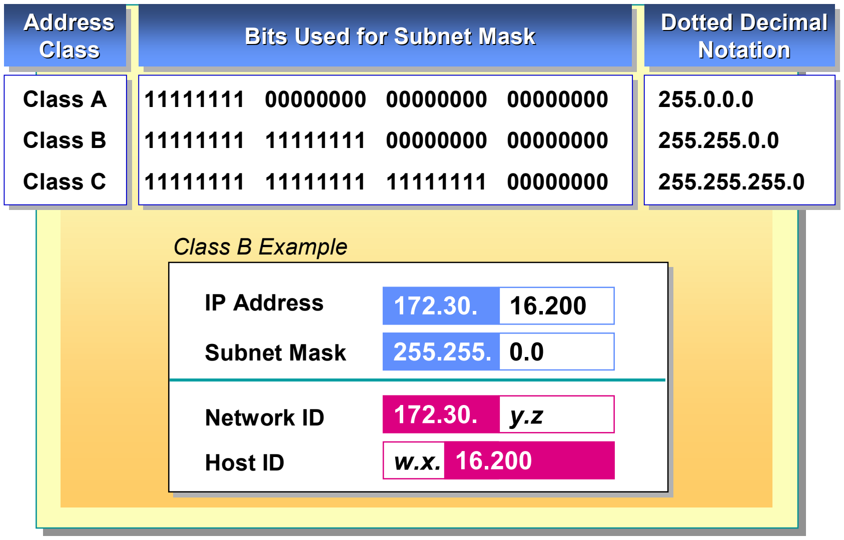
Each host on a TCP/IP network requires a subnet mask—either a default subnet mask, which is used when a network is not divided into subnets, or a custom subnet mask, which is used when a network is divided into subnets.

Dividing the network into subnets requires that each segment use a different network ID. A unique network ID is created for each segment by partitioning the bits in the host ID into two parts. One part is used to identify the segment as a unique network, and the other part is used to identify the hosts.

Specifying Default Subnets

Topic Objective
To define the default subnet masks that are used when networks are divided into subnets.

Lead-in
A default subnet mask is used on TCP/IP networks that are not divided into subnets.



Address Class	Bits Used for Subnet Mask	Dotted Decimal Notation
Class A	11111111 00000000 00000000 00000000	255.0.0.0
Class B	11111111 11111111 00000000 00000000	255.255.0.0
Class C	11111111 11111111 11111111 00000000	255.255.255.0

Class B Example

IP Address	172.30.	16.200
Subnet Mask	255.255.	0.0
Network ID	172.30.	*y.z*
Host ID	*w.x.*	16.200

Delivery Tips
Use the graphic to show the default subnet masks.

Explain how the bits are set in each class.

A default subnet mask is used on TCP/IP networks that are not divided into subnets. All TCP/IP hosts require a subnet mask, even on a single-segment network. The default subnet mask that you use depends on the address class.

All bits that correspond to the network ID are set to 1. The decimal value in each octet is 255. All bits that correspond to the host ID are set to 0.

Every address class has a default subnet mask. The following table lists the default subnet masks for each address class.

IP address class	IP address	Subnet mask	Network ID	Host ID
A	w.x.y.z	255.0.0.0	w.0.0.0	x.y.z
B	w.x.y.z	255.255.0.0	w.x.0.0	x.y
C	w.x.y.z	255.255.255.0	w.x.y.0	z

In classful IP addresses, each of the four octets can assume only the maximum value 255 or the minimum value 0. In a subnet mask, the four numbers are arranged as contiguous maximum values followed by contiguous minimum values.

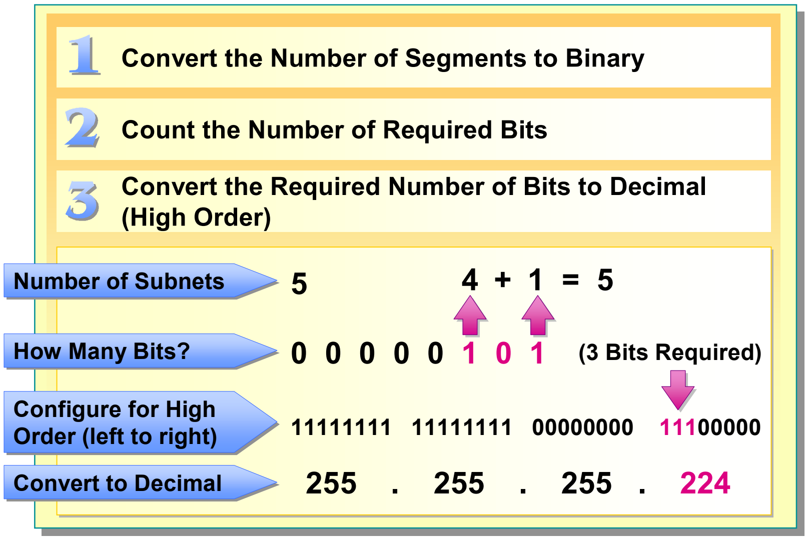
The maximum values represent the network ID and the minimum values represent the host ID. For example, 255.255.0.0 is a valid subnet mask (11111111 11111111 00000000 00000000), whereas 255.0.255.0 is not (11111111 00000000 11111111 00000000). In the second example, the maximum values or the 1s, are not contiguous.

For a class B address, the default subnet mask 255.255.0.0 identifies the network ID as the first two numbers in the IP address.

Determining a Subnet Mask

Topic Objective
To explain the process of defining a subnet mask.

Lead-in
Defining a subnet mask is a three-step process.



1 Convert the Number of Segments to Binary

2 Count the Number of Required Bits

3 Convert the Required Number of Bits to Decimal (High Order)

Number of Subnets 5 4 + 1 = 5

How Many Bits? 0 0 0 0 0 1 0 1 (3 Bits Required)

Configure for High Order (left to right) 11111111 11111111 00000000 11100000

Convert to Decimal 255 . 255 . 255 . 224

Delivery Tip
Use the graphic to explain the process of defining a subnet mask.

Determining a subnet mask is required if you are dividing your network into subnets. For example, you have a Class C address with 256 hosts that needs to be divided into five subnets.

Follow these steps to determine a subnet mask:

1. Once you have determined the number of physical segments in your network environment, convert this number to binary format.
2. Count the number of bits required to represent the number of physical segments in binary. For example, if you need five subnets, the binary value is 101. Representing five in binary requires three bits.
3. Convert the required number of three bits to decimal format in high order (from left to right). For example, if three bits are required, configure the first three bits of the host ID as the subnet ID. The bits must be contiguous, therefore the octet is 11100000. The decimal value for binary 11100000 is 224. The subnet mask is 255.255.255.224 (for a class C address).

Contiguous Mask Bits

Because subnets are defined by the subnet mask, there is nothing to prevent an administrator from using low-order or unordered bits to determine the subnet ID. When subnetting was initially defined, it was recommended that subnet IDs be derived from high-order bits. Today, however, few router vendors support the use of low-order or non-order bits in subnet IDs. Furthermore, it is now a requirement that the subnet ID make use of contiguous, high-order bits of the local address portion of the subnet mask.

Conversion Tables

The following table lists the subnet masks already converted using one octet for class A networks.

Number of subnets	Required number of bits	Subnet mask	Number of hosts per subnet
0	1	Invalid	Invalid
2	2	255.192.0.0	4,194,302
6	3	255.224.0.0	2,097,150
14	4	255.240.0.0	1,048,574
30	5	255.248.0.0	524,286
62	6	255.252.0.0	262,142
126	7	255.254.0.0	131,070
254	8	255.255.0.0	65,534

The following table lists the subnet masks already converted using one octet for class B networks.

Number of subnets	Required number of bits	Subnet mask	Number of hosts per subnet
0	1	Invalid	Invalid
2	2	255.255.192.0	16,382
6	3	255.255.224.0	8,190
14	4	255.255.240.0	4,094
30	5	255.255.248.0	2,046
62	6	255.255.252.0	1,022
126	7	255.255.254.0	510
254	8	255.255.255.0	254

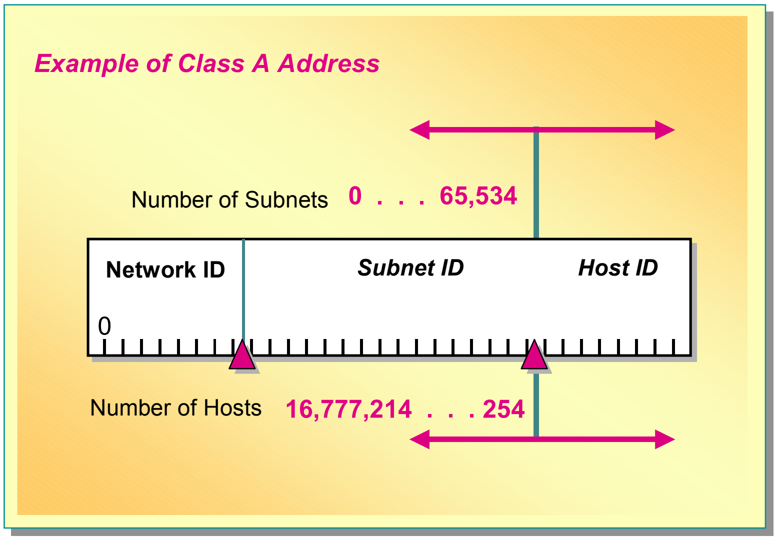
The following table lists the subnet masks already converted using one octet for class C networks.

Required Number of subnets	Required number of bits	Subnet mask	Number of hosts per subnet
Invalid	1	Invalid	Invalid
1–2	2	255.255.255.192	62
3–6	3	255.255.255.224	30
7–14	4	255.255.255.240	14
15–30	5	255.255.255.248	6
31–62	6	255.255.255.252	2
Invalid	7	Invalid	Invalid
Invalid	8	Invalid	Invalid

Subnetting More than One Octet

Topic Objective
To demonstrate subnetting more than one octet, or eight bits, for greater addressing flexibility.

Lead-in
Up to this point, we have worked within one octet to define a custom subnet mask. At times, it may be advantageous to subnet using more than one octet, or more than eight bits.



At times, it may be advantageous to subnet using more than one octet, or more than eight bits.

For example, suppose you are on a team that is responsible for configuring an intranet for a large corporation. The corporation plans to internally connect its sites that are distributed across Europe, North America and Asia. This totals approximately 30 geographical locations with almost 1,000 subnets and an average of 750 hosts per subnet.

It is possible to use several class B network IDs and further subnet them. To meet the host requirements per subnet with a class B network address, you would need to use a subnet mask of 255.255.252.0. Further adding the requirement of subnets, you would need at least 16 class B addresses.

However, there is an easier way. Because you are on an intranet, you can use a private network. If you choose to allocate a class A network ID of 10.0.0.0, you can plan for growth and meet the requirements at the same time. However, subnetting only the second octet will not meet the requirements of one thousand subnets. However, if you subnet both the second octet and a portion of the third octet, you can meet all the requirements with one network ID. The following table is an example of a possible subnet mask and binary representation.

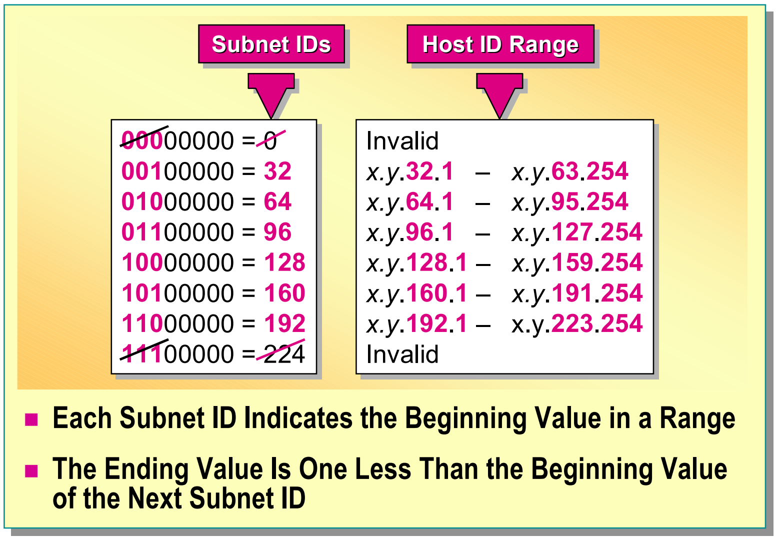
Network ID	Subnet mask	Subnet mask (binary)
10.0.0.0	255.255.248.0	1111111111 11111111 11111000 00000000

Using 13 bits for the subnet ID in a class A address, you have allocated 8,190 subnets, each with up to 2,046 hosts. You have met the requirements with flexibility for growth.

Defining Host IDs for a Subnet

Topic Objective
To define a range of host IDs for a subnet using the subnet ID.

Lead-in
If you have defined your subnet IDs, then you have already defined your host IDs for each subnet.



Subnet IDs

~~00~~000000 = ~~0~~
00100000 = 32
01000000 = 64
01100000 = 96
10000000 = 128
10100000 = 160
11000000 = 192
~~11~~100000 = ~~224~~

Host ID Range

Invalid
x.*y*.32.1 – *x*.*y*.63.254
x.*y*.64.1 – *x*.*y*.95.254
x.*y*.96.1 – *x*.*y*.127.254
x.*y*.128.1 – *x*.*y*.159.254
x.*y*.160.1 – *x*.*y*.191.254
x.*y*.192.1 – x.y.223.254
Invalid

- **Each Subnet ID Indicates the Beginning Value in a Range**
- **The Ending Value Is One Less Than the Beginning Value of the Next Subnet ID**

Delivery Tip
Use the graphic to show students the range of host IDs for each subnet.

Once you have determined the possible subnet IDs, the result of each incremented value indicates the beginning of a range of host IDs for that subnet. If you increment the value one extra time, you can determine the end of the range (one less than the subnet mask).

Note Some subnet IDs are invalid because they are reserved for special purposes. An ID with all ones (1) is reserved for broadcast. An ID with all zeros (0) is used to indicate a default route.

The following table shows the valid range of host IDs on a class B subnet using 3 bits for the subnet mask.

Bit values	Decimal value	Beginning range value	Ending range value
000	0	Invalid	Invalid
001	32	*x*.*y*.32.1	*x*.*y*.63.254
010	64	*x*.*y*.64.1	*x*.*y*.95.254
011	96	*x*.*y*.96.1	*x*.*y*.127.254
100	128	*x*.*y*.128.1	*x*.*y*.159.254
101	160	*x*.*y*.160.1	*x*.*y*.191.254
110	192	*x*.*y*.192.1	*x*.*y*.223.254
111	224	Invalid	Invalid

To determine the number of hosts per subnet:

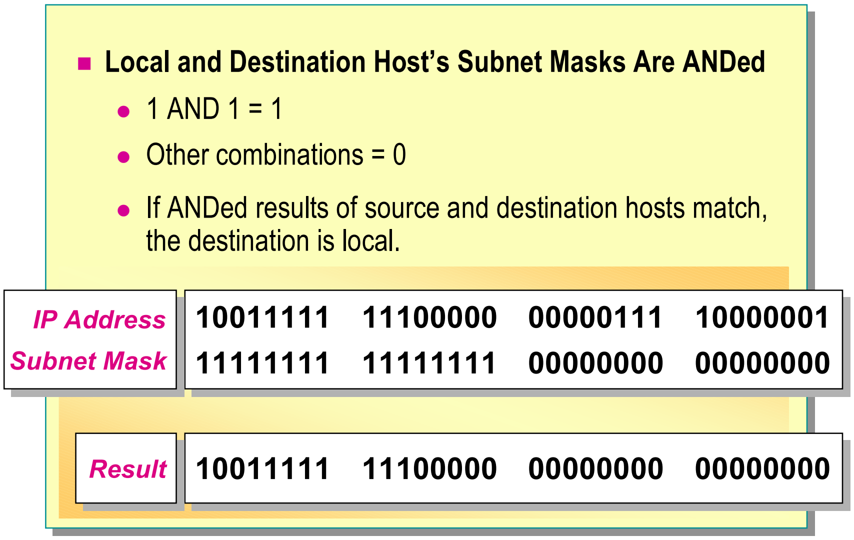
1. Calculate the number of bits available for the host ID. For example, if you are given a class B address that uses 16 bits for the network ID and 2 bits for the subnet ID, you have 14 bits remaining for the host ID.
2. Convert the binary host ID bits to decimal. For example, 11111111111111 in binary is converted to 16,383 in decimal format.
3. Subtract 1 for the mask itself.

Tip If you know the number of host ID bits you need, you can raise 2 to the power of the number of host ID bits, and then subtract 2.

Determining the Destination Packet

Topic Objective
To explain the ANDing process that IP uses to determine whether a packet is destined for a local or remote host.

Lead-in
ANDing is the internal process that IP uses to determine whether a packet is destined for a host on a local or remote network.



- Local and Destination Host's Subnet Masks Are ANDed
  - 1 AND 1 = 1
  - Other combinations = 0
  - If ANDed results of source and destination hosts match, the destination is local.

IP Address	10011111	11100000	00000111	10000001
Subnet Mask	11111111	11111111	00000000	00000000
Result	10011111	11100000	00000000	00000000

Delivery Tips
Use the graphic to explain the ANDing process.

Give students ten minutes to practice the ANDing process.

Ask students the following questions:
1. Do the results match? No
2. Is the destination IP address located on a local or remote network? Remote

Emphasize that ANDing is a process that is used internally by IP and not a process that you would normally have to do.

ANDing is the internal process that TCP/IP uses to determine whether a packet is destined for a host on a local network or a remote network.

When TCP/IP is initialized, the host's IP address is ANDed with its subnet mask. Before a packet is sent, the destination IP address is ANDed with the same subnet mask. If both results match, IP knows that the packet belongs to a host on the local network. If the results don't match, the packet is sent to the IP address of an IP router.

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

Bit combination	Result
1 AND 1	1
1 AND 0	0
0 AND 0	0
0 AND 1	0

Practice

AND the following IP addresses to determine whether the destination IP address belongs to a host on a local network or a remote network.

Source (host) IP address	10011001 10101010 00100101 10100011
Subnet mask	11111111 11111111 00000000 00000000
Result	**1001100110101010**

Destination IP address	11011001 10101010 10101100 11101001
Subnet mask	11111111 11111111 00000000 00000000
Result	**1101100110101010**

1. Do the results match?

   No.

2. Is the destination IP address located on a local or remote network?

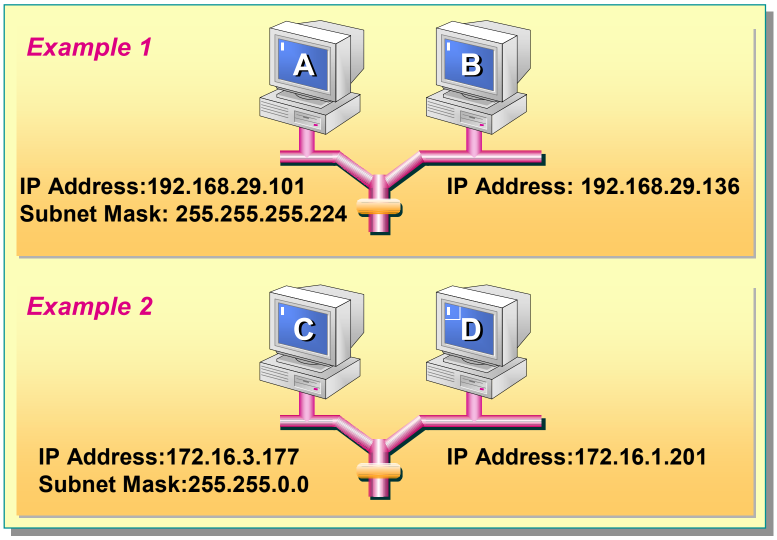
   Remote.

Note ANDing is a process that is used internally by IP and not a process that you would normally do.

Discussion: Determining Local or Remote Hosts

Topic Objective
To practice determining if a hosts are local or remote with respect to each other.

Lead-in
Let's look at some examples and discuss TCP/IP concepts.



Example 1

IP Address:192.168.29.101
Subnet Mask: 255.255.255.224

IP Address: 192.168.29.136

Example 2

IP Address:172.16.3.177
Subnet Mask:255.255.0.0

IP Address:172.16.1.201

Consider the example hosts on the Topic and answer the following questions.

Delivery Tip
Call on students to answer the questions provided. Ask for and answer other questions regarding IP addressing. Keep the discussion focused on identification of host ID, network ID, and determining if hosts are remote or local to each other. This discussion is not intended to provide addressing strategies for various network configurations.

Example 1

1. What IP address class do Computers A and B belong to?

   Class C

2. What can you assume about the size of the network in Example 1 and how did you arrive at your answer?

   A small network; based on the first octet of the IP address.

3. For Computer A, which octets make up the Host ID and which octets make up the Network ID?

   The Network ID is defined by the first three octets and the first three binary numbers from the fourth octet. The Host ID is defined by the remaining five binary numbers from the fourth octet.

4. Are Computers A and B local to each other or remote to each other?

   Remote.

Example 2

1. What IP address class do Computers C and D belong to?

   Class B.

2. What can you assume about the size of the network in Example 2 and how did you arrive at your answer?

   A medium-sized network; based on the first octet of the IP address.

3. For Computer C, which octets make up the Host ID and which octets make up the Network ID?

   Network ID is defined by the first two octets
   Host ID is defined by the last two octets.

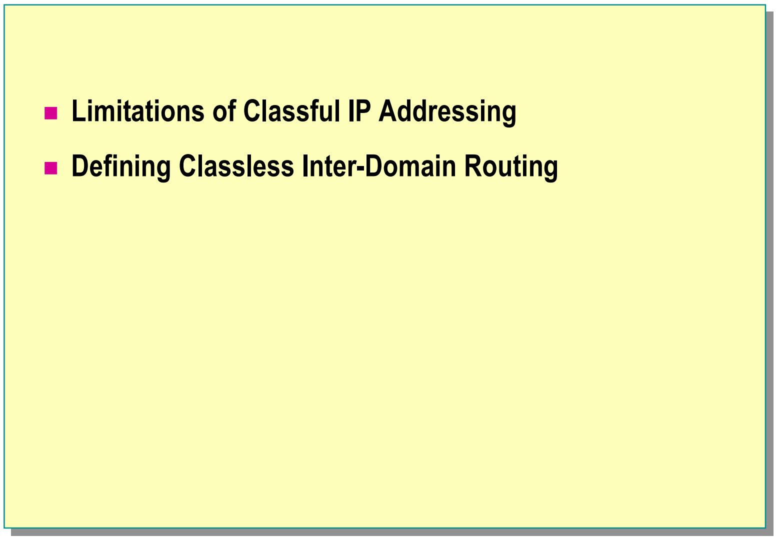
4. Are Computers C and D local to each other or remote to each other?

   Local.

# ◆ Using Classless Inter-Domain Routing

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

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



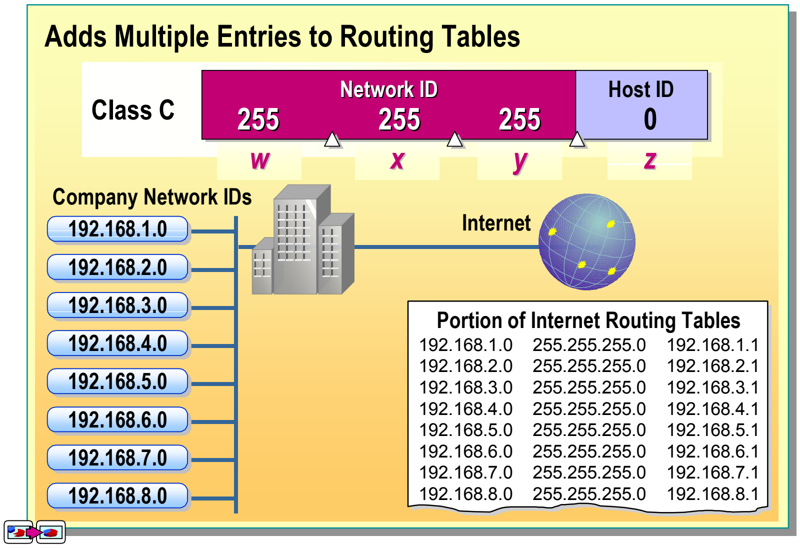
- **Limitations of Classful IP Addressing**
- **Defining Classless Inter-Domain Routing**

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.

Limitations of Classful IP Addressing

Topic 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.



Adds Multiple Entries to Routing Tables

Class C

Network ID: 255 255 255

Host ID: 0

w x y z

Company Network IDs

192.168.1.0
192.168.2.0
192.168.3.0
192.168.4.0
192.168.5.0
192.168.6.0
192.168.7.0
192.168.8.0

Internet

Portion of Internet Routing Tables

192.168.1.0	255.255.255.0	192.168.1.1
192.168.2.0	255.255.255.0	192.168.2.1
192.168.3.0	255.255.255.0	192.168.3.1
192.168.4.0	255.255.255.0	192.168.4.1
192.168.5.0	255.255.255.0	192.168.5.1
192.168.6.0	255.255.255.0	192.168.6.1
192.168.7.0	255.255.255.0	192.168.7.1
192.168.8.0	255.255.255.0	192.168.8.1

Delivery Tip
Use the animated Topics 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. These limitations are further addressed in Module 9, "Examining Web Services," in Course 2151A, *Microsoft Windows 2000 Network and Operating System Essentials*.

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 almost 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.

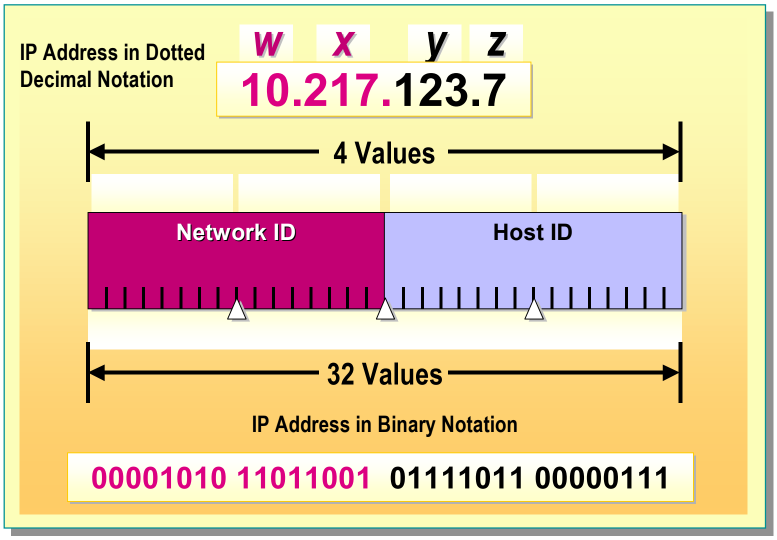
Depletion of All IP Addresses

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 Classless Inter-Domain Routing

Topic 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 Topic 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

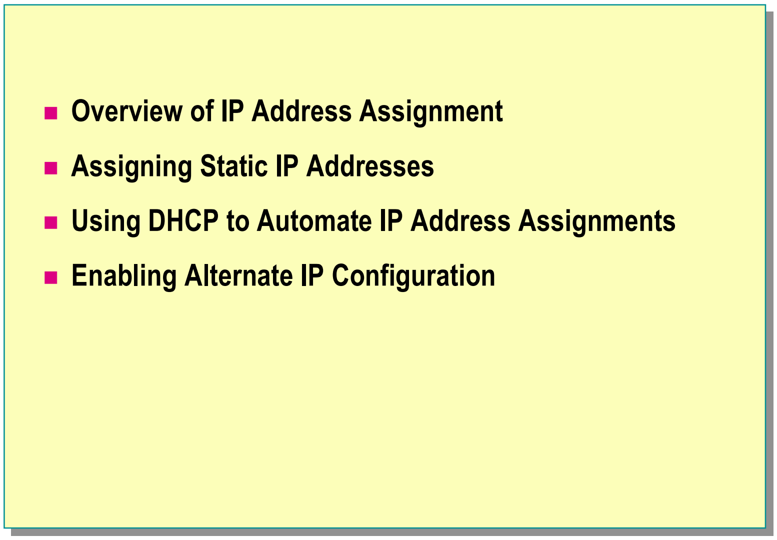
CIDR translates all IP address and subnet masks to binary notation. CIDR divides an IP address into a set of 32 values, in place of the four values used in the classful system. This division allows for more variations in network size and optimizes the allocation of IP addresses. By using CIDR, not many 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.

# ◆ Configuring IP Addresses

Topic Objective
To introduce methods for assigning IP addresses.

Lead-in
Windows XP Professional provides two methods for assigning IP addresses.



- **Overview of IP Address Assignment**
- **Assigning Static IP Addresses**
- **Using DHCP to Automate IP Address Assignments**
- **Enabling Alternate IP Configuration**

Windows XP Professional provides two methods for assigning IP addresses to devices on TCP/IP networks:

- Dynamic addressing by using Dynamic Host Configuration Protocol (DHCP) to assign an IP address.
- Static or manual addressing by physically entering the IP address at the client computer.

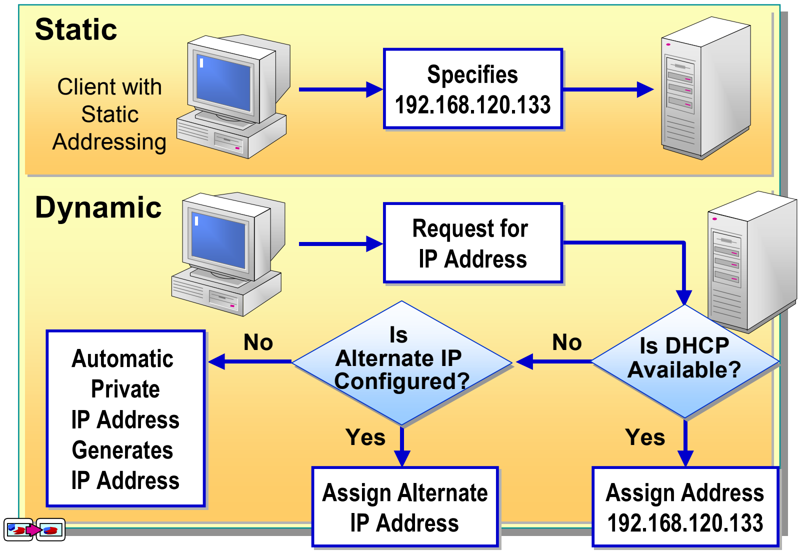
Windows XP Professional provides a new feature for clients using DHCP. If dynamic addressing is enabled on the client computer, you can specify an alternate address for the same client. If the DHCP server is unavailable, the client computer will use the alternate address to connect to the network. This feature can be used by portable computers to easily switch between an environment in which DHCP is available (for example, in your office) and one in which DHCP may not be available (for example, a home Internet Service Provider [ISP]).

After you set the IP address, you can view its TCP/IP configuration by using either the **Internet Protocol (TCP/IP) Properties** dialog box or the **ipconfig** command.

Overview of IP Address Assignment

Topic Objective
To introduce the processes for assigning TCP/IP addresses.

Lead-in
After the network and host IDs are formulated, the next step is to configure the computers accordingly.


Static
Client with Static Addressing
Specifies 192.168.120.133
Dynamic
Request for IP Address
Is DHCP Available?
No
Yes
Assign Address 192.168.120.133
Is Alternate IP Configured?
No
Yes
Assign Alternate IP Address
Automatic Private IP Address Generates IP Address

Selecting the method of IP address assignment depends on your environment and client requirements. There are two methods of assignment: static and dynamic.

If a DHCP server is not available and communication with hosts outside of a single subnet is required, you must use static addressing. If a DHCP server is available, dynamic addressing is generally preferred.

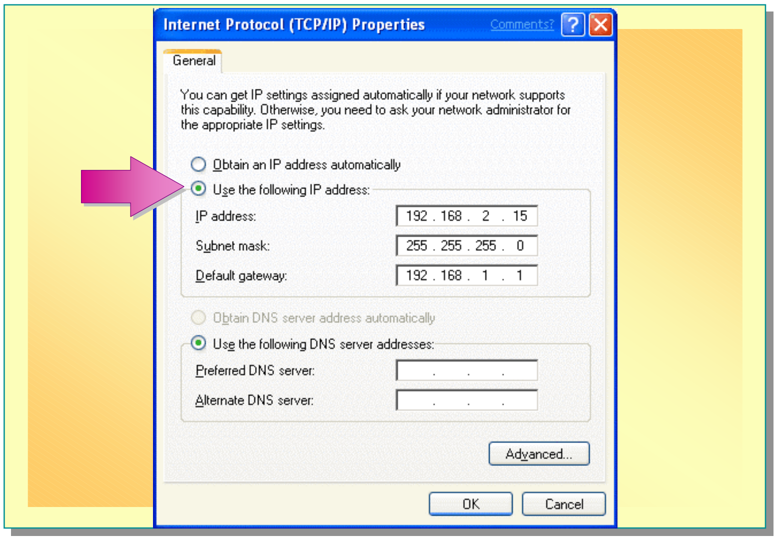
Dynamic addressing is the default addressing method in Windows XP Professional. In dynamic addressing, The DHCP server supplies an IP address to the client. If DHCP is unavailable, an alternate address is provided depending on the client configurations found in the **Internet Protocol (TCP/IP) Properties** dialog box. If an address is provided in the **Alternate static address** option, this will be the address Windows XP Professional uses to communicate. If an alternate static address is not provided, the **Automatic Private IP Addressing** option provides automatic IP address assignments for computers on networks without DHCP servers. The client is assigned an IP address from a reserved class B network.

Note With automatic address assignment, the client cannot communicate with hosts that are outside of the local subnet, including Internet hosts.

Assigning Static IP Addresses

Topic Objective
To illustrate static IP addressing.

Lead-in
You can assign IP addresses manually by using the **Internet Protocol (TCP/IP) Properties** dialog box.



Internet Protocol (TCP/IP) Properties

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically
● Use the following IP address:

IP address:	192 . 168 . 2 . 15
Subnet mask:	255 . 255 . 255 . 0
Default gateway:	192 . 168 . 1 . 1

○ Obtain DNS server address automatically
● Use the following DNS server addresses:

Preferred DNS server:	. . .
Alternate DNS server:	. . .

Delivery Tip
Demonstrate how to use the **Internet Protocol (TCP/IP) Properties** dialog box and explain how to choose the option to set addresses statically.

Static IP addressing refers to configuring IP addresses manually. In this method, you use a utility provided by Windows XP Professional to assign an IP address. Windows XP Professional provides the **Internet Protocol (TCP/IP) Properties** dialog box to manually assign an IP address to a TCP/IP host or device.

To manually configure the IP address:

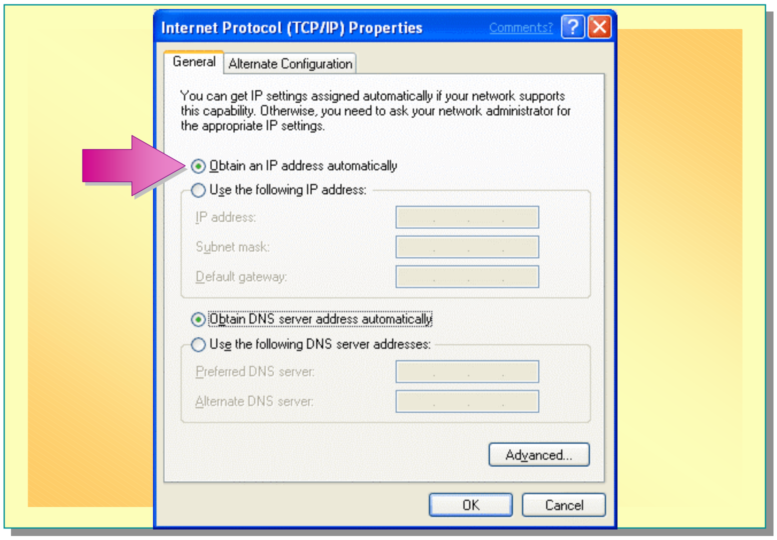
1. On the **Start** menu, click **Control Panel**, click **Network and Internet Connections**, and then click **Network Connections**.
2. Right-click **Local Area Connection**, and then click **Properties**.
3. On the **Local Area Connection Properties** dialog box, click **Internet Protocol (TCP/IP)**, and then click **Properties**
4. On the **Internet Protocol (TCP/IP) Properties** dialog box, click **Use the following IP address** to enter values for the IP address, Subnet mask, and Default gateway, and then click **OK** twice.

Note In general, most computers have only one network adapter installed and therefore require only a single IP address. For devices with multiple network adapters installed, such as a router, each adapter needs its own IP address.

Using DHCP to Automate IP Address Assignments

Topic Objective
To illustrate automatic IP addressing.

Lead-in
By default, Windows XP Professional is configured to obtain an IP address automatically by using DHCP.



Internet Protocol (TCP/IP) Properties

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

● Obtain an IP address automatically

○ Use the following IP address:

IP address: . . .

Subnet mask: . . .

Default gateway: . . .

● Obtain DNS server address automatically

○ Use the following DNS server addresses:

Preferred DNS server: . . .

Alternate DNS server: . . .

Delivery Tip
Demonstrate the **Internet Protocol (TCP/IP) Properties** dialog box and point out the automatic options. The students' computers should be set to automatic if the standard classroom setup is used.

By default, Windows XP Professional is configured to obtain an IP address automatically by using Dynamic Host Configuration Protocol (DHCP).

To use DHCP in a network, hosts in the network must be configured to use DHCP. To enable DHCP, you must click **Obtain an IP address automatically**. If you need to change a host from static to dynamic addressing:

1. On the **Start** menu, click **Control Panel**, click **Network and Internet Connections**, and then click **Network Connections**.
2. Right-click **Local Area Connection**, and then click **Properties**.
3. On the **Local Area Connection Properties** dialog box, click **Internet Protocol (TCP/IP)**, and then click **Properties**.
4. On the **Internet Protocol (TCP/IP) Properties** dialog box, click **Obtain an IP address automatically**, and then click **OK** twice.

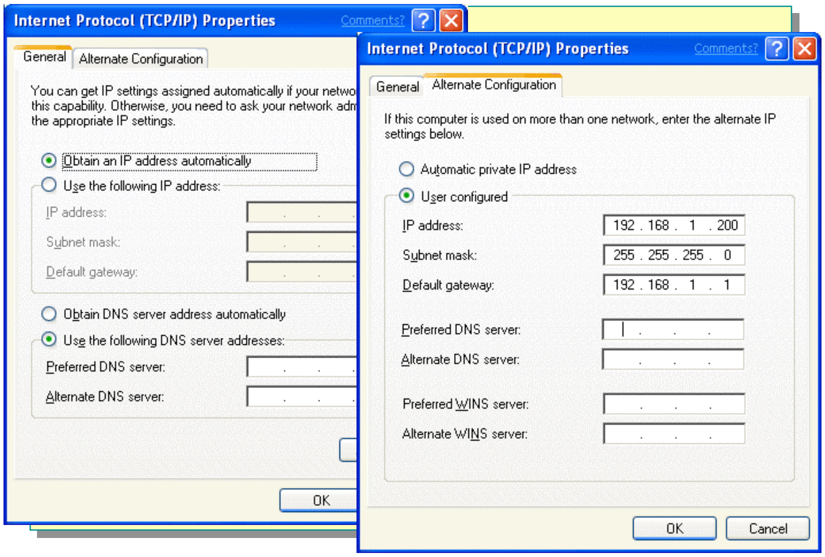
DHCP reduces the complexity and amount of administrative work involved in reconfiguring computers in TCP/IP-based networks. Without DHCP, when you move a computer from one subnet to another, you must change its IP address to reflect the new network and host ID. DHCP enables you to automatically assign an IP address to a host, from a database of addresses assigned to the subnet. Also, when a computer is offline for a specific amount of time, DHCP can reassign its IP address.

If you select dynamic addressing, you can also specify an alternate IP address in the event that DHCP is unavailable. Once you select **Obtain and IP address automatically**, the Alternate Configuration tab becomes available.

Enabling Alternate IP Configuration

Topic Objective
To introduce alternate IP addresses.

Lead-in
Alternate configuration is a new feature for Windows XP Professional enabling users with portable computers to specify a secondary IP configuration for different locations.



Internet Protocol (TCP/IP) Properties

General | Alternate Configuration

You can get IP settings assigned automatically if your netwo
this capability. Otherwise, you need to ask your network adm
the appropriate IP settings.

● Obtain an IP address automatically
○ Use the following IP address:
IP address:
Subnet mask:
Default gateway:

○ Obtain DNS server address automatically
● Use the following DNS server addresses:
Preferred DNS server:
Alternate DNS server:

Internet Protocol (TCP/IP) Properties

General | Alternate Configuration

If this computer is used on more than one network, enter the alternate IP settings below.

○ Automatic private IP address
● User configured
IP address: 192 . 168 . 1 . 200
Subnet mask: 255 . 255 . 255 . 0
Default gateway: 192 . 168 . 1 . 1
Preferred DNS server:
Alternate DNS server:
Preferred WINS server:
Alternate WINS server:

Specifying a secondary static address is ideal for users with portable computers that use DHCP at one location but are required to use a static address at another location. This feature enables a portable computer to operate seamlessly on both networks without manual TCP/IP reconfiguration.

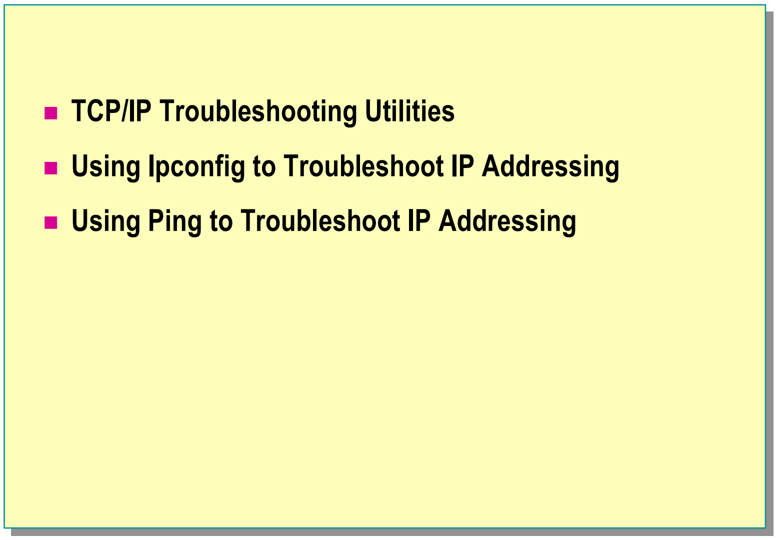
On the **Internet Protocol (TCP/IP) Properties** dialog box, on the **Alternate Configuration** tab, the options for alternative configuration are:

- **Automatic private IP address**. This option determines an address in the reserved for private IP-addressing class that ranges from 169.254.0.1 through 169.254.255.254. This address is used until a DHCP server is located. With this option enabled, DNS, WINS, or a default gateway are not assigned because automatic private IP addressing is designed only for a small network that consists of a single subnet.
- **User configured**. This option causes TCP/IP to use the static IP address on the **Alternate Configuration** tab.

# ◆ Troubleshooting IP Addresses

Topic Objective
To introduce TCP/IP utilities used for troubleshooting IP address problems.

Lead-in
Windows XP Professional comes with a number of TCP/IP utilities that provide network and connectivity functions.



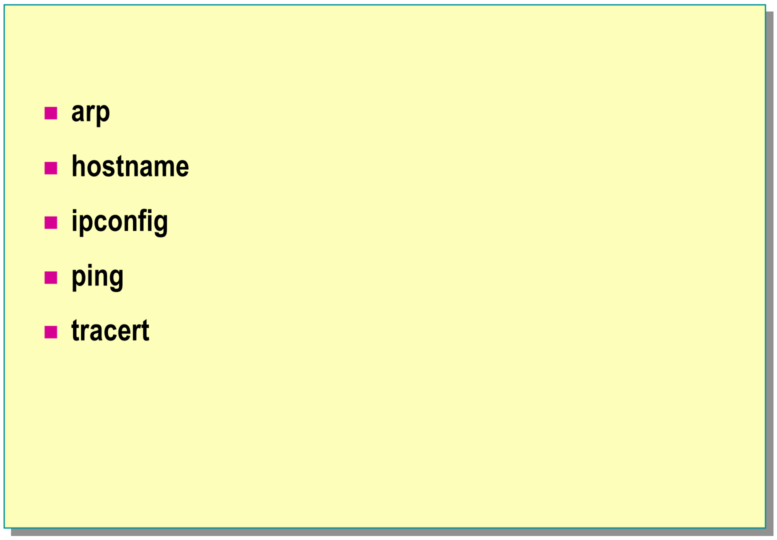
- TCP/IP Troubleshooting Utilities
- Using Ipconfig to Troubleshoot IP Addressing
- Using Ping to Troubleshoot IP Addressing

Microsoft packages TCP/IP with a set of utilities to provide a set of basic services for communications between computers, and for connections between networks. Using these utilities will assist you in performing management and troubleshooting tasks for your Windows XP Professional-based computer.

Using TCP/IP Troubleshooting Utilities

Topic Objective
To introduce the types of diagnostic TCP/IP utilities.

Lead-in
A number of TCP/IP diagnostic utilities come with Windows XP Professional.



- **arp**
- **hostname**
- **ipconfig**
- **ping**
- **tracert**

Windows XP Professional provides a number of TCP/IP diagnostic utilities that enable users to detect and resolve networking problems. Some of the common diagnostic utilities are:

- *arp*. Displays and modifies the Address Resolution Protocol (ARP) cache. Type **arp -a** at the command prompt to display the information in your ARP cache.
- *hostname*. Displays the host name of your computer. To gain access, type **hostname** at the command prompt.
- *ipconfig*. Displays and updates the current TCP/IP configuration, including the IP address. To gain access, type **ipconfig /all** at the command prompt to produce a detailed configuration report for all interfaces.
- *ping*. Tests IP connectivity between two computers. **ping** sends an ICMP (Internet Control Message Protocol) request from the source computer, and the destination computer responds with an ICMP reply. To test connectivity by using an IP address or computer name, type **ping** *IP_address* or type **ping** *computer_name* at a command prompt.

  To test the TCP/IP configuration of your own computer, you use local loopback. *Local loopback* is the IP address 127.0.0.1. To test system configuration by using local loopback, type **ping 127.0.0.1**
- *tracert*. Traces the route that a packet takes to a destination. The **tracert** command displays a list of IP routers that are used to deliver packets from your computer to the destination, and the amount of time that the packet remained at each hop or the destination between two routers. If the packets cannot be delivered to the destination, you can use the **tracert** command to identify the last router that successfully forwarded the packets.

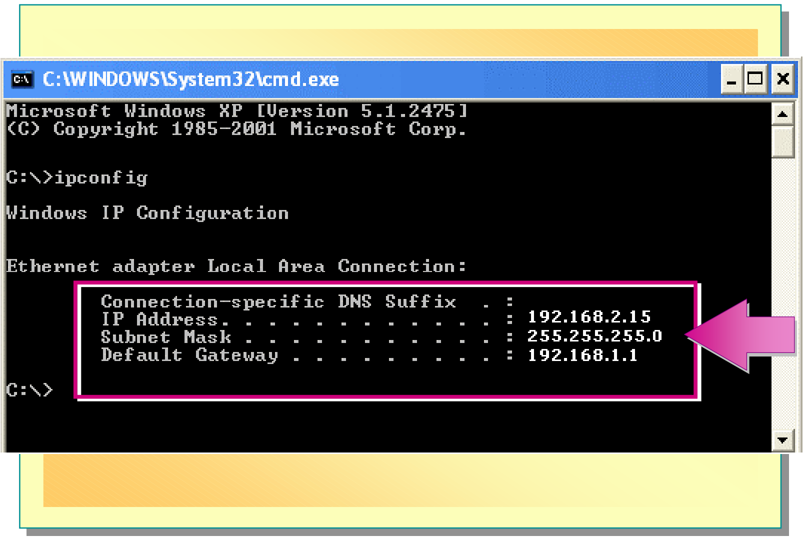
Using ipconfig to Troubleshoot IP Addressing

Topic Objective

To illustrate the syntax of the **ipconfig** command for viewing TCP/IP information.

Lead-in

After you set a TCP/IP configuration on a computer, you can view the configuration at any time.



```
Microsoft Windows XP [Version 5.1.2475]
(C) Copyright 1985-2001 Microsoft Corp.

C:\>ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : 192.168.2.15
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 192.168.1.1

C:\>
```

Delivery Tip

Demonstrate the procedure to gain access to the **ipconfig** command. Emphasize to the students that the **ipconfig** command cannot be used to set the TCP/IP configuration. Question the students about the differences between the information displayed by using **ipconfig** and the **Internet Protocol (TCP/IP) Properties** dialog box. These differences will be the focus of the following lab.

Windows XP Professional provides **ipconfig** to view TCP/IP information. The **ipconfig** command is used to verify, but not set, the TCP/IP configuration options on a host, including the IP address, subnet mask, and default gateway.

To start the **ipconfig** command, type **ipconfig** at the command prompt. The values of the three primary configuration parameters are displayed. However, if you use this command, you cannot determine whether the static or dynamic method has been used to assign the IP address. View the TCP/IP Properties dialog box to determine whether static or dynamic method is used to assign IP addresses.

You can obtain more information on the TCP/IP configuration settings, type **ipconfig** /**all** at the command prompt. The screen displays the information about all TCP/IP configuration options. Using this command, you can determine whether DHCP is enabled at the client computer or not.

A DHCP server leases an IP address to a client for a specific length of time. The Lease Obtained and Lease Expires labels display information about when the lease was obtained and when it will expire, respectively.

Additional commands useful for troubleshooting an IP address are:

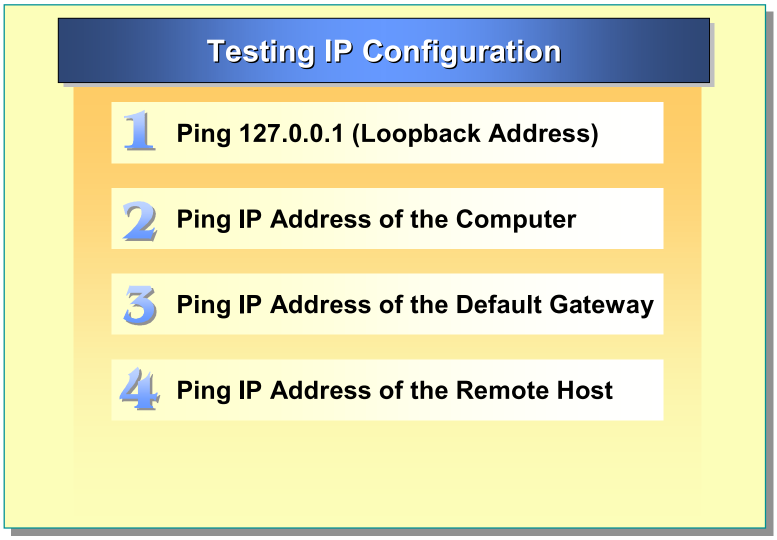
- **ipconfig /release**. This releases all connections for the computer's adapter.
- **ipconfig /renew**. This renews the connections for the computer's adapter according to the **Internet Protocol (TCP/IP) Properties** dialog box.

These commands are useful when moving from a static address to a dynamic address with DHCP. The release command releases the static address from the adapter, and the renew command sends a request to DHCP to assign an address.

Using ping to Troubleshoot IP Addressing

Topic Objective
To list the steps that are performed when testing IP configuration.

Lead-in
You can use the **ping** command to test TCP/IP connectivity to other host computers on the network.



Testing IP Configuration

1. Ping 127.0.0.1 (Loopback Address)
2. Ping IP Address of the Computer
3. Ping IP Address of the Default Gateway
4. Ping IP Address of the Remote Host

The **ping** command is a diagnostic tool that you can use to test TCP/IP configuration and diagnose connection failures by sending an Internet Control Message Protocol (ICMP) Echo Request to a target host name or IP address. Use the **ping** command to determine whether a particular TCP/IP host is available and functional.

Testing Network Connections

To verify that a route exists between the local computer and a network host, at a command prompt, type **ping** *IP_ address* (where *IP_address* is the IP address of the network host to which you want to connect). By default, the following message appears four times after a successful **ping** command:

```
Reply from IP_address
```

Delivery Tip
Demonstrate the result of successful and unsuccessful **ping** commands by typing the following commands: **ping 127.0.0.1** (which is successful) and **ping 192.168.0.108** (which is not successful).

Testing TCP/IP Configuration and Connections

Perform the following tasks to test TCP/IP configuration and connections:

1. Use the **ping** command with the loopback address (**ping 127.0.0.1**) to verify that TCP/IP is correctly installed and bound to your network adapter.

   If you do not receive a reply, the transceiver on your network card is not operating correctly and may need to be reconfigured to use the proper connection type, or in older cards, may need to be configured to use different IRQ (Interrupt Request) resources.

2. Use the **ping** command with the IP address of the local computer to verify that the computer was added to the network correctly and does not have a duplicate IP address. If configured correctly, the **ping** command simply forwards the packet to the loopback address of 127.0.0.1.

3. Use the **ping** command with the IP address of the default gateway to verify that the default gateway is operational and that your computer can communicate with a host on the local network.
4. Use the **ping** command with the IP address of a remote host to verify that the computer can communicate through a router.

   If the **ping** command is successful after using step 4, steps 1 through 3 are successful by default. If the **ping** command is not successful, use the **ping** command with the IP address of another remote host because the current host might be turned off.

Delivery Tip
Demonstrate how to verify TCP/IP properties by opening the **Internet Protocol** (**TCP/IP) Properties** dialog box and clicking the **IP Address** tab.

Verifying TCP/IP Properties

If you cannot use **ping** successfully at any point, verify that the local computer's IP address is valid and appears correctly on the **IP Address** tab of the **Internet Protocol** (**TCP/IP) Properties** dialog box. You can also use the **ipconfig**command to verify the IP address of the local computer. **ipconfig** is the only way to view IP configuration data when the IP address is assigned by DHCP or Auto Private IP.

Lab 7A: Configuring IP Addresses for Windows XP Professional

Topic Objective
To introduce the lab.

Lead-in
In this lab, you configure Windows XP Professional to use TCP/IP.

Objectives

The goal of this lab is for the students to successfully configure Windows XP Professional to use TCP/IP.

After completing this lab, you will be able to:

- Configure static IP addresses.
- Configure Windows XP Professional to use DHCP for IP Address assignment.
- Configure an alternate TCP/IP configuration.
- Configure additional IP addresses and default gateways for Windows XP Professional.

Prerequisites

Before working on this lab, you must have:

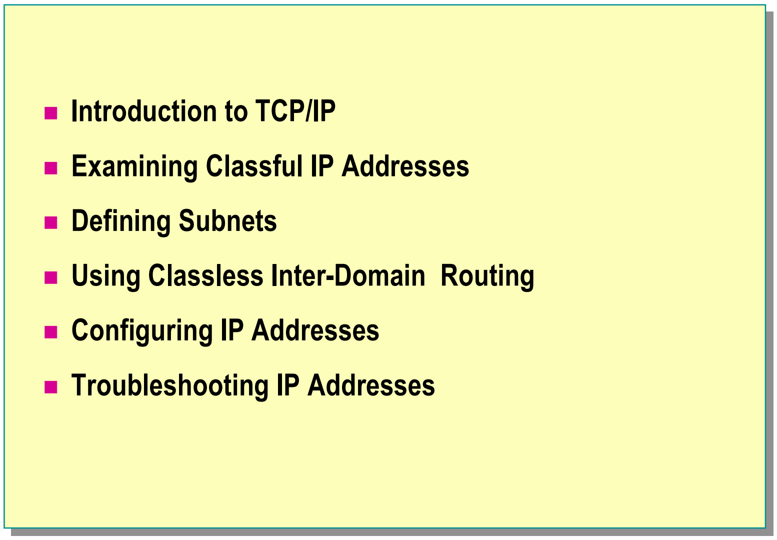
- Knowledge of the basic principles of TCP/IP.
- Understand the most common methods of assigning IP addresses.
- Basic understanding of name resolution, and name resolution services, such as DNS and WINS.
- Basic TCP/IP troubleshooting knowledge.

Estimated time to complete this lab: 45 minutes

Review

Topic Objective
To reinforce module objectives by reviewing key points.

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



- Introduction to TCP/IP
- Examining Classful IP Addresses
- Defining Subnets
- Using Classless Inter-Domain Routing
- Configuring IP Addresses
- Troubleshooting IP Addresses

1. You have a network with 5,000 host computers and 50 network segments. Your organization is planning to connect to the Internet. You have been assigned 137.25.0.0 as your only network address. What is required in order to use the single network address for multiple network segments?

   You must subnet in order to create unique network IDs for each network segment.

2. You have a network with 500 host computers. Your organization is planning to connect to the Internet. You have been assigned 137.25.0.0 as your only network address. What would the default subnet mask be?

   255.255.0.0

3. You have recently installed DSL at home. Your Internet service provider requires you to have static IP address. Your laptop is running Windows XP Professional, and is a DHCP client when connected to the company intranet. How can you configure your laptop to only use the static IP address assigned by your Internet service provider when you are at home?

   Use the Alternate Configuration tab in the TCP/IP properties for your network adapter. Select User configured and supply the Internet service provider's static IP address information.

4. A user reports they are unable to connect to their file server. You suspect their TCP/IP address information is configured improperly. How can you quickly verify the IP address for the user's computer?

   Use ipconfig.

5. A user reports being unable to connect to a file server. You suspect that the TCP/IP address information is configured improperly. How can you easily verify TCP/IP configuration and network connectivity to the file server?

   Use the ping command to verify communication to the server.

6. You are the administrator for a small network with 20 computers running Windows XP Professional and two Windows 2000 file servers. You do not connect to the Internet or any external networks. What is the easiest way to quickly configure the IP addresses for the computers in your network?

   Configure all the computers to obtain an IP address automatically. No DHCP server is needed. By default, the Automatic private IP address feature is used.