MICROSOFT
TRAINING
AND CERTIFICATION

Microsoft Official Curriculum

# Module 6: Examining TCP/IP

**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:**
**105 Minutes**

**Labs:**
**30 Minutes**

This module provides students with an overview of TCP/IP concepts and how TCP/IP handles network communication. The first section of the module begins with an introduction to the communication process and is followed by an overview of the four layers in the protocol stack. The first section concludes with a discussion of the use of sockets in identifying the applications involved in a communication at any given time.

The next section describes the protocols in the TCP/IP protocol suite and the functions each performs in the communication process. This section concludes with information about some of the various utilities—Hostname, Arp, and Ping—included with the TCP/IP suite. In the lab that follows this section, the students will use the Hostname, Arp, and Ping utilities, as examples of TCP/IP utilities, to test connectivity.

The third section in the module focuses on the name resolution process. It begins with a description of host names and NetBIOS names and then explains static and dynamic mapping and the name resolution process.

The next section examines the TCP/IP data transfer process. This section explains the terminology used to refer to a packet and discusses the frame components and the route taken by the data to travel from the source computer to the destination computer. This continues the analogy introduced in the first section and identifies how the TCP/IP protocols interact to enable communication between computers.

The last section of the module describes the routing process. The types of packet delivery are explained and then the process by which data is routed across a network to its final destination is detailed. The analogy from the first section ends here with an explanation that describes how packets are routed from one router to another until the packet reaches its destination. The module concludes with a lab in which the students identify the processes involved with name resolution and the protocols involved in using the Ping utility across a router.



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

- Describe the TCP/IP communication process.
- Name the protocols in the TCP/IP protocol stack and describe the services they provide.
- Describe the process for resolving user-friendly computer names by mapping them to an IP address.
- Describe the process for sending data packets from one computer to another.
- Describe how the process of routing passes information between two segments so that computers can communicate on a larger scope.

# 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_06.ppt
- Module 6, "Examining TCP/IP"

## Preparation Tasks

To prepare for this module, you should:

- Read all of the materials for this module.
- Read the TCP/IP topic in Windows 2000 Help.
- Read the white papers, *Introduction to TCP/*IP and *TCP/IP Implementation Details for Windows 2000*, on the Trainer Materials compact disc.
- Complete the two 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:

- Introduction to TCP/IP

  Provide an overview of the data communication process. Then briefly introduce the layers in the TCP/IP protocol stack and explain how sockets are used to differentiate one communication connection from another.

- TCP/IP Protocol Suite

  Describe the features and functions of each protocol in the TCP/IP suite and the utilities included in it. Then demonstrate the usage of the Hostname, Arp, and Ping utilities.

- Name Resolution

  Describe the two types of computer names, the concepts of static and dynamic mapping, and the name resolution process for host names as well as NetBIOS names.

- Examining the Data Transfer Process

  Explain the different terms used to refer to the data packet at various stages of preparing it for transmission across the network. Describe the components of a data packet and the process used to prepare it at the source computer and to access the information in it at the destination computer.

- Routing Data

  Describe the process of IP routing and the types of packet delivery. Then explain the process by which data is routed from one segment to another.

# 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 communication in a Windows 2000 network by using the TCP/IP protocol suite.

- Introduction to TCP/IP
- TCP/IP Protocol Suite
- Name Resolution
- Examining the Data Transfer Process
- Routing Data

Transmission Control Protocol/Internet Protocol (TCP/IP) for Microsoft® Windows® 2000 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 of their network traffic.

TCP/IP provides a technology for connecting dissimilar systems. It also provides a robust, scaleable, cross-platform client/server framework and a foundation for gaining access to global Internet services, such as the World Wide Web and e-mail.

The various protocols in the TCP/IP stack function together to make network communication happen. This 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.

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

- Describe the TCP/IP communication process.
- Name the protocols in the TCP/IP protocol stack and describe the services they provide.
- Describe the process for resolving user-friendly computer names by mapping them to an IP address.
- Describe the process for sending data packets from one computer to another.
- Describe how the process of routing passes information between two segments so that computers can communicate on a larger scope.

# Introduction to TCP/IP

**Slide Objective**
To introduce the TCP/IP protocol suite.

**Lead-in**
Windows 2000 supports TCP/IP both as a protocol and as a set of services for connecting and managing networks.

- **The Communication Process**
- **TCP/IP Layers**
- **Identifying Applications**

Transmission Control Protocol/Internet Protocol (TCP/IP) is an industry-standard protocol stack that is used for communication between Windows 2000-based computers. TCP/IP is designed for communication across large-scale networks.

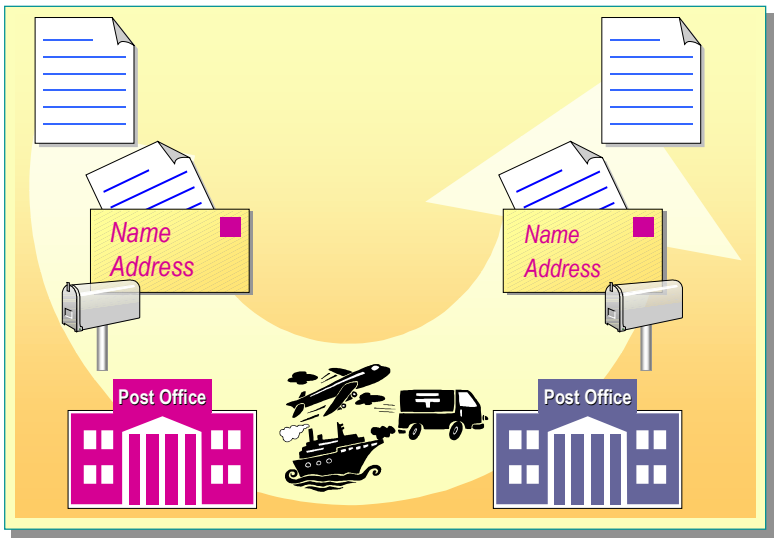
The tasks involved in using TCP/IP in the communication process are distributed between protocols that are organized into four distinct layers of the TCP/IP stack. Each protocol in the TCP/IP stack has a distinct role in the communication process.

During the communication process, many applications may be in communication at the same time. TCP/IP has the ability to differentiate one application from another. TCP/IP identifies an application on one computer and then moves the data from that application to an application on another computer.

# The Communication Process

**Slide Objective**
To illustrate the TCP/IP communication process.

**Lead-in**
TCP/IP enables connectivity on Windows 2000-based computers by using a communication model.




**Delivery Tip**
Explain how communication that uses TCP/IP relates to a common concept, such as the process of sending a letter. Use the animated slides to show the students that the process is parallel at each computer.

The process by which TCP/IP transmits data between two locations is analogous to the procedure used to send a letter from one city to another by postal mail.

## TCP/IP Activities

The TCP/IP communication process is initiated 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 is similar to writing a letter in a language that the recipient can understand. Then the data is associated with the destination application and computer, much like 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.

After these activities are performed, the data and additional information, including a request for confirmation of its delivery, are sent over the network to the destination. The network medium used for transmitting the data is independent of the above activities, just as the means of transport that transfers the letter from one post office to another is independent of the letter's content or address.

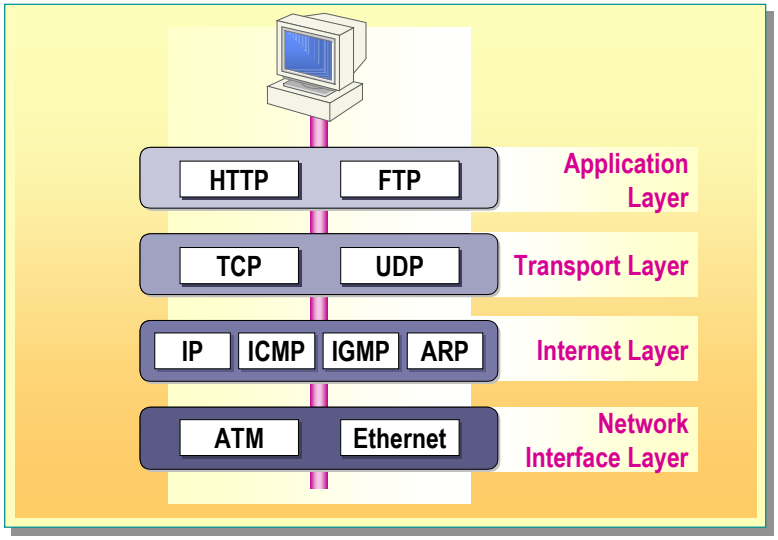
## TCP/IP Protocols and Layers

TCP/IP organizes the communication process outlined here by assigning these activities to various protocols in the TCP/IP stack. To increase the efficiency of the communication process, the protocols are arranged in layers. The addressing information is placed last, so that the computers on a network can quickly check whether the data is meant for them. Only the computer that is the destination computer opens and processes all of the data.

# TCP/IP Layers

**Slide Objective**
To illustrate the TCP/IP layers.

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




**Delivery Tip**
Explain that the four layers of the TCP/IP protocol stack are a specific implementation of the seven layers of the OSI model. Relate the information in this module to the content on protocol stacks in module 5, "Examining Network Protocols," in course 2151A, *Microsoft Windows 2000 Network and Operating System Essentials*. Do not spend a large amount of time on the specific protocols, as these are discussed later. Instead, use this page to point out the division of tasks.

TCP/IP uses a four-layer communication model to transmit data from one location to another. The four layers in this model are application, transport, Internet, and network interface. All protocols that belong to the TCP/IP protocol stack are located in these 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. They include:

- Hypertext Transfer Protocol (HTTP)

  HTTP is used to transfer files that make up the Web pages of the World Wide Web.
- File Transfer Protocol (FTP)

  FTP is used for interactive file transfer.

## Transport Layer

The transport layer provides the ability to order and guarantee communication between computers and passes the data up to the application layer or down to the Internet layer. 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. They are:

- Transmission Control Protocol (TCP)

  TCP guarantees the delivery of data through an acknowledgement.

- User Datagram Protocol (UDP)

  UDP provides fast delivery of data but does not guarantee data delivery.

## 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 to be transmitted and getting it to its destination.

- Address Resolution Protocol (ARP)

  ARP is responsible for identifying the media access control (MAC) address of the network adapter on the destination computer.

- Internet Control Message Protocol (ICMP)

  ICMP is responsible for providing diagnostic functions and reporting errors due to unsuccessful delivery of data.

- Internet Group Management Protocol (IGMP)

  IGMP is responsible for the management of multicasting within TCP/IP.

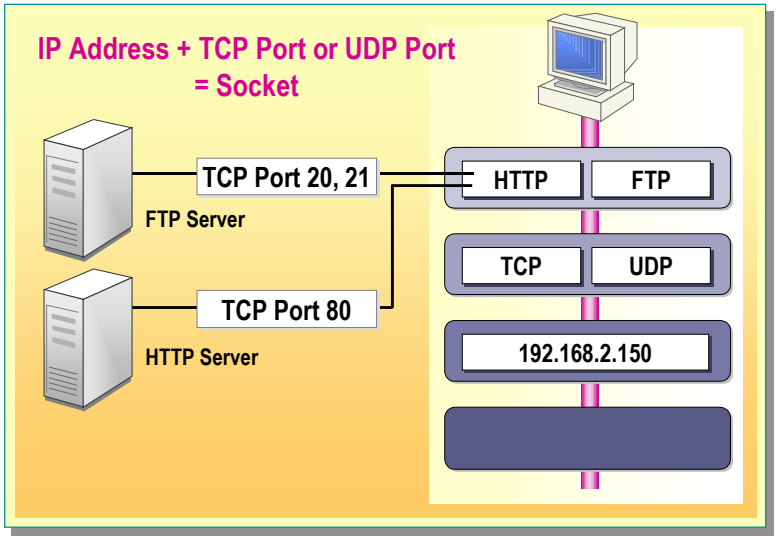
## Network Interface Layer

The network interface layer is responsible for placing data on the network medium and receiving data off the network medium. This layer contains such physical devices as network cables and network adapters. The network adapter has a unique 12-character hexadecimal number, such as B5-50-04-22-D4-65, which is known as the media access control (MAC) address. The network interface layer does not contain the type of software-based protocols that are included in the other three layers, but it does contain such protocols as Ethernet and asynchronous transfer mode (ATM), which define how data is transmitted on the network.

# Identifying Applications

**Slide Objective**
To explain the role of sockets and ports in network communication.

**Lead-in**
How do multiple applications communicate simultaneously in a network?




**Delivery Tip**
Use this page to inform the students that they must identify both the computer to which they are sending the data as well as the specific application on that computer.

In a network, many applications are in communication at the same time. When multiple applications are active on a single computer, TCP/IP requires a method for differentiating one application from another. For this purpose, TCP/IP uses a socket, also known as an end point in network communication, to identify a specific application.

## IP Address

To start a network communication, the location of the source and destination computers in the network must be known. The location is identified by a unique number, known as an IP address, which is assigned to each computer on the network. An example of an IP address is 192.168.2.200.

## TCP/UDP Port

A *port* is an identifier for an application within a computer. A port is associated with either TCP or UDP transport layer protocols and is referred to as a TCP port or UDP port. A port can be any number between 0 and 65,535. Ports for common server-side TCP/IP applications, referred to as well-known port numbers, are reserved to numbers below 1,024 in order to avoid confusion with other applications. For example, the FTP Server application uses the TCP port numbers 20 and 21.

## Socket

A *socket* is the combination of an IP address and the TCP port or UDP port. An application creates a socket by specifying the IP address of the computer, the type of service (TCP for guarantee of data delivery, otherwise UDP), and the port that the application monitors. The IP address component of the socket helps to identify and locate the destination computer, and the port determines the specific application to which the data is to be sent.

# TCP/IP Protocol Suite

**Slide Objective**
To introduce the core protocols in the Microsoft TCP/IP protocol suite.

**Lead-in**
The TCP/IP protocol suite consists of six core protocols and a set of utilities.

- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP)
- Internet Protocol (IP)
- Internet Control Message Protocol (ICMP)
- Internet Group Management Protocol (IGMP)
- Address Resolution Protocol (ARP)
- TCP/IP Utilities

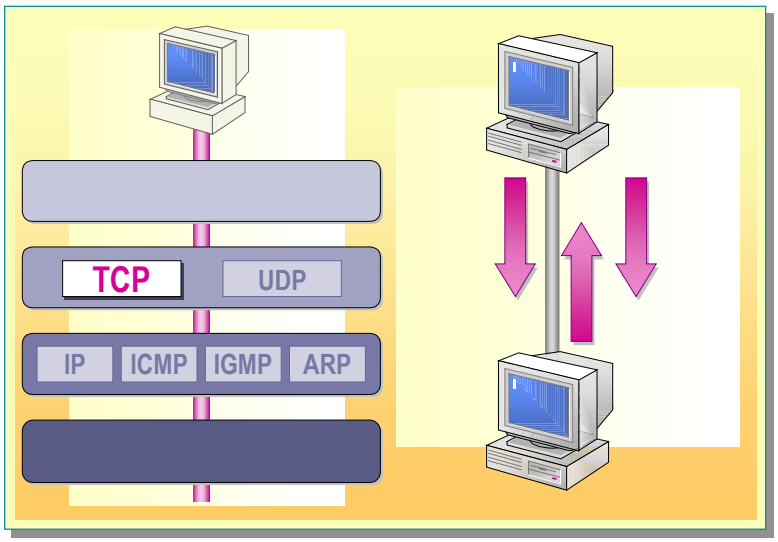
The Microsoft TCP/IP protocol *suite* enables enterprise networking and connectivity on Windows 2000-based computers. A suite is created by a vendor or organization to customize a protocol stack for its requirements. Therefore, a protocol suite is a set of protocols designed and built as complementary parts of a complete, smoothly functioning set.

The TCP/IP protocol suite includes six core protocols and a set of utilities. The six core protocols—TCP, UDP, IP, ICMP, IGMP, and ARP—provide a set of standards for communications between computers and for connections between networks. All applications and other protocols in the TCP/IP protocol suite rely on the basic services provided by these core protocols.

# Transmission Control Protocol (TCP)

**Slide Objective**
To describe the features of TCP.

**Lead-in**
TCP is one of the two core protocols in the transport layer.




**Delivery Tip**
Use the slide to illustrate the three-way handshake process that initiates a TCP session. Also emphasize the unicast nature of TCP. You can mention that a credit card transaction is an example of how TCP is used.

Transmission Control Protocol (TCP) is a required TCP/IP standard protocol that provides a reliable, connection-oriented data delivery service between only two computers. Such a communication is known as a unicast. In connection-oriented communication, the connection must be established before data can be transmitted between the two computers.

After the connection is established, data is transmitted over this single connection only. Connection-oriented communication is also referred to as reliable communication because it guarantees the delivery of the data at the destination.

On the source computer, TCP organizes the data to be transmitted into packets. On the destination computer, TCP reorganizes the packets to recreate the original data.



## Data Transmission Using TCP

TCP transmits packets in groups to increase efficiency. It assigns a sequence number to each packet and uses an acknowledgment to verify that the destination computer has received a group of packets. If the destination computer does not return an acknowledgment for each group of packets sent within a specified period of time, the source computer retransmits the data.

In addition to adding the sequencing and acknowledgement information to the packet, TCP also adds the port information for both the source and the destination applications. The source computer uses the destination port to direct the packet to the proper application at the destination computer, and the destination computer uses the source port to return information to the correct source application.

## Three-Way Handshake

Because TCP is a reliable protocol, two computers using TCP for communication must establish a connection before exchanging data. This connection is a virtual connection and is known as a *session*. Two computers using TCP establish a connection, or TCP session, through a process known as a three-way handshake. This process synchronizes sequence numbers and provides other information needed to establish the session.

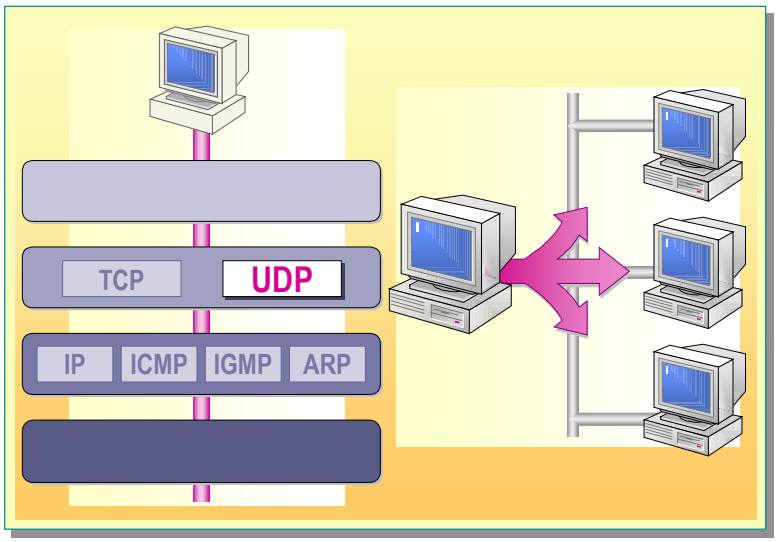
The three-way handshake is a three-step process:

1. The source computer initiates the connection by transmitting the session information, including the sequence number and size of the packet.
2. The destination computer responds with its session information.
3. The source computer agrees with and acknowledges the received information.

# User Datagram Protocol (UDP)

**Slide Objective**
To describe the features of UDP.

**Lead-in**
Along with TCP, UDP is another core protocol in the transport layer.




**Delivery Tip**
Although UDP is capable of unicast transmissions, use the slide to emphasize its broadcast or multicast nature as well. Be sure to contrast it with TCP. Additional examples could include information about the need to communicate with more than one computer at a time or about small pieces of data that would not benefit from the overhead of TCP.

User Datagram Protocol (UDP) is a transport layer protocol that identifies the destination application in network communications. UDP provides a connectionless packet delivery service that offers fast but unreliable, best-effort delivery of the data. UDP does not require an acknowledgment for the data received and does not attempt to retransmit data that is lost or corrupted. This means that less data is sent, but neither the arrival of packets nor the correct sequencing of delivered packets is acknowledged or guaranteed.

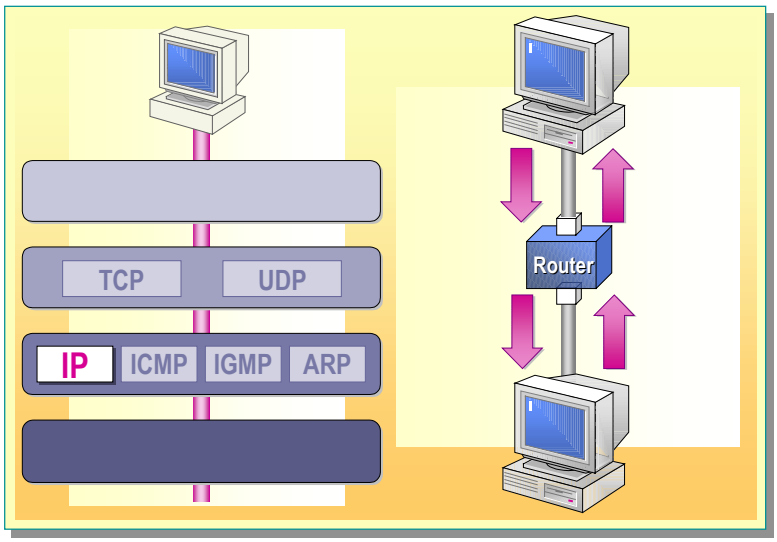
UDP is used by applications that transmit data to multiple computers by using broadcast or multicast transmissions. It is also used for transmitting small amounts of data or data that is not of high importance. Example uses of UDP include multicasting streaming media, such as during a live videoconference, and broadcasting a list of computer names, which are maintained for local communication.

To use UDP, the source application must supply its UDP port number as well as that of the destination application. It is important to note that UDP ports are distinct and separate from TCP ports, even though some of them use the same numbers.

# Internet Protocol (IP)

**Slide Objective**
To describe the features of IP.

**Lead-in**
IP is one of the three core protocols of the Internet layer.




**Delivery Tip**
Use the slide to illustrate the importance of IP in moving data across routers. Remind students that although the network adapters use the MAC address to identify a computer on the local segment, it is IP that moves the data to the local segment.

Internet Protocol (IP) helps to identify the location of the destination computer in a network communication. IP is a connectionless, unreliable protocol that is primarily responsible for addressing packets and routing them between networked computers. Although IP always attempts to deliver a packet, a packet may be lost, corrupted, delivered out of sequence, duplicated, or delayed. However, IP does not attempt to recover from these types of errors by requesting retransmission of the data. Acknowledging the delivery of packets and recovering lost packets is the responsibility of a higher-layer protocol, such as TCP, or of the application itself.

## Activities Performed by IP

You can visualize IP as the mailroom of the TCP/IP stack, where packet sorting and delivery take place. The packets are passed down to IP by UDP or TCP from the transport layer or passed up from the network interface layer. The primary function of IP is to route the packets until they reach their destination.

Each packet includes the source IP address of the sender and the destination IP address of the intended recipient. These IP addresses in a packet remain the same throughout the packet's journey across a network.

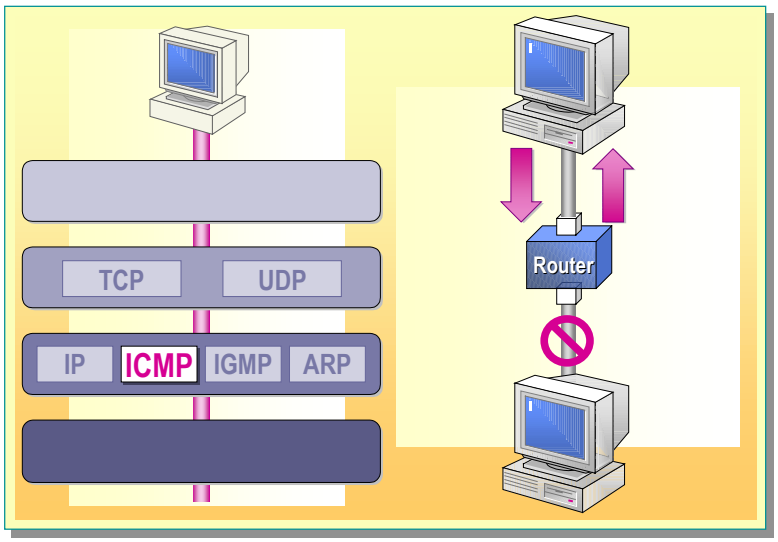
If IP identifies a destination address as an address from the same segment, it transmits the packet directly to that computer. If the destination IP address is not on the same segment, IP must use a router to send the information.

IP is also responsible for ensuring that a packet does not remain on the network forever by limiting the number of networks across which the packet can travel. This is done by assigning a Time to Live (TTL) number to every packet. A TTL specifies the maximum length of time that the packet can travel on the network before being discarded.

# Internet Control Message Protocol (ICMP)

**Slide Objective**
To describe the features of ICMP.

**Lead-in**
ICMP is another core protocol in the IP layer.




**Key Point**
Remind the students that unreliable (not guaranteed) delivery is not bad; it is just not needed in many cases. And the overhead involved to send a simple error message across the network is unnecessary. However, when a packet must reach a specific destination, TCP is used. The above graphic provides an example of an ICMP message being returned.

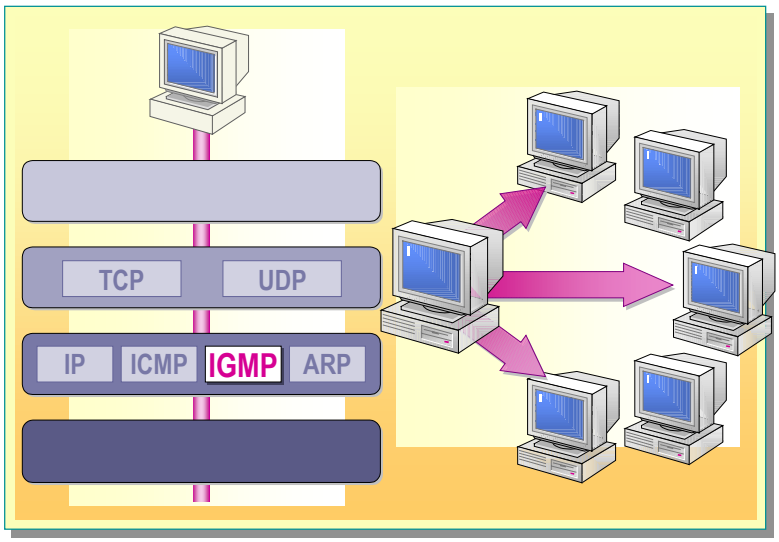
Internet Control Message Protocol (ICMP) provides troubleshooting facilities and error reporting for undeliverable packets. With ICMP, computers and routers that use IP communication can report errors and exchange limited control and status information. For example, if IP is unable to deliver a packet to a destination computer, ICMP sends a Destination Unreachable message to the source computer.

Although the IP protocol is used to move data across routers, ICMP reports errors and control messages on behalf of IP. ICMP does not attempt to make IP a reliable protocol, because ICMP messages are unacknowledged and therefore unreliable. It only attempts to report errors and provide feedback on specific conditions. Although this may not seem effective, it is much more efficient than using bandwidth to acknowledge each ICMP message.

# Internet Group Management Protocol (IGMP)

**Slide Objective**
To describe the features of IGMP.

**Lead-in**
IGMP is also one of the core protocols in the Internet layer.




**Delivery Tip**
Make sure that the students are not confused by the similarity between IGMP and ICMP. A good way to ensure this is to remind the students that G = group with IGMP.

Internet Group Management Protocol (IGMP) is a protocol that manages the membership lists for IP multicasting in a TCP/IP network. IP multicasting is a process by which a message is transmitted to a select group of recipients, known as a multicast group. IGMP maintains the list of members who subscribe to each multicast group.

## Managing IP Multicasting

All of the members of a multicast group listen for IP traffic directed to a specific multicast IP address and receive the packets sent to that IP address. However, because multicasting involves multiple computers, the packets are sent using the unreliable UDP protocol, which does not guarantee the delivery of the packets to the multicast group.

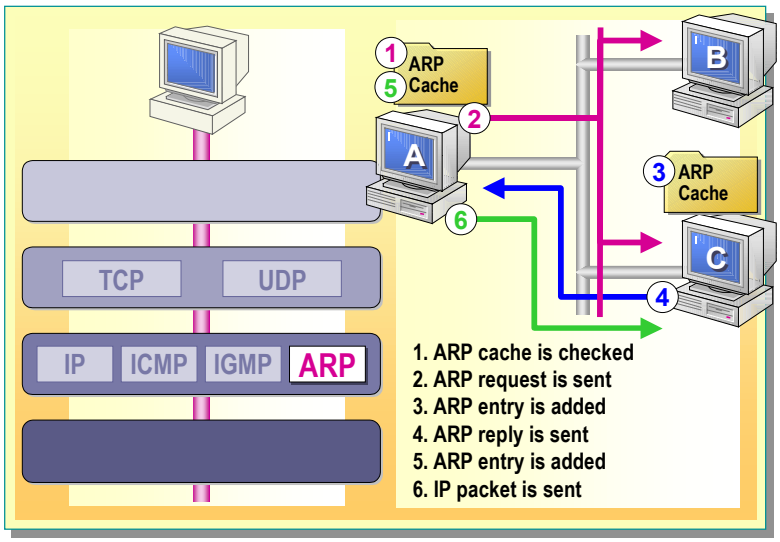
When multiple computers need to access information, such as streaming media, an IP address reserved for multicasting is used. Routers that are configured to process multicast IP addresses pick up this information and forward it to all subscribers of the multicast group associated with the multicast IP address.

For multicast information to reach its recipients, it is important that each router in the path of communication supports multicasting. Windows 2000-based computers can both send and receive IP multicast traffic.

# Address Resolution Protocol (ARP)

**Slide Objective**
To describe the features of ARP.

**Lead-in**
ARP is another of the three core protocols in the Internet layer.




**Key Point**
Point out to the students that ARP can occur only between computers on the same network segment and that a router is needed if the destination computer is remote.

**Delivery Tip**
Test the students' understanding by asking questions about broadcast messages and whether they are routable.

Located in the Internet layer of the TCP/IP suite, Address Resolution Protocol (ARP) performs address resolution for outgoing packets. Address resolution is the process by which IP addresses are mapped to MAC addresses. The network adapters use the MAC address to determine if a packet is meant for that computer.

Without the MAC address, the network adapters do not know if they are to pass the data to a higher layer for further processing. As the outgoing packets in the IP layer are being readied for transmission on the network, the source and destination MAC addresses must be added.

## ARP Cache

ARP stores a table containing IP addresses and their corresponding MAC addresses. The area of memory where this table is stored is referred to as the ARP cache. The ARP cache for any computer contains the mappings for only computers and routers that reside on the same segment.

## Physical Address Resolution

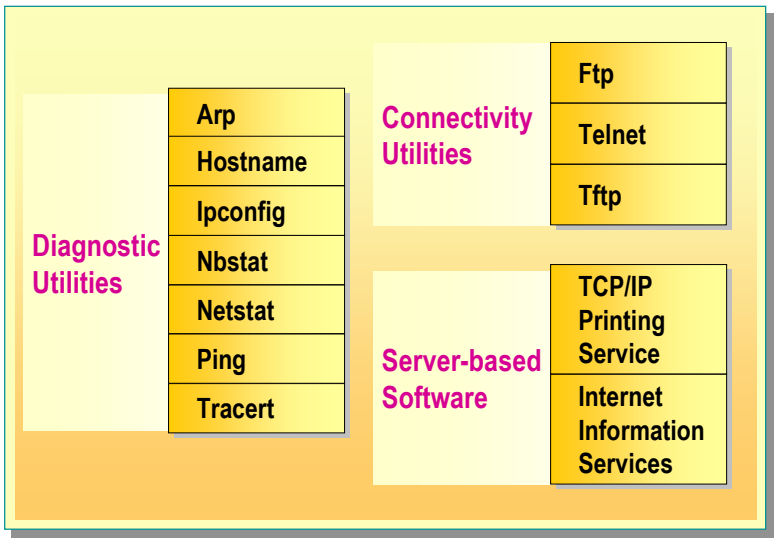
ARP compares every outbound packet's destination IP address with the ARP cache to determine the MAC address to which the packet will be sent. If there is a matching entry, the MAC address is retrieved from the cache. If not, ARP broadcasts a request for the computer owning the IP address in question to reply with its MAC address. Next, the computer with the corresponding IP address adds the initial computer's MAC address to its cache and then replies with its own MAC address. When an ARP reply is received, the ARP cache is updated with the new information and the packet can then be sent.

If the packet is going to another segment, ARP resolves the MAC address for the router responsible for that segment, rather than resolving the address for the final destination computer. The router is then responsible for either finding the MAC address of the destination or forwarding the packet to another router.

# TCP/IP Utilities

**Slide Objective**
To introduce the types of TCP/IP utilities.

**Lead-in**
The TCP/IP protocol suite consists of a number of utilities that allow users to access information on the network.




**Delivery Tip**
This is not an inclusive list of utilities. It only illustrates the various types of utilities that are included with the TCP/IP suite. Remind the students that these utilities are all found on the application layer. Demonstrate the Hostname, Arp, and Ping utilities. Point out that Arp is both a utility and a protocol (ARP). Also point out that Ping uses another protocol, ICMP, to accomplish its work. Note that Ipconfig is discussed in module 7, "Examining IP Addressing," in course 2151A, *Microsoft Windows 2000 Network and Operating System Essentials* and the above utilities are used in the following lab.

The Microsoft TCP/IP suite provides basic TCP/IP utilities that enable a computer running Windows 2000 to access a wide variety of information on the network. Their capabilities range from determining if a specific computer on the network is accessible to downloading multimedia documents from the Internet.

Windows 2000 includes three types of TCP/IP-based utilities: diagnostic utilities, connectivity utilities, and server-based software.

## Diagnostic Utilities

Diagnostic utilities allow users to detect and resolve networking problems. Some of the common diagnostic utilities are:

- Arp: This utility displays and modifies the Address Resolution Protocol (ARP) cache.
- Hostname: This utility displays the host name of your computer.
- Ipconfig: This utility displays and updates the current TCP/IP configuration, including the IP address.
- Nbstat: This utility displays the local NetBIOS name table, which is a table of user-friendly computer names mapped to IP addresses.
- Netstat: This utility displays the TCP/IP protocol session information.
- Ping: This utility verifies configurations and tests IP connectivity between two computers. Ping sends an ICMP request from the source computer, and the destination computer responds with an ICMP reply.
- Tracert: This utility traces the route that a packet takes to a destination.

## Connectivity Utilities

Connectivity utilities allow users to interact with and use resources on a variety of Microsoft and non-Microsoft hosts, such as UNIX systems. Some of the common connectivity utilities are:

- Ftp: This utility uses TCP to transfer files between Windows 2000 and computers running File Transfer Protocol (FTP) server software.
- Telnet: This utility remotely accesses network resources on computers running Telnet server software.
- Tftp: This utility uses UDP to transfer small files between Windows 2000 and computers running Trivial File Transfer Protocol (TFTP) server software.

## Server-based Software

This software provides printing and publishing services to TCP/IP-based clients on Windows 2000.

- TCP/IP Printing service: This utility provides standard TCP/IP printing services. It allows computers running operating systems other than Windows 2000 to print to a printer attached to a Windows 2000-based computer.
- Internet Information Services: Internet Information Services (IIS) offers Web, news, e-mail, and file transfer server software for TCP/IP-based publishing services.

## Examples of Common Utilities

Hostname, Arp, and Ping are three common TCP/IP utilities. Because they are frequently used, it is recommended that you know how to access them.

### Hostname

The syntax to use this utility is *hostname*. To access this utility, type **hostname** at the command prompt. The system displays the host name of your computer.

### Arp

The syntax to access information from the ARP cache is *arp -a*. Type **arp -a** at the command prompt to display the information in your ARP cache.

### Ping

The syntax to test connectivity is *ping*. To test connectivity by using an IP address or computer name, type **ping** [*IP_address* or *computer_name*]

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**

# Lab A: Using TCP/IP Utilities

**Slide Objective**
To introduce the lab.

**Lead-in**
In this lab, you use the Ping utility to test the TCP/IP configuration of your computer. You will also use Ping to test connectivity with other computers, the Hostname utility to identify your computer, and the Arp utility to identify your MAC address.

## Objectives

After completing this lab, you will be able to:

- Use the Ping utility to test configuration and connectivity.
- Use the Hostname utility to obtain your computer name.

## 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
# Using TCP/IP Utilities

## Scenario

You are an administrator of a small company and want to ensure that your server is properly configured with TCP/IP and to test that it can communicate with other computers on the network.

## Goal

In this exercise, you will use the Ping utility to verify TCP/IP configuration and then use the Hostname utility to identify your computer name. You will then use Ping to test connectivity with your partner to ensure that you can communicate on the network.

| Tasks | Detailed Steps |
|---|---|
| **1.** Log on as Administrator with a password of **password**. Use the Hostname utility to verify your computer's host name and Ping to test the TCP/IP configuration. | **a.** Log on to Windows 2000 as Administrator with a password of **password**.<br>**b.** Click **Start**, point to **Programs**, point to **Accessories**, and then click **Command Prompt**.<br>**c.** Type **PING 127.0.0.1** in the Command Prompt window. |
| How many packets were sent, received, and lost?<br>**Four packets were sent, four packets were received, and no packets were lost, unless there is a problem with your TCP/IP installation.** | |
| Is TCP/IP functioning properly?<br>**Yes, if all four packets were received.** | |
| **2.** Use the Hostname utility to obtain your host name, and then use the Ping utility with your host name to return your IP address. | **a.** In the Command Prompt window, type **hostname**<br>**b.** In the Command Prompt window, type **ping** *computer* (where *computer* is the host name that was returned in step a). |

(*continued*)

| Tasks | Detailed Steps |
|---|---|
| What is your computer's host name?<br>**Answers will vary; for the Instructor, it will be London.** | |
| What is your computer's IP address?<br>**192.168.x.y (where x is the room number and *y* is between 1 and 199).** | |
| **3.** Use the Ping utility with the instructor's computer name to verify that your computer can communicate on the network. | **a.** In the Command Prompt window, type **ping London** |
| What is London's IP address?<br>**192.168.1.200.** | |
| How do you know that you are able to communicate with London?<br>**Because no packets were lost.** | |
| **4.** Close all windows and log off from Windows 2000. | **a.** Close all windows and log off from Windows 2000. |

# Name Resolution

**Slide Objective**
To introduce the factors involved in name resolution.

**Lead-in**
All user-friendly names need to be mapped to their IP addresses to provide for communication.

- Types of Names
- Static IP Mapping
- Dynamic IP Mapping
- Name Resolution in Windows 2000

TCP/IP identifies source and destination computers by their IP addresses. However, users are much better at remembering and using words (user-friendly names) than numbers (IP addresses). There are different types of user-friendly names by which a computer can be addressed.

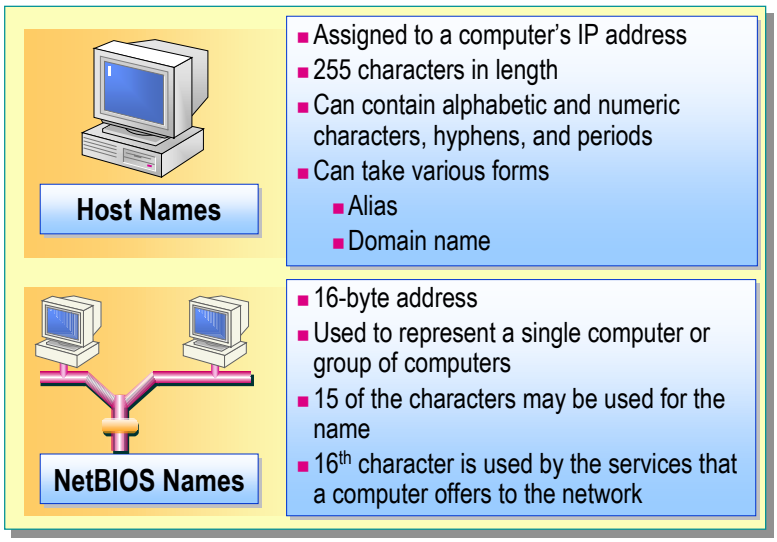
The Windows 2000 operating system has different storage locations where it keeps a record of user-friendly names mapped to their corresponding IP addresses. This mapping of the IP address of a computer can be stored in either a static or a dynamic file, based on the type of name used.

Some applications, such as Microsoft Internet Explorer and the Ftp utility, can use either the IP address or a user-friendly name to establish communication. When a user-friendly name is specified, a Windows 2000-based computer uses a process called name resolution to identify the appropriate IP address before TCP/IP-based communication with the desired resource can begin. However, if the IP address is specified, communication can happen immediately.

# Types of Names

**Slide Objective**
To describe the characteristics of the two types of user-friendly names.

**Lead-in**
There are two types of popular user-friendly names.




**Delivery Tip**
Ask students what NetBIOS means. Use the definition of NetBIOS when explaining a NetBIOS name. Tell the students that some applications, including NetBIOS resources, identify themselves by using a NetBIOS name instead of the more common host name. Host names are used in such products as browsers and are standard in Windows 2000.

There are two types of user-friendly names: host names and NetBIOS names.

## Host Names

A host name is a user-friendly name that is assigned to a computer's IP address to identify it as a TCP/IP host. The host name can be up to 255 characters in length and can contain alphabetic and numeric characters, hyphens, and periods.

Host names can take various forms. The two most common forms are alias and domain name. An alias is a single name associated with an IP address, such as London. A domain name is structured for use on the Internet and includes periods as separators. An example of a domain name is london.nwtraders.msft.

## NetBIOS Names

A NetBIOS name is a 16-character name that is used to identify a NetBIOS resource on the network. A NetBIOS name can represent a single computer or a group of computers, but only the first 15 of the characters may be used for the name. The final character is used to identify the resource or service that is being referred to on the computer.

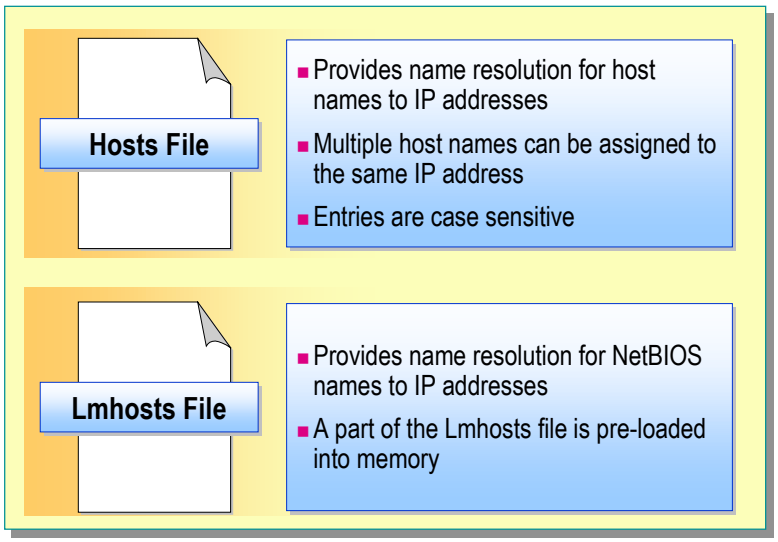
An example of a NetBIOS resource is the File and Printer Sharing for Microsoft Networks component on a computer running Windows 2000. When your computer starts, this component registers a unique NetBIOS name, based on the name of your computer and one character identifier that represents the component.

**Important** In Windows 2000, the NetBIOS name uses up to the first 15 characters of the host name and cannot be configured separately. Although Windows 2000 does not require NetBIOS names, previous versions of Windows require NetBIOS names to support networking capabilities.

# Static IP Mapping

**Slide Objective**
To describe the components of the static IP mapping method.

**Lead-in**
Static IP mapping is a method to store information about user-friendly names and their corresponding IP addresses.




When users specify a user-friendly name to communicate with a destination computer, TCP/IP still requires an IP address for transmission to occur, so the computer name is mapped to an IP address. This mapping is then stored in either a static or dynamic table. In a static table, mappings are stored in one of two text files: the Hosts file or the Lmhosts file.

The advantage of using a static table is that, because it is a text file located on each computer, it is customizable. Each user can create any number of required entries, including easy-to-remember aliases for frequently accessed resources. However, it is difficult to maintain and update static tables if the tables contain a large number of IP address mappings or if the IP addresses change often.



## Hosts File

The Hosts file is a text file that contains IP address-to-host name mappings. Within the Hosts file:

- Multiple host names can be assigned to the same IP address. A server at the IP address 167.91.45.121 can be referred to by its domain name (london.nwtraders.msft) or by an alias (London). This allows a user at this computer to refer to this server by using the alias London rather than by typing the entire domain name.
- Entries are case-sensitive, depending upon the platform. Hosts file entries for computers running Windows 2000 and Microsoft Windows NT® version 4.0 are not case-sensitive.

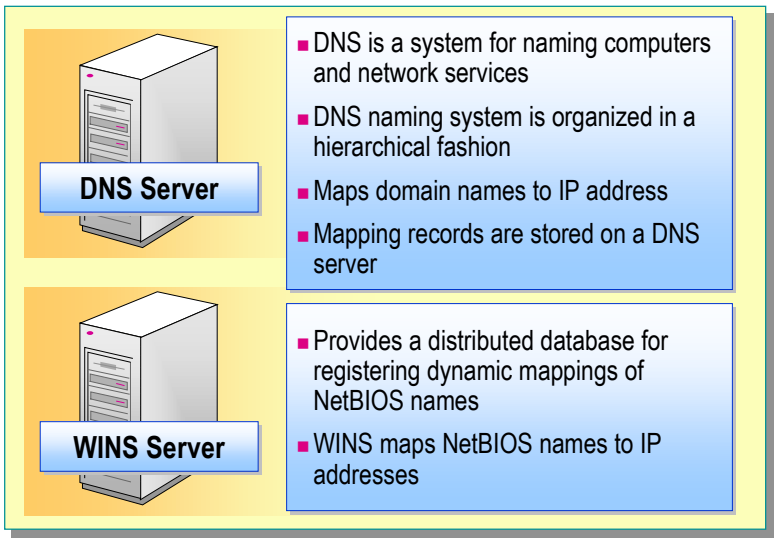
## Lmhosts File

The Lmhosts file is a text file that contains the IP address-to-NetBIOS name mappings. A portion of the Lmhosts file is pre-loaded into memory and is referred to as the NetBIOS name cache.

# Dynamic IP Mapping

**Slide Objective**
To describe the features of dynamic routing.

**Lead-in**
Manually updating mapping entries is not efficient.




The advantage of dynamic tables that store IP mappings is that they are updated automatically. To accomplish this, the dynamic tables use two services: Domain Name System (DNS) and Windows Internet Name Service (WINS). DNS and WINS perform the same functions as the Hosts and Lmhosts files, but without the need for manual configuration.

## Domain Name System (DNS)

DNS is a method for naming computers and network services. TCP/IP networks use the DNS naming convention to locate computers and services through user-friendly domain names. When a user enters a domain name in an application, the DNS service maps the name to an IP address.

The DNS naming system is organized in a hierarchical fashion to allow scalability to large systems, such as the Internet. By using a hierarchical system to create domain names, the computers that store the domain name–to–IP address mapping records have mappings for only their area. These computers, known as DNS servers, only process queries for computers located in their respective areas. As the mappings in the area change, Windows 2000-based DNS servers are automatically updated with the new information.

## Windows Internet Name Service (WINS)

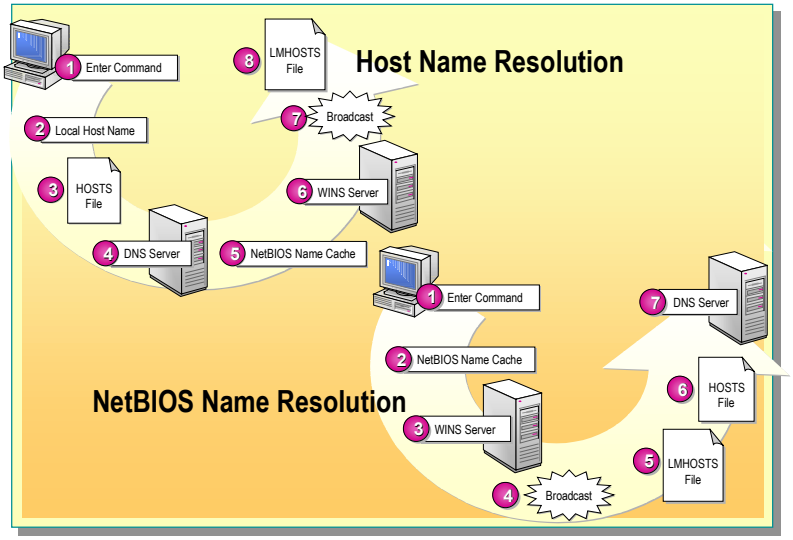
WINS provides a distributed database for registering dynamic mappings of NetBIOS names used on a network. WINS maps NetBIOS names to IP addresses and allows NetBIOS names to be used across routers.

**Note** A WINS server is not required for a pure Windows 2000 network but is recommended for use in a mixed environment.

# Name Resolution in Windows 2000

**Slide Objective**
To illustrate the name resolution process.

**Lead-in**
Windows 2000 resolves both host names and NetBIOS names.




**Key Point**
Windows 2000 applications are designed to use host names, but both NetBIOS and host names can be resolved using either process if configured properly. It is the order of resolution and the time to resolve that differs.

Name resolution is the procedure by which a name is resolved, or mapped, to an IP address. When you enter a user-friendly name in an application, the application determines whether the name is a host or NetBIOS name. Current applications in Windows 2000 use the host name resolution process, but some older applications, such as those designed for Microsoft Windows NT, Windows 95, and Windows 98, still use NetBIOS names. If the name resolution process fails, then the application cannot communicate with the desired destination. If you use an IP address, name resolution is not needed.

## Host Name Resolution Process

Host names can be resolved directly by the Hosts file or by a DNS server. The default name resolution procedure is as follows:

1. Computer A enters a command, such as **FTP**, by using the host name of Computer B.
2. Computer A checks to see if the specified name matches its local host name.
3. If not, then Computer A checks its Hosts file looking for Computer B's host name. If it finds the host name, it resolves it to an IP address.
4. If Computer A does not find Computer B's host name in the Hosts file, it sends a query to the DNS server. If the host name is found, it is resolved to an IP address.

5. If the host name is not found on the DNS server, Windows 2000 checks for the name in the NetBIOS name cache. It does this because Windows 2000 treats the NetBIOS name as the host name.
6. If the NetBIOS name cache does not have the host (NetBIOS) name, a query is sent to the WINS server.
7. If the WINS server cannot resolve the name, a broadcast message is sent out on the network.
8. If no host responds to the broadcast, the Lmhosts file is checked for the host (NetBIOS) name.

## NetBIOS Name Resolution Process

By default, NetBIOS names do not function over a TCP/IP network. Windows 2000 enables NetBIOS clients to communicate over TCP/IP by providing the NetBT protocol. NetBT is an acronym for NetBIOS over TCP/IP. This protocol allows NetBIOS-based applications to communicate using TCP/IP by translating the NetBIOS name to an IP address. If WINS is configured for use, then the procedure for resolving NetBIOS names is as follows:

1. Computer A enters a command, such as **net use**, by using the NetBIOS name of Computer B.
2. Computer A checks to see if the specified name is in its NetBIOS name cache.
3. If not, then Computer A queries a WINS server.
4. If the WINS server cannot locate the name, then Computer A uses a broadcast on the network.
5. If a broadcast does not resolve the name, then Computer A checks its Lmhosts file.
6. If the above NetBIOS methods do not resolve the name, then Computer A checks the Hosts file.
7. Finally, Computer A queries a DNS server.



**Caution** The order in which Windows 2000 uses these mechanisms depends on how the Windows 2000-based computer is configured.

# Examining the Data Transfer Process

**Slide Objective**
To introduce the role of packets in data transfer.

**Lead-in**
Data transfer is a complex process that involves many activities.

- Packet Terminology
- Frame Components
- Data Flow

TCP/IP transmits data on the network by dividing it into smaller portions called packets. Packets are often referred to by different terms based on the protocol with which they are associated. The division of data into packets is necessary because a large unit of data takes a long time to move on the network and can clog the network. While the large unit is being transmitted, no other computer can transmit data. Also, if an error occurs, the entire unit of data must be retransmitted.

In contrast, if small packets are sent on the network, they move quickly. Because small packets don't clog the network, other computers can also transmit data. If any packet becomes corrupted, only the corrupted packet needs to be retransmitted, instead of the entire data.

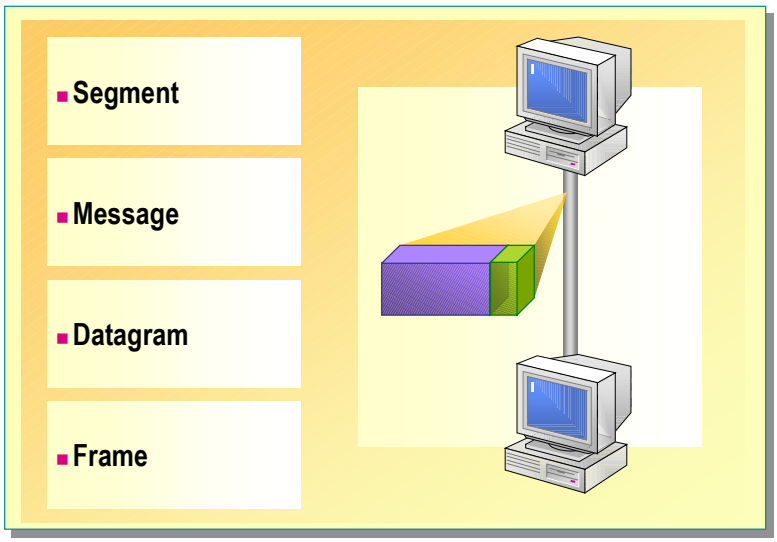
When a packet is transmitted in the network interface layer, it is referred to as a frame. A frame consists of different components that have specific functions in the flow of data in the network interface layer.

The data flow process involves a number of steps, including the organization of data into small packets at the source computer and their reassembly in the original form at the destination computer. Each layer of the TCP/IP protocol stack is involved in these activities at both the source and destination computers.

# Packet Terminology

**Slide Objective**
To introduce terminology frequently associated with packets.

**Lead-in**
At each layer of the TCP/IP stack, the packet is referred to by a different name.




**Delivery Tip**
This page should be used to clarify the fact there is more than one term associated with a packet of data. Remind students of the meaning of segment as it applies to the network.

As a packet of data moves from one layer to another in the TCP/IP stack, each protocol adds its own header information. The packet, along with the information added to it, is referred to by a different technical name as it is identified with different protocols. These names are segment, message, datagram, and frame.

## Segment

A segment is the unit of transmission in TCP. It contains a TCP header, accompanied by application data.

## Message

A message is the unit of transmission in unreliable protocols, such as ICMP, UDP, IGMP, and ARP. It consists of a protocol header, accompanied by application or protocol data.

## Datagram

A datagram is the unit of transmission in IP. It consists of an IP header, accompanied by transport layer data, and is also considered unreliable.

## Frame

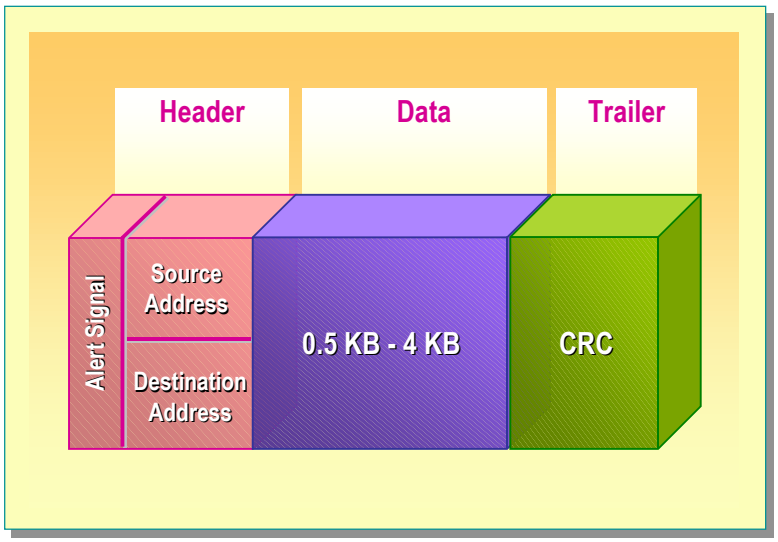
A frame is the unit of transmission in the network interface layer and consists of a header added at the network interface layer, accompanied by IP layer data.

**Note** As the name for UDP (user datagram protocol) suggests, it can also be referred to as a datagram. However, UDP message is the generally accepted term. The term segment is applied when a physical device is used to divide a network. In the context of a packet, the term segment is often referred to as a TCP segment.

# Frame Components

**Slide Objective**
To illustrate the three components of a data packet.

**Lead-in**
When data is broken up into packets, various types of information are added to it to enable it to reach its destination.




A frame (the term for a data packet in the network interface layer) consists of three components: the header, the data, and the trailer.

## Header

The header includes:

- An alert signal to indicate that the packet is being transmitted.
- The source address.
- The destination address.

## Data

This is the actual information sent by the application. This component of the packet varies in size, depending on the size limits set by the network. The data section on most networks varies from 0.5 kilobytes (KB) to 4 KB. With Ethernet, the size of the data is approximately 1.5 KB.

Because most original data strings are much longer than 4 KB, data must be broken into pieces small enough to be put into packets. It takes many packets to complete the transmission of a large file.

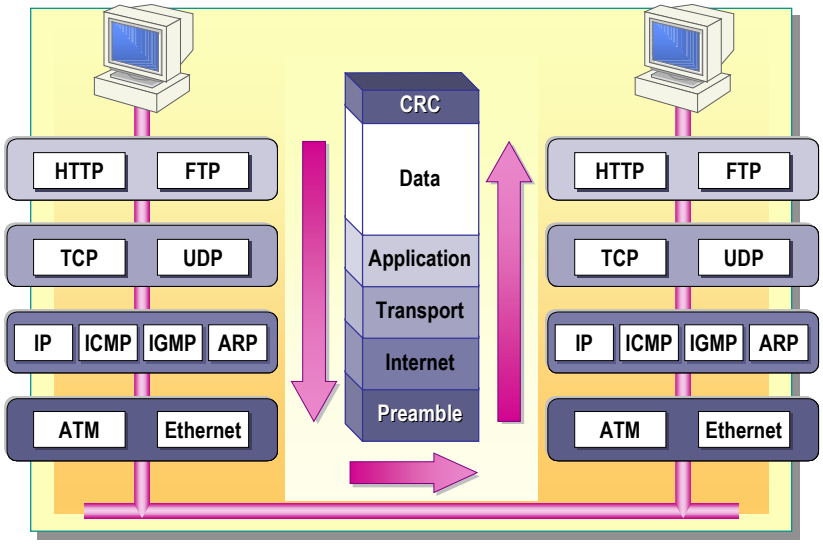
## Trailer

The exact content of the trailer varies depending on the network interface layer *protocol*. However, the trailer usually contains an error-checking component called a *cyclical redundancy check (CRC)*. The CRC is a number produced by a mathematical calculation on the packet at its source. When the packet arrives at its destination, the calculation is made again. If the results of both calculations are the same, this indicates that the data in the packet has remained stable. If the calculation at the destination differs from the calculation at the source, this means that the data has changed during transmission. In that case, the source computer retransmits the data.

# Data Flow

**Slide Objective**
To illustrate the process of data transfer.

**Lead-in**
Transferring packets from one computer to another involves the TCP/IP layers at both the source and destination computers.




**Delivery Tip**
The slide on this page furthers the analogy that was introduced in the first section. The students should now be familiar with the various tasks involved with TCP/IP, and this page integrates them by using a single segment approach. Also compare the CRC of the frame with the checksum added by the protocols at each layer. Remember that this is not a comprehensive list of items added by each protocol.

The packets of data being transmitted from one computer to another travel through the layers of the TCP/IP protocol stack. As the data packets pass through each layer, the protocols in that layer attach specific information to the header. The information added by every protocol includes error-checking information, known as the *checksum*. The checksum is used to verify whether the header information added by the protocol arrived intact at the destination protocol, compared with the CRC, which verifies the whole packet.

The information added by the protocols in one layer is encapsulated as data by the protocols 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. The packet is then passed up the protocol stack to the appropriate protocol.

## 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—TCP or UDP—and the checksum is added for both TCP and UDP.

If selected, TCP:

- 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.

If selected, UDP:

- Adds the UDP 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 header information:

- The source IP address
- The destination IP address
- The transport protocol
- The 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 MAC address. The ARP performs this resolution. The MAC address is added to the packet header and the packet is handed down to the network interface layer.



## Network Interface Layer

The network interface layer adds two types of information—a preamble and a cyclical redundancy check (CRC)—to the packet that it receives from IP. The preamble is a sequence of bytes that identifies the beginning of a frame. The CRC is a mathematical computation that is added to the end of the frame to verify that the frame has not been corrupted.

After the information is added to the frames at the network interface layer, they are merged onto the network. The frames are sent to all computers on 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 MAC address on the frame is examined.

If the MAC address is a broadcast address or if the MAC 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 the TCP at the source computer. Then, it uses the TCP port information on the packet to send it onwards to the appropriate application in the application layer above.

If UDP receives the packet from the Internet layer, it uses the UDP port information on the packet to send it to the appropriate application in the application layer without sending an acknowledgement to the source computer. After the application receives the data, it processes it as required.

# Routing Data

**Slide Objective**
To introduce the role of routers in transmitting data.

**Lead-in**
On most large networks, data needs to be routed from one portion of the network to another.

- IP Routing
- Data Transfer Across Routers

Data flow in a network that consists of a single segment is simple. Each computer that transmits data can broadcast a request over the network for the MAC address of the destination computer and send the data to it. However, in networks that have multiple segments, the data transmission process is more complex. In such environments, TCP/IP provides for multiple paths between computers and prevents unnecessary communication from crossing segment boundaries.

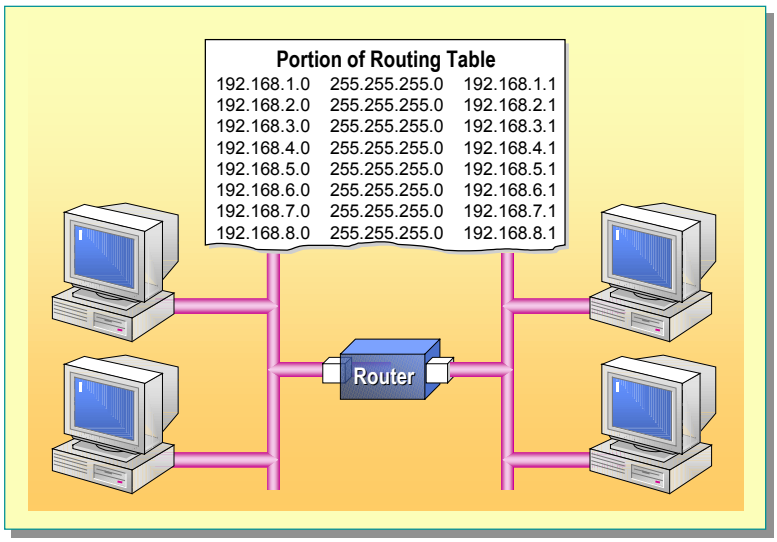
In an environment that has connected networks, the source and destination computers may not be on the same segment. IP determines whether the destination computer is local or remote in relation to the source computer. If the destination computer is remote, the data cannot be sent to it directly. Instead, IP sends it to a router, which then forwards the packet to its destination.

In this section, you will learn about the role of IP in the routing process and the process by which data is transmitted across routers.

# IP Routing

**Slide Objective**
To illustrate the role of IP in transferring data.

**Lead-in**
IP plays a very important role in transferring data across multiple network segments.




**Key Point**
Introduce the term internetwork here because the primary function of the TCP/IP protocol stack is to enable communication over a large area. Point out that routers are the key to understanding how an internetwork functions. Make sure students understand the difference between direct and indirect delivery, as this is an important aspect of the IP protocol.

Large TCP/IP networks, referred to as internetworks, are broken up into smaller segments to reduce the amount of communication within the segment. An *internetwork* is a network consisting of multiple segments that are connected by routers. Routers are basically computers with two network adapters that provide the primary means of joining two or more physically separated segments.

Routers pass IP packets from one network segment to another. This process of forwarding IP packets is known as routing. Routers are attached to two or more IP network segments, enabling packets to be forwarded from one segment to another.

## Packet Delivery

Forwarded IP packets use at least one of two types of delivery, based on whether the IP packet is forwarded to the final destination or whether it is forwarded to a router. These two types of delivery are known as direct and indirect delivery.

- Direct delivery occurs when a computer forwards a packet to a final destination on the same segment. The computer encapsulates the IP packet in a frame format for the network interface layer and addresses the packet to the destination's MAC address.
- Indirect delivery occurs when a computer forwards a packet to a router because the final destination is not on the same segment. The computer encapsulates the IP packet in a frame format for the network interface layer addressed to the IP router's MAC address.

## Routing Table

To determine where a packet is to be forwarded, routers use routing tables to send data between network segments. A routing table is stored in memory and maintains information about other IP networks and hosts. In addition, a routing table provides information to each local host about how to communicate with remote networks and hosts.

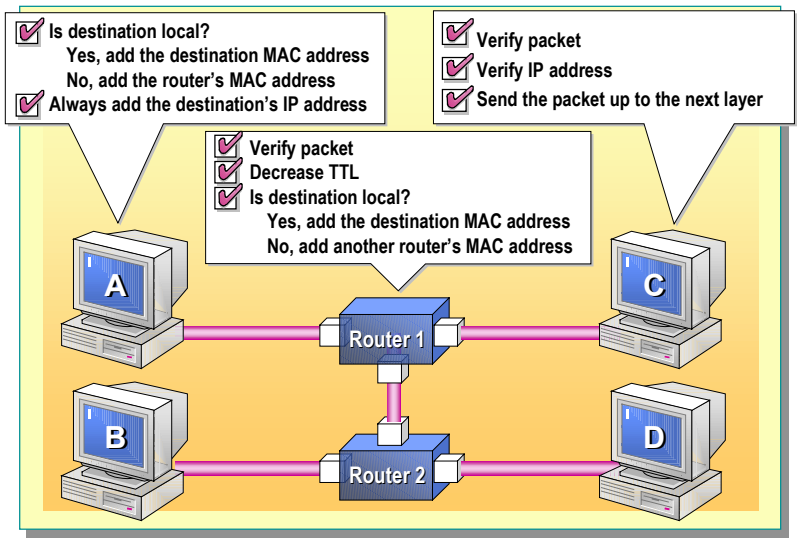
For each computer on an IP network, you can maintain a routing table that contains an entry for every other computer or network in communication with the local computer. However, this is not practical for large networks and a default router is used to maintain the routing table.

Routing tables can be either static or dynamic, depending upon the way they are updated. You update a static routing table manually. Because updating cannot be done often, the information in the routing table may not be current. On the other hand, a dynamic routing table is automatically updated as new information becomes available.

# Data Transfer Across Routers

**Slide Objective**
To illustrate the process of data transfer across routers.

**Lead-in**
Data is transmitted across interconnected networks through routers.




**Delivery Tip**
Use this slide as a conclusion to the analogy begun in the first section. Use the slide to first explain the process of transmitting data from computer A over router 1 to computer C. Then question the students about the process of transmitting data from computer A to computer B or D. Make sure to emphasize the placement of the second MAC address in the second example.

IP plays an important role in transmitting data across internetworks. Packets are exchanged and processed on each computer by using IP at the Internet layer on the source computer, at routers along the path to the destination, and at the destination computer.

To send data between two computers that are on different network segments, IP consults a local routing table for a route to the remote computer. If it finds a route, it sends the packet using that route. Otherwise, it forwards the data packets to its default router.

## IP at the Source Computer

In addition to adding such information as the TTL, IP always adds the IP address of the destination computer to the packet. In the case of a direct delivery, ARP is used to add the MAC address of the destination computer. In the case of indirect delivery, ARP is used to add the MAC address of the router to which the packet will be forwarded.

## IP at the Router

After the packet reaches a router, IP on the router determines where the packet is to be sent next. For this purpose, IP performs the following steps:

1. IP verifies the checksum and destination IP address.

   If the IP address is the router's IP address, the router processes the packet as the destination computer (IP at the destination).

2. IP then decreases the TTL and checks its routing table for the best route to the destination IP address.
3. In the case of a direct delivery, ARP is used to add the MAC address of the destination computer. In the case of indirect delivery, ARP is used to add the MAC address of the router to which the packet will be forwarded.

This entire process is repeated at each router in the path between the source and destination computer until the packet reaches a router on the same segment as the destination computer.

### Fragmentation and Reassembly

When a packet that is too large to be transmitted on the network arrives at a router, IP breaks up the packet into smaller packets before transmitting it onward. This process is known as fragmentation.

All of the small packets are then routed to the remote network. Even if they travel through multiple routers, the fragments are reassembled only when all of the small packets that make up the entire data transmission reach the destination computer. This process is known as reassembly.

## IP at the Destination

When a packet is received at the destination computer, it is passed up to IP. IP on the destination computer verifies the checksum and destination IP address. IP then passes the packet to either TCP or UDP. Finally, the packet is passed to the destination application, based on the port number, for final processing.

**Note** If at any time the TTL drops below zero or a step fails, such as if the destination application is not found, the packet is dropped and an ICMP packet may be returned. Although the delivery of an ICMP packet is not guaranteed, if TCP is used, then the original packet will be retransmitted.

# Lab B: Identifying Processes and Protocols in TCP/IP

**Slide Objective**
To introduce the lab.

**Lead-in**
In this lab, you will identify the processes that are involved in name resolution and the protocols involved in using the Ping utility across a router.

## Objectives

After completing this lab, you will be able to:

- Identify the processes involved with name resolution.
- Identify the protocols involved in using the Ping utility across a router.

## 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 (16-bit recommended).

### ► To start the lab

1. Log on 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 **Identifying Processes and Protocols in TCP/IP**.
5. Read the introduction information, and then click the link to begin the simulation.

**Estimated time to complete this lab: 15 minutes**

# 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.

- Introduction to TCP/IP
- TCP/IP Protocol Suite
- Name Resolution
- Examining the Data Transfer Process
- Routing Data

1. TCP/IP uses a four-layer communication model to transmit data from one location to another. What are the layers in the four-layer model used by TCP/IP?

   **Internet layer, application layer, transport layer, and network interface layer.**

2. When one application needs to communicate with an application on another computer, what does TCP/IP use to differentiate applications from one another, as well as to identify which computer they belong to?

   **It uses a socket.**

3. What three elements make up a socket?

   **An IP address, port, and the transport layer protocol.**

4. What are the six core protocols provided in the Microsoft TCP/IP suite?

   **TCP, UDP, ICMP, IGMP, IP, and ARP.**

5. Which of the six core protocols would you want to use if you need to have an application that accepts credit cards across the network and want to guarantee that the data arrives?

   **TCP.**

6. Which of the six core protocols is responsible for addressing and routing the data to its final destination?

   **IP.**

7. As the administrator of a network, you want to verify that the TCP/IP suite is installed properly and to test communications on the network. Which TCP/IP utility would you use?

   **Ping.**

8. If you wanted to use a user-friendly computer name versus an IP address to identify a computer, what are some of the storage locations that could map the computer names to IP addresses?

   **Hosts file, Lmhosts file, DNS, and WINS.**

9. When using indirect delivery to send a packet from a source computer to a destination computer, the source computer must first determine the MAC address belonging to _____.

   **A router.**