Installing, Configuring, Managing, Monitoring, and Troubleshooting Network Protocols

CERTIFICATION OBJECTIVES

6.01 Windows 2000 TCP/IP
6.02 IP Addressing
6.03 Installing, Configuring, Managing, and Monitoring TCP/IP
6.04 Installing and Configuring NWLink

Q&A Two-Minute Drill
Q&A Self Test
The Windows NT 4.0 certification track included an elective exam (taken by the vast majority of MCSE candidates) on the intricacies of the TCP/IP protocol suite and its implementation in various network environments. The new Windows 2000 track does not include an exam on this topic, which may lead some candidates to mistakenly believe that it is no longer necessary to study and understand TCP/IP. Nothing could be farther from the truth.

**Introduction**

There is a good reason for the fact that there is no TCP/IP elective in the Windows 2000 MCSE track: a thorough knowledge and understanding of this set of protocols is no longer optional. TCP/IP is the foundation upon which Windows 2000 networking is built, and it is the default protocol stack installed with the operating system. The components of the suite and how to use them will be integral topics in the Windows 2000 core exams, especially Exam 70-216, “Implementing and Administering a Windows 2000 Network Infrastructure.”

TCP/IP is not peculiar to Microsoft networks. In fact, not so long ago, TCP/IP was regarded as a somewhat sluggish, difficult-to-configure protocol used primarily by university or government networks participating in an exotic wide area networking project called ARPAnet. Few private organizations used it for their LANS because it was considered too slow and complex.

There were other protocols available that seemed to offer many advantages over the TCP/IP suite. Microsoft and IBM workgroups could use NetBEUI, a fast and simple transport protocol that could be set up easily and quickly by someone without a great deal of expertise. Novell NetWare networks, prior to NetWare 5.0, required the IPX/SPX stack, which was routable and thus could be used with larger server-based networks. Few business networks had any need for a powerful but high-overhead set of protocols like TCP/IP.

That was before the explosive popularity of the global wide area network we call the Internet. Inexpensive and easy to implement instant worldwide connectivity changed the lives of many people—and it changed the nature of networking. The growth of the Internet, more than any other single phenomenon, was responsible for the popularity enjoyed by TCP/IP in networking today. TCP/IP is the protocol stack on which the Internet runs.

In this chapter, we will focus primarily on the TCP/IP protocol suite. We’ll briefly discuss the history and evolution of TCP/IP, and the purposes of the various protocols included in the suite. We’ll pay particular attention to how to configure and use TCP/IP in a Windows 2000 network, and we’ll examine some common troubleshooting scenarios.
An important part of implementing and administering a TCP/IP-based network is working with IP addressing issues, and we will delve into this topic in some depth. We will include information on IP subnetting and supernettting, and talk about new developments on the TCP/IP front, such as Classless InterDomain Routing (CIDR) and IPv6, the future incarnation of the Internet Protocol.

Although TCP/IP is Microsoft's obvious protocol of choice for Windows 2000 local area networks, there are still many hybrid networks in existence, in which Windows 2000 machines must coexist with current and older versions of Novell NetWare. Microsoft provides support for NWLink, its own implementation of the IPX/SPX protocol stack that is necessary for connection to NetWare networks prior to version 5. We will look briefly at how to install and configure the NWLink protocol, and discuss the importance of binding orders when running multiple protocols on a Windows 2000 computer.

CERTIFICATION OBJECTIVE 6.01

**Windows 2000 TCP/IP**

The Transmission Control Protocol/Internet Protocol (TCP/IP) stack is often called the protocol of the Internet. It is also the protocol of choice for Windows 2000. The Windows 2000 implementation of TCP/IP is based on industry standards and designed to support networks of all sizes, up to the largest enterprise environments, as well as providing connectivity to the Internet.

Windows 2000 TCP/IP also includes a variety of built-in utilities used to configure, maintain, and troubleshoot the protocols, and to provide connectivity to many different types of systems, such as:

- Internet host computers
- Apple Macintosh systems
- IBM mainframe systems
- UNIX systems
- Open VMS systems
- Microsoft Windows NT and Windows 2000 computers
- Microsoft Windows 95 and 98 computers
Microsoft Windows for Workgroups computers

Microsoft LAN Manager networks

Network-ready printers, such as HP JetDirect-equipped printers

Windows 2000 TCP/IP also includes utilities such as File Transfer Protocol (FTP) and Telnet. FTP is a character-based application protocol that allows you to connect to FTP servers and transfer files. Telnet is an application that allows you to log in to remote computers and issue commands as if you were sitting at the keyboard of the remote computer. There are many variations of FTP, Telnet, and other programs based on earlier Internet standards available on the Internet as freeware/shareware or for purchase from third-party vendors.

We will take a more detailed look at these utilities and how they are used, later in this chapter.

Introduction to TCP/IP

Before we can understand the function of the TCP/IP protocols, we must first understand the role of protocols in computing and computer networking. A protocol is sometimes likened to a language that computers “speak” to communicate with one another, but a better analogy would be to think of the protocol as the syntax of a language. Protocols are sets of rules that specify the order and manner in which processes occur (in this case, the elements of the network communications process). A common protocol is necessary for two computers to “understand” one another.

Although TCP/IP is often referred to as “a” LAN protocol, in reality it is a set of protocols, also called a protocol stack or protocol suite. A stack consists of two or more protocols working together to accomplish a purpose (communication with another computer across a network). A suite is a more elaborate collection of communication protocols, utilities, tools, and applications. TCP and IP make up the stack, which handles the most important tasks of communication such as handling addressing and routing issues, error checking, and flow control. The suite includes a large number of additional protocols, used in various situations and for different purposes. Different vendors may include different tools and utilities in their implementations of the TCP/IP suite.

Some protocols that were developed specifically for the TCP/IP suite include Simple Mail Transfer Protocol (SMTP), Simple Network Management Protocol (SNMP), and File Transfer Protocol (FTP).
TCP/IP is known as an open standard protocol. In other words, it does not “belong” to any specific vendor, but is open to implementation by different companies. Thus we have not only Windows 2000 TCP/IP, but Novell’s TCP/IP stack, UNIX stacks, TCP/IP stacks that are designed to run on Macintosh systems or mainframe computers, and so on. In order to maintain compatibility across these different operating systems and environments, all these vendors must adhere to certain standards.

TCP/IP Standards
Standards and specifications relating to various aspects of the TCP/IP suite are published as RFCs (Requests for Comments) on the Internet and serve as guidelines to promote standardization. The Windows 2000 implementation of Microsoft TCP/IP supports a large number of RFCs that define how the protocols work. These documents are used to describe Internet standards, and go through a formal approval process before being adopted.

FROM THE CLASSROOM

RFCs
RFCs are submitted by any interested party and assigned an RFC number. Not all RFCs describe standards, but if a document is to become a standard, it goes through three stages: Proposed Standard, Draft Standard, and Internet Standard (RFC 2226, “Instructions to Authors,” contains information on how to write and format a draft). The Internet Engineering Steering Group (IESG) then reviews the document. The IESG is a part of the Internet Engineering Task Force (IETF). The IETF’s working groups (WGs) create a large number of the Internet Drafts. (For more detailed information, see www.ietf.org/home.html).

After review and approval, it is edited and published. The RFC editor, employed by the Internet Society, maintains and publishes a master list of RFCs, and is also responsible for final editing of the documents. The RFC Editor’s homepage is located at www.rfc-editor.org/. For more information about the RFC submission and approval process, see RFC 2026, at ftp://ftp.isi.edu/in-notes/rfc2026.txt. Request For Comments (RFC) 1180, available on the Web, provides an authoritative tutorial on the TCP/IP protocol suite.

—Debra Littlejohn Shinder, MCSE, MCP+I, MCT
Standards are also supported through the use of common networking models, such as the Open Systems Interconnection (OSI) model, developed by the International Organization for Standardization, and the DoD (Department of Defense) model, developed by the U.S. government in conjunction with the design of the TCP/IP protocols themselves during the creation of the ARPAnet. ARPAnet was a wide area network of U.S. military installations and major educational institutions that was the predecessor to today’s Internet.

Brief History of TCP/IP
In the 1960s, at the height of the cold war, the U.S. Department of Defense recognized that it would be valuable to establish electronic communications links between its major military installations, to ensure continued communication capabilities in the event of the mass destruction that would prevail if a nuclear war occurred. Major universities were already involved in their own networking projects. The DoD established the Advanced Research Projects Agency (ARPA), which funded research sites throughout the United States. In 1968, ARPA contracted with a company called BNN to build a network based on packet-switching technology.

A packet-switching network is one in which there is no dedicated pathway or circuit established. It is also known as a “connectionless” technology. If you send data from your computer to your company’s national headquarters in New York over a packet-switched network, each individual packet, or chunk of data, can take a different physical route to get there. Most traffic sent across the Internet uses packet switching.

The ARPAnet grew, as nodes (computers attached to the network) were added each year. Eventually the military network split off, calling itself MILNET. The remaining membership of the ARPAnet consisted primarily of an elite group of academics at major universities. However, in the late 1980s and early 1990s, the international network caught the eye of the business world. As commercial enterprises moved onto the network (which changed its name again during this period), access became less expensive and widely available to companies and individuals. The original ARPAnet thus evolved into today’s global Internet. According to most estimates, by 1999 there were over 50 million host computers connected to the Internet. Figure 6-1 illustrates the growth of the Internet.

From its ARPAnet beginnings, the Internet has grown into a huge, world-wide network. The protocol suite upon which the ARPAnet was built was the TCP/IP suite. The DoD designed it for that purpose, and the focus was reliability, rather
than speed. Despite later efforts to replace it with “better” protocols, such as the OSI suite, TCP/IP has endured. In addition to the Internet, the majority of medium to large networks today run on some implementation of the TCP/IP protocols.

**Advantages and Disadvantages of TCP/IP vs. Other LAN Protocols**

As noted, TCP/IP is relatively slow, it requires more overhead than most other LAN protocols, and it is also more difficult to configure and more complex to troubleshoot. Why, then, has it become so popular? Despite its shortcomings, the TCP/IP suite offers several advantages over other common LAN protocols such as IPX/SPX and NetBEUI.

**The TCP/IP Advantages**  
Reliability is TCP/IP’s strong suit—the Department of Defense designed it that way. It is highly appropriate for mission-critical
communications. But there are many other ways in which TCP/IP outdoes the competition and justifies the extra effort required to implement it:

- **Compatibility**  
  TCP/IP could almost be considered the universal protocol. It is supported by most operating systems and platforms, and allows highly diverse systems—such as Macintosh workstations, UNIX servers, and Windows computers—to communicate with one another. Connection to the Internet requires the TCP/IP protocols.

- **Scalability**  
  More than any other set of protocols in use, TCP/IP can scale from the smallest home network to the largest network of all: the Internet. Because of its unique addressing scheme, TCP/IP is especially suitable for large internetworks (networks that are interconnected with other networks).

- **Routability**  
  Closely related to scalability is the protocol stack’s capability of spanning subnets. Unlike unroutable protocols such as NetBEUI, its data packets can cross from one network, or subnet, to another by traveling through devices called routers. Internet communication often involves a journey through many different networks before the data reaches its destination.

**Disadvantages of TCP/IP**  
As already mentioned, compared to NetBEUI and NWLink (IPX/SPX), the TCP/IP protocols are slow. More resource overhead is required, and configuring the protocols correctly (IP address, subnet mask, default gateway) requires more knowledge and expertise. Most networking professionals feel that these are small prices to pay for TCP/IP’s flexibility and power.

**The TCP/IP Protocol Suite**
TCP and IP make up the protocol “stack” that gets the messages to their destination, and ensures that they get there reliably. However, an entire suite of protocols has come to be associated with the name and included with most vendors’ implementations.

Some of these are used to provide additional services, while others are useful primarily as information-gathering or troubleshooting tools. The different members of the suite work at different “layers” of the networking process. In order to understand this, let’s take a look at the concept of layered networking models.
TCP/IP and the DOD/OSI Networking Models

In the early days of computer networking, protocols were proprietary, that is, each vendor of networking products developed its own set of rules. This meant that computers using the same vendor’s products would be able to communicate with each other, but not with computers that were using the networking product of a different vendor.

The solution to this problem was to develop protocols that are based on open standards. Organizations such as the International Organization for Standardization (also called the ISO, which derives from the Greek word for “equal”) took on the responsibility of overseeing the definition and control of these standards, and publishing them so that they would be available to any vendor who wanted to create products that adhered to them. This is an advantage to consumers, because they are no longer forced to use the products of only one vendor. It can also benefit the vendors, in that its products are more widely compatible and can be used in networks that started out using a different vendor’s products.

Graphical models were developed to represent these open standards. Models provide an easy-to-understand description of the networking architecture and serve as the framework for the standards. The ISO’s OSI model has become a common reference point for discussion of network protocols and connection devices. Another widely used model is the DoD (Department of Defense) networking model, on which TCP/IP is based. Both of these are layered models that represent the communication process as a series of steps or levels. This layered approach provides a logical division of responsibility, where each layer handles prescribed functions.

The Open Systems Interconnection Model

The OSI model consists of seven layers. The data is passed from one layer down to the next lower layer at the sending computer, until the Physical layer finally puts it out onto the network cable. At the receiving end, it travels back up in reverse order. Although the data travels down the layers on one side and up the layers on the other, the logical communication link is between each layer and its matching counterpart, as shown in Figure 6-2.

As the data is passed down through the layers, it is enclosed within a larger unit as each layer adds its own header information. When it reaches the receiving computer, the process occurs in reverse; the information is passed upward through each layer, and as it does so, the encapsulation information is stripped off one layer at a time. After processing, each layer removes the header information that was added by its corresponding layer on the sending side.
Chapter 6: Installing, Configuring, Managing, Monitoring, and Troubleshooting Network Protocols

The process of enclosing data within a larger unit, with header information added by the protocol that is doing the enclosing, is called encapsulation.

By the time the data is finally presented to the Application layer, which then passes it up to the user application at the receiving computer, the data is once again in the form it was in when it was sent by the user application at the sending machine. Figure 6-3 shows how the header information is added to the data as it moves down through the layers.

The layers of the OSI model (from the top down), and a brief summary of the functions of each, are as follows:

- **Application**: This is the part of the networking component that interfaces with the user application.
- **Presentation**: This layer handles issues such as compression and encryption.
- **Session**: This layer is responsible for establishing a one-to-one connection, or session, between computers.
- **Transport**: The protocols at this layer handle error checking, flow control, and acknowledgments.
The Department of Defense Networking Model  

TCP/IP is often discussed in reference to the OSI networking model. However, the protocol suite was developed prior to development of the OSI model, and in conjunction with the DoD model. Therefore, the TCP/IP protocols do not map exactly to the seven OSI layers, but do map directly to the four layers of the DoD model.

Figure 6-4 shows how the OSI and DoD layers correlate.

Again beginning at the top and working our way down, the four DoD layers are as follows:

<table>
<thead>
<tr>
<th>Layer</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network</td>
<td>This layer is responsible for routing and logical addressing issues.</td>
</tr>
<tr>
<td>Data Link</td>
<td>This layer deals with the physical addressing and link establishment.</td>
</tr>
<tr>
<td>Physical</td>
<td>This layer interfaces with the hardware, and does not add headers to the data.</td>
</tr>
<tr>
<td>Application/Process</td>
<td>The Application layer of the DoD model corresponds to the top three layers of the OSI model, and handles the functions performed by the Application, Presentation, and Session layers. A number of the protocols included in the TCP/IP suite operate here, including File Transfer Protocol (FTP), Telnet, HyperText Transfer Protocol (HTTP), Simple Mail Transfer Protocol (SMTP), and others. Two Application Programming Interfaces (APIs) also reside here: NetBIOS and WinSock</td>
</tr>
</tbody>
</table>
Since the Presentation layer handles the very important task of protocol translation, this layer is where many gateways operate. The gateway acts as a translator, and allows computers using different protocols to communicate with one another. There are many types of gateway programs available that you may encounter in the field, such as:

**E-mail gateway**  This software translates the messages from diverse, noncompatible e-mail systems into a common Internet format such as the Simple Mail Transfer Protocol (SMTP).

**SNA gateway**  Systems Network Architecture is a proprietary IBM architecture used in mainframe computer systems such as the AS/400. An SNA gateway allows personal computers on a local area network to access files and applications on the mainframe computer.

**Gateway Services for Netware (GSNW)**  This software is included with Windows 2000 (and Windows NT) Server operating systems to allow the Windows server’s clients to access files on a Novell NetWare server. It translates between the SMB (Server Message Block) used by Microsoft to NCP (Netware Core Protocol) used by NetWare.
Host to Host (Transport)  This layer is basically the same as the Transport layer in the OSI model. It is responsible for flow control, acknowledgments, sequencing (ordering) of packets, and establishment of end-to-end communications. TCP and the User Datagram Protocol (UDP) operate at this level.

Internetwork  This layer matches the Network layer in the OSI model. The Internet Protocol (IP) works here to route and deliver packets to the correct destination address. Other protocols that operate at this layer include the Address Resolution Protocol (ARP), Reverse Address Resolution Protocol (RARP), and the Internet Control Message Protocol (ICMP).

Network Interface  This bottom layer of the DoD model corresponds to both the Data Link and Physical layers of OSI. It provides the interface between the network architecture (Ethernet, Token Ring, AppleTalk, etc.) and the upper layers, as well as the physical (hardware) issues.

The most critical members of the TCP/IP suite are the Network and Transport layer protocols (or Internetwork and Host-to-Host): TCP/UDP and IP.

The Internetwork Layer Protocols

The protocols that operate at the Internet layer of the DoD model (Network layer of the OSI model) handle logical (IP) addressing issues and routing.

Routers work at the Internetwork layer. A router can be a dedicated device, or you can configure a Windows NT or Windows 2000 computer to route IP packets by installing multiple network interface cards and enabling IP forwarding. Routers are necessary for communication to take place between computers that are not on the same network (subnet).

IP routing involves discovering a pathway from the sending computer (or forwarding router) to the destination computer whose address is designated in the IP header.

The protocol most commonly associated with this layer is IP, the Internet Protocol (IPX, as part of the IPX/SPX stack, also operates at this layer, and Windows 2000 also supports IPX routing).

IP  IP is a connectionless protocol; this means it must depend on TCP at the Transport layer above it to provide a connection if necessary.
A connection-oriented protocol is one that establishes a direct connection before sending data. A connection-oriented protocol works something like a phone call: If you wish to have a conversion with Mr. Smith, you would dial his number, ask for him, and verify that he is the party to whom you’re speaking before plunging into the discussion. Connectionless protocols work more like sending a postcard: You write your message, address it to Mr. Smith, and drop it in the mailbox, hoping it will reach its destination. Mr. Smith was not aware that the message was coming until it arrived, and you have no way of knowing whether or not Mr. Smith received your communication.

Although IP does not establish a connection or acknowledgment receipt of messages, it is able to use number sequencing to break down and reassemble messages, and uses a checksum to perform error checking on the IP header.

ICMP and IGMP

The Internet Control Message Protocol (ICMP) is a TCP/IP standard that allows hosts and routers that use IP communication to report errors and exchange limited control and status information. The PING utility (discussed later in this chapter) works by sending an ICMP echo request message and recording the response of echo replies.

The Internet Group Management Protocol is used for multicasting, which is a method of sending a message to multiple hosts but only addressing it to a single address. Members of a multicast group can be defined, and then when a message is sent to the group address, only those computers that belong to the group will receive it. IGMP is used to exchange membership status information between IP routers that support multicasting and members of multicast groups.

ARP and RARP

The Address Resolution Protocol (ARP) is used to resolve IP (logical) addresses to Media Access Control (MAC) physical hardware addresses. ARP uses broadcasts to discover the hardware addresses, and stores the information in its arp cache.

RARP is the Reverse Address Resolution Protocol, which does the same thing in reverse; that is, it takes a physical address and resolves it to an IP address. The arp –a command can be used to view the current entries in the ARP cache. See Figure 6-5 for an illustration of this IP address to MAC address list.

The Transport Layer Protocols

Remember that the Transport layer’s primary responsibility is reliability; it must verify that the data arrives complete and in good condition. It also must have a way
to differentiate between the communications that may be coming to the same network address (the IP address) from—or to—different applications.

There are two protocols in the TCP/IP suite that operate at the Transport layer: the Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP). TCP is called a connection-oriented protocol, and UDP is a connectionless protocol. A connection-oriented protocol such as TCP offers better error control, but its higher overhead means a loss of performance. A connectionless protocol like UDP, on the other hand, suffers in the reliability department but, because it doesn’t have to bother with error-checking duties, is faster.

**TCP**

TCP is based on point-to-point communication between two network hosts. This means a session is established before data transmission begins. This is done using a process called a three-way handshake. This is a way of synchronizing communications and establishing a virtual connection.

TCP processes data as a stream of bytes, which are divided into groups called segments. TCP bytes are grouped into segments that TCP then numbers and sequences for delivery. TCP sends acknowledgments when segments are received, to let the sending computer know that the data arrived. If data segments arrive out of sequence, TCP/IP can reassemble them in the correct order. If a segment fails to arrive, TCP lets the sending computer know so that segment can be sent again.

**UDP**

UDP provides a service similar to that of TCP, but it does so in a different way. UDP is a connectionless protocol, which offers what is called
Chapter 6: Installing, Configuring, Managing, Monitoring, and Troubleshooting Network Protocols

best-effort delivery. This means that UDP does not guarantee delivery, nor does it verify sequencing. If a sending host needs reliable communication, it should use either TCP or a program that provides its own sequencing and acknowledgment services at the Application level.

Applications that need to send only a small amount of data at a time, or those that place a priority on speed of transmission rather than reliability, use UDP.

Ports and Sockets   Thanks to the multitasking capabilities of Windows 2000 and other modern operating systems, you can use more than one network application simultaneously. For example, you can use your Web browser to access your company’s homepage at the same time your e-mail software is downloading your e-mail. You probably know that TCP/IP uses an IP address to identify your computer on the network, and get the messages to the correct system, but how does it separate the response to your browser’s request from your incoming mail when both arrive at the same IP address?

That’s where ports come in. Remember we said that the two parts of an IP address that represent the network identification and the host (individual computer) identification are somewhat like a street name and an individual street number. In this analogy, the port number designates the specific apartment or suite within the building.

TCP and UDP, the Transport layer protocols, both use port numbers to ensure that the data intended for Apartment A doesn’t get sent to Apartment B instead.

A socket is the combination of an IP address and a port number.

TCP Sliding Windows   TCP is a reliable protocol, and as a result, in a TCP communication, every segment sent must be acknowledged. That way, if one segment doesn’t arrive at its destination (and thus the receiving computer does not send back an acknowledgment for it), it will be sent again.

TCP has to have a way to control the “flow” of data transmission when multiple TCP connections have to share a busy link. Flow control is necessary so that the receiving computer doesn’t get “overwhelmed” by a sending computer that deluges it with data faster than it can be processed, or alternately so that the receiver doesn’t sit around waiting for the data to “trickle” in.

Flow control is the process of matching the outflow of data from the sending computer to the receiving computer’s inflow. This is done by setting a limit on the number of packets that can be sent before acknowledgment is required, which signals the sender to slow down (or stop and wait) if data is “piling up” in the
receiver’s buffer. If the buffer overflows, data will be lost and must be retransmitted. Think of flow control as the effective management of the data flow between devices in a network so that the data can be handled at an efficient pace.

In the TCP communication process, those bytes of data that can be considered active are called the “window.” These are the bytes that are ready to be sent, or they have been sent and are awaiting acknowledgment. As acknowledgments are received, the window “slides” past those bytes, to send additional bytes. The sliding window protocol determines how much data is being transmitted based on actual bytes, rather than segments. See Figure 6-6 for an illustration of how the sliding window concept works.

**Other Members of the Suite**

There are several other protocols that belong to the TCP/IP suite. Many of these operate at the Application layer, and are used for such tasks as transferring files, remote terminal emulation, messaging, and network management. Some of these protocols include:

- **File Transfer Protocol (FTP)** Used to download files from another computer, or to upload files to another computer.
- **Telnet** Used to connect to a remote computer and run programs or view files.

![Figure 6-6](image-url)
Simple Mail Transfer Protocol  Used for sending Internet mail (usually used in conjunction with the Post Office Protocol (POP), which is used to retrieve incoming mail from the mail server.

Simple Network Management Protocol (SNMP)  Used to monitor and manage TCP/IP networks. SNMP has two components, the SNMP Agent and the SNMP Management System, which use SNMP messages sent using UDP to communicate host information, which is stored in a Management Information Base (MIB).

The Application Programming Interfaces, APIs, are called boundary layers in Microsoft's own Windows networking model. The two supported APIs in Windows 2000 networking are NetBIOS and WinSock. NetBIOS communications use a destination name (called a NetBIOS name) and a message location to get the data to the correct destination. NetBIOS supports a session mode, for establishing a connection and transfer of large messages, and a datagram mode, for connectionless transmissions such as broadcast messages. A WinSock program handles input/output requests for Internet applications in a Windows operating system, using the sockets convention for connecting with and exchanging data between two processes. WinSock runs as a .dll file (dynamic link library). A .dll file is a collection of small programs, any of which can be loaded when an application needs to use it, but it isn’t required to be included as part of the application.

The following answers some common questions about the responsibilities of various layers of the networking models.

### SCENARIO & SOLUTION

<table>
<thead>
<tr>
<th>Question</th>
<th>Layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>At what networking layer do encryption and data compression take place?</td>
<td>The OSI Presentation layer</td>
</tr>
<tr>
<td>Which networking layer is responsible for acknowledgment of receipts, flow control, and sequencing of packets?</td>
<td>The OSI Transport layer</td>
</tr>
<tr>
<td>At what layer are hardware issues handled?</td>
<td>The OSI Physical layer</td>
</tr>
<tr>
<td>Which layer deals with routing and logical addressing?</td>
<td>The OSI Network layer</td>
</tr>
</tbody>
</table>

**CertPrs8 / MCSE: Windows 2000 Network Administration Study Guide (Exam 70-216) / Syngress / 2383-4 / Chapter 6**
EXERCISE 6-1

Using the Windows 2000 FTP Client

To use the built-in FTP client in Windows 2000, perform the following steps:

1. Click the Run selection on the Start menu.
2. Type cmd in the Run box to invoke a command prompt.
3. At the command line, type ftp.
4. You should see a prompt displayed as ftp>.
5. At the prompt, type open ftp.microsoft.com.
6. You will be prompted for a username. Type anonymous.
7. You will receive a message that anonymous connections are allowed, and asked to give your e-mail address. Type in your e-mail address.
8. You will be welcomed to the Microsoft FTP site.
9. To get a list of files available for download, type dir. You will see a list such as that shown in the following illustration.
You can use the GET command to specify a file to be downloaded, or the PUT command to specify a file to be uploaded. For a complete listing of FTP commands, type help at the ftp prompt.

CERTIFICATION OBJECTIVE 6.02

IP Addressing

The IP address is a logical address, assigned by the network administrator. It bears no direct relation to the network interface card’s physical address (called the MAC address because it is used at the Media Access Control sublayer of the OSI’s Data Link layer). The MAC address is hard-coded into a chip on the network card in the typical Ethernet network. The Address Resolution Protocol (ARP), which was discussed earlier in this chapter, has the task of translating IP addresses to MAC addresses.

Locating IP Addressing Information

There are a couple of ways to find out what a computer’s IP address is. TCP/IP configuration information is found in the Properties box for the protocol. Windows 2000 TCP/IP also includes a utility, IPCONFIG, which displays the computer’s IP address and other TCP/IP configuration information.

If your network contains client computers that use older Microsoft operating systems, you should be aware that the IPCONFIG command also works with Windows NT (all versions) and with Windows for Workgroups 3.11. However, to view IP configuration information in Windows 95 or 98, you must use a different command: WINIPCFG. This will display the same information, but in a Graphical User Interface (GUI) dialog box.
Finding Your Computer's IP Addressing Information

To determine your computer’s IP address and related information (subnet mask and default gateway), follow these steps:

Locating IP information via the TCP/IP Properties box.

1. Select Start | Settings | Network and Dial-up Connections.
2. In the Network and Dial-up Connections Folder, right-click on your local area connection and select Properties, shown as follows.

3. In the Properties dialog box, double-click on Internet Protocol (TCP/IP), or highlight it and click Properties.
4. You will see a dialog box displayed similar to the one shown next.

5. Note the assigned IP address, subnet mask, and default gateway information. If the “Obtain an IP address automatically” radio button is selected, a DHCP (Dynamic Host Configuration Protocol) server on the network will assign the IP address and other TCP/IP configuration information. In that case, you will have to use the second method to determine the IP address being used by the computer.

Locating IP information via the IPCONFIG command.

The best way to determine IP addressing information is by using the TCP/IP command-line utility IPCONFIG. To do so, follow these steps:

6. Bring up a command window (select Start | Run and type cmd).
7. Type ipconfig at the prompt.
8. You will see information displayed similar to that shown in the following illustration.

The IP address and subnet mask are always required for TCP/IP communication. The default gateway value is required to communicate on a routed network.

How IP Addressing Works

In order to communicate over the network using the TCP/IP protocols, a computer must have an IP address that is unique on that network. A network administrator can manually assign the IP address, or it can be automatically assigned by an addressing service such as DHCP, APIPA (Automatic Private IP Addressing), or ICS (Internet Connection Sharing).

The IP address is usually represented as shown in the screenshots, in “dotted decimal” (also called “dotted quad”) notation with four sections, called octets, separated by dots. This decimal notation is merely a “user friendly” way to express the binary number used by the computers to communicate. The octets are called that because each represents eight binary digits.
To identify which octet we’re talking about, they are often referred as the “W,” “X,” “Y,” and “Z” as follows:

w.x.y.z

**Ones and Zeros: Binary Addressing**

Let’s take a look at how IP addresses look in binary. This will help you to understand what the numbers really represent and how the computer uses them for communication.

For example: The IP address 192.168.1.185 really represents the following binary number: 11000000.10101000.00000001.10111001.

This number is made up of four groups of eight binary digits, the octets mentioned earlier. Binary uses only two digits—0 and 1—to represent all numerical
values. Binary is a *base two* system, as opposed to decimal, which is a *base ten* system because it uses ten digits—0 through 9—to represent all numerical values.

How do you convert decimal to its binary equivalent? Well, you could just use the Windows calculator in scientific mode (choose “Scientific” from the View menu). Check the dec radio button and enter the number in decimal, then click on the bin radio button and Tada! As if by magic, you have the binary equivalent.

But you also need to know how to perform the calculation without the assistance of a calculator. It’s really not as difficult as you may think.

**Converting Decimal to Binary**

Let’s take a look at an octet:

```
11111111
```

We have eight binary digits, and each of them represents a decimal value, beginning with the rightmost digit and working our way back to the leftmost. The rightmost digits are sometimes referred to as the low order bits, and the leftmost as the high order bits.

Each bit that is “turned on” (that is, shows a 1 instead of a 0) represents the value of that bit as shown in Table 6-1.

You’ll notice that the value increases by a power of 2 as you move from right to left. A bit that is turned “off” (represented by a 0) counts as 0. All we have to do then is add up the values of the bits that are “on.”

If the octet we wish to convert is 10011011, we would add 1 + 2 + 8 + 16 +128 (the values of all the bits that are turned “on”), for a total of 155. So, 10011011 equals 155 in decimal notation.

<table>
<thead>
<tr>
<th>TABLE 6-1</th>
<th>Values of Binary Digits in an Octet</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 1 1 1 1 1 1 1</td>
<td>128 64 32 16 8 4 2 1</td>
</tr>
</tbody>
</table>

The Components of an IP Address

What does this mean, then, in terms of the IP addresses we work with every day? Generally, when we configure TCP/IP properties, we enter IP addresses in dotted decimal notation. An IP address in its “pure” binary form consists of four octets (each octet being made up of eight binary digits), or 32 bits. The dotted decimal form shows the octets converted to their decimal equivalent with each octet separated by a dot. Thus, the address that the computer sees as 10011110 11101000 00011001 11111001 will be expressed as 158.232.25.249 (do the calculations as shown earlier or use the scientific calculator to make the conversion).

Network and Host ID

This address is really made up of two parts, just as your street address contains both the house number and the street name; for example, 123 Main Street. Many houses share the “street name” portion of the address (everyone else on your street). There may also be other houses in your neighborhood that have the house number “123,” but they will be on different streets. It’s the combination of the house number and street name that makes up the unique identifier that describes to others which house is yours.

IP addresses work in a similar fashion. Part of the address is the network ID, which identifies the network (or subnet) on which the computer is located. All computers on the subnet share this part of the address. The second part of an IP address is called the host ID, and identifies the individual computer on that network or subnet. Combined, they create a unique address that differentiates this computer from all others on the internetwork.

The Role of the Subnet Mask

How do we know which of the octets, or parts of the IP address, indicate the network ID, and which ones indicate the host ID? It would be easier if, for instance, the first octet always indicated the network, and the last three always indicated the host. Unfortunately, it’s not that simple. However, we can determine what part of the IP address pertains to which by taking a look at our subnet mask.

The subnet mask is another 32-bit binary number, expressed in the same form as an IP address, but its purpose is to tell us (and more importantly, to tell the computers) which part of the IP address is masked (and thus represents the network ID). In the binary form of the subnet mask, the masked bits are those that are “on,” or set to 1.

If the first eight bits from the left in the subnet mask (the first octet) are all ones, and the rest of the bits are zeros, that means the first octet represents the network ID
and the remaining three octets represent the host ID. Let’s convert that to decimal, since we usually see the subnet mask expressed in dotted decimal in the TCP/IP configuration.

11111111.00000000.00000000.00000000 = 255.0.0.0.

This generally means the first portion of the IP address identifies the network on which the computer “lives,” and the last three parts identify the specific computer (host) on that network. In other words, if our IP address is 103.24.125.6 with a subnet mask of 255.0.0.0, the first octet (103) identifies the network, and the remaining three (24.125.6) identify the host computer on that network.

Certain addresses are used for special purposes. A host number of all 0s is used to identify the network, and a host number of all 255s is used as the broadcast address, to send messages to all computers on that network.

There is one more thing we must factor in: the address class to which the IP address belongs.

**Address Classes**

In order for computers to communicate on a worldwide global internetwork like the Internet, which requires that each computer have a unique IP address, there must be some centralized authority in charge of assigning addresses and ensuring that none are duplicated. This has been handled by the Internet Assigned Numbers Authority (IANA) and the InterNIC, a company tasked with that responsibility. Traditionally, blocks of IP addresses have been assigned in “lumps” to organizations and Internet Service Providers (ISPs), depending on how many host addresses were needed for their networks.

These blocks of addresses came in three basic sizes: large, medium, and small. The networks for which these blocks of addresses were assigned were called Class A, B, and C networks.

**Class A Addresses**

Class A addresses are for the “large size” networks, those that have a tremendous number of computers, and thus a need for many host addresses. Class A addresses always begin with a 0 in the first octet (also called the W octet). This will be the first
bit on the left. This leaves seven bits for the individual network ID, and 24 bits to identify the host computers. When we convert to decimal, we see that this means a Class A address will have a decimal value in the first octet of 127 or less.

**Class A networks can be assigned addresses with a first octet of 1–126. The 127.0.0.0 network, although technically a Class A, is reserved for use as the “loopback Network ID.” This is a test Network ID used to troubleshoot TCP/IP connectivity. The address 127.0.0.1 is generally known as the “loopback” address, but a message sent to any valid IP address with the loopback network will “loop back” to the sender (regardless of the sender’s IP address). Unfortunately, this means the more than 24 million additional addresses in the 127.0.0.0 network cannot be assigned and are wasted.**

Class A addresses, because they use only the first octet to identify the network, are limited in number. However, each Class A network can have a huge number of host computers, over 16 million. The Class A network numbers were all used up some time ago; they have been assigned to very large organizations such as IBM, MIT, and General Electric.

**Class B Addresses**

Class B networks are the “medium size” networks. Class B networks use the first two octets (the 16 leftmost bits) to identify the network, and the last two octets (or the 16 rightmost bits) to identify the host computers. This means there can be far more Class B networks than Class As (over 16,000), but each can have fewer hosts (“only” 65,535 each). Class B addresses always begin with a 10 for the two leftmost bits in the W octet, and the network is defined by the first two octets, which translates to decimal values of 128 through 191 for the first octet. 16 bits identify the Network ID, and the remaining 16 bits identify the Host ID. Microsoft’s network is an example of a Class B network.

**Class C Addresses**

The smallest sized block of addresses designated by a class is the Class C network, each of which can have only 254 hosts. However, there can be over 2 million Class C networks. A Class C network always has 110 as its first three bits. This leaves 24
bits to identify the network, with only 8 bits to use for host IDs. A Class C network, in decimal notation, will have a first octet decimal value of 192 through 223.

**Don't be confused if you read in some texts that the network ID in a Class B network is identified by 14 bits rather than 16, or in a Class C by 21 instead of 24. Technically this is correct—the first 2 bits define the address class, and the next 14 define the individual network. To simplify our understanding of addressing, these two are usually referred to together as the “network ID.”**

There are many, many class C networks. Most Internet Service Providers (ISPs) have been assigned Class C network numbers.

---

**The address ranges 10.x.x.x, 172.16.x.x, and 192.168.x.x are reserved for use as private addresses. That is, these address ranges cannot be assigned by the Internet authorities to any network connected to the public Internet, but can be used as internal addresses that are not connected to the public network, without being required to be registered. Private addresses cannot send to or receive traffic from the Internet—at least, not directly. If a LAN is using private addresses, and the computers on the LAN need to communicate with Internet locations, the private addresses must be translated to a public address. NAT (Network Address Translation) software is used for this purpose, and Windows 2000 includes built-in NAT support.**

---

**Class D and E Addresses**

We said there are three network sizes, so where do the Class D and E addresses fit in? These two classes are **not** assigned to networks, but are reserved and used for special purposes.

Class D addresses, whose four high order (leftmost) bits in the W octet are 1110, are used for **multicasting.** This is a method of sending a message to multiple computers simultaneously.

Class E addresses, with four high order bits of 1111, are reserved to be used for experimental and testing purposes.

The following provides a quick reference for defining the ranges of IP address classes.
Chapter 6: Installing, Configuring, Managing, Monitoring, and Troubleshooting Network Protocols

Default Subnet Masks

When an entire block of addresses from a specified class is assigned and used as one network (either a Class A, B, or C), the subnet mask is easy to determine and understand. Either the first, first two, or first three octets are “masked”; that is, all bits in those octets are 1s (turned “on”), indicating that those bits represent the network ID. The subnet masks used in these cases are called the default subnet masks for each address class. The default masks are as follows:

- **Class A** 255.0.0.0 (11111111.00000000.00000000.00000000)
- **Class B** 255.255.0.0 (11111111.11111111.00000000.00000000)
- **Class C** 255.255.255.0 (11111111.11111111.11111111.00000000)

Often, however, a block of addresses (such as the 254 addresses available in an assigned Class C network) needs to be split into two or more smaller networks. This is called subnetting. There are many reasons for subnetting a network, one of which is to cut down on broadcast traffic (broadcast messages go only to the computers on the same subnet) and make better use of network bandwidth.

It is also possible to do the opposite: combine two or more Class C networks together to create a larger network. This is referred to as supernetting.

### SCENARIO & SOLUTION

<table>
<thead>
<tr>
<th>Question</th>
<th>Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is the first octet range for Class A networks?</td>
<td>1–126</td>
</tr>
<tr>
<td>What is the first octet 127 used for?</td>
<td>The loopback address, for verifying that the TCP/IP stack is installed and configured properly.</td>
</tr>
<tr>
<td>What is the first octet range for Class B networks?</td>
<td>128–191</td>
</tr>
<tr>
<td>What is the first octet range for Class C networks?</td>
<td>192–223</td>
</tr>
<tr>
<td>What does a first octet value that exceeds 223 indicate?</td>
<td>Addresses used for special purposes, such as multicasting and experimental use, which are not assigned to networks (these are called Class D and E addresses).</td>
</tr>
</tbody>
</table>
Remember that the default subnet masks indicate unsubnetted networks only when applied to the network class listed. This means the subnet mask of 255.255.0.0 when applied to a Class B network indicates an unsubnetted network. However, the same mask of 255.255.0.0, if applied to a Class A network, would be a subnetted network. The network class is always determined by the high order (leftmost) bits, as discussed earlier. A common mistake for new administrators is to assume that if the subnet mask is 255.255.255.0, for example, the network is a Class C network.

Subnetting and Supernetting

Both subnetting and supernetting are ways of modifying the IP address by “stealing” bits from one portion (network ID or host ID) to “give” to the other. To do this, you must use a variable length (or custom) subnet mask to indicate which bits in the IP address pertain to the network ID and which to the host. Routers use the subnet mask to determine to which subnetwork a data packet should be sent.

Subnetting a network turns it into a routed network, as an IP router (either a dedicated device or a computer configured to function as a router) will be required for computers on one subnet to communicate with the computers on other subnets.

Subnetting Basics

If you are allocated an entire Class C network, remember that the default subnet mask is 255.255.255.0, or in binary, 11111111.11111111.11111111.00000000. The eight bits on the right, represented as zeroes, are “yours.” You can use all of them for host addresses, or you can “loan” some of them to the network ID, to divide your Class C network into two or more smaller networks.

To understand variable-length subnet masks, which indicate that the network is divided into subnets, you must work with the binary or you will probably end up hopelessly confused. Variable-length subnet masks are created by taking bits from the portion of the IP address normally used for the host ID and using it for the
network (or subnet) ID. For instance, if you borrow four bits from the host portion of a class C network address, your subnet mask will look like this:

11111111 11111111 11111111 11110000

or, in decimal:

255 255 255 240

This technique allows us to divide our Class C network into 14 usable subnets with 14 hosts on each subnet, using the following formula:

Number of subnets = \(2^x - 2\), where \(x\) = the number of bits borrowed from the host ID.

Number of hosts = \(2^x - 2\), where \(x\) = the number of unmasked host ID bits remaining.

Note that we subtract two from the number of subnets, because conventional IP subnetting rules say we can’t have a subnet ID that is all 1s or all 0s. Thus we must “throw out” the first and last subnet IDs. We also subtract two from the number of hosts, because two host addresses are always reserved for use as the network ID and the broadcast address.

**Determining the Number of Subnets** The first step in creating a subnetted network is to decide how many subnets you want to define. Remember that the more bits you “steal” from the host ID portion of the address, the more subnets you can create—but this reduces the number of hosts you can have per subnet. See Table 6-2.

<table>
<thead>
<tr>
<th>Subnets</th>
<th>Bits Needed</th>
<th>Mask</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2</td>
<td>192</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>224</td>
</tr>
<tr>
<td>14</td>
<td>4</td>
<td>240</td>
</tr>
<tr>
<td>30</td>
<td>5</td>
<td>248</td>
</tr>
<tr>
<td>62</td>
<td>6</td>
<td>252</td>
</tr>
<tr>
<td>126</td>
<td>7</td>
<td>254</td>
</tr>
<tr>
<td>254</td>
<td>8</td>
<td>255</td>
</tr>
</tbody>
</table>
Table 6-2 illustrates how many new subnets can be created for each bit that you “steal” from the host ID. Use the formula $2^n - 2$ (where $n$ is the number of bits that are available to be used for the host ID) to figure out the number of host addresses you will have. Remember that Class A addresses have 24 bits minus the number of bits used for the mask, Class B addresses have 16 bits minus the number used in the mask, and Class C addresses have 8 bits minus the number used in the mask.

**Determining the Mask**  There are three basic steps involved in determining the appropriate subnet mask:

1. Determine the number of subnets you want.
2. Convert the number to binary. Notice how many bits were required.
3. Convert the number of bits required to decimal.

**About Supernetting**  
Supernetting is a way of combining several small networks into a larger one. For example, a company may need a Class B network, but because those have all been assigned, it can’t get one. However, Class C networks are available, so the company can assign multiple Class C networks with contiguous addresses. By “stealing” bits again, but in the opposite direction (sort of like taking from the poor and giving to the rich instead of vice versa), you can use some of the bits that originally represented the network ID to represent host IDs, reducing the number of networks but increasing the number of hosts available per network.

For instance, you can combine two Class C networks using a subnet mask of 255.255.254.0, to provide for 512 hosts on the network instead of the 254 to which a Class C network is traditionally limited. Or, we could combine 1024 Class C networks with a subnet mask of 255.252.0.0 and obtain 262,144 host addresses (although we probably wouldn’t want to).

The addresses of the two Class C networks must be contiguous for this to work.

Supernetting often is used in conjunction with an IP addressing scheme called CIDR.
Classless Addressing: CIDR

The use of address classes is the traditional way of working with IP addressing and subnetting. A more recent development is called Classless InterDomain Routing, abbreviated as CIDR (and pronounced “cider”).

One Internet resource describes CIDR as “subnetting on steroids.” CIDR networks are referred to as “slash x” networks, with the “x” representing the number of bits assigned originally as the network ID (before subnetting). Think of this as the number of bits that don’t “belong” to you.

With CIDR, the subnet mask actually becomes part of the routing tables. CIDR allows us to break networks into subnets and combine networks into supernets.

A traditional Class C network, you’ll recall, contained 8 bits in the IP address that you could use as you wished for host IDs or subnetting, leaving 24 bits that were not under your control. Using CIDR, this would be designated as a “slash 24” network. Thus, a CIDR IP address would look like this: 192.168.1.27/24. Using the same formula, a traditional unsubnetted Class A network would be a /8, and a traditional unsubnetted Class B would be designated as /16. Of course, subnetted networks that would use variable length subnet masks are also designated in the same way. See Table 6-3 for the correlation of the subnet masks to the “slash x” designations.

### Table 6-3

<table>
<thead>
<tr>
<th>“Slash x” Designation</th>
<th>Subnet Mask</th>
</tr>
</thead>
<tbody>
<tr>
<td>/8</td>
<td>255.0.0.0</td>
</tr>
<tr>
<td>/12</td>
<td>255.240.0.0</td>
</tr>
<tr>
<td>/16</td>
<td>255.255.0.0</td>
</tr>
<tr>
<td>/20</td>
<td>255.255.240.0</td>
</tr>
<tr>
<td>/21</td>
<td>255.255.248.0</td>
</tr>
<tr>
<td>/22</td>
<td>255.255.252.0</td>
</tr>
<tr>
<td>/23</td>
<td>255.255.254.0</td>
</tr>
<tr>
<td>/24</td>
<td>255.255.255.0</td>
</tr>
<tr>
<td>/25</td>
<td>255.255.255.128</td>
</tr>
<tr>
<td>/26</td>
<td>255.255.255.192</td>
</tr>
<tr>
<td>/27</td>
<td>255.255.255.224</td>
</tr>
<tr>
<td>/28</td>
<td>255.255.255.248</td>
</tr>
<tr>
<td>/29</td>
<td>255.255.255.252</td>
</tr>
<tr>
<td>/30</td>
<td>255.255.255.254</td>
</tr>
</tbody>
</table>
CERTIFICATION OBJECTIVE 6.03

Installing, Configuring, Managing, and Monitoring TCP/IP

In order to put all this theory into practice, you must first install (if you haven’t already) and configure the TCP/IP protocol on your Windows 2000 computer. Network protocols are installed via the Network and Dial-up Connections window (this is different from NT 4.0, where you could right-click on Network Neighborhood and bring up the Properties sheet to install new protocols). In this case, you will select your local area connection, right-click, and choose Properties.

Installing TCP/IP in Windows 2000

The Properties sheet will list the networking protocols and components that are already installed, and will allow you to install, uninstall, and configure the properties of your networking components (Figure 6-7).

In the following exercise, we will walk through the steps of installing and configuring TCP/IP on a Windows 2000 computer. Note that you must be logged on with administrative privileges to install TCP/IP or other network protocols.

FIGURE 6-7
The Properties sheet for the Local Area Connection
EXERCISE 6-3

Installing TCP/IP in Windows 2000

1. Select Start | Settings | Network and Dial-up Connections.
2. In the windows displaying the contents of the Network and Dial-up Connections folder, right-click on your local area connection and choose Properties.
3. Click Install, and you will see a dialog box as shown in the following illustration.

4. A list of available protocols will be shown, as shown next.

5. Select Internet Protocol (TCP/IP).
6. Click OK. You may be prompted for the Windows 2000 installation CD. The protocol will be installed, and can now be configured.
Configuring TCP/IP in Windows 2000

Once TCP/IP has been installed, you must enter the proper configuration information before the computer can communicate on the network using the TCP/IP protocols. The following exercise will walk you through the process.

Configuring TCP/IP

Return to the Properties sheet for the local area connection, and select Internet Protocol (TCP/IP), then double-click or click Properties. You will then see the TCP/IP Properties dialog box, as shown in the following illustration.

1. If your network has a DHCP server that will be used to obtain an IP address for this computer, select the “Obtain an IP address automatically” radio button. The DHCP server will provide both the IP address and other configuration information, such as the subnet mask and default gateway. If the network does not have a DHCP server, or you will not use the DHCP server to obtain an address automatically, skip to Step 4.
The default gateway address must be on the same subnet as the computer’s IP address.

Advanced TCP/IP Properties
Windows 2000 will allow you to more finely tune your TCP/IP settings. When you click Advanced in the Properties box, you will see the tabbed Advanced TCP/IP Settings Properties sheet shown in Figure 6-8. The Advanced Settings allow you to further configure the IP settings, DNS, WINS, and other options.

IP Settings Under Advanced IP Settings, you can configure the computer to use more than one IP address or default gateway. Note that you can add, remove, or edit the properties of both in this dialog box.
Assigning Multiple IP Addresses  You can use multiple IP addresses in various situations, such as public addresses used for the Internet and private addresses used for an internal network, or for multiple logical IP networks on the same physical network segment.

Assigning Multiple Default Gateways  Windows 2000 supports a feature called dead gateway detection, which is used to detect routers that have gone down. If multiple default gateways are configured, a failing TCP connection will update the IP routing table with the next default gateway in the list. Although you can assign multiple gateways, the second (or subsequent) gateway(s) will be used only if the first fails. In other words, more than one default gateway cannot be active simultaneously.

The Interface Metric  You can specify a custom metric for the connection by typing a value in this field (the default value is 1). A metric is the cost of using a particular route from one destination to another. Generally this will be the number of hops to the IP destination. Anything on the local subnet is one hop, and every time a router is crossed, this adds 1 to the hop count. The value of this is that it lets Windows 2000 select the route with the lowest metric if there are multiple routes to the same destination.
Advanced DNS Settings  The DNS tab on the Advanced Settings sheet is shown in Figure 6-9.
You can configure the following Advanced settings for DNS:

- **Multiple DNS servers**  If there are multiple DNS servers configured on the network, and TCP/IP doesn’t receive any response from the current DNS server, the next DNS server will be used.

- **Unqualified name resolution**  You can configure TCP/IP to resolve unqualified names by either (1) appending the primary and connection-specific DNS suffixes to the unqualified name for DNS queries, or (2) appending a series of configured DNS suffixes to the unqualified name for DNS queries.

- **Connection-specific DNS suffixes**  Each connection in the Network and Dial-up Connections can be set up to have its own DNS suffix, along with the primary DNS suffix that is configured for the computer on the Network Identification tab in the System applet (in Control Panel).
DNS dynamic update behavior If you have DNS servers that support DNS dynamic update (DDNS), you can enable the DNS dynamic update of the domain name and IP addresses for the computer. Windows 2000 DNS Servers support dynamic update.

Advanced WINS Settings You can also make advanced settings to WINS, using the WINS tab shown in Figure 6-10. Some settings that can be configured include:

- Multiple WINS servers If you have multiple WINS on the network, and TCP/IP fails to receive any response from the current WINS server, the next WINS server in the list will be tried.
- Enabling and disabling the use of the LMHOSTS file You can use this selection to enable or disable the LMHOSTS file. If it is enabled, TCP/IP will use the LMHOSTS file found in the systemroot\System32\Drivers\Etc folder during the process of NetBIOS name resolution. The LMHOSTS file is enabled by default.
- Enabling and disabling the use of NetBIOS over TCP/IP You can enable or disable the use of NetBIOS over TCP/IP here. When it is disabled, NetBIOS programs cannot run over TCP/IP, which means you may not be able to connect to computers that are running downlevel operating systems such as Windows 95 or NT. NetBIOS over TCP/IP should be disabled only if all computers on your network have been upgraded to Windows 2000 and your network is not using NetBIOS-based applications.

Other Advanced Options The last tab in the Advanced TCP/IP Settings sheet is the Options tab, shown in Figure 6-11. The two available options allow you to do the following:

- Enable Internet Protocol security (IPSec) You can provide for secure end-to-end communication of IP-based traffic on a private network or the Internet by enabling IPSec (it is disabled by default). When IPSec is enabled, you can also specify an IPSec security policy.
Enable TCP/IP filtering  This option allows you to enable filtering of TCP/IP packets. If TCP/IP filtering is enabled, you can specify what types of TCP/IP traffic are processed.

Although the help file indicates that the settings apply to all adapters, the filtering is specific for the adapter for which you are adjusting the filters. TCP/IP filtering specifies the types of incoming traffic destined for this adapter and passed up to the TCP/IP protocol for processing.

To configure TCP/IP filtering, select this check box and specify the types of allowed TCP/IP traffic for all adapters on this computer in terms of IP protocols, TCP ports, and UDP ports. The protocols are determined by the protocol number. You can determine protocol numbers by looking at the contents of the RFC 1700.

Configuring TCP/IP filtering can get a little tricky, because when you first enable it, all traffic is filtered out. You add filters for protocols and ports that will want to
allow traffic to pass through. If you run into problems with network communications after enabling the filters, you should first disable TCP/IP filtering and see if that fixes the problem. If so, there is a protocol or port that you must add in order to resume normal network functioning.

**TCP/IP Best Practices**

Microsoft recommends the following best practices when setting up a TCP/IP-based Windows 2000 network:

- If your local network will be connected to the Internet, either obtain registered public IP addresses for all computers that will access the Internet and use an IP router to send traffic to the public network, or establish the
Internet connection using one computer, install *Network Address Translation* (NAT) on that computer, assign private (nonregistered) IP addresses to the other computers on the internal network, and let NAT translate the private addresses to the public address to provide access to the Internet for all the computers on the LAN.

- If you assign private addresses, use the address ranges in each class that are designated as reserved for that purpose by IANA.

**Troubleshooting TCP/IP**

TCP/IP is a powerful, flexible, and reliable protocol suite. Generally it works well, but because it is so complex, there are many ways in which it can be misconfigured or, for other reasons, fail to provide network communications.

Troubleshooting TCP/IP is thus a complicated topic, and an entire book could be written about it (and in fact, several have been). This chapter can only touch on the basics of TCP/IP troubleshooting. For more detailed information, see our book *Troubleshooting Windows 2000 TCP/IP* by Debra Littlejohn Shinder and Thomas W. Shinder, published by Syngress Media.

Windows 2000 TCP/IP comes with several useful utilities that can provide you with valuable troubleshooting information and help you to diagnose the problem when your TCP/IP network experiences connectivity problems. These include the command-line utilities IPCONFIG, PING/PATHPING, TRACERT, and NETSTAT/NBTSTAT, as well as Windows 2000’s protocol analysis tool, Network Monitor.

**Using TCP/IP Utilities**

The command-line utilities included with Windows 2000 are useful in identifying and resolving TCP/IP problems.

**IPCONFIG**

This utility displays current TCP/IP configuration values, and can be used to manually release and renew (with the /release and /renew switches, respectively) a TCP/IP configuration lease assigned by a DHCP server. It can also be used to reset DNS name registrations with the /registerdns switch. Typing **IPCONFIG** provides basic configuration information: IP address, subnet mask, and default gateway. Typing **IPCONFIG /ALL** provides more detailed information, as shown in Figure 6-12.
As you can see, the `/all` switch with the `ipconfig` command provides more detailed information, including the host name, DNS suffix, node type being used, whether IP routing and WINS proxy are enabled, as well as the MAC (physical) address of the network card and the addresses of assigned DNS and WINS servers.

**PING and PATHPING** The PING command is used to verify whether TCP/IP is configured correctly and to test connectivity to other host systems. It is often used as the first step in diagnosing network problems. Microsoft recommends that PING be used, in the following order, to isolate a connectivity problem on the TCP/IP network:

1. First, ping the loopback address (127.0.0.1) to verify that TCP/IP is installed and configured correctly on the local computer.
2. Next, try pinging the IP address of the local computer itself, to verify that it was added to the network correctly.
3. Then ping the IP address of the default gateway (sometimes called the “near side of the router”) to verify that the default gateway is functioning and that you can communicate with a local host on the local network.
4. Finally, ping the IP address of a remote host (called the “far side of the router”) to verify that packets are being forwarded and you can communicate through the router.
If the TCP/IP connection is working, PING will return a response as shown in Figure 6-13.

PATHPING was not available in Windows NT, but is provided in Windows 2000 to allow you to trace the route a packet takes to a destination and display information on packet losses for each router in the path as well as the links between routers. You can also use PATHPING to troubleshoot Quality of Service (QoS) connectivity. PATHPING combines features of both the PING and the TRACERT commands, and includes additional information that neither of those tools provides.

TRACERT  TRACER traces the network route taken by an IP datagram to its destination. It does this by sending Internet Control Message Protocol (ICMP) echo packets with varying Time-To-Live (TTL) values to the destination address. Each router along the path must decrease the TTL on a packet by at least 1 before forwarding it, so the TTL is basically a hop count. When the TTL on a packet reaches 0, the router is supposed to send back an ICMP Time Exceeded message to the source system. TRACERT determines the route by sending the first echo packet with a TTL of 1 and then incrementing the TTL by 1 on each additional transmission until a response is received or the maximum TTL count occurs. Examining the ICMP Time Exceeded messages sent back by intermediate routers allows the route to be determined.

The results of a TRACERT command are shown in Figure 6-14.
You can ping or trace by either the IP address or by the fully qualified domain name. If you are able to get a response using the IP address but not by using the host name, you should suspect a problem with the name resolution server or the computer’s DNS configuration.

**NETSTAT and NBTSTAT**

NETSTAT displays protocol statistics and information on current TCP/IP connections. There are several options available for the NETSTAT command, listed as follows.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-a</td>
<td>Causes all connections and listening ports to be displayed (server connections are not normally shown).</td>
</tr>
<tr>
<td>-e</td>
<td>To display Ethernet statistics.</td>
</tr>
<tr>
<td>-n</td>
<td>Used to display addresses and port numbers in numerical form instead of by name.</td>
</tr>
<tr>
<td>-s</td>
<td>To display statistics on a per-protocol basis (type <code>netstat -p &lt;protocol type&gt;</code>).</td>
</tr>
<tr>
<td>-r</td>
<td>To display the routing table.</td>
</tr>
</tbody>
</table>
NBTSTAT is used to check the state of current NetBIOS over TCP/IP connections (also called NetBT connections). It can also update the NetBIOS Remote Name Cache and determine the registered names and scope IDs.

NBTSTAT can be used to troubleshoot NetBIOS name resolution problems and, like NETSTAT, includes a number of options:

- **-n** Used to display names that were registered locally by programs and services
- **-c** Used to display the NetBIOS Remote Name Cache (a mapping of names to addresses for other computers)
- **-R** Used to purge the name cache and reload it (from the LMHOSTS file)
- **-RR** Used to release the NetBIOS names registered with the WINS server, then renew their registrations with the server
- **-a** Used in conjunction with a computer name (in the format `nbtstat -a <computername>`) to return the NetBIOS name table for the computer named, as well as the MAC address of the network adapter
- **-s** Used to list current NetBIOS sessions and the status of each, with statistical information as shown in Figure 6-15

**NETDIAG** The Resource Kit for Windows 2000 Professional includes the NETDIAG utility. This is a command-line diagnostic tool that helps isolate networking and connectivity problems. It does this by performing a series of tests designed to determine the state of the network client software, and ascertain whether it is functional. This tool does not require that parameters or switches be specified,
which means support personnel and network administrators can focus on analyzing
the output, rather than training users on how to use the tool.

The resource kits for both Windows 2000 Professional and Server contain
many useful tools for isolating, diagnosing, and sometimes even fixing
problems. It is an excellent idea to familiarize yourself with the tools included
in the resource kits, as well as those that ship with the product. Third parties
also provide a variety of troubleshooting utilities for Windows 2000.

Managing and Monitoring Network Traffic

Because TCP/IP is such a widely used set of protocols, there are numerous hardware
and software tools available for monitoring and managing traffic on a TCP/IP
network. You can practice preventative maintenance, establish baselines, or
troubleshoot existing problems. One of the most popular types of tools is called a
protocol analyzer. This is usually a software program that allows you to copy the
individual packets and analyze their structure. The process of copying frames for
review and analysis is called capturing.

Windows 2000 Server includes a “lite” version of the protocol monitor that
comes with Systems Management Server (SMS), Network Monitor. This tool can
be used to capture and display the frames or packets that a Windows 2000 Server
receives from a LAN. You can use Network Monitor to detect and troubleshoot
many networking problems.

Installing the Windows 2000 Network Monitor

The Network Monitor tool can be installed on Windows 2000 Server from the
Add/Remove Programs applet in Control Panel. Open the applet and click on
Add/Remove Windows Components, select Management and Monitoring Tools,
and then click Details. Select the Network Monitor check box. You may be
prompted to insert your Windows 2000 installation CD.

After installation is complete, the Network Monitor will appear in the
Administrative Tools submenu of the Programs menu.

Using the Windows 2000 Network Monitor

The version of Network Monitor included in Windows 2000 has some limitations.
It cannot run in what is referred to as promiscuous mode. This is a state in which the
network card can listen to all the traffic on the network, not just that which is sent
from or received by the computer running the Network Monitor software. The
more sophisticated version of Network Monitor that comes with SMS is capable of
promiscuous mode. Even so, there is a great deal of information that can be collected with Windows 2000’s Network Monitor utility. Figure 6-16 shows the Network Monitor interface, with information from a capture.

As you can see, there is a great deal of statistical information available, including such useful values as the percentage of network utilization during the capture, the number of frames and bytes transferred, as well as how many frames were dropped.

After you have captured the data, you can view it by selecting Display Captured Data from the Capture menu. You will then see detailed information for each frame (packet) as shown in Figure 6-17.

**Network Monitor Best Practices**

Microsoft recommends that you run Network Monitor at off-peak (low-usage) times or only for short periods of time, in order to decrease the detrimental impact on system performance that can be caused by Network Monitor.

You can filter the data captured, as well as the data displayed. A capture filter works somewhat like a database query; you can use it to specify the types of network information you want to monitor. For example, you can capture packets based on the protocol or based on the addresses of two computers whose interactions you wish to monitor. When a capture filter is applied, all packets are examined and compared to the filter’s parameters; those that do not fulfill the filter requirements...
are dropped. This can be a processor-intensive activity during periods of moderate or high network utilization when the network card is placed in promiscuous mode.

Display filters work on data that has already been captured. They do not affect the contents of the Network Monitor capture buffer. You can use a display filter to determine which frames you want to display. The frames can be filtered by source or destination address, protocols used to send it, or the properties and values it contains.

You can also use capture triggers to specify a set of conditions that will cause (“trigger”) an event to occur in a Network Monitor capture filter. Triggers allow Network Monitor to respond to events on your network. For example, you can set triggers so that if Network Monitor detects a particular set of circumstances on the network, it will start an executable file.

CERTIFICATION OBJECTIVE 6.04

Installing and Configuring NWLink

Despite Microsoft’s emphasis on TCP/IP for Windows 2000 networks, there are situations in which other LAN protocols are required or desirable. For instance, connectivity to NetWare servers (prior to version 5) requires that the connecting
computer(s) be running an IPX/SPX protocol stack. NWLink is Microsoft’s implementation of IPX/SPX (the Internetwork Packet Exchange/Sequenced Packet Exchange protocols). You can install NWLink on Windows 2000 machines to allow them to access NetWare servers.

Be aware that although NWLink and IPX/SPX are often thought of in Microsoft networking circles as “the protocols used to connect to NetWare networks,” you can also use these protocols to connect Windows machines to one another. Because NWLink is faster and easier to configure than TCP/IP and routable (unlike NetBEUI), it can be an appropriate LAN protocol choice for a small or medium network that is not connected to the Internet, even if there are no NetWare servers anywhere in sight.

Installing the NWLink Protocol

NWLink is installed in the same way as TCP/IP, through the Network and Dial-up Connections properties box. Note that you must be logged on with administrative privileges to install NWLink or other network protocols. When you install the NWLink protocols, they will be installed on all your network connections. However, you can change this. After the installation is completed, if you do not want to use NWLink on a particular connection, you can right-click that connection, click Properties and uncheck the NWLink IPX/SPX/NetBIOS Compatible Transport Protocol check box on either the General or Networking tab.

Configuring NWLink

There is less configuration information required to set up NWLink than is necessary for TCP/IP, but there are still a few pieces of information you will need to have on hand. The configuration box asks you for a network number; if you are uncertain of this number, the default value of 00000000 will usually work.

Another setting that can be configured is the frame type. You can use automatic frame type detection, however. In most cases, this is preferred.

You can use the IPXROUTE CONFIG command to get information about the internal network number being used by the routers on the network, as well as the frame type being used by the servers.
Installing and Configuring NWLink in Windows 2000

To install NWLink on your Windows 2000 computer, follow these steps:

1. Select Start \ Settings \ Network and Dial-up Connections.
2. Right-click on the icon for the local area connection, and click Properties.
3. On the General tab, click Install.
4. In the Select Network Component Type dialog box, click Protocol, and then click Add.
5. In the Select Network Protocol dialog box, click NWLink IPX/SPX/NetBIOS Compatible Transport Protocol, and then click OK.

To configure NWLink, follow these steps:

1. In the Local Area Connection properties box, select NWLink IPX/SPX NetBIOS Compatible Transport Protocol, and click Properties. You will see the General Properties sheet displayed, as shown in the following illustration.
In order for Windows 2000 clients to access a NetWare server directly, you will also need to install Client Services for NetWare (CSNW) on the client computer(s). The users must also have valid user accounts on the NetWare server to log on.

Windows 2000 machines that are clients to a Windows 2000 Server can also access the NetWare server through the Windows 2000 Server, without having extra client software installed, if you install Gateway Services for NetWare (GSNW) on the Windows 2000 Server and configure the NetWare server to allow access through a special gateway account. To accomplish this, you must create a group on the NetWare server called NTGATEWAY and create a NetWare user account in that group. On the Windows 2000 Server, you must then enable the gateway service and enter the name of the gateway group and the password for the gateway account. All Windows 2000 clients can now connect to the NetWare server through this same account. Note, however, that all will be using the same user account for access, and thus will all have the same permissions. If you wish for a user to have different permissions, that user must have a valid individual user account on the NetWare server.
server through which he or she can access the server’s resources and client software installed on the client machine.

**Protocol Binding Order**

Network protocols are bound to network services installed on a particular client or server. Multiple protocols can be bound to multiple network services. For example, you can bind NetBEUI, TCP/IP, and NWLink to the Microsoft network client.

When a client attempts to establish a session with a server, it will request to use the protocol on the top of its binding order for the particular service. If the service is the Microsoft Network Client, then the protocol binding order for that list is processed. Just as was the case with Windows NT 4.0, the client determines the preferred protocol.

For example, the Microsoft network client service on the client has NetBEUI and TCP/IP installed, with NetBEUI on the top of the binding order. The destination server also has NetBEUI and TCP/IP installed, but has TCP/IP on top of the binding order. Since the caller determines the binding order, they will negotiate using TCP/IP first, because that is on top of the client’s binding order.

To change the binding order you must open the Network and Dial-up Connections folder first. After opening the folder, click on the Advanced menu and click Advanced Settings. On the Adapters and Bindings tab, you will be able to use the UP and DOWN arrows to change the binding order.

**CERTIFICATION SUMMARY**

This chapter has covered a wide scope of topics pertaining to the networking protocols used for communication by the Windows 2000 operating systems, particularly the members of the TCP/IP protocol suite. In fact, we started the chapter with an introduction to the Windows 2000 implementation of TCP/IP and a brief history of the protocols, discussed the advantages and disadvantages of TCP/IP and where it fits into the popular networking models, the Open Systems Interconnection (OSI) model and the Department of Defense (DoD) model.

TCP/IP is the protocol stack of the global Internet, so we discussed the growth of the Internet and how TCP/IP played a part in that growth. We talked about IP
addressing, including subnetting and supernetting, and we walked through an exercise on how to install and configure the TCP/IP protocols in Windows 2000.

Then we briefly discussed troubleshooting TCP/IP connectivity problems, and some of the tools and utilities included in Windows 2000 or available elsewhere that make troubleshooting easier. We examined the Network Monitor protocol analysis tool that Windows 2000 includes, and learned how to install it and how to use it to capture packets on the network.

Finally, we turned our attention to another popular protocol stack, IPX/SPX, which is used in many NetWare networks and is implemented in Windows 2000 as NWLink. We completed an exercise in installing and configuring NWLink, and we discussed the services—Client Services for NetWare and Gateway Services for NetWare—that are used, along with the NWLink protocols, to allow Windows 2000 machines to connect to and access the resources of NetWare servers.
TWO-MINUTE DRILL

Windows 2000 TCP/IP

- TCP/IP is the default network transport protocol in Windows 2000, and is required for connectivity to the Internet.
- The TCP/IP protocol suite is relatively slow and difficult to configure, but has the advantages over NWLink and NetBEUI of a flexible, scalable addressing scheme, compatibility with many platforms and operating systems, and routability.
- The TCP/IP suite consists of many different protocols that operate at different layers of the networking model, including IP, ICMP and IGMP, ARP/RARP, TCP, UDP, FTP, Telnet, SMTP, SNMP, and others.

IP Addressing

- IP addresses are 32-bit binary numbers (often expressed as their dotted decimal equivalent), in which one portion represents the network ID and the other portion represents the host ID.
- The subnet mask is a 32-bit binary number used to indicate which bits in the IP address represent the network and which represent the host.
- IP networks were traditionally divided into three classes: Class A for large networks, Class B for medium-sized networks, and Class C for small networks.
- Classless InterDomain Routing (CIDR) is a newer method of allocating addresses to networks that does not use the class system, but instead appends a “slash x” number to the end of the IP address that indicates how many bits are used for the network ID.
- Subnetting is a method of “stealing” bits from the designated host ID and adding them to the network ID in order to create more networks, but with fewer hosts each.
- Supernetworking is a method of combining two or more smaller networks into one larger network, by “stealing” bits from the network ID portion of the IP address and adding them to the host portion, to create a network that allows for more hosts.
Installing, Configuring, and Monitoring TCP/IP

Utilities included with TCP/IP that are useful for troubleshooting include IPCONFIG, PING and PATHPING, TRACERT, NETSTAT and NBTSTAT, and a resource kit utility called NETDIAG.

Network Monitor is a software protocol analyzer included with Windows 2000 Server that allows you to capture and analyze packets sent to or from the Windows 2000 machine on which the software is installed.

Windows 2000 supports assigning multiple IP addresses, multiple gateways, and setting the interface metric under Advanced TCP/IP settings.

Disabling of NetBIOS over TCP/IP (NetBT) is done via the Advanced WINS configuration options, and should only be done if your network does not have any non-Windows 2000 machines and does not use NetBIOS applications.

TCP/IP filtering allows you to specify what types of TCP/IP traffic are processed.

Installing and Configuring NWLink

NWLink is Microsoft’s implementation of the IPX/SPX protocol stack used by Novell for its NetWare networks. It can be installed, along with NetWare client or gateway software, to provide access by Windows 2000 machines to a NetWare server, or it can be used as the LAN protocol for Microsoft networks that don’t need to connect to the Internet.

Connectivity to NetWare servers (prior to version 5) requires that the connecting computer(s) be running an IPX/SPX protocol stack.

Configuration settings for NWLink include the internal and external network numbers and the frame type.

The protocol binding order on the client machine determines which LAN protocol will be used for the connection if multiple protocols are installed and configured.
SELF TEST

The following questions will help you measure your understanding of the material presented in this chapter. Read all of the choices carefully, as there may be more than one correct answer. Choose all correct answers for each question.

Windows 2000 TCP/IP

1. Which of the following is an advantage of the TCP/IP protocol stack over other common LAN protocols such as NetBEUI and IPX/SPX (NWLink)?
   A. TCP/IP is faster than other LAN protocols.
   B. TCP/IP is more scalable than other LAN protocols.
   C. TCP/IP is easier to configure than other LAN protocols.
   D. TCP/IP has lower overhead than other LAN protocols.

2. Your network is experiencing problems that seem to be related to IP addressing and routing. At which layer of the OSI networking model should you start the diagnostic procedure?
   A. The Physical Layer
   B. The Transport Layer
   C. The Application Layer
   D. The Network Layer

3. You have an application that needs to send messages quickly, in real time. Speed of transfer is more important in this case than reliability. Which Transport layer protocol will this application most likely use to communicate?
   A. UDP
   B. TCP
   C. IP
   D. SNMP

4. You do not know the IP address of your computer. You look at the TCP/IP properties settings for your local area connection, but discover that your computer is obtaining an IP address from a DHCP server. How can you find out what IP address has been assigned by the DHCP server?
   A. Click Advanced on the TCP/IP properties sheet.
   B. Use the IPCONFIG command-line utility.
   C. Right-click on Network Neighborhood and select “IP address” from the right context menu.
   D. Reboot the computer and enter Setup to locate your IP address in the CMOS settings.
Chapter 6: Installing, Configuring, Managing, Monitoring, and Troubleshooting Network Protocols

IP Addressing

5. You have a Class C network, which you wish to divide into six usable subnets with an equal number of hosts on each. Which of the following default subnet masks would you use to identify the network and host portions of the IP address?
   A. 255.255.255.252
   B. 255.255.255.248
   C. 255.255.255.224
   D. 255.255.255.128

6. Which of the following “slash x” designations would represent a network with a subnet mask of 255.255.252.0 in CIDR terminology?
   A. /12
   B. /16
   C. /18
   D. /22

7. Using traditional classful addressing, which of the following would be an IP address of a computer on a Class B network?
   A. 192.168.2.34
   B. 14.244.22.9
   C. 185.21.1.1
   D. 224.32.128.4

8. Which of the following defines the cost of using a particular route from a source to a destination?
   A. The default route
   B. The metric
   C. The address class
   D. The TTL

9. You wish to configure your TCP/IP computer so there will be fault tolerance if the assigned default gateway (router) should fail. How can you accomplish this?
   A. Configure a secondary gateway in the TCP/IP properties sheet.
   B. Configure a static route in the routing table.
   C. Configure multiple default gateways in the Advanced TCP/IP properties.
   D. There is no way to provide fault tolerance in case of failure of a default gateway.
Installing, Configuring, and Monitoring TCP/IP

10. You need to provide end-to-end security for your TCP/IP communications. How can you accomplish this in Windows 2000, using no third-party utilities?
   
   A. Enable IP forwarding.
   B. Enable IP filtering.
   C. Enable RIP.
   D. Enable IPsec.

11. You have a small private unsubnetted, unrouted Class C network. One of your users is unable to connect to the other computers on the network. You run IPCONFIG and it shows the following: IP address 192.168.1.2, subnet mask 255.0.0.0, default gateway (none). Which, if any, of these settings is your first suspect as to the source of the trouble?
   
   A. Invalid IP address.
   B. Incorrect subnet mask.
   C. Missing default gateway.
   D. None of these is the source of the problem.

12. You have a medium size Class B network that is connected to the Internet via a routed connection. One of the computers on the network is unable to communicate with computers on its subnet or with the Internet. You run IPCONFIG and it displays the following: IP address – 192.168.255.255, subnet mask 255.255.0.0, default gateway – 161.23.2.1. Which, if any, of these settings do you first suspect as the source of the problem?
   
   A. Invalid IP address.
   B. Incorrect subnet mask.
   C. Incorrect default gateway.
   D. None of these is the source of the problem.

13. You need to know what node type is being used by your computer. Which of the following commands could you use to display that information?
   
   A. IPCONFIG
   B. NETSTAT -R
   C. NBTSTAT
   D. IPCONFIG /ALL
Chapter 6: Installing, Configuring, Managing, Monitoring, and Troubleshooting Network Protocols

Installing and Configuring NWLink

14. You have a small private network running Windows 2000 and Windows 9x computers. You wish to have routing capability, but you do not need to connect to the Internet. You need a protocol stack that is fast and easy to configure. Which of the following would best fit that requirement?
   A. NWLink
   B. NetBEUI
   C. TCP/IP
   D. DLC

15. At which layer of the OSI networking model does IPX operate?
   A. Physical
   B. Data Link
   C. Network
   D. Presentation

16. Which of the following information do you supply when configuring the NWLink protocol? (Select all that apply.)
   A. DNS server
   B. External network number
   C. Internal network number
   D. Frame type

17. You have a Windows 2000 client computer that has NWLink, TCP/IP, and NetBEUI installed, with the binding order set in that order. This machine is connecting to a Windows 2000 Server that has TCP/IP and NWLink installed, bound in that order. Which protocol will be used to communicate between the machines?
   A. TCP/IP
   B. NWLink
   C. NetBEUI
   D. None of the above
LAB QUESTION

You have two laptop computers belonging to two employees in the Accounting department, Fred and Joe. Both are configured with both TCP/IP and NWLink. Fred complains that when he connects to the network, he is able to communicate with Joe using NWLink, but not TCP/IP. He is also unable to communicate with other computers using TCP/IP on the network. Joe tells you the same thing happened to him earlier in the week, although today he is able to communicate with other computers on the network using TCP/IP. Which of the following do you first suspect as the source of the problem?

A. Fred’s and Joe’s computers have the same NetBIOS name.
B. Fred’s and Joe’s computers have the same subnet mask.
C. Fred’s and Joe’s computers have the same default gateway.
D. Fred’s and Joe’s computers have the same IP address.
450 Chapter 6: Installing, Configuring, Managing, Monitoring, and Troubleshooting Network Protocols

SELF TEST ANSWERS

Windows 2000 TCP/IP

1. ✗ B. TCP/IP is more scalable than other LAN protocols.
   ✓ A, C, and D are incorrect, because although TCP/IP is more flexible, routable, and scalable and thus appropriate for large networks, and although it is the protocol required to connect to the global Internet, it is slower than the other two common LAN protocols, is more difficult to configure, and has higher overhead.

2. ✓ D. The Network Layer of the OSI model (which corresponds to the Internetwork layer of the DoD model) is responsible for logical addressing and routing tasks.
   ✗ A, B, and C are incorrect because the Physical layer handles hardware, media and signaling issues; the Transport layer is responsible for reliability and acknowledgments; and the Application layer interfaces between the networking components and the user applications.

3. ✓ A. The User Datagram Protocol, UDP, is the Transport layer protocol used for sending data when speed is more important than reliability.
   ✗ B is incorrect because TCP is a slower, but more reliable Transport layer protocol. C is incorrect because IP is a Network layer protocol, responsible for routing and logical addressing. D is incorrect because SNMP is an Application layer protocol, used for monitoring and managing TCP/IP networks.

4. ✓ B. Type IPCONFIG at the command line and your IP address, subnet mask and default gateway will be displayed.
   ✗ A is incorrect because the IP address will not be indicated in the Advanced properties. C is incorrect because there is no Network Neighborhood in Windows 2000, and if you right-click on its replacement (My Network Places), there is no “IP address” selection in the right context menu. D is incorrect because the IP address is assigned in the operating system software and is not displayed in the CMOS settings for the computer.

IP Addressing

5. ✓ C. In order to create six subnets, you will need to “borrow” three bits from the host ID to create the subnet ID. This makes your last octet 11100000. If you convert the binary number to decimal by adding the values of the bits that are “on,” 128+64+32, you end up with 224.
   ✗ A, B, and D are all incorrect.
6. **D.** The subnet mask 255.255.252.0 signifies that 22 bits are used for the network ID, and the remaining 10 are used for the host ID. In CIDR terms, this is indicated as a “slash 22” network, and would be expressed with “/22” following the IP address.

   ✗ A, B, and C are incorrect.

7. **C.** Network class is indicated by the high order, or leftmost bits. Class A networks always start with 0 as the high order bit, Class B with 10, and Class C with 110. This means when we convert to binary, Class A addresses have 1–126 as their first octet, Class B addresses have 128–191 as the first octet, and Class C addresses have 192–223 as the first octet. Only C, which begins with 185, falls into the Class B range.

   ✗ A, B, and D are incorrect. A would be a member of a Class C network, B would be a member of a Class A network, and D would be a multicast address.

8. **B.** The metric is the term used to indicate the cost of a route.

   ✗ A is incorrect; the default route is the route to be used when there is no route to a particular network specified in the routing table. C is incorrect because the address class indicates the size of the network, not the cost of a route. D is incorrect because the TTL is the Time to Live, used to indicate how long a receiving computer or router should hold or use the packet before it expires.

9. **C.** If you configure multiple default gateways, the first will always be used unless it fails, in which case the next on the list will be tried.

   ✗ A is incorrect because there is no option to configure a “secondary” gateway. B is incorrect because configuring a static route will not provide fault tolerance for a gateway failure. D is incorrect because there is a way to provide fault tolerance in case of a gateway failure.

**Installing, Configuring, and Monitoring TCP/IP**

10. **D.** IP Security (IPSec) can be used to provide for secure end-to-end communications.

   ✗ A is incorrect because IP forwarding is used to route messages from one subnet to another but does not provide security. B is incorrect because IP filtering will allow you to block certain messages but does not provide end-to-end security for your communications. C is incorrect because RIP (Routing Information Protocols) allows for dynamic routing, but does not address end-to-end security.
Chapter 6: Installing, Configuring, Managing, Monitoring, and Troubleshooting Network Protocols

11. B. 255.0.0.0 is the default subnet mask for a Class A network. Because the network is not subnetted, the default Class C mask 255.255.255.0 should be used.
    A is incorrect because this IP address is a valid private class C address. C is incorrect because no default gateway is needed for an unsubnetted, unrouted network. D is incorrect because B is most likely the problem. You should check the other computers on the network to verify that they are using the standard Class C default mask.

12. A. The IP address has several problems: First, the 192.168.x.x address range is reserved for private addresses and cannot be used on the Internet. Second, the 255 host address is reserved for broadcast messages.
    B is incorrect because 255.255.0.0 is the correct default subnet mask for a Class B network. C is incorrect because this is a valid default gateway address for a Class B network (although if the IP address were valid, you would want to check the router and ensure this is the correct gateway address). D is incorrect because A is a likely cause of the problem.

13. D. IPCONFIG with the /all switch will display information that includes host name, DNS suffix, node type, MAC address, and DNS and WINS server addresses.
    A is incorrect because IPCONFIG alone only displays IP address, subnet mask, and default gateway information. B and C are incorrect because the NETSTAT and NBTSTAT commands do not display node type information.

Installing and Configuring NWLink

14. A. Although usually used for connecting to NetWare networks, NWLink can be used as the LAN protocol for a Windows 2000 network if Internet connection is not required (you must have TCP/IP for Internet connection).
    B, C, and D are incorrect. NWLink is faster and easier to configure than TCP/IP, and unlike NetBEUI, can be routed. DLC is used to communicate with mainframes and some network printers and is not routable.

15. C. The IPX protocol, like IP, operates at the Network layer to provide routing and addressing.
    A, B, and D are incorrect because IPX does not operate at these layers.
16.  ✓  B, C, D. Configuration for the NWLink protocol requires an internal and external network number, and a frame type designation. However, in most cases the defaults will work. Frame type must be configured if there is more than one frame type being used on the network.

   ✘  A is incorrect because the NWLink protocol does not include configuration of a DNS server.

17.  ✓  B. Because NWLink is at the top of the binding order on the client machine, it will first try to connect using NWLink if that protocol is installed on the server. Because NWLink is installed, the connection will be made with that protocol stack. To ensure that the two connect using TCP/IP, you should move TCP/IP to the top of the binding order on the client machine.

   ✘  A is incorrect because although TCP/IP is installed on both machines, it is the binding order on the client machine that determines the connection. C is incorrect because NetBEUI is not installed on the server, so the two machines cannot connect using that protocol. D is incorrect because B is correct.

LAB ANSWER

✓  D. The most likely cause of the problem is an IP address conflict. When Joe connects to the network first, he is able to use the IP address for TCP/IP communications, but when Fred then attempts to connect, he is unable to use TCP/IP because the IP address configured in his TCP/IP properties is still in use. If both had the same NetBIOS name, the second user to attempt to connect would receive an error message that the name already exists. Fred’s and Joe’s computers should have the same subnet mask and default gateway if they are on the same subnet; this would not hinder communications. To solve the problem, you should change the IP address on one of the machines to one not in use. A better solution might be to configure all the machines to use DHCP if there is a DHCP server on the network. This reduces the likelihood of duplicate addresses.