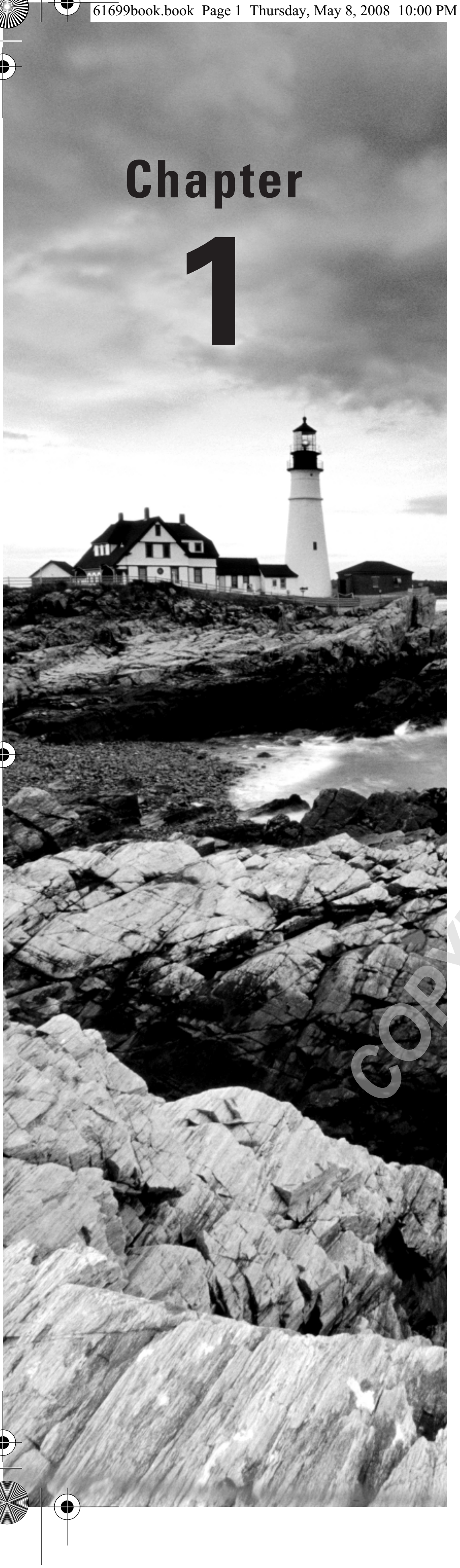


Chapter 1

Understanding Windows Server 2008 Networking

Microsoft has put an immense amount of time and effort into building Windows Server 2008. Much is new in the operating system, but it still retains a great deal of core code from Windows 2000, Windows 2003, and even Windows NT, Internet Information Services, and Exchange Server. Windows Server 2008 is a large, complicated, and very powerful operating system. To use it effectively, you have to understand how it works and how to make it do what you want it to do.

This book is a study guide for the Microsoft Windows Server 2008 exam 70-642. So, it only makes sense to start with a discussion of network protocols and which protocols have been included and removed in Windows Server 2008.

Having a good frame of reference helps when discussing network protocols. To establish such a framework, this chapter will start with the OSI network model, a sort of idealized way to stack various protocols together. Then the chapter will cover the different Microsoft Windows network models and which one would work best for your company.

While discussing these topics, we will cover the necessary background information in order for you to be successful on the 70-642 exam as well as work with Windows Server 2008 on the network.

Understanding the OSI Model

The International Organization for Standardization (ISO) began developing the Open Systems Interconnection (OSI) reference model in 1977. OSI has since become the most widely accepted model for understanding network communication; once you understand how the OSI model works, you can use it to compare network implementations on different systems.

When you want to communicate with another person, you need to have two things in common: a communication language and a communication medium. Computer networks are no different; for communication to take place on a network composed of a variety of different network devices, both the language and the medium must be clearly defined. The OSI model (and networking models developed by other organizations) attempts to define rules that cover both the generalities and the specifics of networks:

- How network devices contact each other and, if they have different languages, how they communicate with each other
- Methods by which a device on a network knows when to transmit data and when not to transmit it

- Methods to ensure that network transmissions are received correctly and by the right recipient
- How the physical transmission media is arranged and connected
- How to ensure that network devices maintain a proper rate of data flow
- How bits are represented on the network media

The OSI model isn't a product. It's just a conceptual framework you can use to better understand the complex interactions taking place among the various devices on a network. It doesn't do anything in the communication process; appropriate software and hardware do the actual work. The OSI model simply defines which tasks need to be done and which protocols will handle those tasks at each of the seven layers of the model. The seven layers are as follows:

- Application (layer 7)
- Presentation (layer 6)
- Session (layer 5)
- Transport (layer 4)
- Network (layer 3)
- Data-Link (layer 2)
- Physical (layer 1)

TIP

You can remember the seven layers from top to bottom using a handy mnemonic, such as "All People Seem To Need Data Processing."

Each of the seven layers has a distinct function, which we'll explore a little later in the chapter.

The True IP Protocol Suite

There is another model to represent these same concepts that is truly what the Internet was built upon. The model is known by a few names, including the TCP/IP model, the IP model, or the DoD model (after its designers, the U.S. Department of Defense). The TCP/IP model, as we'll call it, contains only four layers:

- Application
- Transport (sometimes called Host to Host)
- Internet
- Link (also called Network Access)

This model was the one originally used for the design of the Internet. You won't encounter this model on the 70-642 exam, but as a network administrator, you should know that this model exists when you hear your peers talking about it.

Protocol Stacks

The OSI model splits communication tasks into smaller pieces called *subtasks*. Protocol implementations are computer processes that handle these subtasks. Specific protocols fulfill subtasks at specific layers of the OSI model. When these protocols are grouped together to complete a whole task, the assemblage of code is called a *protocol stack*. The stack is just a group of protocols, arranged in layers, that implements an entire communication process. Each layer of the OSI model has a different protocol associated with it. When more than one protocol is needed to complete a communication process, the protocols are grouped together in a stack. An example of a protocol stack is TCP/IP, which is widely used by Unix and the Internet—the TCP and IP protocols are implemented at different OSI layers.

NOTE

Windows Server 2008 and Windows Vista include the Next Generation TCP/IP protocol stack, which is for both version 4 (IPv4) and version 6 (IPv6) of the Internet Protocol.

Why You Should Care about Protocol Stacks

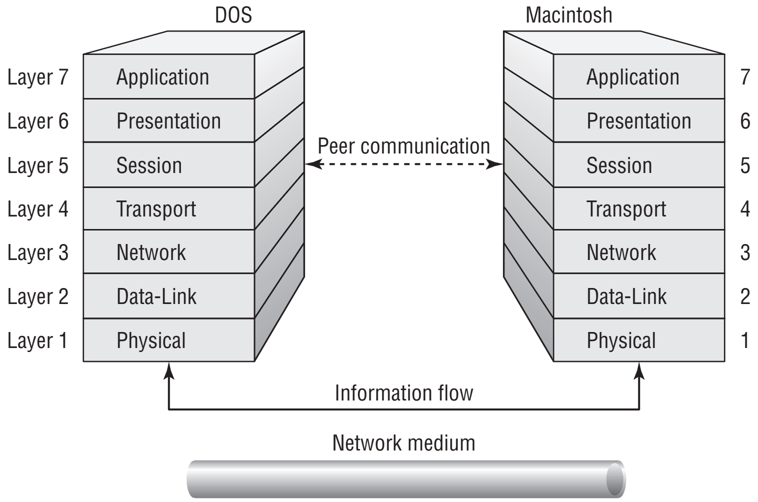
It may not be incredibly clear right now why you should care about protocol stacks and the OSI model. However, the OSI model and the protocol stacks for Internet communication are the basis upon which the 70-642 exam is built. All things about today's modern network stem from either the OSI or IP model.

Knowing the layers of both models is essential to sound troubleshooting, and though it may not be obvious as you trudge through this background material, knowing the OSI model will help when managing a Windows network. You'll see additional examples of how knowing the OSI model helps when troubleshooting throughout this chapter.

Each layer in the protocol stack receives services from the layer below it and provides services to the layer above it. It can be better explained like this: layer N uses the services of the layer below it (layer N – 1) and provides services to the layer above it (layer N + 1).

For two computers to communicate, the same protocol stacks must be running on each computer. Each layer on both computers' stacks must use compatible protocols in order for the machines to communicate with each other. The computers can have different operating systems and still be able to communicate if they are running the same protocol stacks. For example, a DOS machine running IP can communicate with a Macintosh machine running IP (see Figure 1.1).

FIGURE 1.1 Each layer communicates with its counterparts on other network hosts.

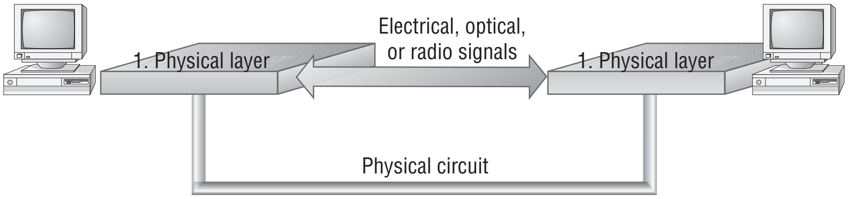


When sending data, each layer in the OSI model places its own information onto the data as it passes down the stack in a process called *encapsulation*. Encapsulation takes place when each layer adds its own header and sometimes trailer information onto the data. When the data is received, it works its way back up the protocol stack, and the corresponding layer of the protocol stack reads this information.

The Physical Layer

The Physical layer is responsible for using electric (or sometimes other types of) signaling to get bits from one computer to another. Physical layer components don't care what the bits *mean*; their job is to get the bits from point A to point B, using whatever kind of optical, electrical, or wireless connection that connects the points. This level defines physical and electrical details, such as what will represent a 1 or a 0, how many pins a network connector will have, and when the network adapter can or cannot transmit the data (see Figure 1.2).

FIGURE 1.2 The Physical layer makes a physical circuit with electrical, optical, or radio signals.



The Physical layer addresses all the small details of the actual physical connection between the computer and the network medium, including the following:

- Network connection types, including multipoint and point-to-point connections.
- Physical topologies, or how the network is physically laid out (for example, bus, star, or ring).
- Which analog and digital signaling methods are used to encode data in the analog and digital signals.
- Bit synchronization, which deals with keeping the sender and receiver in sync as they read and write data.
- Multiplexing, or the process of combining several data channels into one.
- Termination, which prevents signals from reflecting back through the cable and causing signal and packet errors. It also indicates the last node in a network segment.

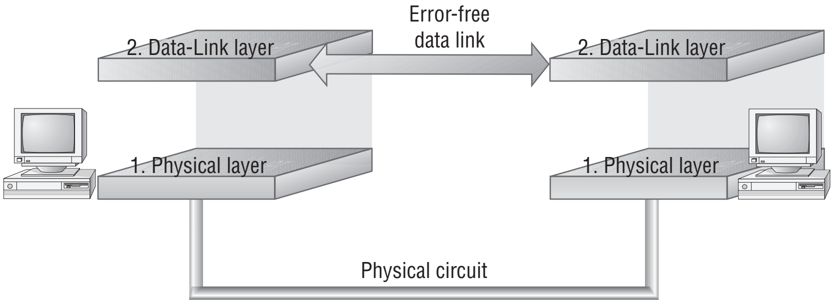
The Data-Link Layer

The Data-Link layer provides for the flow of data over a single physical link from one device to another. It accepts packets from the Network layer and packages the information into data units called *frames*; these frames are presented to the Physical layer for transmission. The Data-Link layer adds control information, such as the frame type, to the data being sent.

This layer also provides for the error-free transfer of frames from one computer to another. A *cyclic redundancy check* (CRC) added to the data frame can detect damaged frames, and the Data-Link layer in the receiving computer can request that the CRC information be present so that it can check incoming frames for errors. The Data-Link layer can also detect when frames are lost and request that those frames be sent again.

In broadcast networks such as Ethernet, all devices on the LAN receive the data that any device transmits. (Whether a network is broadcast or point-to-point is determined by the network protocols used to transmit data over it.) The Data-Link layer on a particular device is responsible for recognizing frames addressed to that device and throwing the rest away, much as you might sort through your daily mail to separate good stuff from junk. Figure 1.3 shows how the Data-Link layer establishes an error-free connection between two devices.

FIGURE 1.3 The Data-Link layer establishes an error-free link between two devices.

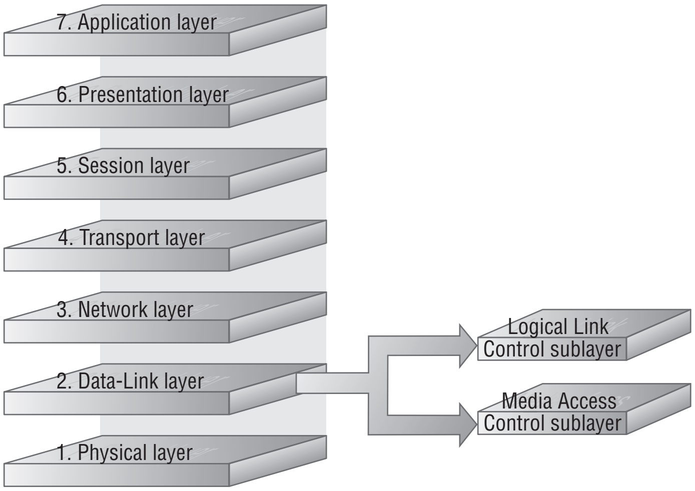


The Institute of Electrical and Electronics Engineers (IEEE) developed a protocol specification known as IEEE 802.X. (802.2 is the standard that divides this layer into two sublayers. The Media Access Control layer, more commonly called the MAC layer, varies depending on the network type and is described further in standards 802.3 through 802.5.) As part of that specification (which today we know as Ethernet), the Data-Link layer is split into two sublayers:

- The Logical Link Control (LLC) layer establishes and maintains the logical communication links between the communicating devices.
- The Media Access Control (MAC) layer acts like an airport control tower—it controls the way multiple devices share the same media channel in the same way that a control tower regulates the flow of air traffic into and out of an airport.

Figure 1.4 illustrates the division of the Data-Link layer into the LLC and MAC layers.

FIGURE 1.4 The IEEE split the ISO Data-Link layer into the LLC sublayer and the MAC sublayer.



The LLC sublayer provides *service access points* (SAPs) that other computers can refer to and use to transfer information from the LLC sublayer to the upper OSI layers. This is defined in the 802.2 standard.

The MAC sublayer, the lower of the two sublayers, provides for shared access to the network adapter and communicates directly with network interface cards. A unique 48-bit address, commonly represented as a 12-digit hexadecimal MAC address (frequently called the *hardware Ethernet address*), is assigned to network interface cards before they leave the factory where they are made. The LLC sublayer uses MAC addresses to establish logical links between devices on the same LAN. Ethernet is an example of a protocol that exists at the Data-Link layer.

MAC Address Conflicts and Limitations

Historically, one of the more difficult problems to diagnose and fix has been a conflict of MAC addresses between two (or more) devices on a network. Since the MAC address is specific to the hardware, it's sometimes referred to as the *burned-in address*. The MAC consists of 48 bits represented as a 12-digit hexadecimal number. Of those 12 digits, 6 are specific to the vendor that produced the card. For example, the first six numbers on a network card produced by Intel are 00AA00, while on Cisco devices they are 00000C.

On many network cards there is no way to change this address, because it is set at the factory (although some cards do enable the administrator to change the MAC address). When two devices with the same MAC address are connected to the same network segment, a conflict will occur that can be quite difficult to diagnose. Many times the normal troubleshooting techniques won't work since things like pings respond normally. Resolving a MAC conflict sometimes comes down to looking at the MAC address for each device currently reporting problems. On Windows this is accomplished with the `ipconfig /all` command, as you'll see in Chapter 2, "TCP/IP."

MAC addresses don't cross network boundaries. Therefore, when troubleshooting a problem related to a MAC address with a tool called a *sniffer* (a packet sniffer is a utility used to extract packets from a network cable so that the packet information may be examined), you might see one MAC address showing up more often than others. Chances are that this is the address of the router or gateway boundary of the network. Since all traffic coming into the network goes through the router, it assigns its own MAC address to all conversations coming into the network. Many an intrusion analyst or administrator on a network has been confused when seeing large amounts of traffic apparently coming from a single MAC source.

The Network Layer

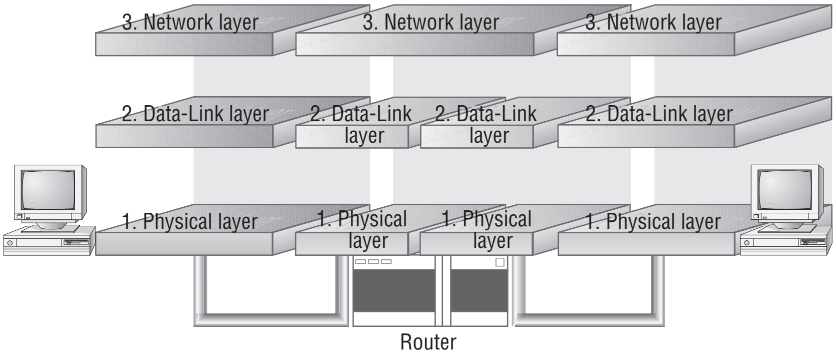
The Network layer handles moving packets between devices. It makes routing decisions and forwards packets as necessary to help them travel to their intended destination. In larger networks, there may be intermediate devices and subnetworks between any two end systems. The network layer makes it possible for the Transport layer (and layers above it) to send packets without being concerned with whether the end system is on the same piece of network cable or on the other end of a large wide area network.

To do its job, the Network layer translates logical network addresses into physical machine addresses (MAC addresses, which operate at the Data-Link layer). The Network layer also determines the quality of service (such as the priority of the message) and the route a message will take if there are several ways a message can get to its destination.

The Network layer also may split large packets into smaller chunks if the packet is larger than the largest data frame the Data-Link layer will accept. The Network layer reassembles the chunks into packets at the receiving end.

Intermediate systems that perform only routing and relaying functions and do not provide an environment for executing user programs can implement just the first three OSI network layers. Figure 1.5 shows how the Network layer moves packets across multiple links in a network.

FIGURE 1.5 The Network layer moves packets across links to their destination.



The Network layer performs several important functions that enable data to arrive at its destination. The protocols at this layer may choose a specific route through an internetwork to avoid the excess traffic caused by sending data over networks and segments that don't need access to it. The Network layer serves to support communications between logically separate networks. This layer is concerned with the following:

- Addressing, including logical network addresses and service addresses
- Circuit, message, and packet switching
- Route discovery and route selection
- Connection services, including Network layer flow control, Network layer error control, and packet sequence control
- Gateway services

The Internet Protocol (IP) resides on the Network layer.

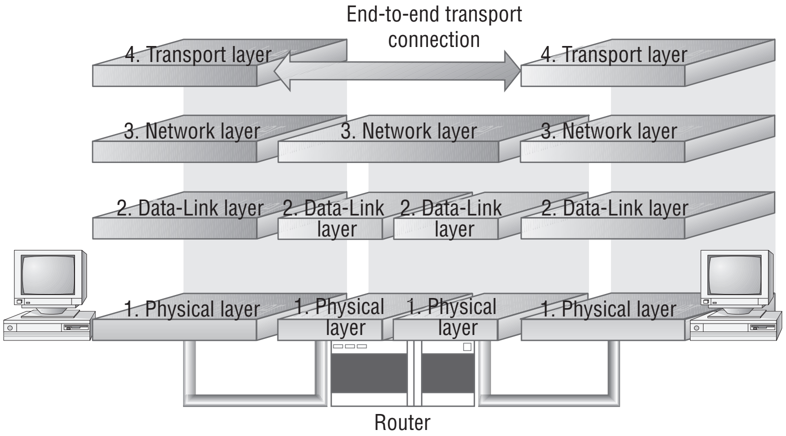
The Transport Layer

The Transport layer ensures that data is delivered error-free, in sequence, and with no losses or duplications. This layer also can break large messages from the Session layer into smaller segments to be handed down to the Network layer and sent to the destination computer; it then reassembles segments into messages to be presented to the Session layer. The Transport layer can send an acknowledgment to the originator for messages received (as in Figure 1.6). Most of these services are optional and are not required in the implementation of all Transport layer protocols. The one feature common to all protocols at the Transport layer is upper-layer protocol multiplexing, allowing multiple higher-layer protocol flows to operate simultaneously.

In terms of TCP/IP, this means you could, for example, navigate to a website and download a file at the same time.

Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) are examples of protocols that exist at the Transport layer.

FIGURE 1.6 The Transport layer provides end-to-end communication with integrity and performance guarantees.



The Session Layer

The Session layer allows applications on separate computers to share a connection called a *session*. This layer provides services that allow two programs to find each other and establish the communication link, such as name lookup and security. The Session layer also provides for data synchronization and check pointing so that in the event of a network failure, only the data sent after the point of failure would need to be re-sent. This layer also controls the dialogue between two processes and determines who can transmit and who can receive at what point during the communication (see Figure 1.7).

NetBIOS, RPC, Named Pipes, PPTP, and SQL are examples of protocols on the Session layer.

The Presentation Layer

The Presentation layer translates data between the formats the network requires and the formats the computer expects. The Presentation layer performs protocol conversion; data translation, compression, and encryption; character set conversion; and the interpretation of graphics commands.

The network redirector, long a part of Windows networking, operates at this level. The redirector is what makes the files on a file server visible to the client computer. The network redirector also makes remote printers act as though they were attached to the local computer. Figure 1.8 shows the Presentation layer's role in the protocol stack.

Graphic formats such as PICT, TIFF, and JPEG are examples of Presentation layer protocols.

FIGURE 1.7 The Session layer allows applications to establish communication sessions with each other.

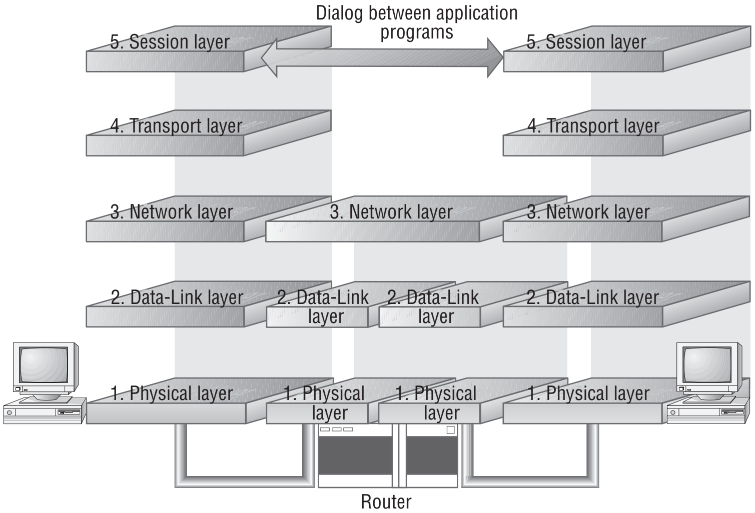
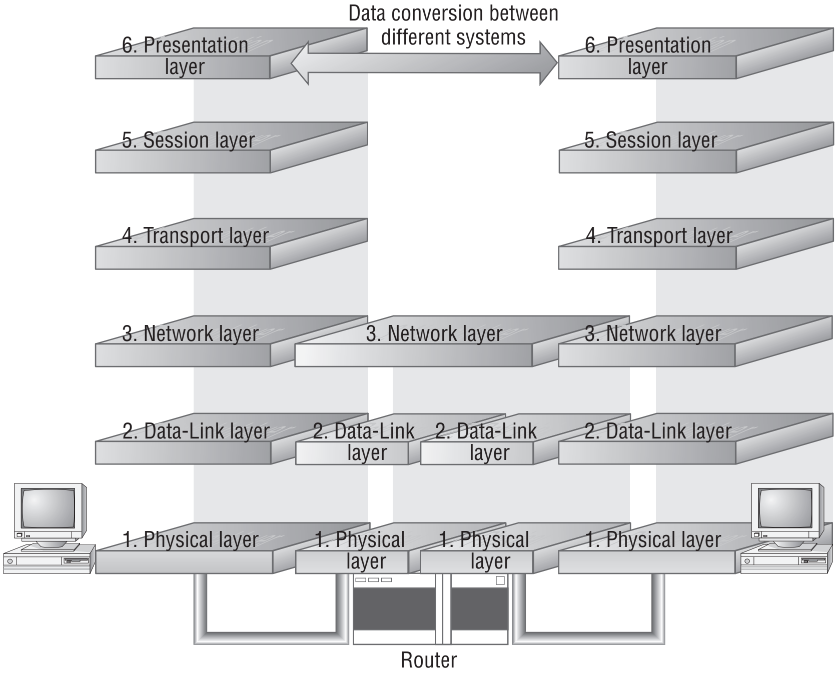


FIGURE 1.8 The Presentation layer allows applications to establish communication sessions with each other.

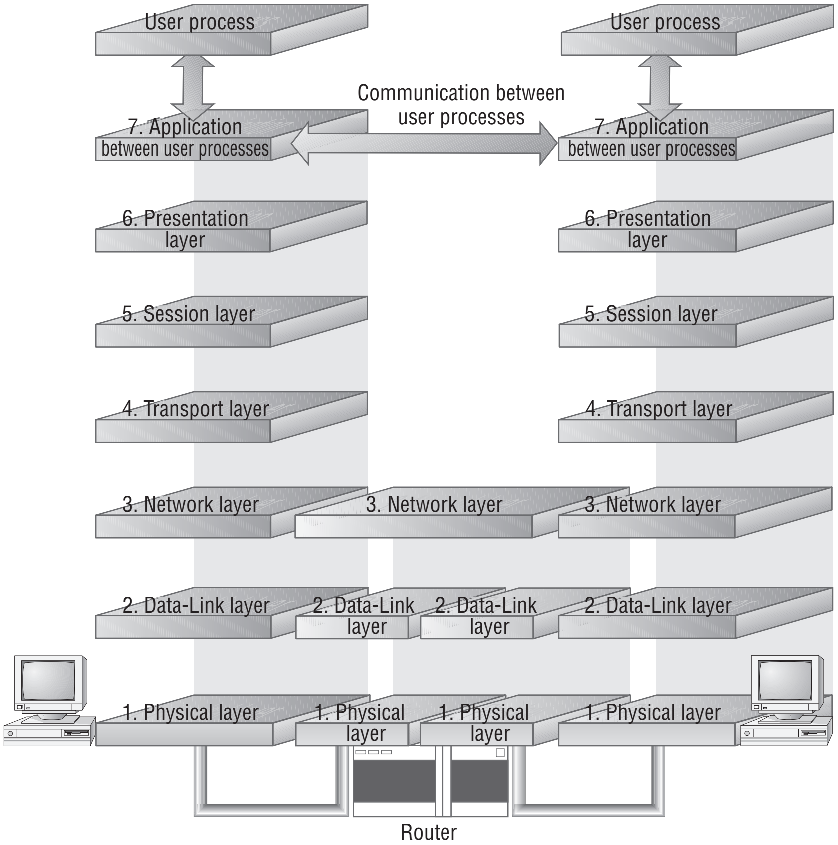


The Application Layer

The Application layer is the topmost layer of the OSI model, and it provides services that directly support user applications, such as database access, email, and file transfers. It also allows applications to communicate with applications on other computers as though they were on the same computer. When a programmer writes an application that uses network services, this is the layer the application will access. For example, Internet Explorer uses the Application layer to make its requests for files and web pages; the Application layer then passes those requests down the stack, with each succeeding layer doing its job (as in Figure 1.9).

File Transfer Protocol (FTP), Hypertext Transfer Protocol (HTTP), Simple Mail Transfer Protocol (SMTP), and others are examples of protocols at the Application layer.

FIGURE 1.9 The Application layer is where the applications function, using lower levels to get their work done.



Communication Between Stacks

When a message is sent from one machine to another, it travels down the layers on one machine and then up the layers on the other machine, as shown in Figure 1.10.

As the message travels down the first stack, each layer it passes through (except the Physical layer) adds a header. These headers contain pieces of control information that are read and processed by the corresponding layer on the receiving stack. As the message travels up the stack of the other machine, each layer removes the header added by its peer layer and uses the information it finds to figure out what to do with the message contents (see Figure 1.11).

FIGURE 1.10 Traffic flows down through the stack on one computer and up the stack on the other.

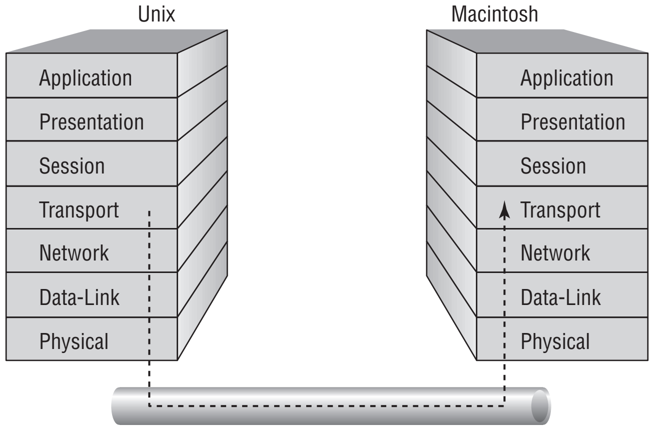
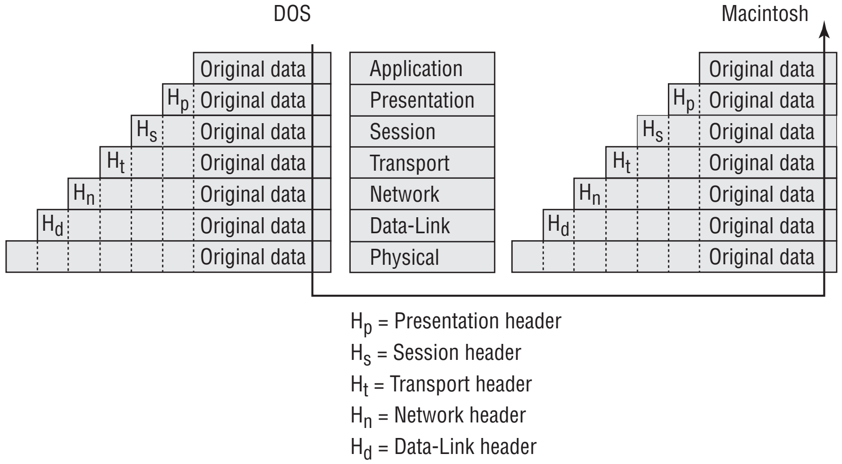


FIGURE 1.11 As packets flow up and down the stacks, each layer adds or removes necessary control information.



How Microsoft's Network Components Work with the OSI Model

Because the OSI model is so abstract, it can be hard to tell how its concepts relate to the actual network software and hardware you use in the real world. The following sections will make the link clearer. We will introduce you to some network protocols and show how they apply to the various layers of the OSI model.

NOTE

In the following sections, we will discuss some protocols that Microsoft Windows Server 2008 no longer supports. We will talk about some of these protocols to help give you a wider range of networking knowledge and to help explain some of the networking concepts in previous Microsoft Windows Server versions.

Device Drivers and the OSI Model

Every hardware device in a computer requires a software-based device driver to make it work. Some drivers—for instance, the driver for an Integrated Device Electronics (IDE) hard disk or for the keyboard—are built into the operating system. Other devices require that drivers be installed separately when the device is attached or installed in the computer. Windows Server 2008 really blurs this distinction because it includes drivers for several hundred different network cards, but if your card isn't on the list, you will need to install the driver provided by the manufacturer.

When Windows 3.11 was introduced, network drivers were vendor-specific for both the operating system and the card. You might, for instance, have had a difficult time if you wanted to put a 3Com Ethernet card and an IBM Token Ring card in the same server. Worse yet, most drivers could be bound only to a single protocol stack and a single card, so you couldn't have two cards using TCP/IP on one server.

A variety of vendors tried to solve this problem by developing driver interfaces that allowed multiple cards to be bound to multiple protocols. Apple and Novell developed the Open Datalink Interface (ODI), and Microsoft countered with the Network Driver Interface Specification (NDIS). Microsoft's operating systems have supported NDIS ever since, making it possible to bind either multiple protocols to one card or the same protocol to multiple cards.

NOTE

Windows Server 2008 and Windows Vista include Network Driver Interface Specification (NDIS) 6.0.

Network adapter cards and drivers provide the services corresponding to the Data-Link layer in the OSI model. In the IEEE model, the Data-Link layer is split into the LLC sublayer, which corresponds to the software drivers, and the MAC sublayer, which corresponds to the network adapter. You can think of the drivers as intermediaries between the higher layers and the card hardware that handles the business of forming packets and stuffing them into a wire.

Network Protocol Basics

Protocols are nothing more than an agreed-upon way for two objects (people, computers, home appliances, and so on) to exchange information. There are protocols at various levels in the OSI model. In fact, it is the protocols at a particular level in the OSI model that provide that level's functionality. Protocols that work together at one or more layers of the OSI model are known as a *protocol stack* or *protocol suite*. The following sections explain how network protocols move data between machines.

How Protocols Work

A *protocol* is a set of basic steps two or more parties perform according to a predefined or agreed-upon set of standards. A good example of a protocol that follows some unwritten but largely agreed-upon standards is a telephone conversation. When one person places a phone call, they dial the number of another party. The person on the other end answers the phone and says something akin to "Hello," at which point the calling party responds with a similar greeting. The conversation ensues from there. When the conversation is complete, each party (usually) ends the call with some parting words such as "Good-bye." This telephone call followed a routine protocol:

1. Say "Hello."
2. Converse.
3. Say "Goodbye."

In the realm of computers, a protocol follows the same concept. A protocol is a set of predefined standards that both computers must perform in the right order. For instance, for one computer to send a message to another computer, the first computer must perform the steps given in the following general example:

1. Break the data into small sections called *packets* (or *segments*, or another name depending on the layer involved).
2. Add addressing information to the packets, identifying the destination computer.
3. Deliver the data to the network card for transmission over the network.

The receiving computer must perform these steps:

1. Accept the data from the network adapter card.
2. Remove the transmitting information that was added by the transmitting computer.
3. Reassemble the packets of data into the original message.

Each computer needs to perform the same steps, in the same way and in the correct order, so that the data will arrive and be reassembled correctly. If one computer uses a protocol with different steps or even the same steps with different parameters (such as different sequencing, timing, or error correction), the two computers won't be able to communicate with each other.

Network Packets

Ethernet networks running IP and using TCP as the transport protocol primarily send and receive small chunks of data called *packets*. Network protocols construct, modify, and disassemble packets as they move data down the sending stack, across the network, and back up the OSI stack of the receiving computer. An IP packet has the following components:

- A source address specifying the sending computer
- A destination address specifying where the packet is being sent
- Instructions that tell the computer how to pass the data along
- Reassembly information (if the packet is part of a longer message)
- The data to be transmitted to the remote computer (often called the *packet payload*)
- Error-checking information to ensure that the data arrives intact

These components are assembled into slightly larger chunks; each packet contains three distinct parts (listed here and shown in Figure 1.12), and each part contains some of the components listed previously:

Header A typical header might include an identifier, source and destination addresses, and other options, depending on the protocol.

Data This is the actual data being sent.

Trailer The contents of the trailer (or even the existence of a trailer) vary among network types, but it typically includes a CRC. The CRC helps the network determine whether a packet has been damaged in transmission.

FIGURE 1.12 A packet consists of a header, the data, and a trailer.

Header
Data
Trailer

Protocols and Binding

Many different protocol stacks can perform network functions, and many different types of network interface cards can be installed in a computer. A computer may have more than one card, and a computer may use more than one protocol stack at the same time.

The *binding* process is what links the protocol stack to the network device driver for the network interface adapter. Several protocols can be bound to the same card. In addition, one computer with several interface adapters—for instance, a server that must be able to communicate with both a local area network and a network backbone—can have the same protocol bound to two or more network cards.

The binding process can be used throughout the OSI layers to link one protocol stack to another. The device driver (which implements the Data-Link layer) is bound to the network interface card (which implements the Physical layer). TCP/IP can be bound to the device driver, and the NWLink Session layer can be bound to the device driver.

Determining Connections

Communication between computers can be arranged in two ways:

- Using connectionless protocols
- Using connection-oriented protocols

It's important to understand the differences between them because different Windows Server 2008 services use both types.

Connectionless Protocols

It might seem odd to talk about a connectionless protocol for networks, but you use at least two of them just about every day: radio and television. Connectionless systems assume that all data will get through, so the protocol doesn't guarantee delivery or correct packet ordering. Think of shouting a message out of your window to someone walking by outside—there's no guarantee that they'll hear you, but it's quick and easy. These optimistic assumptions mean that there's no protocol overhead spent on these activities, so connectionless protocols tend to be fast. The User Datagram Protocol (UDP), which is part of the IP protocol suite, is an example of a connectionless Internet transport protocol. In fact, IP itself is connectionless, relying on upper-layer protocols such as TCP to provide the connection. The Domain Name System uses the UDP protocol.

Connection-Oriented Protocols

Connection-oriented systems work more like your telephone—you have to dial a number and establish a connection to the other end before you can send a message. Connection-oriented protocols pessimistically assume that some data will be lost or disordered in most transmissions. They guarantee that transmitted data will reach its destination in the proper sequence and that all data will get through. To accomplish this, connection-oriented protocols that also are considered *reliable* retain the transmitted data and negotiate for a retransmission when needed. Once all the needed data has arrived at the remote end, it can be reassembled into its proper sequence and passed to the higher-level protocols. This means that any application can depend on a connection-oriented transport to reliably deliver data exactly as it was transmitted. TCP is an example of a reliable connection-oriented Internet protocol. Frame Relay is an example of an *unreliable* connection-oriented protocol. Unreliable does not imply undependable. It just means that the protocol does not support the retransmission of lost or errored data.

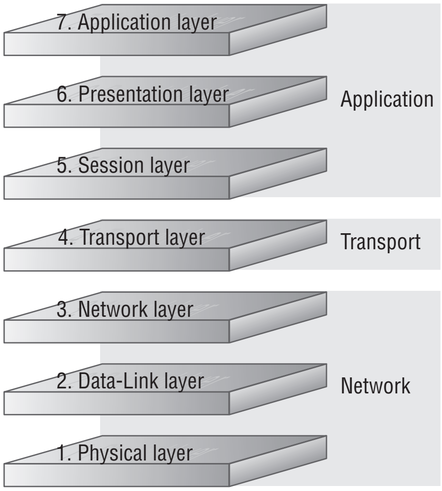
For local area systems where data isn't likely to be dropped, it makes sense to push serialization and guaranteed delivery up to higher-level protocols that are less efficient because they won't be used often anyway. But in wide area networks like the Internet, it would simply take too much time for higher-level protocols to sort out what data had been sent and what was missing, so the transport protocol takes measures to guarantee that all the data gets through in order.

Network Protocols and Windows Server 2008

A number of protocol stacks are used in the world's networks today. Besides NetWare, AppleTalk, NetBIOS, and TCP/IP, there are a bunch of specialty protocols such as IBM's Systems Network Architecture (SNA), Digital's (now HP/Compaq's) DECnet, and others. Even though these protocols actually work at different levels of the OSI model, they fall neatly into three distinct groups, as shown in the following list and in Figure 1.13:

- Application protocols provide for application-to-application interaction and data exchange.
- Transport protocols establish communication sessions between computers.
- Network protocols handle issues such as routing and addressing information, error checking, and retransmission requests.

FIGURE 1.13 Each OSI protocol works within one of three groups: Application, Transport, or Network.



Network Protocols No Longer Supported in Windows Server 2008

In Windows Server 2003, Microsoft supported many different protocol types such as NWLink, AppleTalk, Serial Line Interface Protocol (SLIP), and TCP/IP. In Windows Server 2008, Microsoft supports a different set of protocols.

Support for the following technologies has been removed from Windows Server 2008:

- Bandwidth Allocation Protocol (BAP)
- X.25
- Serial Line Interface Protocol (SLIP)

  SLIP-based connections will automatically be updated to PPP-based connections.

- Asynchronous Transfer Mode (ATM)
- IP over IEEE 1394
- NWLink IPX/SPX/NetBIOS Compatible Transport Protocol
- Services for Macintosh (SFM)
- The Open Shortest Path First (OSPF) routing protocol component in Routing and Remote Access
- Basic Firewall in Routing and Remote Access

  This protocol has been replaced with Windows Firewall.
- Static IP filter APIs for Routing and Remote Access

  This protocol has been replaced with Windows Filtering Platform APIs.
- The SPAP, EAP-MD5-CHAP, and MS-CHAP v1 authentication protocols for PPP-based connections

NOTE

This list of removed technologies is always subject to change. Microsoft may remove support for additional technologies because of technical or other reasons.

TCP/IP Preferred for Windows Server 2008

Microsoft Windows 2008 supports TCP/IP version 4 and TCP/IP version 6 networking protocols. TCP/IP is a complex transport sufficient for globe-spanning networks such as the Internet, and Microsoft is doing everything possible to position TCP/IP as a one-size-fits-all network protocol. TCP/IP is required to use Active Directory and is the default protocol for Windows Server 2008.

Understanding TCP/IP

TCP/IP is actually two sets of protocols bundled together: the Transmission Control Protocol and the Internet Protocol. The U.S. Department of Defense's Advanced Research Projects Agency (ARPA, or later DARPA) developed TCP/IP and its suite of protocols beginning in 1969. The original goal was to develop network protocols that were robust enough to route communications around damage caused by nuclear war. That design goal was never tested, but some aspects of that design have led to the redundant, distributed whole we call the Internet.

IP is by far the most widely used protocol for interconnecting computers, and it is the protocol of the Internet. This is because although ARPA originally created IP to connect military networks, it provided the protocol standards to government agencies and universities free of charge. The academic world leapt at the chance to use a robust protocol to interconnect their networks, and the Internet was born. Many organizations and individuals collaborated to create higher-level protocols for everything from newsgroups, mail transfer, and file transfer to printing, remote booting, and even document browsing.

NOTE

Although you'll see it commonly referred to as TCP/IP in Windows and throughout this book, TCP/IP really means the TCP/IP or IP protocol suite, and not necessarily TCP, the Transport layer protocol. UDP is also used, thus making it sort of a misnomer to say TCP/IP while ignoring UDP/IP.

IP is currently the protocol of choice for most networks because of its rapid and widespread adoption. IP is used for networks that span more than one metropolitan area or to connect to (or over) the Internet.

Real World Scenario

Understanding the OSI Model and Troubleshooting

The company you work for has several regional offices spread around the country. Your job is to make sure the resources on the Windows Server 2008 network, which include manufacturing, inventory, and sales information, are available at all times. If the sales information from the regional offices isn't collected and updated to the manufacturing and inventory programs, the company won't be able to supply its customers effectively. The users of the network aren't particularly interested in the technical nuts and bolts of the system, but they do care when the system is down.

At the same time, you're studying for your MCSE certification and wondering how the abstract notions of the OSI model are relevant to your job. A support call comes in from a user who can't connect to a printer on a Windows Server 2008 machine in another region where an executive management meeting is taking place. The user is down the hall from you, so you drop everything and run down to take a look.

With the OSI model fresh in mind, you approach the problem in terms of layers of functionality. You ping the address of your router, and it comes back fine. You now know that the Physical, Data-Link, and Network layers are working fine, which means you have eliminated cable and basic protocol problems. Your browser also seems to work fine because you can reach random sites. When you ping the name of the Windows Server 2008 machine that hosts the printer, you get the "request timed out" message. But when you ping the IP address directly, the reply shows a healthy connection, implying that you have a name resolution problem. You begin the task of looking at your DNS server.

By breaking down your troubleshooting tactics into the general OSI layers, you can better gauge where the problem lies and which services to look at, depending on where in the OSI model the symptoms appear. Although the OSI model is fairly abstract, when it's applied appropriately, it gives you a structure for thinking about your overall network and provides a framework for following methodical troubleshooting tactics.

IP has some significant advantages:

- Broad connectivity among all types of computers and servers, including direct access to the Internet
- Strong support for routing, using a number of flexible routing protocols
- Support for advanced name and address resolution services (which will be covered in more depth in Chapter 3, “Domain Name System (DNS)”): the Domain Name System (DNS), the Dynamic Host Configuration Protocol (DHCP), and the Windows Internet Name Service (WINS)
- Support for a wide variety of Internet-standard protocols, including protocols for mail transport, web browsing, and file and print services
- Centralized network number and name assignment, which facilitates internetworking between organizations

TCP/IP is the core protocol that Windows Server 2008 depends on for all its network services. In fact, most of this book focuses on TCP/IP and its related services.

NOTE Chapter 2 covers TCP/IP in detail.

About Windows Network Models

Now that you have learned how computers communicate with each other, we’ll discuss how you choose what kind of network to set up. Before you start setting it up, you have to make an important decision. Specifically, you need to decide what type of network model you need to install. The choice you make here is going to determine how you set up the rest of the computers and servers on your network.

The following sections describe two network models used by Microsoft Windows.

NOTE Since this is a Microsoft Windows certification book, the network models that are discussed in this book are Microsoft Windows–related models.

Windows Peer-to-Peer Network

In a Microsoft Windows peer-to-peer network (also referred to as a *workgroup*) all computers on the network are equal. All computers (also referred to as *nodes*) simultaneously act as both clients and servers. This is an advantage for small networks that have 12 or fewer users.

New Features of Windows Peer-to-Peer Network

Windows Vista includes some of the following enhancements to the Windows peer-to-peer network:

People Near Me This new Windows Vista feature provides four important services:

- Discovery of users on the same subnet
- Ability to invite users to an application
- Publication of objects
- Contact management through the use of the Windows Address Book

New application programming interface (API) Microsoft has improved and simplified its API's abilities to access Windows peer-to-peer networking capabilities such as name resolution, group creation, and security. This makes it easier for developers to create peer-to-peer applications for Windows.

Group Policy configuration support Windows Vista allows you to set up a Group Policy to configure Windows peer-to-peer networking settings.

New version of Peer Name Resolution Protocol (PNRP) Microsoft Windows Vista includes version 2 of the Peer Name Resolution Protocol (PNRP). This version uses less network bandwidth and is more scalable.

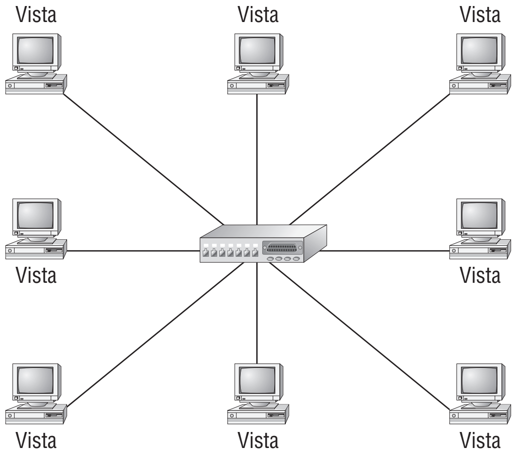
NOTE The new version of the PNRP protocol is incompatible with the version that came with the Windows XP Advanced Networking Pack. You can download the PNRP v2 upgrade for Windows XP through the Windows Update utility.

Windows Meeting Space Windows Vista has included a new peer-to-peer application called Windows Meeting Space. Windows Meeting Space addresses the needs of information workers by providing a means to invite, track, and detect the presence of attendees in a meeting. It also allows screen and window sharing among laptops, tablets, and projectors and file sharing with other meeting attendees.

Windows Peer-to-Peer Network Scenario

Imagine you are planning a network for a small real-estate office with five realtors. Should you set up a client/server-based network and spend money on a powerful machine, Windows 2008 Server, client access licenses (CALs), and a consultant who knows how to set up a Microsoft Active Directory domain? Or should you set up a Microsoft Windows peer-to-peer network? With a peer-to-peer network, all Windows Vista or XP Professional machines can be linked with each other through a small hub, and all users can share resources across the network (see Figure 1.14).

FIGURE 1.14 Microsoft Windows peer-to-peer network



A Microsoft Windows peer-to-peer model has some disadvantages. All data is stored on individual workstations, and the local workstation's owner controls the security. Each user needing to log onto a machine in a peer-to-peer network must have a local username and password. Returning to our example, let's suppose the real-estate office grows to employ 12 realtors. Each realtor needs to be able to log onto any machine in the network:

12 users × 12 computers = 144 user accounts that need to be created

The following are some other disadvantages to a peer-to-peer network:

Scalability A peer-to-peer network is limited to 12 computers.

Backups Backups must be done locally on each computer, instead of backing up one server.

Decentralized security Security is controlled by the local workstation's owner. Imagine having 200 files on 12 different workstations. If one user goes on vacation, you might not be able to access a needed file on that user's workstation.

NOTE

Now, at this point you may be saying to yourself, "We can just install a Windows 2008 file server, and that will solve the backup and decentralized security issues." True. A Windows 2008 file server has a central location to place files for backups and security. But, if you were going to install a Windows 2008 Server anyway, why would you even need to stay on a peer-to-peer network? You could make that server a domain controller and take advantage of all the benefits of a domain (see the next section of this chapter).

A Microsoft Windows peer-to-peer network does have its place in the vast computer universe. It's great for a small company that is trying to save money while still sharing network resources. But if a peer-to-peer network does not fit into your part of the computer universe, you have another Microsoft networking option.

Windows Server 2008 Active Directory Network

IT departments for companies are responsible for maintaining the security of the company's information. This involves planning for, implementing, and managing various network resources. Servers, workstations, and routers are common infrastructure devices that are used to connect users with the information they need to do their jobs. In all but the smallest environments, the effort required to manage each of these technological resources can be great.

That's where Windows Server 2008 and Microsoft Active Directory come in. Active Directory is a data store that allows administrators to manage various types of information within a single distributed database. This is no small task, but many features of this directory services technology allow it to meet the needs of organizations that are small or large in size.

In its most basic definition, a directory is a repository that records information and makes it available to users. The overall design goal for Active Directory is to create a single centralized repository of information that administrators can work with to securely manage a company's user accounts, security, applications, and more.

An Active Directory setup consists of one or more *domains*. A domain is a logical grouping of objects within your organization. Objects within a domain do not have to be physically located near each other.

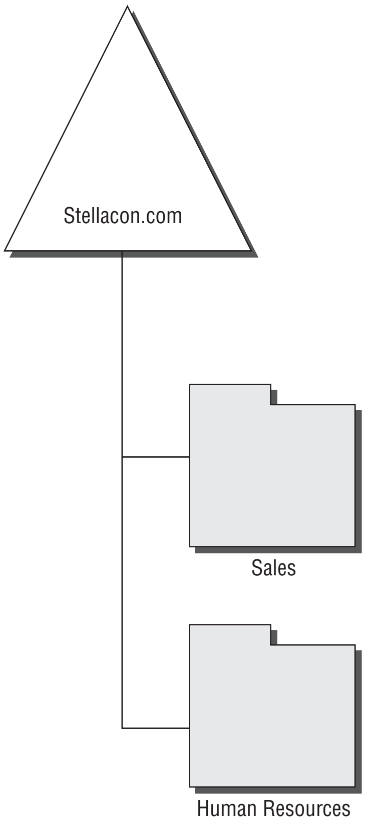
Active Directory's features include the following:

Hierarchical organization Active Directory is based on a hierarchical layout. Through the use of various organizational components (or *objects*), a company can create a network management infrastructure and directory structure that mirrors the business organization. For example, if a company called Stellacon.com had several departments (such as sales and human resources), the directory services model could reflect this structure through the use of various objects within the directory (see Figure 1.15). Stellacon.com could then organize its users into the appropriate department containers.

The directory structure can efficiently accommodate the physical and logical aspects of information resources, such as access to other databases, user permissions, and computers. Active Directory also integrates with the network naming service, the DNS. *The* DNS provides for the hierarchical naming and location of resources throughout the company and on the public Internet.

Centralized data storage All the information within Active Directory resides within a single, distributed, data repository. Users and systems administrators can easily access the information they need wherever they may be within the company. This is one of the most important design goals of the directory service—to provide a secure and centralized location for all your data. The benefits of centralized data storage include reduced administrative requirements, less duplication, higher availability, and increased visibility and organization of data.

FIGURE 1.15 Directory service model



Ease of administration To accommodate various business models, Active Directory can be configured for centralized or decentralized administration. This gives network and systems administrators the ability to delegate authority and responsibilities throughout the organization while still maintaining security. They allow for making companywide changes with just a few mouse clicks.

Network security Through the use of a single logon and various authentication and encryption mechanisms, Active Directory can facilitate security throughout an entire enterprise. Through the process of *delegation*, higher-level security authorities can grant permissions to other administrators. For ease of administration, objects in the Active Directory tree inherit permissions from their parent objects. Application developers can take advantage of many of these features to ensure that users are identified uniquely and securely. Network administrators can create and update permissions as needed from within a single repository, thereby reducing the chances of an inaccurate or outdated configuration.

Scalability Large organizations often have many users and large quantities of information to manage. Active Directory was designed with scalability in mind. Not only does it allow for storing millions of objects within a single domain, it also provides methods for distributing the necessary information between servers and locations. These features relieve much of the burden of designing a directory services infrastructure based on technical instead of business factors.

The biggest disadvantage to an Active Directory model is cost. The following are some of the cost items to consider:

- A powerful enough computer to handle Windows Server 2008
- The cost of personnel (staff IT or consultants) needed to plan, implement, and maintain the Windows Server 2008 Active Directory model

This is why it is important for you to decide which network model is right for your organization. The choice you make here will determine how your network is going to function and grow down the road.

Summary

This chapter covered how the OSI networking model is organized into seven layers (Physical, Data-Link, Network, Transport, Session, Presentation, and Application) and described each level of the OSI stack. You also learned that Windows Server 2008 includes support for TCP/IP v4 and v6. TCP/IP is the primary protocol in use today, and Microsoft encourages you to use TCP/IP exclusively, if possible.

Finally, the chapter discussed the two different Microsoft Windows network models (Windows peer-to-peer and Windows Active Directory networks) and the advantages and disadvantages of these models.

Exam Essentials

Know which protocols Windows Server 2008 supports. Previous versions of Microsoft Windows Server supported many networking protocols. Windows Server 2008 no longer supports many of these protocols, including BAP, X.25, SLIP, ATM, IP over IEEE 1394, NWLink IPX/SPX/NetBIOS Compatible Transport Protocol, SFM, several components of Routing and Remote Access (OSPF, Basic Firewall, and static IP filter APIs), SPAP, EAP-MD5-CHAP, and MS-CHAP v1 authentication protocols for PPP-based connections.

Review Questions

1. A packet is sent from one computer to another across a network. Various protocols move the packet down the OSI stack from the sending computer and up the OSI stack to the receiving computer. How do the protocols know where to send the packet?

   A. Each packet has a trailer that contains source and destination addresses.

   B. Each packet has a header that contains an alert signal and source and destination addresses.

   C. The data portion of every packet stores all the source and destination information.

   D. Special packets, called *header packets*, that contain only source and destination addresses are sent first. Every packet that follows the header packet is sent to the destination address contained in the header packet.

2. You are the administrator for a software development house that writes small utility programs for a wide range of networks. In addition to supporting its Windows Server 2008 network, you are responsible for verifying that some of the applications that are developed function properly. During these tests, you have to install transport protocols from other development houses that are used by various systems. You have worked out the issues surrounding the different protocols working on Windows Server 2008 by requiring the protocol developers to make sure their protocols are compliant with what standard?

   A. ODI

   B. DLC

   C. NDIS

   D. NetBIOS

3. You administer a very large network that consists of Windows Vista, Windows XP Professional, and Windows Server 2008 computers. You want to implement DNS, DHCP, and WINS, and every computer must have access to the Internet and services on non-Windows machines. You want to be able to configure the network from a central location. Which network protocol provides the ability to do all these things?

   A. NetBEUI

   B. NWLink

   C. TCP

   D. TCP/IP

4. You are the administrator for a Windows NT network that has been internally focused on basic file and print services. You have been charged with upgrading your network to Windows Server 2008 and also allowing the users of the network to find information on the Internet. Currently, the network is running NWLink because of routing needs between two locations and a lack in the IT department of IP experience. You need to change the network protocol to TCP/IP to support Internet connectivity. What layers in the OSI model do you need to consider to allow the workstations to access the Internet for simple browsing? (Choose all that apply.)

   A. Network layer

   B. Application layer

   C. Presentation layer

   D. Transport layer

5. You have just been asked to troubleshoot intermittent communication problems on a fairly old network for a company that builds and repairs elevator motors. You have determined that the network is a straightforward thin-coax Ethernet Windows NT LAN running TCP/IP. The company wants to upgrade to Windows Server 2008, hoping that the now-stable platform will resolve the intermittent problems. You perform the upgrade; all goes smoothly, and initially everything seems to function properly. However, the intermittent problems show up again. What layer in the OSI model is the most likely place for the problems to be occurring?

   A. Physical layer

   B. Data-Link layer

   C. Network layer

   D. Transport layer

   E. Session layer

6. This sublayer, the lower of two sublayers, provides for shared access to the network adapter and communicates directly with network interface cards. A unique 48-bit address, commonly represented as a 12-digit hexadecimal address, is assigned to network interface cards before they leave the factory where they are made. What is the name of this sublayer?

   A. The TCP/IP sublayer

   B. The ISO sublayer

   C. The MAC sublayer

   D. The ARP sublayer

   E. The LLC sublayer

7. This OSI layer handles moving packets between devices. It makes routing decisions and forwards packets as necessary to help them travel to their intended destination. In larger networks, there may be intermediate devices and subnetworks between any two end systems. This OSI layer makes it possible for the Transport layer to send packets without being concerned with whether the end system is on the same piece of network cable or on the other end of a large wide area network. What layer in the OSI model makes this possible?

A. Physical layer

B. Data-Link layer

C. Network layer

D. Transport layer

E. Session layer

8. You are the administrator for a small company that makes video games. The company uses Windows XP, Windows Vista, and Windows Server 2008. In the last two days users have been complaining the network has been running much more slowly than usual. You monitor the performance of the server, but the server is showing no signs of problems. You believe that there might be a computer or device on the network that is sending large amounts of broadcast traffic. What type of utility can you use to examine the packets on your network?

A. Performance Monitor

B. Ethernet cable tester

C. `ipconfig /all`

D. Packet sniffer

9. The OSI layer is responsible for using electric (or sometimes other types of) signaling to get bits from one computer to another. This layer's components don't care what the bits mean; their job is to get the bits from point A to point B, using whatever kind of optical, electrical, or wireless connection that connects the points. What layer in the OSI model are we referring to?

A. Application layer

B. Data-Link layer

C. Network layer

D. Transport layer

E. Physical layer

10. You have been hired by a small travel agency to set up a Microsoft Windows network. Currently the travel agency has six travel agents who use Windows XP Professional and Vista. The computers are stand-alone machines with no network connection. The owner of the travel agency wants all the agents to be able to share resources (files and printers), but they are very concerned with minimizing costs. What type of network should you install?

A. Stand-alone network

B. Active Directory domain-based network

C. Peer-to-peer network

D. Client/server-based network

11. This OSI layer provides for the flow of data over a single physical link from one device to another. It accepts packets from the Network layer and packages the information into data units called *frames*; these frames are presented to the Physical layer for transmission. This OSI layer adds control information, such as the frame type, to the data being sent. This layer also provides for the error-free transfer of frames from one computer to another. Which OSI layer does these tasks?

    A. Application layer

    B. Data-Link layer

    C. Network layer

    D. Transport layer

    E. Physical layer

12. You have been hired by a large food chain to set up a Microsoft Windows network. The food chain currently has 100 computer users, and it plans on increasing that number by 50 percent. Scalability, network security, centralized data storage, and administration are all objectives that must be achieved. What type of network should you install?

    A. Stand-alone network

    B. Active Directory domain-based network

    C. Peer-to-peer network

    D. Client/server-based network

13. This OSI layer ensures that data is delivered error free, in sequence, and with no losses or duplications. This layer also can break large messages from the Session layer into smaller segments to be handed down to the Network layer and sent to the destination computer and then reassemble segments into messages to be presented to the Session layer. This OSI layer can send an acknowledgment to the originator for messages received. Which OSI layer does these tasks?

    A. Application layer

    B. Data-Link layer

    C. Network layer

    D. Transport layer

    E. Physical layer

14. What are the seven OSI Layers in order from top to bottom?

    A. Application, Physical, Session, Transport, Network, Data-Link, Presentation

    B. Application, Physical, Session, Network, Transport, Data-Link, Presentation

    C. Application, Presentation, Session, Network, Transport, Data-Link, Physical

    D. Application, Presentation, Session, Transport, Network, Data-Link, Physical

15. This OSI layer translates data between the formats the network requires and the formats the computer expects. This OSI layer also performs protocol conversion; data translation, compression, and encryption; character set conversion; and the interpretation of graphics commands. What OSI layer are we referring to?

A. Application layer

B. Data-Link layer

C. Network layer

D. Transport layer

E. Presentation layer

16. You are the administrator for a Windows peer-to-peer network. All users have Windows Vista on their computers. The owner tells you he needs his employees to have the ability to invite other employees via computer to a meeting where they can share files with other meeting attendees. What application can be used to solve this problem?

A. Windows Explorer

B. Windows Meeting Space

C. Windows Contact Manager

D. Exchange Server

17. This OSI layer allows applications on separate computers to share a connection. This layer provides services that allow two programs to find each other and establish the communication link, such as name lookup and security. This OSI layer also provides for data synchronization and check pointing so that in the event of a network failure, only the data sent after the point of failure would need to be resent. This layer controls the dialogue between two processes and determines who can transmit and who can receive at what point during the communication. Which OSI layer does this?

A. Session layer

B. Data-Link layer

C. Network layer

D. Transport layer

E. Physical layer

18. Which one of the following uses the UDP connectionless protocol?

A. TCP

B. FTP

C. DNS

D. None of the above

19. What is the name of the process that links the protocol stack to the network device driver for the network interface adapter?

A. Linking

B. Binding

C. Joining

D. LNIC

20. The TCP/IP model (sometimes called the IP model or the DoD model) is truly what the Internet was built upon. The TCP/IP model has four layers. What are the four layers, in order from top to bottom?

A. Internet, Application, Transport, and Link

B. Transport, Internet, Link, and Application

C. Application, Transport, Internet, and Link

D. Application, Link, Internet, and Transport

Answers to Review Questions

1. B. Each packet has a header and data and is typically placed in a frame that consists of three parts: a header, data, and a trailer. The packet header includes the source and destination logical addresses.

2. C . Network Driver Interface Specification (NDIS) provides a standard way for protocols to bind to the Data-Link drivers. As long as a developer supports NDIS, the protocol will load in Windows Server 2008. However, this will not make it interoperate with the Windows Server 2008 services. The applications will have to be written to the specific protocols.

3. D. TCP/IP is the most widely used protocol for interconnecting computers and networks. It is the only protocol used on the Internet and the only one compatible with the other protocols mentioned. It works well with very large internetworks.

4. A, B, C, D. TCP sits at the Transport layer, and IP sits at the Network layer; both are necessary to route requests through the Internet. However, you also need a browser such as Netscape or Internet Explorer to provide the HTTP calls to actually connect to the various websites; the browser sits at the Application layer. But any end-to-end communication uses all the levels of the OSI model because each layer communicates with the layer below and the layer above to form the complete chain.

5. A. The Physical layer is concerned with signaling, specifically through electrical, optical, or radio signals. The high voltage associated with large motors can easily cause an interruption in the signaling of coax cable. There have been many cases of people running network cable through elevator shafts in a building because of the ease of access, only to have the network malfunction every time someone summons the car. The other layers are associated with software and are beyond the reach of most electrical interference unless it affects the entire workstation.

6. C. The IEEE split the ISO Data-Link layer into the LLC sublayer and the MAC sublayer. The MAC sublayer, the lower of the two sublayers, provides for shared access to the network adapter and communicates directly with network interface cards. A unique 48-bit address, commonly represented as a 12-digit hexadecimal MAC address (frequently called the *hardware Ethernet address*), is assigned to network interface cards before they leave the factory where they are made.

7. C. To do its job, the Network layer translates logical network addresses into physical machine addresses (MAC addresses, which operate at the Data-Link layer). The Network layer also determines the quality of service (such as the priority of the message) and the route a message will take if there are several ways a message can get to its destination. The Network layer also may split large packets into smaller chunks if the packet is larger than the largest data frame the Data-Link layer will accept. The Network layer reassembles the chunks into packets at the receiving end.

8. D. A packet sniffer is a utility used to extract packets from a network cable so that the packet information may be examined. There are many different packet sniffers on the market, but most will be able to determine what types of packets are being sent. Using a packet sniffer will allow you to determine whether a computer or device is sending high amounts of broadcast traffic.

9. E. The Physical layer is concerned with signaling, specifically through electrical, optical, or radio signals. This layer's components don't care what the bits mean; their job is to get the bits from point A to point B via the optical, electrical, or wireless connection between the points.

10. C. A peer-to-peer network works well with a company that has 12 or fewer users. The advantage of a peer-to-peer network is that users can share resources across a network without spending money on powerful equipment and server software.

11. B. The Data-Link layer provides for the flow of data over a single physical link from one device to another. It accepts packets from the Network layer and packages the information into data units called *frames*; these frames are presented to the Physical layer for transmission. The Data-Link layer adds control information, such as frame type, to the data being sent.

12. B. An Active Directory domain-based network covers all the objectives that the food chain requires. An Active Directory–based network provides security, scalability, centralized data, and administrative ease.

13. D. The Transport layer ensures that data is delivered error free, in sequence, and with no losses or duplications. The Transport layer also can break large messages from the Session layer into smaller segments to be handed down to the Network layer and sent to the destination computer; the Transport layer then reassembles the segments into messages to be presented to the Session layer. The Transport layer can send an acknowledgment to the originator for messages received.

14. D. The OSI layers, in order from top to bottom, are Application, Presentation, Session, Transport, Network, Data-Link, and Physical. The easiest way to remember the seven layers is to use a mnemonic such as "All People Seem To Need Data Processing."

15. E. The Presentation layer translates data between the formats the network requires and the formats the computer expects. The Presentation layer also performs protocol conversion; data translation, compression, and encryption; character set conversion; and the interpretation of graphics commands.

16. B. Windows Vista includes a new peer-to-peer application called Windows Meeting Space. Windows Meeting Space helps information workers address their needs by providing a means to invite, track, and detect the presence of attendees in a meeting. It also allows screen and window sharing between laptops, tablets, and projectors, as well as file sharing with other meeting attendees.

17. A. The Session layer allows applications on separate computers to share a connection called a *session*. This layer provides services that allow two programs to find each other and establish the communication link, such as name lookup and security. The Session layer also provides for data synchronization and check pointing so that in the event of a network failure, only the data sent after the point of failure would need to be resent. This layer also controls the dialogue between two processes and determines who can transmit and who can receive at what point during the communication.

18. C. Domain Name System (DNS) uses the UDP protocol. The *User Datagram Protocol (UDP)*, which is part of the IP protocol suite, is an example of a connectionless Internet transport protocol. In fact, IP itself is connectionless, relying on upper-layer protocols such as TCP to provide the connection.

19. B. The *binding* process links the protocol stack to the network device driver for the network interface adapter.

20. C. The TCP/IP model (also called the IP model or the DoD model) was truly what the Internet was built upon. The TCP/IP model contains four layers: Application, Transport (sometimes called Host to Host), Internet, and Link (also called Network Access).