Exam Topics in This Chapter

12 UNIX

13 Windows (NT/95/98/2000)

45 Cisco Secure UNIX

46 Cisco Secure NT

48 Cisco Secure Policy Manager (formerly Cisco Security Manager)

49 Cisco Secure Intrusion Detection System (formerly NetRanger)

50 Cisco Secure Scanner (formerly NetSonar)

CHAPTER 6

Operating Systems and Cisco Security Applications

This chapter reviews two of today's most common end user applications, UNIX and Windows NT systems. Cisco security applications are also covered.

This chapter covers the following topics:

- **UNIX**—The UNIX operating system and some of the most widely used operating commands. The section looks at the files that are manipulated in UNIX to monitor and maintain usernames and passwords.
- **Microsoft NT Systems**—Windows NT 4.0 and some of the concepts used to manage users and domains.
- **Cisco Secure for Windows and UNIX**—Cisco Secure Access Control Server (ACS), the Cisco security application that is available on Windows and UNIX platforms.
- **NetSonar and NetRanger**—Cisco supported applications, NetSonar (Cisco Secure Scanner) and NetRanger (Cisco Secure Intrusion Detection System), to ensure that networks are secured and tested for vulnerabilities.

"Do I Know This Already?" Quiz

The purpose of this assessment quiz is to help you determine how to spend your limited study time. If you can answer most or all these questions, you might want to skim the "Foundation Topics" section and return to it later, as necessary. Review the "Foundation Summary" section and answer the questions at the end of the chapter to make sure that you have a strong grasp of the material covered. If you intend to read the entire chapter, you do not necessarily need to answer these questions now. If you find these assessment questions difficult, you should read through the entire "Foundation Topics" section and review it until you feel comfortable with your ability to answer all these and the "Q & A" questions at the end of the chapter.

Answers to these questions can be found in Appendix A, "Answers to Quiz Questions."

1 What UNIX command implements a trace route to the remote network www.guitar.com?

a. **trace www.guitar.com** if DNS is enabled with the IOS **command dns server** *ip-address*.

b. **traceroute www.guitar.com**

c. **trace guitar.com**

d. UNIX does not support the **traceroute** command.

2 What UNIX command copies a file?

a. **copy**

b. **cpy**

c. **cp**

d. **pc**

3 A Cisco router network manager wants to copy the configuration in RAM to a UNIX server. What needs to be accomplished before this can occur?

a. Issue **copy run tftp**.

b. Modify the .rhosts file.

c. Modify the rcmd.allow file.

d. Erase the .rhosts.allow file.

e. Enable TFTP on the UNIX server.

4 Which of the following is not a UNIX file flag parameter?

a. Execute

b. Write

c. Read

d. Read/Write

e. Authenticate

5 Which of the following is not a UNIX file type?

a. Normal

b. Directories

c. Special

d. Link

e. Medium

6 NetBIOS over TCP/IP operates at what layer of the OSI model?

a. 1

b. 2

c. 3

d. 4

e. 5

f. 6

g. 7

7 In Windows NT, what is a domain that is trusted by all remote domains called?

a. Local

b. Remote

c. Single

d. Global

e. Master

f. Slave

8 In Windows NT, what is a domain that is trusted automatically called?

a. Local

b. Remote

c. Single

d. Global

e. Master

f. Slave

9 Which of the following is not an NTFS permission type?

a. R

b. W

c. D

d. P

e. O

f. M

10 In Windows NT, when in a DOS command window, what command displays the local IP ARP entries?

a. **arp**

b. **rarp**

c. **rarp –b**

d. **arp –n**

e. **arp –a**

11 What devices can the Cisco Secure Policy Manager remotely manage? (Select the best three answers.)

a. Routers

b. Switches

c. NMS workstations

d. PIX Firewalls

12 NetRanger LAN interface supports all but which one of the following?

a. Ethernet

b. Fast Ethernet

c. Token Ring

d. Serial WAN interfaces

e. FDDI

13 Which of the following is not a component of the security wheel?

a. Develop

b. Secure

c. Monitor

d. Manage

e. Increase

14 Which of the following is false in regards to NetRanger?

a. NetRanger examines the IP header.

b. NetRanger examines the TCP header.

c. NetRanger examines the entire IP frame.

d. NetRanger monitors TCP or UDP port scans.

15 How many phases are completed with NetSonar?

a. 1

b. 2

c. 3

d. 4

e. 5

f. 6

Foundation Topics

UNIX

The UNIX operating system was developed in 1969 at Bell Laboratories. UNIX has continued to develop since its inception. AT&T, for example, released UNIX 4.0.

UNIX was designed to be a multiuser system (more than one user can connect to the host at one time), and it is used usually for multiuser systems and networks.

Because most engineers are more familiar with DOS (and Windows NT) than UNIX, this section presents some analogies to demonstrate the UNIX command structure.

The operating system DOS used in the early days is similar to UNIX in terms of architecture. For example, the command syntax to list the directories in DOS is **dir**, and in UNIX, it is **ls**.

Table 6-1 displays some of the common commands between UNIX and DOS.

Table 6-1 *DOS Versus UNIX Commands*

DOS/Windows NT Command	UNIX Command	Purpose
attrib +h/-h	All files starting with a dot (for example .hosts) are hidden automatically. The UNIX command **mv** renames a file. For example, **mv hosts .hosts** hides the file named hosts.	Either hides (**+h**) or uncovers (**-h**) files from directory lists when the command **dir** is used. The **attrib** command also displays the file attributes. In UNIX, the . (dot) automatically hides files.
cd *dirname*	**cd** *dirname*	Moves the user to a specific directory.
chkdsk	**Df**	Checks the disk for logical problems; only admin users can perform this command in UNIX. UNIX commands are case-sensitive.
copy/**xcopy** *dirname/filename*	**cp** *dirname/filename*	Allows you to copy files.
del/**erase** *filename*	**rm** *filename*	Erases files from the disk.
dir	**ls**	Lists the files in the current directory.
help *command name*	**man** *command name*	Displays information about the specified command.
rename *oldfilename newfilename*	**mv** *oldfilename newfilename*	Renames a file. In UNIX, it can also be used to move the file to a different directory.

Table 6-1 *DOS Versus UNIX Commands (Continued)*

DOS/Windows NT Command	UNIX Command	Purpose
ping *ip-address*	**ping** *ip-address*	Pings a local or remote host.
tracert	**traceroute**	Windows sends ICMP requests with varying time to live (TTL) values. UNIX sends UDP probes, varies the TTL values, and watches for any ICMP messages returned.

NOTE The Windows DOS-based **attrib** command is a widely used command that modifies file attributes. In a Windows environment, the options include the following:

```
C:\ >help attrib
Displays or changes file attributes.
ATTRIB [+R ¦ -R] [+A ¦ -A ] [+S ¦ -S] [+H ¦ -H] [[drive:] [path] filename]
       [/S [/D]]
  +   Sets an attribute.
  -   Clears an attribute.
  R   Read-only file attribute.
  A   Archive file attribute.
  S   System file attribute.
  H   Hidden file attribute.
  /S  Processes matching files in the current folder
      and all subfolders.
  /D  Processes folders as well
```

The **attrib** command allows files to be read only, archived, made a system file, or hidden.

In UNIX, you use the **man** command for command syntax help:

```
Simonunixhost% man
Usage: man [-M path] [-T macro-package] [ section ] name ...
or: man -k keyword ...
or: man -f file ...
```

UNIX Command Structure

UNIX servers and hosts are managed using files. To manage the files, you need to be aware of the UNIX command structure.

A UNIX command contains three basic parts:

- Command
- Flags
- Arguments

Figure 6-1 displays the parts of a UNIX command.

Figure 6-1 *Three Parts of a UNIX Command*

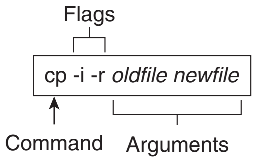


Figure 6-1 displays the copy request command (**cp**). Notice that most UNIX commands are abbreviations of English words. For example, the copy command is defined by **cp**. The first part of any UNIX command tells the device to run a specific program or process, such as the copy function. The second part identifies any flags, which directly follow the UNIX process commands; dashes (-) identify flags. The flags in Figure 6-1 are defined as the **-i** flag, telling the UNIX host to confirm before it overwrites any files in this process, and the **-r** flag, telling the UNIX host to copy any files in subdirectories if you are copying directories.

Finally, the last part is the argument, which, in most cases, is the name of a file or directory. In Figure 6-1, for example, the old filename and the new filename must be specified.

Table 6-2 displays some common UNIX commands and their meanings.

Table 6-2 *Common UNIX Commands*

Command	Description	Example
cp -i/-r *oldfile newfile*	Makes a copy of a file. You must specify the name of the file to be copied and the name of the new file to be created. The **-i** flag tells the computer to ask before it overwrites any files in this process. The **-r** flag copies any files in subdirectories if you are copying directories.	**cp -i simon.doc henry.doc**
rm -i/-r *filename*	Erases the specified file. The **-i** flag asks you for confirmation before a file is deleted. The **-r** flag erases directories/ subdirectories and all the files they might contain.	**rm -i cisco**

Table 6-2 *Common UNIX Commands (Continued)*

Command	Description	Example
rmdir -p *directoryname*	Erases directories. The **-p** flag allows you to erase a directory and all its contents. Without this flag, the directory must be empty before you erase it.	**rmdir –ptomII**
mv -i *filename1 filename2*	Renames a file. The **-i** flag asks for confirmation before overwriting a file if you attempt to use a filename that is already taken. Without the flag, the original file with the same name is automatically erased.	**mv 2002ccie 10000ccie**
mv -i *filename directoryname/filename*	Moves a file to another directory. The flag serves the same purpose as in the other **mv** command.	**mv index.html index1.html**
man *command*	Displays a description and usage instructions for a specified command. This command is similar to **help** in a Windows environment.	**man ls**
grep -i	Allows you to search for a string in files. The flag –**i** tells the UNIX server to ignore upper- or lowercase.	**grep -i myword *.txt** Searches for the keyword myword in all files that end in .txt.
netstat -s	Displays a description and usage instructions for a specified command. The **netstat -s** displays statistics for network interfaces and protocols, such as TCP.	**netstat -s**
ifconfig -a	Displays the current interfaces that are configured. Displays the IP address and subnet mask.	**ifconfig –a**

NOTE All UNIX commands are in lowercase and are case-sensitive. For a free tutorial on UNIX, visit www.ee.surrey.ac.uk/Teaching/Unix/.

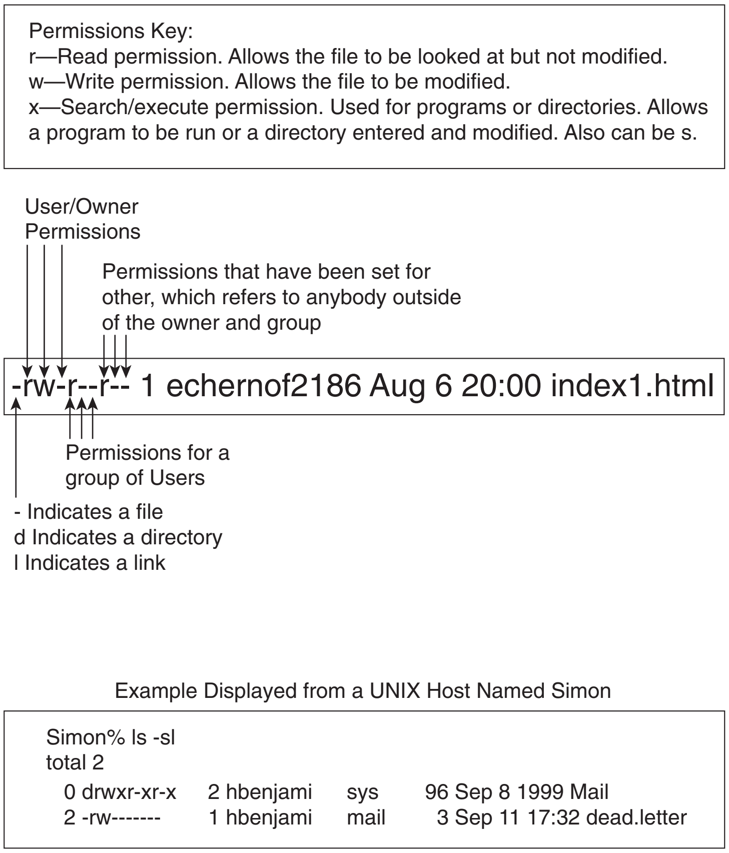
UNIX Permissions

UNIX allows certain users access to files and commands by setting permissions to ensure that only legitimate users are permitted access to files and directories.

To view information about each file, use the **-l** flag with the UNIX command **ls** (for example, **ls –l**). The command **ls –s** lists the current UNIX permissions. To display both the file permissions and file information, combine the flags –**s** and –**l** with the command **ls** (for example, **ls –sl** or **ls –ls**). Figure 6-2 displays a sample output for the command **ls -ls** for a UNIX host named Simon.

Figure 6-2 also displays a sample output of the command **ls -sl** and explains the meaning of this output.

Figure 6-2 **ls -sl** *Command Output*



Permissions Key:
r—Read permission. Allows the file to be looked at but not modified.
w—Write permission. Allows the file to be modified.
x—Search/execute permission. Used for programs or directories. Allows a program to be run or a directory entered and modified. Also can be s.

User/Owner Permissions
Permissions that have been set for other, which refers to anybody outside of the owner and group
-rw-r--r-- 1 echernof2186 Aug 6 20:00 index1.html
Permissions for a group of Users
- Indicates a file
d Indicates a directory
l Indicates a link

Example Displayed from a UNIX Host Named Simon

```
Simon% ls -sl
total 2
   0 drwxr-xr-x    2 hbenjami    sys       96 Sep 8 1999 Mail
   2 -rw-------    1 hbenjami    mail       3 Sep 11 17:32 dead.letter
```

When a new file is created in UNIX, the default is to define read and write access to the owner. To set new or modify permissions, use the command **chmod** *flag filename*.

The **chmod** flag is always three numbers. The first number affects the owner permissions (U), the second number affects the group permissions (g), and the third number affects the other (o) permissions. Each number can be a number between 0 and 7; Table 6-3 displays the possible values for each flag.

Table 6-3 **chmod** *Flag Definitions*

Number	Value
0	No permissions
1	Execute only
2	Write only
3	Write and execute
4	Read only
5	Read and execute
6	Read and write
7	Read, write, and execute

NOTE The network administrator is typically given the root password allowing configuration changes, program execution, and file management. For example, to connect a new hard drive, the installation engineer requires the root password. The administrator types in the root password first. After entering the root password, the administrator types the UNIX command **mount** to attach or detach a file system, also known as the super user.

UNIX File Systems

UNIX can consist of four main files types:

- **Normal files**—Contain user data
- **Directories**—Containers that hold files
- **Special files**—Input and output devices, such as a disk drive, printer, or CD-ROM
- **Links**—Pointers to another file

UNIX stores files and important information in directories. The following are some common examples (might vary according to the UNIX version):

- **/bin/**—Executable system utilities, such as **sh**, **cp**, and **rm**.
- **/etc/**—System configuration files and databases.

- **/lib/**—Operating system and programming libraries.
- **/tmp/**—System scratch files (all users can write here).
- **/lost+found/**—Where the file system checker puts detached files.
- **/usr/bin/**—Additional user commands.
- **/usr/include/**—Standard system header files.
- **/usr/lib/**—More programming and system call libraries.
- **/usr/local/**—Typically a place where local utilities go.
- **/usr/man**—The manual pages are kept here.

NOTE Certain system files created by UNIX store important details about the operational characteristics, such as the password lists for all users.

The file named shadow in the /etc directory is a read only, protected file referenced by the program login.

The file named passwd contains the passwords for all users.

The file named wtmp contains an account of all users that logged into the UNIX host.

The file named lastlog contains details of when a user logged out of a UNIX host.

The file .rhosts contains information permitting remote devices, such as routers, the capability to TFTP or Remote Copy Protocol (RCP) files to a UNIX host.

Microsoft NT Systems

This section briefly covers Windows NT 4.0. Cisco Systems requires you to have no more than a conceptual overview on Windows NT systems, so the detail in the next section is only provided to give you the required foundations to pass the CCIE Security written exam.

Windows NT allows clients and servers to be grouped into domains or workgroups. A *domain* is typically a large group of devices under a common administration. A *workgroup* usually describes a smaller group of Windows devices or any logical collection of computers. A domain is managed by a primary domain controller (PDC), which is a Windows-based server that stores and controls security and user account information for an entire domain. Each domain must have at least one PDC. A backup domain controller (BDC) maintains a copy of the database in the event the PDC is unavailable.

NetBEUI was first developed by IBM in the mid 1980s to provide an interface for applications that were currently using Network Basic Input/Output System (NetBIOS).

Before routing became popular, NetBEUI was developed as a Layer 2 protocol that allowed devices, such as PCs, to communicate over a broadcast medium, such as Ethernet. NetBEUI was also designed for earlier versions of Windows (Windows 3.1 and MS-DOS-based clients).

NetBEUI is not routable and must be bridged when networks are not locally reachable. NetBEUI is still used today.

NetBIOS is a session layer protocol that allows communication between PCs in domains or workgroups.

NetBIOS provides the following functions:

- Authentication
- Connection management
- Error control
- File sharing
- Flow control
- Full-duplex transmissions
- Name resolution
- Print sharing
- Session management

NOTE NetBIOS over IPX is called NWLink, and NetBIOS over TCP/IP is called NetBT.

Next, you learn how Windows devices can find network resources by browsing and using Windows name resolution.

Browsing and Windows Names Resolution

Network Neighborhood, Windows NT's browsing service, provides end users with a list of all devices available in their network. Before a user's PC can browse the network or Network Neighborhood, the Windows-based PC must register its name periodically by sending a broadcast to the master browser. The master browser contains a list of all devices available on the network. This service, called *browsing*, is supported by three methods—NetBEUI, NWLink, and NetBT. In addition to accessing the Network Neighborhood services, Windows devices require name resolution so that network names can be translated to protocol addresses, either IP or IPX.

Networking administrators have four options for name resolution, which are similar to the Domain Name System (DNS) provided by TCP/IP. These four name resolution options for Windows NT network administrators are as follows:

- **Broadcasts**—This method enables end stations to broadcast their names to a designated master browser (typically a Windows NT server). The master browser collects the names of available devices and maintains a list. The list is then sent to all devices that request it. This allows communication between servers and clients.
- **LMhosts file**—This simple method enables local PCs to maintain a static list of all Windows computers available in the network. The file typically contains the name and protocol addresses of all servers available in the domain. For large networks, the file might become too large and unusable, so a service called Windows Internet Naming Services (WINS) was developed (as described in the next entry).
- **Windows Internet Naming Services (WINS)**—This was developed so Windows network administrators could avoid dealing with a large amount of broadcasts or statically defined lists. WINS allows client PCs to dynamically register and request name resolution by a specific server running the WINS services. Instead of sending broadcasts, the client sends unicasts. WINS typically runs on a Windows NT server and has an IP address. Clients are statically or dynamically configured to use the server's IP address.
- **Dynamic Host Configuration Protocol (DHCP)**—In large networks (which contain thousands of PCs), a static IP address configuration can cause scalability issues because all devices in the network would require file modification. DHCP was developed to dynamically allocate IP addresses and many other parameters, such as subnet masks, gateways, and WINS server addresses. When you use DHCP, a Windows client sends out a broadcast for an IP address, and the DHCP server (a Windows NT server or compatible device) provides all the necessary TCP/IP information. The client then registers its names with the WINS server so browsing can take place. Cisco IOS routers can relay DHCP clients' requests (because Cisco IOS routers drop broadcast packets by default) with the **ip helper-address** *remote dhcp servers ip address* command.

NOTE DHCP is an IP address assignment and management solution rather than a name resolution. The DHCP server pushes the WINS/DNS/Gateway addresses to the client making it easier for the client to resolve names.

Scaling Issues in Windows NT

In larger Windows NT environments, you can have many domains. Windows NT allows information sharing between domains with the use of trusted domains. A *trusted domain* grants or denies access to clients without having to manage each user individually. Each domain can exchange information and form a trust relationship. Based on these trust relationships, end users from each domain can be allowed or denied access. Creating trust relationships allows secure data to flow between different domains and ensures adequate security for data files and application files in any Windows-based network.

Windows NT supports several domain models, including the following:

- **Single domain**—Used in small networks.
- **Global domain**—Automatically trusts every domain.
- **Master domain**—Trusted by all remote domains but does not trust the remote domains.
- **Multiple master domains**—Used in large networks where the master domain is trusted by other master domains, which in turn trust smaller domains.

Login and Permissions

NT users must log in to the domain. Pressing Control-Alt-Delete together displays the login utility.

After a valid username and password pair are entered, the verification process starts by comparing the username/password pair with the data stored in the Security Accounts Manager (SAM), which is stored on the NT server in the form of a database.

This database also contains a list of privileges for each user. For example, the database might contain the following permissions:

- User_1 is permitted access to group Cisco_Icon.
- User_2 is permitted access to group APAC.
- Directory d:\data has read and write access to both groups Cisco_Icon and APAC.
- The Word documents stored in d:\data\word are owned by group APAC only.
- The Excel documents stored in d:\data\excel are owned by group APAC, and read access is granted to all other users.

When a user or client attempts to access objects shared by other users in the domain, permissions are used to authorize or deny services.

The Windows NT file system is called New Technology File System (NTFS). NTFS is a naming file system that allows extra security. Earlier versions of Windows, such as 95, did not support NTFS and do not support file permissions.

The following are six NTFS permissions:

- **R**—Read only. The data or object can only be viewed.
- **W**—Write access. The data can be changed.
- **X**—Execute. The data can be executed. (For example, a directory can be viewed or a program can be executed.)

- **D**—Delete. The data can be deleted.
- **P**—Change Permissions. The data access permissions can be altered.
- **O**—Take Ownership. The ownership can be altered.

The NTFS permissions can also be combined for certain files and directories. For example, RX (read/execute) allows a client to view and execute the data.

NOTE Computers running DOS/Windows 3.X, 95, 98, or ME/Windows NT with FAT partition do not provide any file permissions. They can provide only share-level permission. (Remote users can be permitted or denied access.) File permissions for local users can be implemented only in an NTFS file system.

Windows NT Users and Groups

The following is an explanation of the groups:

- **Global Groups**—A global group contains only individual user accounts (no groups) from the domain in which it is created. It can be added to a local group. After created, a global group can be assigned permissions and rights, either in its own domain or in any trusting domain. Global groups are available only on Windows NT Server domains. Domain Admins and Domain Users are two built-in groups.
- **Local Groups**—Local groups are created on a Windows NT Server or Workstation computer and are available only on that computer. A local group can contain user accounts or global groups from one or more domains. They cannot contain other local groups. Backup Operator and Guests are examples of built-in local groups.

The permissions for a user of multiple groups will be additive of all permissions except for NO PERMISSION, which overrides all other permissions.

Windows NT Domain Trust

Setting up trust among multiple NT domains allows the users of one domain to use resources from another domain. The trusting domain trusts the trusted domain to manage users, groups, and resources. The trusting domain contains the resources that validated users need to access. Trust relationships aren't transitive. In other words, if the A domain trusts B, and B trusts C, A doesn't necessarily trust C. A domain's administrator must explicitly grant a trust to another domain to establish a trust relationship. Trust is one way; if A trusts B, B does not necessarily trust A.

Common Windows DOS Commands

The following are some of the most widely used DOS operating commands in Windows environments along with sample displays:

- **ipconfig**—Displays IP address and subnet mask:

```
C:\>ipconfig
Ethernet adapter Local Area Connection:

       Connection-specific DNS Suffix  . : cisco.com
       IP Address. . . . . . . . . . . . : 150.100.1.253
       Subnet Mask . . . . . . . . . . . : 255.255.255.0
       Default Gateway . . . . . . . . . : 150.100.1.240
```

- **ipconfig /all**—Displays more detailed information about TCP/IP configurations, such as DNS and domain names:

```
C:\>ipconfig /all

Windows 2000 IP Configuration

       Host Name . . . . . . . . . . . . : c03298157693425
       Primary DNS Suffix  . . . . . . . : cisco.com
       Node Type . . . . . . . . . . . . : Hybrid
       IP Routing Enabled. . . . . . . . : No
       WINS Proxy Enabled. . . . . . . . : No
       DNS Suffix Search List. . . . . . : cisco.com

Ethernet adapter Local Area Connection:

       Connection-specific DNS Suffix  . : cisco.com
       Description . . . . . . . . . . . : 3Com 10/100 Mini PCI Ethernet Adaptr
       Physical Address. . . . . . . . . : 00-00-86-48-7B-35
       DHCP Enabled. . . . . . . . . . . : No
       IP Address. . . . . . . . . . . . : 150.100.1.253
       Subnet Mask . . . . . . . . . . . : 255.255.255.0
       Default Gateway . . . . . . . . . : 150.100.1.240
       DNS Servers . . . . . . . . . . . : 64.104.200.116
                                            171.68.10.70
       Primary WINS Server . . . . . . . : 64.104.193.200
```

- **arp –a**—Displays ARP entries on the local machine:

```
C:\>arp  -a

Interface: 150.100.1.253 on Interface 0x1000003
  Internet Address      Physical Address      Type
  150.100.1.240         00-60-09-c4-34-17     dynamic
  150.100.1.254         00-b0-64-46-a8-40     dynamic
```

- **hostname**—Displays the local host name:

```
C:\>hostname
c03298157693425
```

- **nbtstat**—Displays the NetBIOS over TCP/IP statistics. A number of options are displayed:

```
C:\>nbtstat

Displays protocol statistics and current TCP/IP connections using NBT
(NetBIOS over TCP/IP).
```

```
NBTSTAT [ [-a RemoteName] [-A IP address] [-c] [-n]
        [-r] [-R] [-RR] [-s] [-S] [interval] ]

  -a   (adapter status) Lists the remote machine's name table given its name
  -A   (Adapter status) Lists the remote machine's name table given its
                        IP address.
  -c   (cache)          Lists NBT's cache of remote [machine] names and their
                        IP addresses
  -n   (names)          Lists local NetBIOS names.
  -r   (resolved)       Lists names resolved by broadcast and via WINS
  -R   (Reload)         Purges and reloads the remote cache name table
  -S   (Sessions)       Lists sessions table with the destination IP addresses
  -s   (sessions)       Lists sessions table converting destination IP
                        addresses to computer NETBIOS names.
  -RR  (ReleaseRefresh) Sends Name Release packets to WINs and then starts
                        Refresh

  RemoteName   Remote host machine name.
  IP address   Dotted decimal representation of the IP address.
  interval     Redisplays selected statistics, pausing interval seconds
               between each display. Press Ctrl+C to stop redisplaying
               statistics.
```

- **ping**—Provides a means to test and verify remote locations. An example ping to www.cisco.com follows:

```
C:\>ping www.cisco.com
Pinging www.cisco.com [198.133.219.25] with 32 bytes of data:
Reply from 198.133.219.25: bytes=32 time=182ms TTL=248
Reply from 198.133.219.25: bytes=32 time=180ms TTL=248
Reply from 198.133.219.25: bytes=32 time=180ms TTL=248
Reply from 198.133.219.25: bytes=32 time=181ms TTL=248
Ping statistics for 198.133.219.25:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 180ms, Maximum = 182ms, Average = 180ms
C:\>
```

- **tracert**—Provides a method to list next hop addresses for remote networks. The following is a sample Windows output when **tracert** routing to the URL www.smh.com.au:

```
C:\>tracert www.smh.com.au
Tracing route to smh.com.au [203.26.51.42]
over a maximum of 30 hops:
  1    <1 ms    <1 ms    <1 ms  c6k-bbn1-vlan105.cisco.com [64.105.208.2]
  2    <1 ms    <1 ms    <1 ms  c6k-bbn1-msfc-v161.cisco.com [10.66.2.2]
  3    <1 ms    <1 ms    <1 ms  sydneycisco-wall-1-f0-1.cisco.com [10.166.128.15]
  4    41 ms   236 ms    <1 ms  telstra-gw.cisco.com [103.141.98.141]
  5     1 ms     1 ms    <1 ms  FastEthernet6-1-0.chw12.Sydney.telstra.net
[149.130.85.3]
  6     1 ms     1 ms     1 ms  FastEthernet1-0-0.ken4.Sydney.telstra.net
[203.50.19.14]
```

- **route**—Provides a method to define static routing entries (Windows NT supports RIP and 2000 supports OSPF). The following example adds a static route for the network 150.100.100.0/24 via the next hop address 131.108.1.1:

```
c:\>route add 150.100.100.0 mask 255.255.255.0 131.108.1.1
```

- **nslookup**—Provides a DNS query for any host names. The following displays the use of **nslookup** for the host name www.cisco.com:

```
C:\>nslookup www.cisco.com
Server:  dns-sydney.cisco.com
Address:  64.104.200.248

Name:    www.cisco.com
Address:  198.133.219.25
```

Cisco Secure for Windows and UNIX

Cisco Systems has developed a number of scalable security software products to help protect and ensure a secured network in relation to Cisco products.

Cisco Secure Access Control Server (ACS), commonly referred to as Cisco Secure, provides additional network security when managing IP networks designed with Cisco devices.

Cisco Secure can run on Windows NT/2000 and UNIX platforms.

Three versions of Cisco Secure are listed here:

- **Cisco Secure ACS for NT**—This powerful ACS application for NT servers runs both TACACS+ and RADIUS. It can use NT username/password database or Cisco Secure ACS database.
- **Cisco Secure ACS for UNIX**—This powerful ACS application for UNIX includes support for TACACS+ and RADIUS. It supports SQL applications such as Oracle and Sybase.
- **Cisco Secure Global Roaming Server**—This performs TACACS+ and RADIUS proxy functions. It is a standalone server for large ISP networks.

NOTE Cisco also has a UNIX-based freeware TACACS+ server available for download.

NOTE Cisco Secure topics are tested in the CCIE Security lab exam (particularly Cisco Secure for Windows 2000 server). The written exam does not require you to have a detailed understanding of this application.

The main features of Cisco Secure ACS include the following:

- Supports centralization of AAA access for all users, including routers and firewalls
- Can manage Telnet access to routers and switches

- Can support an unlimited number of network access servers
- Supports many different Cisco platforms, including PIX access servers and routers

Figure 6-3 displays a typical centralized Cisco Secure ACS performing functions such as user authentication, authorization, and accounting.

Figure 6-3 *Cisco Secure Example*

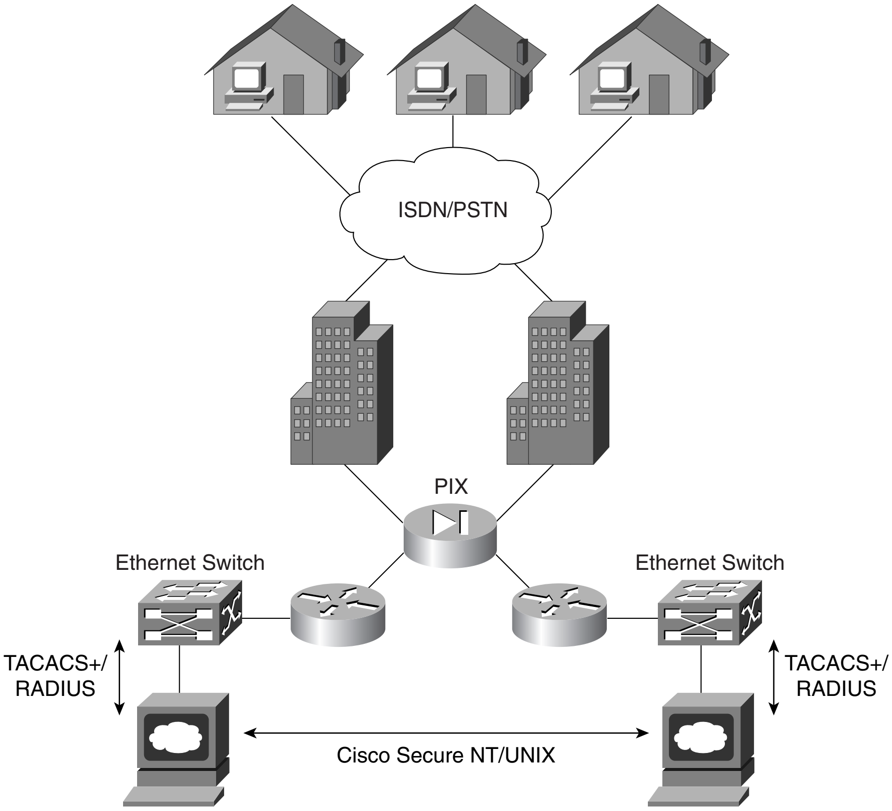


Figure 6-3 displays a typical application where ISDN/PSTN users are authenticated by RADIUS or TACACS+ via the Cisco Secure ACS server.

In addition to simultaneous support for RADIUS/TACACS+, Cisco Secure also supports the following AAA features:

- TACACS+ support for the following:
  - — Access lists
  - — Privilege level support
  - — Time restrictions where access to network is controlled during the day and night

- RADIUS support for the following:
  - Cisco RADIUS AV pairs
  - IETF support (RADIUS is a defined standard)
- Others include the following:
  - Support for virtual private networking
  - The ability to disable accounts after a set number of failed attempts

Further description of the Cisco ACS application and screenshots are shown in the sample CCIE Security lab in Chapter 9, "CCIE Security Self-Study Lab."

Cisco Secure Policy Manager

Cisco Secure Policy Manager (CSPM) provides a scalable and comprehensive security management system for Cisco Secure PIX Firewalls and Cisco Secure Integrated Systems.

Cisco Secure Policy Manager, formerly known as the Cisco Security Manager, is a policy-based security management system for Cisco security technologies and network devices.

Policy-based management allows a network administrator to define a set of high-level rules that control the deployment of and access to services, such as FTP and HTTP.

CSPM enables the management of remote Cisco Secure PIX and IOS Firewalls. CSPM allows you to configure and edit configurations remotely. CSPM only runs over Microsoft Windows operating systems.

NOTE Cisco PIX Firewalls running version 6.2 and above have a built-in, Java-based PIX Device Manager (PDM). PDM allows browser-based management and configuration of PIX Firewalls.

Cisco Secure Intrusion Detection System and Cisco Secure Scanner

This section covers network security tools that are useful for managing network security. Cisco Secure Intrusion Detection System (IDS), formerly known as NetRanger, and Cisco Secure Scanner, formerly known as NetSonar, are two security applications that allow network monitoring.

NOTE The CCIE Security written exam still refers to the terms NetRanger and NetSonar, so this guide refers to NetRanger and NetSonar as well.

NetRanger (Cisco Secure Intrusion Detection System)

NetRanger is an enterprise intrusion detection system designed to detect, report, and, in the event of unauthorized access, terminate data sessions between users and host devices.

NetRanger is an application designed to detect unauthorized access. Users are not aware that NetRanger is watching data across the network; it is transparent to all systems.

NetRanger has two components:

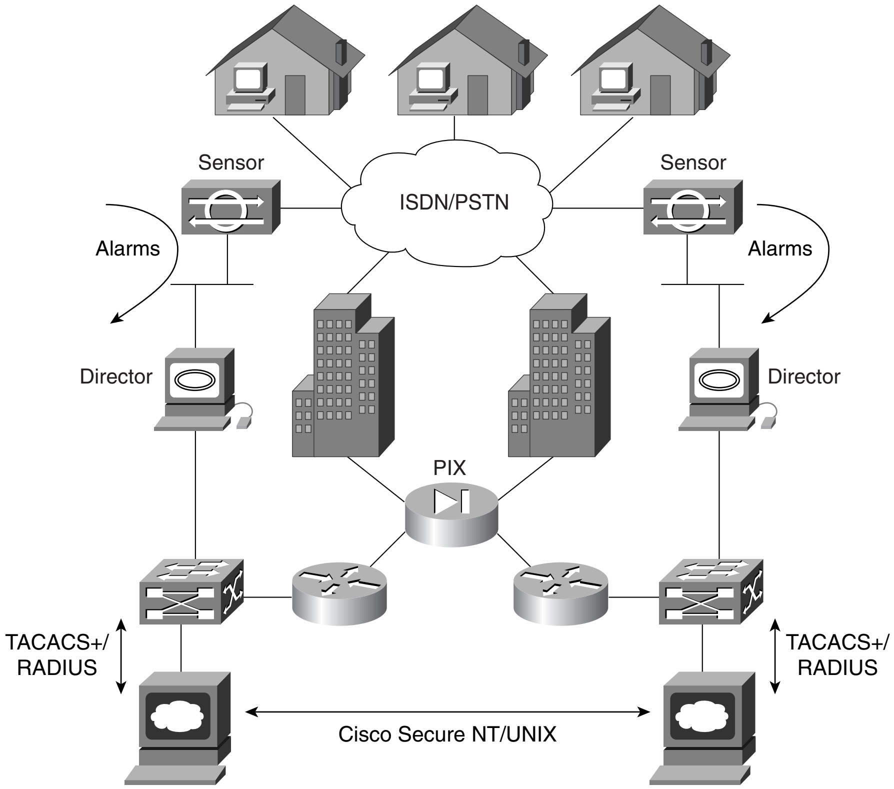
- **NetRanger Sensor**—High-speed device that analyzes the contents of data being transported across a network and determines whether that traffic is authorized or unauthorized. Unauthorized traffic includes ping requests from intruders. Traffic detected from unauthorized sources is sent directly to the NetRanger Director, and the intruder is removed from the network (optional setting to remove host).
- **NetRanger Director**—Provides real-time response to intruders in the network by blocking access to the network and terminating any active data sessions. The Director collects the real-time information from the Sensor.

Figure 6-4 displays the typical network placement of NetRanger products.

NetRanger Sensors can be located anywhere in the network. They are typically located close to hosts or entry points to a network, such as dial-in users or Internet connections. Alarms are logged on the Sensor and Director. The alarms are displayed or viewed on the Director. Optional configuration settings include killing an active TCP session or reconfiguring access lists (termed shunning).

The sensor can detect the intruder's IP address and destination ports, and buffer up to 256 characters entered by the illegal devices. NetRanger supports Ethernet (10/100), Token Ring, and FDDI LAN interfaces. NetRanger Sensors can modify predefined access lists on Cisco IOS routers and change the definitions of permitted networks in response to an attack. NetRanger Sensors cannot modify the IP routing table nor reload or shutdown interfaces. When illegal activity is discovered, an alarm is sent directly to configured directors, including multiple directors. The software used on the sensors can be loaded from a central director, allowing easier software upgrades. The GUI interface on the Director also allows network monitoring from one central location, ensuring that one central group within an organization can be directly responsible for monitoring and acting on alarms. GUI interfaces and colored alarms indicate possible vulnerabilities.

Figure 6-4 *Typical NetRanger Design*



The following platforms support NetRanger Sensor applications:

- IBM PC Pentium II or higher with the following specifications:
  - — 32 MB RAM
  - — At least 2 GB hard drive
  - — Ethernet, Token Ring, or FDDI
  - — Windows-based software
- Ultra Sparc Based UNIX station with the following specifications:
  - — 167 MHz Clone or higher
  - — 64 MB RAM
  - — 2 GB hard drive
  - — Ethernet or FDDI
  - — Solaris version 2.6 or higher software; and HP OpenView installed prior to loading NetRanger software

NetRanger Director can send out an alarm when certain configuration changes are made on Cisco routers, can send e-mail messages when particular alarm levels are reached, and can ensure a TCP attack is thwarted by sending TCP reset segments to unauthorized sources. When a NetRanger Sensor communicates with the Director, if the network is down, up to 255 alternate route paths can be attempted. Packets can be buffered and sent when the network is restored and communications occur (there are no keepalive communications; rather, one device sends and the other waits and listens) to ensure that alarms are sent.

The following platforms support NetRanger Director applications:

- HP UNIX, Ultra UNIX workstations (not PC-based)
- Software: Solaris 2.6, HP UNIX
- 128 MB RAM, CD-ROM drive, 4 GB of hard disk space
- Example machines include Sun Ultra 170 and HP 725

NOTE NetRanger examines only the IP or TCP header and not actual data. Intruders usually use an attack based on large ICMP traffic, typically fragmented, to discover the behavior of routers in a network. When a router that is set for a particular MTU size receives a fragmented packet, it sends all fragments to the destination, assuming that the end device can reassemble the packet.

Intruders typically also use context-based attacks by scanning TCP or UDP ports in use.

For more details on how Cisco IOS supports NetRanger, visit

www.cisco.com/univercd/cc/td/doc/product/iaabu/csids/csids3/index.htm

NetSonar (Cisco Secure Scanner)

NetSonar is a Cisco Systems-developed product, now named Cisco Secure Scanner. NetSonar is a software tool designed to investigate vulnerable systems within a network and report the vulnerabilities to the network administrator.

NetSonar scans the network to uncover systems that might be vulnerable to security threats by performing a number of predefined steps:

- **Network mapping**—NetSonar compiles an electronic inventory of all host devices on the network.
- **Security assessment**—NetSonar identifies potential security holes by probing and confirming vulnerabilities in the network.

- **Reports**—NetSonar communicates results to the administrator detailing the assessment, such as detailing what operating systems are in use, what the host addresses are, and the associated vulnerabilities.
- **Network security database**—This database lists the critical problems and organizes them by operating system, system services, and device types.

Figure 6-5 displays the process completed by NetSonar.

Figure 6-5 *NetSonar Phase Functions*

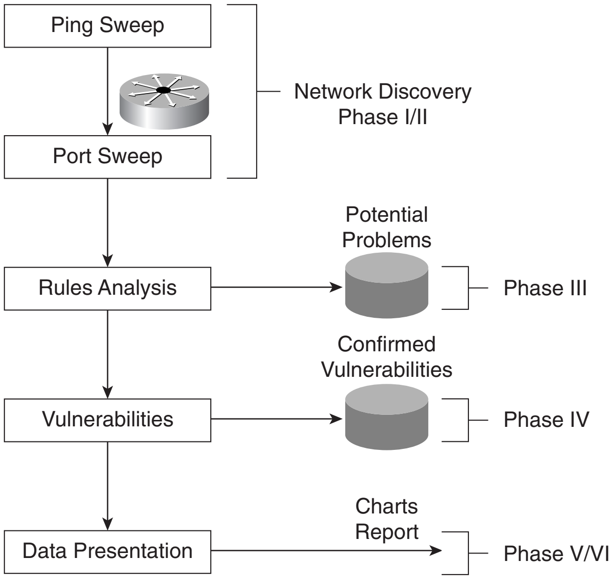


Figure 6-5 displays the six phases completed by NetSonar:

Phase I—NetSonar sends out ICMP echo requests (pings) to query hosts.
Phase II—All live hosts are collected and stored on particular port numbers.
Phase III—NetSonar identifies the hardware devices that might be vulnerable, such as routers, switches, firewalls, printers, desktops, and hosts that responded to ping requests. Operating systems and network services are documented and labeled as potential vulnerabilities.
Phase IV—Vulnerabilities are confirmed. This phase is intrusive.
Phase V—The data is charted for presentation. The data can also be charted graphically as line or 3D bar graphs.
Phase VI—The data is reported in a number of different formats, including a summary report, a short and detailed report, or a full technical report.

NetSonar software has the following hardware requirements:

- Intel Pentium I or higher
- 64 MB RAM
- 2 GB hard drive
- TCP/IP software or Sun Sparc Solaris with version 2.5 and higher

Any HTTP browser can be used to manage the NetSonar server, which can be located anywhere in the IP network.

Cisco Systems details more security products at the following URLs:

www.cisco.com/en/US/netsol/ns110/ns129/net_solution_home.html
www.cisco.com/univercd/cc/td/doc/product/vpn/ciscosec/index.htm

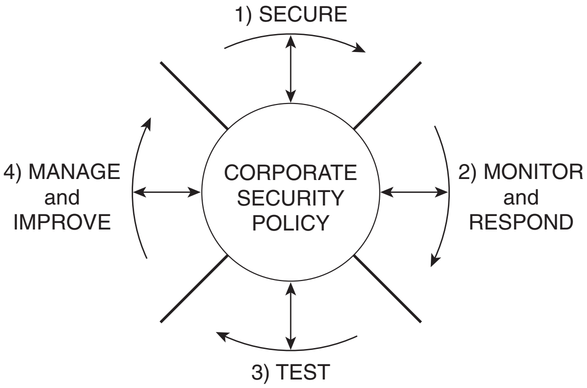
Cisco Security Wheel

Cisco defines a *Security Wheel* concept that outlines the critical steps to ensuring that data and networks are secured correctly. The Security Wheel revolves around a strong, well-defined corporate policy. The Security Wheel consists of the following:

- **Secure**—After defining a strong corporate policy, you should secure your network by deploying the products necessary in the appropriate places to achieve your corporate security goals.
- **Monitor and respond**—Continuously monitor using NetRanger tools at strategic points in the network to discover new vulnerabilities.
- **Test**—On a regular and formal basis, test all network components.
- **Manage and improve**—Analyze all the reports and metrics supplied by NetSonar and continue to cycle through the Security Wheel by going through all these steps continuously.

Figure 6-6 displays the Cisco Security Wheel graphically.

Figure 6-6 *Cisco Security Wheel*



Foundation Summary

The Foundation Summary is a condensed collection of material for a convenient review of key concepts in this chapter. If you are already comfortable with the topics in this chapter and decided to skip most of the "Foundation Topics" material, the "Foundation Summary" section can help you recall a few details. If you just read the "Foundation Topics" section, this review should help further solidify some key facts. If you are doing your final preparation before the exam, the "Foundation Summary" section offers a convenient way to do a quick final review.

Table 6-4 summarizes important UNIX commands.

Table 6-4 *UNIX Commands*

Command	Description
cp -i/-r *oldfile newfile*	Makes a copy of a file. You must specify the name of the file to be copied and the name of the new file to be created. The **-i** flag tells the computer to ask before it overwrites any files in this process. The **-r** flag copies any files in subdirectories if you are copying directories.
rm -i/-r *filename*	Erases the specified file. The **-i** flag asks you for confirmation before a file is deleted. The **-r** flag erases directories or subdirectories and all the files they contain.
rmdir -p *directoryname*	Erases directories. The **-p** flag allows you to erase a directory and all its contents. Without this flag, the directory must be empty before you can erase it.
mv -i *filename1 filename2*	Renames a file. The **-i** flag asks for confirmation before overwriting a file if you attempt to use a filename that is already taken. Without the flag, the original file with the same name will be automatically erased.
mv -i *filename directoryname/filename*	Moves a file to another directory. The flag serves the same purpose as in the other **mv** command.
man *command*	Displays a description and usage instructions for a specified command. This command is similar to **help** in a Windows environment.

continues

Table 6-4 *UNIX Commands (Continued)*

Command	Description
grep -i	Allows you to search for a string in files. The flag **–i** tells the UNIX server to ignore upper- or lowercase.
netstat -s	Displays a description and usage instructions for a specified command. The **netstat -s** displays statistics for network interfaces and protocols, such as TCP.
ifconfig -a	Displays the current interfaces that are configured (displays the IP address and subnet mask).

Table 6-5 summarizes the main Windows DOS commands.

Table 6-5 *DOS Commands*

Command	Meaning
ping	Provides a means to test and verify remote locations.
nslookup	Provides a DNS query for any host names.
route	Provides a method to define static routing entries (Windows NT supports RIP and 2000 supports OSPF).
tracert	Provides a method to list next hop addresses for remote networks.

Table 6-6 Summarizes NetRanger's two components.

Table 6-6 *NetRanger Components*

Component	Meaning
NetRanger Sensor	High-speed device that analyzes the contents of data being transported across a network and determines whether that traffic is authorized or unauthorized. Unauthorized traffic includes ping requests from intruders.
NetRanger Director	Provides real-time response to intruders in the network by blocking access to the network and terminating any active data sessions. The director collects the real-time information from the sensor.

Table 6-7 defines the NetSonar Phase functions.

Table 6-7 *NetSonar Phase Functions*

Phase Number	Function
I	Sends ICMP echo requests (ping) to query hosts.
II	Collects and stores all live hosts on particular port numbers.
III	Identifies the hardware devices that might be vulnerable, such as routers, switches, firewalls, printers, desktops, and hosts that responded to ping requests.

Table 6-7 *NetSonar Phase Functions (Continued)*

Phase Number	Function
IV	Confirms vulnerabilities. This phase is intrusive.
V	Charts data for presentation. The data can also be charted graphically as line or three-dimensional bar graphs.
VI	Reports data in a number of different formats, including a summary report, a short and detailed report, or a full technical report.

Table 6-8 displays the Cisco Security Wheel model and functions.

Table 6-8 *Cisco Security Wheel*

Cisco Security Wheel	Meaning
Secure	After defining a strong corporate policy, you should secure your network by deploying the products necessary in the appropriate places to achieve your corporate security goals.
Monitor and respond	Continuously monitor using NetRanger tools at strategic points in the network to discover new vulnerabilities.
Test	On a regular and formal basis, test all network components.
Manage and improve	Analyze all the reports and metrics supplied by NetSonar and cycle through the Security Wheel by going through all these steps continuously.

Q & A

The Q & A questions are designed to help you assess your readiness for the topics covered on the CCIE Security written exam and those topics presented in this chapter. This format is intended to help you assess your retention of the material. A strong understanding of the answers to these questions can help you on the CCIE Security written exam. You can also look over the questions at the beginning of the chapter again for additional review. As an additional study aid, use the CD-ROM provided with this book to take simulated exams, which draw from a database of over 300 multiple-choice questions—all different from those presented in the book.

Select the best answer. Answers to these questions can be found in Appendix A, "Answers to Quiz Questions."

1 What UNIX command displays the files in the current directory?

____

____

____

2 What UNIX command changes a directory from etc/ to bin/?

____

____

____

3 What does the following UNIX command accomplish?

```
cp -i simon.doc henry.doc
```

____

____

____

4 To define a permission for a UNIX file, what command line interface is required?

____

____

____

5 The **chmod** UNIX command can define what levels of access or permissions on a UNIX host?

6 In a Windows NT environment, what is a domain, primary domain controller, and backup domain controller?

7 What functions does the protocol NetBIOS provide in a Window NT environment?

8 What is the function of the lmhosts file on a Windows platform device?

9 Name and define the six NTFS permission types.

10 In Windows NT 4.0, what DOS command displays any local ARP entries?

11 Define the terms NetRanger Sensor and Director and their uses?

12 What LAN interfaces can be supported on a NetRanger Sensor?

13 What are the six phases completed by Cisco NetSonar?

14 What is the meaning of the term Security Wheel?

Scenarios

Scenario 6-1: NT File Permissions

A group of users in a Windows NT environment are members of the domain CISCO_CCIE. You are supplied the following details regarding file permissions:

- PC1 and PC2 are authenticated in domain CISCO.
- The CISCO domain is trusted by the CISCO_CCIE domain.
- The directory d:\data has a file named ccielab35.doc and has access for users in the CISCO domain set to read only access.
- A user named hbenjamin in the CISCO domain owns the Word document ccielab3.doc.

With these details, can PC1 open and read the file named ccielab35.doc?

Scenario 6-2: UNIX File Permissions

A newly created program file is on a UNIX server in the etc/bin named simon.exe directory. The root user creates the file simon.exe after compiling some UNIX C-based code. The root user password is set to guitar. How can you allow all users who are authenticated and authorized to view the etc/bin directory access to the file named simon.exe?

Scenario Answers

Scenario 6-1 Solution

The CISCO domain is part of the large domain CISCO_CCIE. Because the directory d:\data is set to read only, users from the CISCO domain are permitted to open the document in read-only mode. User hbenjamin is permitted to open and write to the document because Windows NT sets the privilege for the owner as read/write by default.

Scenario 6-2 Solution

If the users know the root password, they can enter the root mode by typing **root** and then the password **guitar**. This allows the user access. If the root password is not known, the file permissions can be modified with the command **chmod 777 simon.exe**, and because users can already view the directory etc/bin, access to the file named simon.exe is now permitted.