Chapter 1

Installing Windows Server 2008

This chapter reviews the installation of Windows Server 2008. It discusses a number of hardware configurations and setup options and reviews potential obstacles. Several recipes are discussed in this chapter, and most of them use minimum hardware requirements; keep that in mind when ordering your server. We also explain how to achieve a fresh install or upgrade with different server configurations, installation of the Server Core image, and installation of the base OS with Windows Server 2008 GUI. We will have a look at Server Manager and the variety of server roles and features it allows you to install. Several other topics, including SQL Server, ASP, IIS, and Exchange, are also covered to help you understand how they are incorporated into Windows Server 2008.

IN THIS CHAPTER

It's All About the Core

Before we begin, let's review Microsoft's so-called Core Server installation paradigm, a new type of barebones OS that can also be headless, keyless, and mouseless . . . and Windowless. During the years of Windows NT, Windows 2000, and Windows Server 2003 (pre-R2) installing the operating system was a nail-biting event. We would always stand and gawk at the screen and hold our breath as certain stages in the installation were completed. Once we got through to the restart procedure it was high-fives all round.

By Windows 2000, installing on various hardware platforms was a lot easier. Gone, for the most part, were blue screens during installation or mysterious restarts that had everyone scratching their heads. But another problem arose. The Internet exploded in popularity and along with it the scourge of viruses and hostile cyberspace junk.

The problem was compounded by the need to connect a server to the Internet and patch it with all manner of fixes and updates that Microsoft issued after release to manufacturing (RTM). The result was that unless you sealed the server on a secure network and applied all the patches from a local update server, the chances that the server would be infected and compromised were almost 100 percent.

One of the problems of updating a new server is that the process would usually include patches, fixes, and configuration for just about every service running or not running on a server. It was a tedious and time-consuming task. But that has now been changed with Windows Server 2008.

You no longer need to install the OS and worry about all the services and functionality on it that you will not be using. Now you can install Server Core (a special build of the OS) or base OS with GUI, and incrementally install and apply only the services and bits needed for specific server functions. Server Core is like the birth of a baby, all naked and uncorrupted but not exposed, while the base OS installation with Windows user interface is like a 3-year-old, ready for "intelligence" to be added but with some exposure required.

What Is Server Core?

The Server Core installation lets you install a minimal OS for running just the chosen server roles that would not even need a GUI. This means that you don't have the huge "attack" surface that will ensue from all the service requirements. One more thing: Once you install just Server Core you can stand your server up in a secure environment, both physical and online, and worry only about securing the services you are actually running. Once Server Core has been installed you can then open Server Manager (remotely or via scripting) and install, among many others, the following server roles:

- Active Directory Domain Services (AD DS)
- Application Server
- DHCP Server
- DNS Server
- File Services
- Print Services

Here are some more benefits of the Server Core installation alternative:

- **Lower maintenance.** You only need to maintain on the server what is actually installed on the server. Why worry about maintaining File Services on a server that is nothing more than a simple domain controller?
- **You need less disk space.** The Server Core requires only about 1 gigabyte (GB) of disk space to install and approximately 2GB for operations after the installation.
- **Less management.** Management costs in realms like security, availability, and service level are far less than previous installation scenarios. You would not have to worry about supporting a bunch of services and code that you are not using.

More details of the server roles are presented throughout this chapter and in Chapter 2. We will return to actual installation of the Server Core OS later in this chapter.

Installation and Configuration Strategy

If you have done your homework and have an installation plan and architecture document ready, you are now ready to begin installing Windows Server 2008 in your lab. For help creating an installation plan and an architecture document (including an architecture template) visit `www.misiq.com/whitepapers`. You may be tempted (or you may have an urgent need) to go directly to a working or production system in a production environment. Perhaps your DHCP server died or a new DNS server is needed urgently. Resist — or stick with what you know. If you have a Windows Server 2003 network and need to raise a new service to fill an urgent need, stick with Windows Server 2003 until you have fully implemented a test lab and are familiar with the way things work under Windows Server 2008. In many respects it is a very different operating system. Microsoft Server 2008 has the same core kernel and presentation functionality as Windows Vista, but with server bits added in. So in many respects it is new code from the ground up compared to Windows Server 2003.

Conversely, if you are a seasoned administrator and you know what you're doing, you probably have issues such as a hardware checklist, remote or unattended installation, hot standby, and so on well taken care of. Proceed directly to a production system only if you know what you are doing and the production system is part of a conversion and rollout project.

Generally, you should always raise servers in a lab. Then you should burn them in (run them continually) for about a week; hit them with work for at least another week. After that, and if all test items check off, ship or go live. No two environments are the same. The following sections look at various installation and configuration scenarios.

NOTE **A lot of people want to know how to burn in a server that is standing idle and has no users connected to it. One simple way is to set up Windows Server Backup to run continually. Running Backup is great physical therapy for a server. It works the hard disks, memory, system buses, access control, permissions and the NTFS, removable storage functions, transactional file system, and more. You can also configure Backup (or any other backup utility, for that matter) to perform both pre- and post-backup routines, such as sending alerts and moving files around the house. Depending on your stress test, you may need to write a backup script to automatically overwrite media and so on. In addition, if you want to test disk I/O and other routines, you may need to write some custom software for the job.**

Getting psyched up about installing

This chapter takes you through the basic install routines and then to rollout a sophisticated deployment strategy. We are going to help you cook up a variety of server meals. Microsoft has spent many millions on the installation and configuration process, so Windows Server 2008 generally installs easily and performs well considering the power that it wields. It is certainly a

lot smoother and friendlier to install than any other server operating system in existence (other than the machine you receive pre-installed from the factory).

We have installed the operating system numerous times and on at least ten different platforms with a variety of hardware, from scrap piles to brand names. We have also deliberately sabotaged our systems (such as taking away drives, drivers, memory, and certain system files) and tried a variety of recovery techniques. Our final verdict? If you experience any difficulty installing Windows Server 2008, you must be using very unconventional methods, thrift store hardware, or you're not paying attention to details and recommended strategy.

Take a moment to sit back, close your eyes, and imagine that you are in a class going through installation training.

Server recipes

In evaluating the various needs in the enterprise, we classify our installation options into various recipes of server installation, which are discussed in the following sections.

Server Core or bare-bones system recipe

This option consists of using minimum hardware requirements as recommended by Microsoft and some testing. All servers require at least 1GHz for x86-based computers except Datacenter Server, which requires 1.4GHz for x86-based computers. These are bare minimums for production servers, but you could get away with less in lab or testing environments. For production servers you'll likely deploy Windows Server 2008 in the 2GHz and higher range, especially for x64. We suggest a bare-bones minimum of 512MB of RAM for all servers except Datacenter Edition, which requires a bare-bones minimum of 1GB of RAM. You also want a DVD-ROM (the OS no longer fits on CDs), a 1.4MB floppy disk drive, a standard network card, and a mouse, keyboard, and monitor.

We have raised servers (Standard Server, Enterprise Server, and Web Server Edition) on CPUs ranging from old Pentium 866s, 1.2s, 1.4s, and so on. You can raise the mentioned servers on less, but we don't recommend it for anything more than the smallest test server, described in the section "Overview of Hardware," later in this chapter. On the other hand, an old horse with a lot of RAM might serve many of your needs. You can usually pick these servers up on the Internet for a song; and if they are good brands, they do well for many years, especially for servers only running Server Core.

Small file and print server recipe

The IT department needs the capability to efficiently utilize file and print resources and keep them available and secure for users. Networks tend to expand, with greater numbers of users located onsite, in remote locations, or even in partner companies, and IT administrators face an increasingly heavier burden. Windows Server 2008 now provides many enhancements to the file and print infrastructure to help solve the never-ending administrators' burden.

You should still use the bare-bones components but add a second large IDE hard-disk drive for file and print services, the usual peripherals, and so on. The amount of RAM that you need depends on the number of connections and users. Printing services require a lot more RAM than file services.

Your hard-disk demands are higher, and you should now consider adding a second drive. You can stick to a cheap IDE disk (even the cheap IDE or EIDE drives are good disks) or begin thinking about SATA and SCSI. Hold the thought about hard disks for the section "Overview of Hardware," later in this chapter.

NOTE **You may have read information elsewhere calling for more firepower in Windows Server 2008. My assessment is based on various experiments, projects, pilot systems, and deployments. Every situation is different, and the only way to really know what you need to throw at a situation is to test.**

Application-server installation recipe

The Windows Server 2008 application environment builds on the solid enterprise capabilities of Windows Server 2003 Server security, availability, reliability, scalability, and manageability. The application development seems to be more dependable because the environment can be managed by fewer people, and it delivers lower TCO with better performance. Developers are one of the most highly leveraged resources in IT. By integrating .NET Framework into the Windows Server 2008 application-server development environment, developers are now freed from writing "plumbing" code and can instead focus their efforts on delivering business solutions.

You may want to install applications on servers for users who load them into local memory at their workstations. The application is thus loaded across the network, but the "footprint" and ensuing resource consumption is local to the user's hardware.

You may also have applications that are server-based or server-oriented. These may include database front ends, communications software, processing-oriented software, and network-management applications. Hundreds of applications may be suited to server-side execution and need no user interaction, such as process-control applications and data processing.

You could use the recipe for file and print servers that we give in the preceding section; raising the ideal configuration for your purpose takes some testing. Depending on the availability requirements, you may need to add RAID, hot-swap drive-bays, and so on, which are discussed in the section "Partitioning Hard-Disk Drives," later in this chapter.

Terminal Services installation recipe

A Terminal Services application server is a whole new ball game. The WinFrame licensing arrangement between Citrix Systems, Inc., and Microsoft was the origin of Terminal Services. Terminal Server, under the Hydra project name, first made its debut in Windows NT 4.0 in late 1997. It was then launched as a separate NT 4.0 operating system called Windows NT 4.0 Terminal Server Edition (TSE). It was even further enhanced in Windows Server 2003. Terminal Server in Windows Server 2008 is still called Terminal Services but it is now installed

as role. However, you can still connect to a server using Remote Desktop Connection (RDC) in administrative mode. RDC provides substantial improvements over previous releases. RDC provides administrators and users with a simplified user interface that still connects to previous versions of Terminal Services (Windows NT 4–Terminal Server Edition and Windows 2000). See Chapter 2 for more information about the Terminal Services and RDC.

Windows Server 2008 Terminal Server supports more users on each high-end server than previous editions. Windows Server 2008, Enterprise Edition, provides a superior load-balancing support than previous editions. Session Directory maintains a list of indexed sessions by user-name, enabling users to reconnect to the Terminal Server and resume working in that session. It also provides unsurpassed remote manageability by taking advantage of technologies such as Group Policy and Windows Management Instrumentation (WMI), which provides management with complete remote capabilities through a comprehensive read/write system.

With Windows Server 2008 acting as a Terminal Services application server, all your users run all their applications on the server. No such thing as a local Terminal Services client even exists. The client can be a browser, a fat client running a Terminal Services terminal application (such as a TN3270 character-based terminal running on Windows and accessing a DB2 database on the mainframe), a dumb terminal (known as a Windows-based terminal), or terminals running on the Windows CE or Pocket PC platforms. Your users' terminals can also be installed on any non-Windows platform, such as Macintosh, DOS, and Unix, but these require extras from Citrix, which uses the ICA protocol.

Terminal Servers can be raised with any of the recipes discussed so far. What matters is not what you start up with but what the terminal users do after they are attached to the server. We have tested these services and deployed them in vigorous real-life situations with every version of the OS, and the following configuration pointers, which apply to a different configuration recipe that we discuss shortly, are key:

- Restrict your users from having more than four applications open at one time. Make sure, for example, that they can comfortably open and run a database application, a word-processing application, e-mail, and a Web browser.
- Configure the applications to run without fancy splash screens, animations, or any resource-intensive software.
- Assign and enforce hard-disk quotas. This is important to do for all users but is especially useful if you are dealing with terminal users.

A server hosting no more than RDC users should be running on a CPU of no less than 1.4GHz. Each user (depending on the applications and the type of processing) should be assigned no less than 32MB of RAM. 128MB and higher should be your goal to cope with the high demand of memory from many of today's memory hungry applications, especially applications from suites like Office 2007. You should also install fast SATA or SCSI drives and support them in hardware RAID configurations on fast controller cards. In short, no bare-bones situation is possible for Terminal Services and application hosting. After all, if you were deploying to standard clients, they would likely each have more than 1.6GHz with 1 or 2 GB of RAM.

At 128MB each, the recipe thus calls for the following total server RAM:

- Operating system = 1GB
- Five users at 128MB each = 640MB
- Total RAM needed = 2GB

You're likely to have a hard time adding a small amount of RAM into a modern motherboard. Your configuration would thus be at least 2GB for a modern application server hosting one to five RDC users.

Line-of-business role-server installation recipe

Role servers are servers running services such as DHCP, WINS, DNS, and Active Directory. Your application and needs may vary widely, depending on the service and how many subscribers it has. A small company may get away with a lightweight configuration, such as the small file- and print-server recipe offered in the section of that name, earlier in this chapter. In other cases, you may require much more firepower, especially on medium to large intranets. You can easily run DHCP, WINS, and DNS on Windows Server 2008 on 1GHz machines with 1GB of RAM in each, servicing several thousand users across a nationwide WAN, but you have a lot more replication and dynamic configuration overhead with Windows Server 2008, so you may need to shell out for more powerful machines.

High-road, or mission-critical recipe

Mission-critical servers should have no less than 1.6GHz in CPU capability. For the most part, and especially if you have more than a handful of users, your CPU should be more than 1GHz. You may consider equipment running two-CPU configurations or possibly deploy quad systems.

Hard-disk needs may vary, but you need to configure a second drive letter running at RAID-5 under hardware control. (In case you're wondering, these are SCSI devices, which we discuss in the section "Partitioning Hard-Disk Drives," later in this chapter.)

Redundant or standby system recipe

Any of the server recipes mentioned in the preceding sections can be cloned to provide an offline or hot spare. These are obviously not clustered or automatic failover machines. If the primary server goes down, you could pull dynamic volumes out of the primary arrays and install them into the hot spares. A better solution, if you can afford it, is to install Enterprise Server and run cluster services and network load balancing.

Large systems, clusters, and Datacenter Server installations

Advanced clustering (high availability) and Datacenter Server solutions are beyond the scope of this book, although most of the configuration information in this book applies to the high-end operating systems. Any large system calls for an external SCSI-based storage silo under hardware RAID-5.

The various recipes that we've discussed so far are summarized in Table 1-1.

TABLE 1-1

Hardware Guide for Server Recipes

Recipe	CPU/GHz	RAM/GB	HDD
Bare-bones	1.6	1-2	SATA, eSATA or SCSI
Small File and Print	1.6	1-2	SATA, eSATA or SCSI
App server	2	1-2	SATA, eSATA/SCSI
Terminal Services	2	2+	SCSI-RAID
Role server	2	1+	SCSI-RAID
LOB	2	1+	SCSI-RAID
Standby	1	1+	SATA, eSATA/SCSI
Large	2X2 or 2X4	4+	SCSI, eSATA – RAID

Overview of Hardware

Choosing hardware is not a difficult exercise at all for Windows Server 2008. You really don't put a lot into your system. The list of hardware that we discuss in the following sections is as follows:

- Motherboards
- CPU
- Memory
- Hard-disk drives
- HDD controllers
- Network interface cards (NICs)

Hardware compatibility

Before you go buy parts, review the Windows Server Catalog for hardware compatibility at `http://www.windowsservercatalog.com/`. The "Designed for ... " or "Ready for ... " Windows logo identifies software and hardware products that have been designed for and work well with Microsoft products. Software and hardware products displaying the logo must pass rigorous testing to ensure that they provide ease of use and stability and that they take advantage of the new features in Windows products. Software is tested by an independent testing lab. All PCs and peripheral hardware must be tested by Microsoft approved labs.

Businesses that use products meeting Windows logo criteria stand to gain the following benefits:

- Lower support costs
- Support for mixed Windows environments
- Correct use of the operating system
- Compliance with the Americans with Disabilities Act and other equal-rights legislation

According to Microsoft policy, Microsoft does not support you if the item is not on the so-called "HCL," or hardware compatibility list, but not many items may be on the HCL yet. If you offer to spend $195 with Microsoft to figure out whether hardware is the reason a server does not start, do they refuse to take your money? They never have to date. Microsoft's paid support team is very responsive and helps you determine whether hardware is your problem. If they tell you that you have a hardware-compatibility problem, that's probably all the advice that you need.

The compatibility issues aside, you should heed the following advice: Most large companies buy brands from the likes of IBM, Dell, HP, and so on; and if the budget is there, a small company looking for one business server should go this route as well. The servers are burned in and tested, and the manufacturer stands behind the compliance of its product running Windows Server 2008, logo-compliant or not. The servers also come with warranties and various levels of support.

If, however, you plan to build your own server, or if you need to upgrade a machine down the road, by all means, buy your own parts and knock together your own server. For best motherboard results, however, try to stick to made-in-America components or well-known and popular foreign imports. For RAM, only a handful of factories are left, but you're okay buying products from the likes of NEC, HP, IBM, TI, and others. For hard disks, IBM, Quantum, Western Digital, Maxtor, and Seagate are the leaders now, and really the only players. For CPUs, you have Intel and AMD. If you are thinking other marginal CPUs, you need to talk to the likes of IBM or Motorola. The other peripherals do not interfere with your server.

Installing Windows Server 2008

We have found, after dozens of installations, that the best practice for installing Windows Server 2008 is to follow this specific checklist of events:

- Check system requirements — visit Microsoft's site and review the System Requirements for Windows Server 2008.
- Read the setup instructions and release notes included with the Windows Server 2008 Installation DVDs.
- Determine whether to upgrade or install.

- Determine what licensing schema to use: per server or per seat.
- Determine whether you want the capability to choose between different operating systems each time that you start the computer.
- Determine whether you need an NTFS or FAT32 file system.
- Determine whether a special partition is necessary for this installation.
- Choose the correct components to install. Determine the server's purpose.
- Determine how to handle networking, IP, TCP/IP, and name resolution.
- Determine whether you want workgroups or domains.
- Disconnect any UPS devices. The setup process tries to detect devices connected to serial ports or USB ports; therefore, UPS equipment can cause problems with the detection process.

Start setup after you have considered each of the events in the following sections and prepared a checklist that is right for your installation.

Partitioning hard-disk drives

Give Windows Server 2008 a hand, and it takes an arm . . . or at least another drive. Installation assesses all the hard-drive resources in the system, and if you have two drives (or partitions), the OS attempts to use both. The first active partition gets snagged for the system files . . . the minimum required to raise the system to a point where you can run recovery tools or the Recovery Console. Windows Server 2008 calls this volume — you guessed it — the system volume.

Windows Server 2008 then snags a second drive or partition and uses it for the boot files, the files needed to boot the rest of the operating system all the way to the desktop on which you can log in. Windows Server 2008 calls this volume the boot volume. (This is a reversal of the old naming convention for boot and system partitions.)

Two reasons exist for the dual-disk consumption. First, Windows Server 2008 is optimized to use more than one hard-disk drive. Second, a minimum boot disk can be configured to hold just the boot files and can be formatted as FAT or FAT32 instead of NTFS. The theory is that if you lose the base operating system — that is, if you cannot boot to the desktop — you can at least boot to a DOS diskette and then, from DOS, copy new base files over the corrupt ones (or replace a defective drive). Many NT and NetWare systems have been configured this way. However, a well-designed and managed system need not retain a FAT boot disk, which, because of its poor security, is a risk to the entire system because it does not support file-level security.

Windows Server 2008, however, enables you to boot to the Boot Options console (whenever it detects a disaster). Here you have several options, such as Safe Mode with Networking, and from there you can attempt to boot without certain services and debug the problem after you have the OS up and running. You can also boot the Recovery Mode Console, which takes you to a command line that you can use to access NTFS partitions and the boot disks. The practice

of leaving boot or system files on FAT volumes is old-fashioned — the result of bad memories from Windows NT days. We recommend the partition arrangement options described in the following sections.

Option 1: One HDD

This arrangement uses one hard-disk drive, which forces Windows Server 2008 to put both boot files and system files onto the same drive and partition. To use this option, follow these steps:

1. Configure the system with one hard-disk drive of about 12GB in size. (Microsoft's official recommendation is to supply at least a 10GB partition, but with roles and features to be added, as well as patches and fixes and new features coming down the road, you need to leave room for expansion.)
2. Format the partition during the install as NTFS.
3. Have Windows Server 2008 choose the default partition name.

The pros of this partitioning option are as follows: First, you save on hard-disk drives. Second, you can mirror this disk for fault tolerance. (Unfortunately, you can mirror the disk only under hardware disk mirroring because Windows Server 2008 does not enable you to mirror a disk that was installed as a basic partition . . . even if you make the disk a dynamic disk.)

The negatives of this partitioning option are that, if you must format the system or boot volumes as FAT, you end up with a disk consisting of numerous partitions. This is not necessary on a server and can later lead to problems, such as no capability to mirror or diminishing hard-disk space and the advanced features of dynamic disks. You may also have trouble providing dual-boot capability, but dual boot is not recommended, and besides, you have no need to provide dual boot on a production server.

Option 2: Two HDDs

This arrangement uses two hard-disk drives: Windows Server 2008 puts boot files on one disk and system files on the second disk. To use this option, follow these steps:

1. Configure the system with two hard-disk drives of about 2GB each in size.
2. Format the drives as NTFS during the install.
3. Have Windows Server 2008 choose the partition names and the default and put the files where it needs to.

The positive aspect of this partitioning option, as far as we can tell, is that you have the option of leaving the boot volume formatted as FAT (or FAT32) and formatting the rest of the partitions and drives as NTFS.

The negatives of this partitioning option are that you use up a second drive for a small amount of hard-disk space, but if you are bent on dual or multi-boots, the second drive can hold the additional OS.

Although you have a performance incentive to use a second hard disk, the increased performance is not worth the effort and the second drive, considering the speed and response of modern hard disks. We are also talking about Server Core here and not Active Directory, LOB servers, SQL Server, or Exchange, which are built to take advantage of additional drives. You would be better off using a second drive as a mirror of the first to gain a fault-tolerance feature.

Performing a Server Core install

To create a server running on Server Core installation you need to have the following handy:

- The Windows Server 2008 installation media
- The product key
- A computer with the recommended configuration for a Server Core installation

Before you begin, make sure you have clean or newly formatted hard disks or volume that you can allow installation to format for you. You cannot upgrade from a previous version of Windows Server to a Server Core installation. You also cannot upgrade from a full installation of Windows Server 2008 to a Server Core installation. Only a clean installation is supported.

Be sure of your needs and configuration before you start. Once you start a Server Core installation you cannot go back later and try upgrading it to a full installation of Windows Server 2008 with the Windows UI. Microsoft does not support that route and you would have to blow away the Server Core installation and start all over again.

To install a Server Core installation, perform the following:

1. Insert the Server Core Windows Server 2008 installation media into the DVD drive.
2. The auto-run dialog box will now appear. Click the Install Now option.
3. The installation wizard takes you through the instructions to complete Setup.
4. After the installation, press Ctrl+Alt+Delete and click Other User. At the login enter Administrator with a blank password, and then press Enter. You will now be able to log in and you will have the chance to set a password for the Administrator account.

Performing an unattended Server Core install

As with previous versions of the OS, you use an "unattend" file for a Server Core installation or a regular Windows Server 2008 image. The unattended server install enables you to perform most of the initial configuration tasks during Setup. The following section describes an unattended installation of the Server Core image. If you have a number of servers to install, the unattended installation of Server Core can provide a host of benefits.

There is no need to perform initial configuration using command-line tools because you can include options in the unattend file that will enable remote administration. Once Setup completes you will be able to connect with various tools and applications and continue to fine-tune and configure.

To install a Server Core installation by using an unattend file, do the following:

1. First create an `.xml` file titled `unattend.xml`. You can use any text editor or the Windows System Image Manager.
2. Next copy the `unattend.xml` file to a local drive or place it on a shared network resource.
3. Place the Windows Preinstallation Environment (Windows PE), Windows Server 2003, or Windows XP media in the machine's CD drive and start your computer.
4. Next place the CD of the Server Core installation image of Windows Server 2008 into your disk drive. As soon as the auto-run Setup window appears, click Cancel. This will bring you to the command prompt.
5. Next, change to the drive that contains the installation media, enter the following command, and press Enter:

   ```
   setup /unattend:<path>\unattend.xml
   ```

The `<path>` is the path to your `unattend.xml` file described in step 2. Setup will run to completion with whatever you have in the `unattend.xml` file.

Performing a basic install

The CD install consists of several stages, prompting you for information, copying files, and restarting. Setup concludes with the Installation Server Wizard, which guides you through the server configuration.

Initial setup: Using the DVD

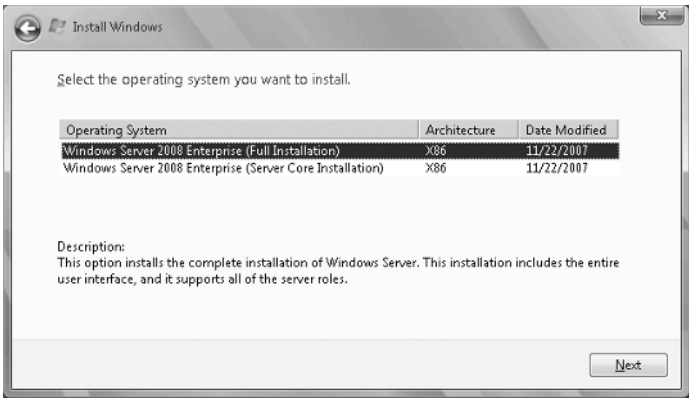
To use the DVD for initial setup, follow these steps:

1. Insert the media into your DVD drive and then reboot your machine. Alternatively, you can run Setup from the DOS command line by typing `D:\setup.exe`. Executing Setup from the command line or from reboot loads a minimal footprint of Windows Server 2008 into memory. The code in memory contains the functions that start the setup program. The machine is rebooted, and the text-based version of Setup starts.
2. Next you can check online if any new bits need to be downloaded for the installation or you can continue with the bits you have on your DVD. Click Next to continue.
3. You are now required to add in your product key. Click Next and you will see the option to install Server Core or the regular Server 2008 image (this is shown in Figure 1-1). Click Next.
4. The next screen gives you the license terms and you have to agree to them to continue. Check the agreement checkbox and click Next.

   If you are installing on a server that already has a Windows Server installed you will be prompted to upgrade or install a fresh image. You should install a fresh copy of Windows Server 2008 on a newly formatted partition.

FIGURE 1-1

Choose standard installation image or Server Core.



5. Setup next asks you to choose the partition on which to install the operating system. You can select existing partitions or choose to create a new partition. (You also have the option to load disk drivers at this point.) After choosing the partition, click Next. Once you have confirmed the file system you are on your way to installation.
6. Setup immediately begins installing files to the partition. If you have more than one disk in the system and want to install to one disk, do not format or partition any other media at this time.
7. Setup then saves the initial configuration and restarts the machine.

On boot up, the first screen that you see is the Windows Server 2008 Setup Wizard. The Windows Server 2008 operating system files are installed in the `C:\Windows` folder.

Running the Setup Wizard: Information to have handy

When you install Windows Server 2008 you will be asked for information about yourself, the organization or company licensing the software, and the computer.

Windows Server 2008 takes this information and begins installation in which it copies software to support machine configuration, installed devices, and so on. After this phase, Windows Server 2008 prompts you for the following information:

- **Language options.** You are asked to customize language, locale, and keyboard settings. If you are installing in the United States, you can, for the most part, leave these at the default settings. You can also configure the server to use multiple languages and regional settings. Choosing multiple languages forces Windows to install the character sets from multiple languages.

- **Name and organization.** Provide the name of the person responsible for the software and the name of the organization that owns the license.
- **Licensing mode.** You can choose to select licensing on a per-server, per-seat, or per-device basis. If you choose to license per seat, you must enter the number of client access licenses (CALs) purchased. If you are going to provide application services by using the Terminal Services in Application mode, choose the CAL option.
- **Computer name.** This is where you get to add the NetBIOS name. Windows Server 2008 chooses a default name for you, which you should change because it doesn't make very much sense. Coming up with a convenient naming convention that your users recognize is far better.

Windows pretty much leaves you to your own devices in naming your computers. The best rule to follow is to name the machine according to any convention you dream up that works for your situation . . . just be consistent. Resist cute names for several reasons: The names may be hard for your users to relate to, and some may find them annoying. (Not everyone loves Disney.) Server names are also the prefixes for the new Dynamic DNS names assigned to the server. A simple machine name for the `genesis.mcity.us` domain name would be `MDENTS02.MCITY.US`, which is far better than `BULLWINKLE.MCITY.US`. Be careful, too, about using names that attract security problems. We once used the name `Checkpointcharlie`, which was subsequently hacked the following week.

- **Password for the Administrator account.** This account is installed into the local domain's Administrator account except for domain controllers.
- **Windows Server 2008 components.** The next step is to add the optional components and services. Ignore most of these services in trial installations and go directly to Networking Options. Here, you must provide DHCP information, the DNS server address, and other information.
- **Terminal Services.** You can also choose the operating mode of Terminal Services. For now leave it as is, in Administration mode. There's no point in installing Application Server mode until you are ready, and the mode can be changed at any time.
- **Display settings.** These settings enable you to configure the screen resolution, number of display colors, and video-related information such as refresh rate. You can leave many of these settings at the default. Change your screen resolution, however, to at least 800 × 600. Many Windows Server 2008 folders and menus are jam-packed with icons and information, and 640 × 480 just does not work. In many cases, you should go with 1,024 × 768 resolution.
- **Time and date.** These settings enable you to set time zones and daylight saving information and to adjust the current date and time. After this information is applied, Windows Server 2008 starts Phase 3 of the installation process: the network install.

Windows network install

This phase installs the networking components. Windows Server 2008 attempts to detect the network interface cards (NICs). If you use standard well-known brands such as Intel, you'll have no problems getting through the installation. The following list describes the steps, both automatic and interactive:

- **Network card detection.** After detecting and installing the drivers for the NICs, Windows Server 2008 attempts to locate a DHCP server on the network. It does this by broadcasting on DHCP Port 75 and then listening for a response from a DHCP server. If Windows Server 2008 cannot obtain an IP address, it uses the auto-configuration protocol and assigns itself an IP address. You can then continue with the installation, installing to a new workgroup, and make the necessary network connections later.
- **Networking components.** Next, you are asked to choose the networking components. The basic options to choose are the client for Microsoft Networks, File and Print Sharing for Microsoft Networks, and TCP/IP. You can install other services and components at any time after installation. If you are installing into an existing NT domain that does not have DNS or WINS servers in place, install NetBIOS as well.
- **Workgroup or domain.** If you are installing into a domain, you need the name of the account and password that has the authority to create new accounts in the domain. If you have problems installing into the domain, install into a workgroup. If you do not have a workgroup, create any workgroup name on the fly, such as awshucks, because you can always change it after installation or change to a domain whenever you are ready, post installation.

Final installation setup

This is the fourth phase of the installation, which involves final file copy, configuration, and removal of temporary files. The Setup program copies all remaining files to the hard disk. These include bitmap files, accessories, and services or component files that are either installed into service or left dormant until activated. Setup then applies configuration settings specified during earlier interactions.

The new configuration is saved in the registry databases and on disk to be used for the configuration after the computer starts anew. At this time, all temporary files are removed from the computer. After this activity, the machine is rebooted.

Installing from the network

You can also install servers from network sharepoints, which are called distribution drives or servers. Network installs should obviously be limited to local area network installation because anything less than the standard 100-Mbit/sec network speed makes installation an excruciatingly slow experience.

If you have not created a distribution share, simply copy the I386, I486, or ia64 (for Itanium-based systems) folder on the Windows Server 2008 DVD to a drive and share it. Apply

the necessary access control to prevent unauthorized users from accessing the distribution files. The process, after you have a distribution point in place, is as follows:

1. Create a FAT partition on the target machine. This partition should be within the earlier recommended parameters. You can use the old faithful DOS `FDISK` command to create the partition, but if you are using a very large disk (more than 2GB), only Windows 98's `FDISK` for FAT32 enables you to configure all the space as one huge drive.
2. Boot to a network client. You can use Windows 95/98 boot disks (if you can still find them), but a simple DOS may be all that you need. Your DOS client contains the following software:
   - TCP/IP protocol files
   - DOS operating system files for minimum machine life
   - Network interface card drivers (another reason to use good cards that require no configuration)
3. You also need to create configuration files that log the target machine onto the network and enable it to use the source distribution sharepoint.
4. After you have connected to the network sharepoint, you start the installation by executing setup from the distribution server.

As Windows Server 2008 performs an upgrade, it first gathers information relating to installed hardware and software and reports this to you before installation begins. If some components preclude Windows Server 2008 from installing, you are given an option to remove those components or circumvent attempts by the installation process to support them in the new environment. After you have removed or dealt with the offending components, Windows Server 2008 enables the installation to proceed.

Roles, Features, and Applications

After you install the operating system and log in for the first time as an administrator, Windows Server 2008 automatically presents you with the Server Manager console. This tool enables you to configure role services such as Active Directory, DHCP, DNS, IIS, and more. If you do not need to use the tool immediately, you can close it. It can be accessed again from the menu items in Administrative Tools, from the Control Panel, and the command line. The OS, however, presents this tool to anyone who logs on to a server interactively in the capacity of an administrator.

Once you have completed a basic or Server Core installation, you have a variety of services and applications that can be installed on the server. These are grouped by roles, features, and applications. We will go into these in more detail shortly.

Standalone servers

Standalone servers do not connect to any domain but rather to a workgroup. You can create a workgroup from one standalone server or join the server to another workgroup, Windows

for Workgroups–style. Standalone servers can share resources with other computers on the network, but they do not receive any of the benefits provided by Active Directory.

For a standalone server, you need the following items:

- Workgroup name
- An administrator's password
- Network protocols
- IP address
- DNS IP addresses and host names
- NetBIOS name of host

Member servers

Member servers are members of domains. A member server is running Windows Server 2008, a member of a domain, and not a domain controller. Because it is not a domain controller, a member server does not handle the account logon process, does not participate in Active Directory replication, and does not store domain security-policy information.

Member servers typically function as the following types of servers:

- File servers
- Application servers
- Database servers
- Web servers
- Certificate servers
- Firewalls
- Remote-access servers
- Print servers

Member servers also have a common set of security-related features, as follows:

- Member servers adhere to Group Policy settings that are defined for the site, domain, or organizational unit.
- Resources that are available on a member server are configured for access control.
- Member server users have user rights assigned to them.
- Member servers contain a local security-account database, the Security Account Manager (SAM).

To install a member server into a domain, you need to add the following items to your checklist:

- Domain name
- Network protocols

- IP address
- NetBIOS name of host

Role servers

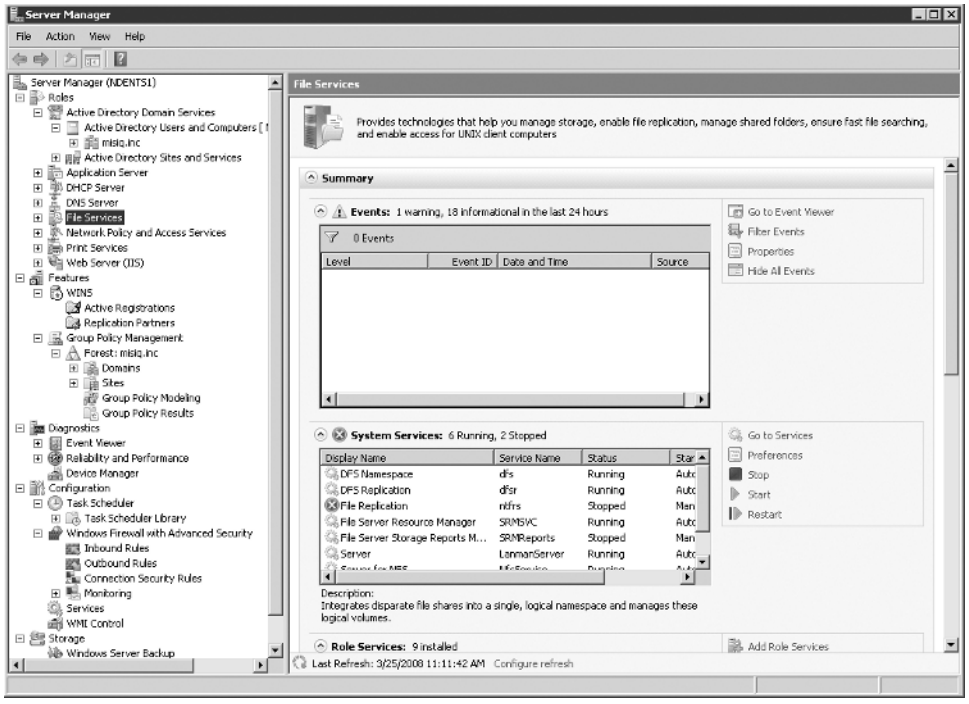
A server on a network — standalone or member — can function in a number of roles.

As the needs of your computing environment change, you may want to change the role of a server. By using the Server Manager and the Add Roles Wizard, you can install Active Directory Domain Servers to promote a member server to a domain controller, or you can install individual roles or combinations of various roles, such as DHCP, WINS, and DNS. It is also relatively straightforward to demote a domain controller to a simple role server or remove any number of roles and features from a server.

Server Manager is the key configuration console you will use for installing server roles and features on your server. It can be configured to open automatically as soon as you log in to the Windows console or desktop. Figure 1-2 shows Server Manager opened to the File Services role details page.

FIGURE 1-2

The Server Manager console.



Types of roles

Let's look at the various roles and features you can install on Windows Server 2008.

- **Active Directory Certificate Services (AD CS).** AD CS role services install on a number of operating systems, including Windows Server 2008, Windows Server 2003, and Windows 2000 Server. Naturally the fullest implementation of AD CS is only possible on Windows Server 2008. You can deploy AD CS as a single standalone certification authority (CA), or you can deploy multiple servers and configure them as root, policy, and certificate issuing authorities. You also have a variety of Online Responder configuration possibilities. AD CS is discussed in depth in Chapter 16.
- **Active Directory Domain Services (AD DS).** This is the role in the Windows Server 2008 operating system that stores information about users, computers, and other resources on a network. AD DS is also used for directory-enabled applications such as Microsoft Exchange Server. AD also stores all information required for Group Policy. See Chapters 17–24.
- **Active Directory Federation Services (AD FS).** AD FS employs technology that allows users over the life of a single online session to securely share digital identity and entitlement rights, or "claims," across security and enterprise boundaries. This role — introduced and supported on all operating systems since Microsoft Windows Server 2003 R2 — provides Web Single Sign-On (SSO) services to allow a user to access multiple, related Web applications.
- **Active Directory Lightweight Directory Services (AD LDS).** This service is ideal if you are required to support directory-enabled applications. AD LDS is a Lightweight Directory Access Protocol (LDAP) compliant directory service.
- **Active Directory Rights Management Services (AD RMS).** This service augments an organization's security strategy by protecting information through persistent usage policies. The key to the service is that the right management policies are bound to the information no matter where it resides or to where it is moved. AD RMS is used to lock down documents, spreadsheets, e-mail, and so on from being infiltrated or ending up in the wrong hands. AD RMS, for example, prevents e-mails from being accidentally forwarded to the wrong people.
- **The Application Server role.** This role supports the deployment and operation of custom business applications that are built with Microsoft .NET Framework. The Application Server role lets you choose services for applications that require COM+, Message Queuing, Web services, and Distributed Coordinated Transactions.
- **DHCP and DNS.** These two roles install these two critical network service services required for every network. They support Active Directory integration and support IPv6. See Chapters 3 and 4 for DNS and DHCP, respectively. WINS is not classified as a key role for Windows Server 2008, and you install it as a feature, discussed later.
- **Fax Server role.** The fax server lets you set up a service to send and receive faxes over your network. The role creates a fax server and installs the Fax Service Manager and the Fax service on the server.

- **File Server role.** This role lets you set up all the bits, bells, and whistles that come with a Windows file server. This role also lets you install Share and Storage Management, the Distributed File System (DFS), the File Server Resource Manager application for managing file servers, Services for Network File System (NFS), Windows File Services, which include stuff like the File Replication Service (FRS), and so on. The File Server role is discussed in Chapters 13 through 15.
- **Network Policy and Access Services.** This provides the following network connectivity solutions: Network Access Protection (NAP), the client health policy creation, enforcement, and remediation technology; secure wireless and wired access (802.1X), wireless access points, remote access solutions, virtual private network (VPN) services, Radius, and more. The Network Policy and Access Service is discussed in Chapter 6.
- **Print Management role.** The print services provide a single interface that you use to manage multiple printers and print servers on your network. Printer management is discussed in Chapter 12.
- **Terminal Services role.** This service provides technologies that enable users to access Windows-based programs that are installed on a terminal server. Users can execute applications remotely (they still run on the remote server) or they can access the full Windows desktop on the target server.
- **Universal Description, Discovery, and Integration (UDDI).** UDDI Services provide capabilities for sharing information about Web services. UDDI is used on the intranet, between entities participating on an extranet, or on the Internet.
- **Web Server role.** This role provides IIS 7.0, the Web server, ASP.NET, and the Windows Communication Foundation (WCF).
- **Windows Deployment Services.** These services are used for deployment of new computers in medium to large organizations.

Features

Server Manager also lets you install dozens of "features" on Windows Server 2008. These so-called features are actually programs or supporting layers that support or augment the functionality of one or more roles, or simply add to the functionality of the server.

A good example of a feature is the clustering service. Now called Failover Clustering, this feature can be used to support mission-critical roles such as File Services, Printer Services, and DHCP Server, on server clusters. This provides for higher availability and performance.

Other features you will likely install include SMTP Server, Telnet Client and Server, Group Policy Management (for use with Active Directory), Remote Assistance, and more.

Net's now look at some specific scenarios.

Windows Server 2008 as a domain controller

Member Servers or just standalone servers can be promoted to domain controller. The Active Directory Wizard can help you install and configure components and enables you to provide

directory service to network computers and users. Before installing or even considering a domain controller, however, review the following checklist:

- Review the Active Directory topic "Introduction to Active Directory" in your Windows Server 2008 Help guide.
- Make sure that you review the role of a domain controller.
- Review concepts about security.
- Review concepts about Domain Name Service (DNS) namespace planning and integration with DNS.
- Verify that the server has an NTFS partition.
- Verify that DNS is correctly configured.

Promoting member servers to domain controllers either creates new domains or adds additional domain controllers to existing domains. In creating the first domain, you must have already created one domain controller in that domain. The act of creating the domain controller also creates the domain.

If your organization needs additional domains, you must create one domain controller for each additional domain. New domains in a forest must be either a new child domain or the root of a new domain tree. If you decide to create a child domain, the name of the new domain must contain the full name of the parent. To hierarchically organize domains within your organization, make sure that you use the domain tree structure. If you would rather create the root of a new domain tree, make sure that its name is not related to the other domains in the forest.

To improve the availability and reliability of network services, add additional domains to a single domain. You can create new domain controllers across the network or from backup media.

Windows Server 2008, Windows Enterprise Server 2008, and Windows Datacenter Server 2008 all support Active Directory. AD uses a structured datastore for logical, hierarchical organization of directory information. The datastore is also known as the directory, and it contains information about Active Directory objects. Active Directory objects include shared resources such as servers, volumes, printers, and the network users and accounts.

Active Directory is tightly integrated with security through logon authentication and access control to objects. This makes managing directory data and organization throughout the network easy for an administrator. Schemas also help administrators with daily tasks by setting constraints and limits on instances of objects. Schemas consist of classes of objects and attributes contained in the directory. Global catalogs consist of the information about each and every object in a directory; therefore, a global catalog provides easy access to directory information regardless of which domain of the directory actually contains the data.

The following list summarizes the Active Directory features that are enabled by default on any domain controller running Windows Server 2008:

- The selection of multiple user objects and the capability to modify common attributes of multiple user objects at one time.
- The capability to drag and drop Active Directory objects from container to container or to a desired location in the domain hierarchy. You also have the capability to drag objects to group membership lists.
- Enhanced search functionality is object-oriented and provides an efficient search that minimizes network traffic associated with browsing objects.
- The capability to save queries, enabling you to save commonly used search parameters for reuse in Active Directory Users and Computers.
- Active Directory command-line tools, which give you the capability to run directory-service commands for administration scenarios.
- You can now create instances of specified classes in the base schema of a forest and instances of several common classes, including country or region, person, organizationalPerson, groupOfNames, device, and certificationAuthority.
- The inetOrgPerson class is added to the base schema and can be used in the same manner as the user class.
- You can configure replication scope for application-specific data among domain controllers running Windows Server 2008.
- The capability to add additional domain controllers to existing domains by using backup media, thus reducing the time necessary for an administrator to create additional domain controllers.
- Universal group membership caching to help prevent the need to locate a global catalog across a WAN.

Active Directory can provide a companywide network solution with one domain, reduced sign-on capabilities, and one single point of management. Active Directory helps eliminate unnecessary domains and reduces server hardware and maintenance costs.

CROSS-REF **Please refer to Chapters 17–22 for a more in-depth view of Active Directory.**

Two approaches to installing a domain controller are possible. First, you can raise the machine as a member server and promote it post-installation — and even post-burn-in. Alternatively, you can promote it to domain controller status during an automated installation. The latter option naturally requires a script.

We don't recommend the latter option unless you are really confident about your machines and their configuration or you have a huge rollout. If you are an Original Equipment Manufacturer (OEM), you would not need to be concerned about domain controllers and Active Directory

because the domain specifics, such as creating a new tree or forest or joining existing trees and forests, is done on the customer's network. Conversely, if you, as a consultant or network engineer, have created an extensive unattended or remote installation regimen that automatically raises the machine as a domain controller, you know what you are doing.

For now, you have several reasons to not promote during or just after initial installation. First, promoting a domain controller is a time-intensive operation. (Active Directory goes through extensive self-configuration before the installation completes.) Second, if you experience a problem with the machine, you must demote the domain controller, which can be a complicated process. Third, after you have installed and raised a domain controller, you do not want to demote it because of a hardware problem or risk trashing your domain controller.

If Active Directory is demoted, it tears down everything that it created and restores the machine to the control of the registry and the local SAM. In fact, it is like watching a movie in reverse. Active Directory asks you for a new administrator account name and password for the rollback. All configuration changes made to the machine, such as desktop settings, are restored to the default, newly created settings. After you reboot the machine, you are back to where you started. You do not even get earlier changes that you made to the registry because the registry is essentially reinstalled after Active Directory comes down (because it is wiped out if you promote the server).

A good reason lies behind this. Everything configured on a domain controller is stored in the directory databases, and after the registry is restored, you can re-promote it from scratch. Promoting a domain controller is dealt with in Chapter 21.

To promote a role server into a domain controller, you need to add the following items to your checklist:

- Domain name
- An administrator's password
- Network protocols
- IP address
- DNS IP addresses and host names
- NetBIOS name of host
- Role service information

The checklist for a domain controller is as follows:

- Domain name. If you are creating a new domain, you need the name of the parent domain that you are installing under or the existing tree name (or the forest name if you are installing a new domain tree). If you are adding a domain controller to an existing domain, you need to have that name handy as well.
- An administrator's password
- Network protocols

- IP address
- NetBIOS name of host
- DNS IP addresses and host names

Windows Server 2008 as a Communications Server and Microsoft Exchange

Microsoft Exchange Server unites users with knowledge anytime, anywhere. Exchange is designed to meet the messaging and collaboration needs of small organizations, large distributed enterprises, and everything in between. Microsoft Exchange integrates with Windows Server 2008, although there have been a few hairy incompatibility problems with Exchange 2007 on the RTM build of Windows Server 2008. We list a few of the Exchange Server main services in the following sections.

Internet Information Services integration

Exchange is also integrated with IIS to provide for high-performance mail protocols, SMTP protocols, and POP protocols. Exchange also provides a browser interface to access the Microsoft Outlook Web Access client.

Active Directory integration

Active Directory, which is covered in more detail in the final chapters of this book, is an enterprise directory service that is highly scalable and fully integrated with Exchange at the system level. Exchange takes full advantage of the Windows Server 2008 Active Directory; with but a single point of administration, it enables users to control all messaging services seamlessly. All directory information, including users, mailboxes, servers, sites, and recipients, is stored in Active Directory. Administrators benefit from the unified administration, experience no user-interface changes, and require no retraining after switching to Active Directory. Integration features of Exchange Server and Active Directory include the following:

- Unified administration of Exchange Server and Windows Server 2008 enables an administrator to manage all user data in one place using one set of tools.
- Security groups in Windows Server 2008 can be automatically used as Exchange distribution lists, removing the need to create a parallel set of distribution lists for each department or group.
- Active Directory's schema extensibility enables the management of distributed information and easily configurable Exchange user and server information.
- Lightweight Directory Access Protocol (LDAP) is a native access protocol for directory information.

Distributed services

Distributed services enable subsystems to use storage, protocol, and directories on different computers, providing for scalability for millions of users. This system is extremely configurable, providing extensibility and flexibility for system architecture.

Security

Exchange Server offers you the only messaging system that is fully integrated with the Windows Server 2008 security model. Administrators use the Windows Server 2008 security model to define the permissions for all messaging and collaboration services, including public folders. This means that administrators can learn a single permissions model for managing both Windows Server 2008 and Exchange and can create a single set of security groups to apply to either Windows Server 2008 resources or Microsoft Exchange objects. This helps simplify your domain administration, and Exchange Server enables permissions to be set at the item or document level. Security descriptors can be set for messages and components. These features provide for new levels of security.

Single-seat and policy-based administration

Microsoft Exchange uses a graphic administration and monitoring system that integrates with Windows Server 2008's Microsoft Management Console (MMC) to provide single-seat administration. The MMC does not provide you with management capabilities, but with a common interface that enables you to manage all your needs. The Microsoft Exchange System Manager, Microsoft Active Directory, and Internet Services Manager are snap-ins that provide the management for Server 2008. Policy-based management provides the administrator with the capability to perform single operations made up of hundreds of objects. Policies are a set of objects defined by the administrator. The administrator can also define recipient policies that could potentially affect hundreds of thousands of users, groups, and contacts in Active Directory.

SMTP message routing

Exchange Server supports SMTP, POP, LDAP, IMAP, HTTP, NNTP, S/MIME, and X.509 version 3. This versatility enables Exchange Server to act as an organization's gateway to the Internet. Providing high-performance routing of e-mail services, SMTP is, by default, the transport protocol for routing all message traffic between servers, within an Exchange site and between sites. Your organization's use of SMTP results in increased performance and new opportunities for integration with the Internet. Exchange Server's message algorithms have been enhanced to provide fault-tolerant message delivery and to eliminate messages that bounce, even when multiple servers or network links are down. This provides for increased message bandwidth and performance. SMTP routing provides customers with considerable flexibility in designing a reliable, high-performance messaging backbone by using Exchange Server.

Internet mail content

Exchange Server can significantly increase performance of e-mail, because you use e-mail clients to store and retrieve Multipurpose Internet Mail Extensions (MIME) content directly from the base, without any form of content conversion. Client software such as Outlook enables you to stream data in and out of the database. This process helps performance immensely.

All the features discussed in the preceding sections provide low cost-of-ownership, which makes Microsoft Exchange Server a valuable asset to every organization.

System Monitoring Using Windows Management Instrumentation

Windows Management Instrumentation (WMI) helps simplify the instrumentation of computer software and hardware. It provides you with a means of monitoring and controlling system components, both locally and remotely. The sole purpose of the WMI is to define a set of environment-independent specifications, thus helping you share management information that works with existing enterprise-management standards, such as Desktop Management Interface and the Simple Network Management Protocol (SNMP). The WMI provides a uniform model that complements these standards.

WMI is fully integrated with Windows Server 2008 to provide a simplified approach to management. Such tools as Microsoft Management Console help simplify the task of developing well-integrated management applications, therefore enabling vendors to provide Windows Server 2008 customers with enterprise-scalable management solutions. Combining local and remote events and the WMI query language provides you with the tools that you need to create complex management solutions.

The WMI also provides you with the Windows Driver Model (WDM), a kernel-level instrumentation technology. This technology provides you with consistent, open access to management data. WMI extensions are available for the following WDM capabilities:

- Publishing kernel instrumentation
- Configuring device settings
- Providing kernel-side event notification
- Publishing custom data
- Enabling administrators to set data security
- Accessing instrumentation by way of WMI

You can run the WMI console from the command line in interactive mode or non-interactive mode. Interactive mode is used for entering commands at the computer, and noninteractive mode is useful for processing batch procedures.

The console installs the first time that you run it. If a change is introduced to the managed object format (MOF) files, the console automatically compiles the alias. To start the WMI console from a command prompt, type **wmic**. The prompt now should look as follows: `wmic:root\cli>`. The WMI console enables you to enter aliases, commands, and global switches, or you can enter /? for Help.

You can also run WMI console in noninteractive mode, whereby the command prompt returns to you after executing the command. An example is as follows:

```
<PROMPT>wmic os get /format:hform>OperatingSystem.html
```

The output from the command is redirected to an HTML file.

Windows Server 2008 for Database Services with SQL Server

If you have a modest understanding of database connectivity, you should find SQL Server's command syntax uncomplicated and easy to use. If you are an experienced developer, you are sure to appreciate the scalable, high-performance access that SQL Server provides.

If you are concerned about backward compatibility, we suggest connecting using ODBC.

You want to avoid any security issues so we are going to go through the steps to create an account for SQL Server to access data on a remote computer. Start by opening Management Studio; then, in the console tree, select Microsoft SQL Server ➢ SQL Server Group ➢ SQLComputerName. Select Databases and then double-click your database; proceed by right-clicking Users ➢ New Database User. The login name should be domain\username, and the Public checkbox should be selected for all the following items:

- db_owner
- db_accessadmin
- db_securityadmin
- db_ddladmin
- db_datareader
- db_datawriter
- db_backupoperator

NOTE **Do not select db_denydatareader or db_denydatawriter. These options, if selected, deny members read and write permissions to the database.**

You can choose between the TCP/IP Sockets and Named Pipes connection methods for accessing a remote SQL Server database. Named Pipes database clients must be authenticated by Windows Server 2008 prior to establishing a connection. Alternatively, connections using TCP/IP Sockets connect directly to the database server without connecting through an intermediary computer. Because connections made with TCP/IP Sockets connect directly to the database server, users can gain access through SQL Server authentication, rather than Windows Server 2008 authentication.

One of the main challenges of designing a sophisticated Web database application seems to involve managing database connections. After you open and maintain a database connection, it can severely strain a database server's resources and result in stability issues. Database servers experiencing a sudden increase in activity can become backlogged, greatly increasing the time necessary to establish a database connection.

Windows Server 2008 for IIS and ASP.NET

Windows Server 2008 offers integration between Visual Studio 2008 and IIS. This tight integration provides developers with very high levels of functionality. Now that the request-processing architecture is integrated with IIS 7.0, it should provide an improved experience for those of you using ASP.NET and the Microsoft .NET Framework. The new Windows Server 2008 Web Edition delivers a single-purpose solution for Internet Service providers, application developers, and others wanting to use only the specific Web functionality.

Windows Server 2008 for Application Services

Windows Server 2008 builds on the core strengths of the Windows family, providing security, manageability, reliability, availability, and scalability across the board. Many advancements were made in Windows Server 2008 that provide benefits for application development, resulting in lower total cost-of-ownership and better performance. The following list describes a few of these benefits:

- Simplified integration and interoperability
- Improved developer productivity
- Increased enterprise efficiency
- Improved scalability and reliability

- End-to-end security
- Efficient deployment and management
- Simplified integration and interoperability

Windows Server 2008 delivers a revolutionary application environment to build, deploy, and run XML Web services. Microsoft has provided integrated support for XML Web services, which enables applications to take advantage of the loosely coupled principles of Internet computing. The Windows Server 2008 application environment improves the productivity of developers by providing integrated application services and industry-leading tool support. The following feature set helps increase the productivity of developers:

- **ASP.NET.** Besides standard Web-based applications ASP.NET XML Web services allows developers to write their business logic in Web services, and the ASP.NET infrastructure is responsible for delivering that service via SOAP and other public protocols.
- **Automatic memory management.** The .NET Framework runs in the common-language runtime, which is a garbage-collected environment. Garbage collection frees applications that are using .NET Framework objects from the need to explicitly destroy those objects, reducing common programming errors dramatically.
- **Industry-leading tools.** Visual Studio 2008 provides an integrated, multilanguage tool for building Web applications.
- **Microsoft .NET Framework.** By integrating the .NET Framework into the Windows Server 2008 application-development environment, developers are freed from writing the day-in, day-out code and can instead focus their efforts on delivering real business value.
- **Reusable code.** Visual Studio 2008 provides an architecture that is easy to learn and that enables improved code reuse.
- **Separation of code from content.** This enables developers and content creators to work in parallel by keeping content separate from application code.
- **Server-side Web controls.** Visual Studio 2008 Web controls are compiled and run on the server for maximum performance, and can be inherited and extended for even more functionality.

Applications that are developed using Windows Server 2008 tend to be more responsive and available because Windows Server 2008 can be managed by so few people. This helps lower the total cost of ownership and provides better performance. Microsoft has also made many programming-model enhancements, providing component aliases, public and private components, process initialization, and services without components. Component aliasing enables you to configure the same physical implementation of a component as many times as you want. This provides component reuse at the binary level. The public and private components enable you to individually mark components as public for use in other applications, or private if the component can be seen and activated only by other components in that same application. Process initialization provides the developer with the capability to execute code as the hosting process starts and finishes. This helps your component take the opportunity to take any action, such as initializing connections, files, caches, and so on. Services without components enable you to

programmatically enter and leave a service domain. This enables you to build components that use transactions without needing to inherit from ServicedComponent.

Building and deploying your application on Windows Server 2008 gives you better performance and more options for the design and architecture of your system.

Windows Server 2008 for Resolutions Services

The following sections describe what you need to create a plan to prepare and configure your server by using DNS, WINS, and DHCP. These sections focus on decisions that you must make for a complete Windows Server 2008 installation.

DNS

Before you begin using DNS on your network, decide on a plan for your DNS domain namespace. Coming up with a namespace plan involves making some decisions about how you intend to use DNS naming and what goals you are trying to accomplish by using DNS. Some questions that you may have at this stage include the following:

- Have you previously chosen and registered a DNS domain name for use on the Internet?
- Are you going to set up DNS servers on a private network or the Internet?
- What naming requirements do you need to follow in choosing DNS domain names for computers?

Choosing your first DNS domain name

In setting up DNS servers, you should first choose and register a unique parent DNS domain name that can be used for hosting your organization on the Internet. Before you decide on a parent DNS domain name for your organization to use on the Internet, search to determine whether the domain name is already registered to another organization.

CROSS-REF DNS, as it relates to Active Directory, is covered in more detail in Chapter 17.

DNS namespace planning for Active Directory

Before a DNS domain namespace can be correctly implemented, the Active Directory structure needs to be available, so you must begin with the Active Directory design and support it with the appropriate DNS namespace.

Active Directory domains are named by using DNS names. In choosing DNS names to use for your Active Directory domains, start with the registered DNS domain-name suffix that

your organization has reserved for use on the Internet and combine this name with something significant in your organization to form full names for your Active Directory domains.

TIP **In planning your DNS and Active Directory namespace, we recommend that you use a different set of distinguished names that do not overlap as the basis for your internal and external DNS use.**

Only use characters in your names that are part of the Internet standard character set permitted for use in DNS host naming. Permitted characters are all letters (a–z), numbers (0–9), and the hyphen (-).

DHCP

If you are still in the decision-making process as you are deciding how many servers your organization needs, consider the locations of the routers on the network and whether you want a DHCP server in each subnet. If you are planning on extending the use of a DHCP server across more than one network, you may need to configure additional DHCP relay agents and use superscopes as well. If DHCP service is provided between segments, transmission speeds may also be a factor. If your WAN links or dial-up links are slower, you may need a DHCP server on both sides of these links to service clients locally. Currently, the only limit that a DHCP server can serve is determined by the number of available IP addresses.

Following are some Windows Server 2008 factors that could enhance DHCP server performance:

- The primary contributing factor to improving DHCP server performance is the amount of random access memory (RAM) and the speed of the server disk drives installed.
- You should carefully evaluate disk-access times and average times for disk read/write operations in sizing and planning for your DHCP-server hardware specifications. You should also try to increase RAM to the point where server performance is maximized.

TIP **For the best possible DHCP server design in most networks, we recommend that you have, at most, 10,000 clients per server.**

Most networks need one primary online DHCP server and one other DHCP server acting as a secondary, or backup, server. If you choose not to implement two DHCP servers, using the 80/20 rule for balancing scopes, but want to continue to provide a measure of potential fault tolerance, you may consider implementing a backup or hot standby DHCP server as an alternative.

WINS

The first decision that you need to make is how many WINS servers your organization needs. A single WINS server can handle NetBIOS name-resolution requests for a large number of computers, but you must also consider the location of the routers on your network and the distribution of clients in each subnet as you decide how many WINS servers are actually required.

Determine whether you want to configure WINS servers as pull or push partners, and set partner preferences for each server. WINS servers are designed to help reduce broadcast traffic between local subnets; WINS creates some traffic between servers and clients. This can be particularly important if you use WINS on routed TCP/IP networks. Consider the effects of slower speed links on both replication traffic between WINS servers and NetBIOS registration and renewal traffic required for WINS clients. In addition, consider how temporarily shutting down a WINS server can affect your network. Use additional WINS servers for failure recovery, backup, and redundancy.

The following two factors can enhance WINS server performance:

- Installing dual processors on the computer running WINS. This can help increase performance by almost 25 percent.
- Installing a dedicated disk drive, separate from the system drive, for the WINS database

TIP **After you establish a WINS server on your intranet, adjusting the renew interval, which is the time between a WINS client-name registration and name renewal, can help you trim server-response times.**

You can also sometimes estimate WINS client traffic based on the behavior of the WINS clients. In estimating WINS client traffic, however, you also need to consider the network topology and the design or configuration of the network routers. In some cases, predicting the traffic load on a specific network router is not possible because the routers are configured to autonomously route traffic based on factors other than traffic load. By testing the performance of your network installation of WINS, you can better identify potential problems before they occur. Use WINS server performance counters, which are available through the use of System Monitor. One last point about WINS: it is not installed as a server role, but rather as an add-on feature.

Summary

This chapter took you through the Windows Server 2008 basic install procedure. We recommend that you install only what you need to get the system up and running. Later, you can begin adding advanced components to the server and establish its role on the network or promote it to an Active Directory domain controller.

We also took you through an exhaustive discussion of hardware. Unless you plan to install complex adapter or interface cards for specialized purposes, such as modems, telephony cards, sound cards, and so on, you won't have problems as long as you stick to tried-and-tested components.

The next chapter provides the information that you now need to configure and deploy your running server.