Microsoft prePress

Installing and Configuring Windows Server® 2012

Ian McLean

Exam Ref

EXAM

70-410

PUBLISHED BY
Microsoft Press
A Division of Microsoft Corporation
One Microsoft Way
Redmond, Washington 98052-6399

With ***Microsoft prePress***, you can access just-written content from upcoming books. The chapters come straight from our respected authors, before they're fully polished and debugged—for critical insights now, when you need them.

Contents at a glance

Note: Chapters included in this file are indicated in black.

CHAPTER 1

Installing and configuring servers

Installing new Windows servers on your network is not something to be done casually—you must plan the installation well in advance. Among other things, you must decide what edition of the operating system to install, whether you are installing the full GUI or the Server Core option, what your virtualization strategy will be, if any, and what roles you intend to implement on the server. If you are installing Windows Server 2012 for the first time, you might also have to decide whether to add the server to your production network or install it on a test network.

This chapter discusses the process of installing Windows Server 2012, using either a clean install or a server upgrade, as well as the server configuration tasks you must perform immediately following the installation. Finally it considers the configuration of various types of hard disk technologies used for local storage, and the deployment of roles to servers all over the network.

Objectives in this chapter:

- Objective 1.1: Install servers
- Objective 1.2: Configure servers
- Objective 1.3: Configure local storage

EXAM TIP

Some exam questions are in a multiple-choice format, where answers are either right or wrong. If, in the exam, you have an option where it seems as though two answers could be right, but you can only choose one answer, you've likely missed a clue in the question text that would allow you to discard one of these answers. When exams are authored, not only does the question writer have to provide good reasons why one answer is correct, but also why the other answers are incorrect. Although there is a small chance that you've come across a bad question that got through proofreading and peer review, it's more likely that in a stressful exam situation you've overlooked a vital bit of evidence that discounts an answer you suspect is correct.

Objective 1.1: Install servers

Installation is a key topic and has been extensively tested in previous Windows Server exams. There is no reason to believe the 70-410 exam will be different. This objective discusses planning a Windows Server 2012 installation. It looks at the preinstallation requirements and how you can prepare your installation hardware. It also considers the server roles you can implement during installation.

The objective takes you through a clean installation of Windows Server Core 2012, and describes how the Features on Demand function enables you to optimize resources by removing all the files associated with a server role or feature you have chosen to delete. The objective also looks at the options for upgrading a Windows Server 2008 or Windows Server 2008 R2 server to Windows Server 2012 and migrating roles from an existing server to a new one.

This objective covers how to:

- Plan for a server installation
- Plan for server roles
- Plan for a server upgrade
- Install Server Core
- Optimize resource utilization using Features on Demand
- Migrate roles from previous versions of Windows Server

Planning for a server installation

In previous versions of Windows Server, installation planning could become a complex task. You had to decide from the outset what edition of the operating system to install, whether to install the 32-bit or 64-bit version, and whether you should perform a Server Core installation or use the full graphical user interface (GUI). All of these decisions affected the server hardware requirements, and all of them were irrevocable. To change the edition, the platform, or the interface, you have to reinstall the server from scratch.

With Windows Server 2012, the options are reduced substantially, and so are the installation decisions. There is no 32-bit version of Windows Server 2012; only a 64-bit operating system is available—reflecting the fact that most major applications are now 64-bit and that modern server configurations are typically supported on hardware that requires 64 bits. There are now only four Windows Server 2012 editions to choose from, down from six in Windows Server 2008 R2. The Server Core and full GUI installation options still remain, along with a third option, called the Minimal Server Interface. However, it is now possible to switch among these options without having to reinstall the operating system.

Selecting a Windows Server 2012 edition

Microsoft releases all of its operating systems in multiple editions, which provides consumers with varying price points and feature sets. When planning a server deployment, the operating system edition you choose should be based on multiple factors, including the following:

- The roles you intend the servers to perform
- The virtualization strategy you intent to implement
- The licensing strategy you plan to use

Compared with Windows Server 2008, Microsoft has simplified the process of selecting a server edition by reducing the available products. As with Windows Server 2008 R2, Windows Server 2012 requires a 64-bit processor architecture. All of the 32-bit versions have been eliminated, and for the first time since the Windows NT Server 4.0 release, there will be no build supporting Itanium processors. This leaves Windows Server 2012 with the following core editions:

- **Windows Server 2012 Datacenter** The Datacenter edition is designed for large and powerful servers with up to 64 processors and fault-tolerance features such as hot add processor support. As a result, this edition is available only through the Microsoft volume licensing program and from original equipment manufacturers (OEMs), bundled with a server.
- **Windows Server 2012 Standard** The Standard edition includes the full set of Windows Server 2008 features, differing from the Datacenter edition only in the number of virtual machine (VM) instances permitted by the license.
- **Windows Server 2012 Essentials** The Essentials edition includes nearly all of the features in the Standard and Datacenter editions, except for Server Core, Hyper-V, and Active Directory Federation Services. The edition is limited to one physical or virtual server instance and a maximum of 25 users.
- **Windows Server 2012 Foundation** A reduced version of the operating system designed for small businesses that require only basic server features such as file and print services and application support. The edition includes no virtualization rights and is limited to 15 users.

These various editions have prices commensurate with their capabilities. Obviously, the goal of administrators planning server deployments is to purchase the most inexpensive edition that meets all of their needs. The following sections examine the primary differences among the Windows Server 2012 editions.

Supporting server roles

Windows Server 2012 includes predefined combinations of services called roles that implement common server functions. Computers running the Windows Server 2012 operating system can perform a wide variety of tasks, using both the software included with the product and third-party applications. The activities Windows Server 2012 performs for network clients

are known as roles. After you install the Windows Server 2012 operating system, you can use Server Manager or Windows PowerShell to assign one or more roles to that computer.

Some of the Windows Server 2012 editions include all of these roles, whereas others include only some of them. Selecting the appropriate edition of Windows Server has always been a matter of anticipating the roles that the computer must perform. At one time, this was a relatively simple process. You planned your server deployments by deciding which ones would be domain controllers, which ones would be web servers, and so forth. Once you made these decisions, you were done, because server roles were largely static.

With the increased focus on virtualization in Windows Server 2012, however, more administrators will be forced to consider not only what roles a server must perform at the time of the deployment, but what roles they will perform in the future as well.

Using virtualized servers, you can modify your network's server strategy at will, to accommodate changing workloads and business requirements, or to adapt to unforeseen circumstances. Therefore, the process of anticipating the roles a server will perform must account for the potential expansion of your business, as well as possible emergency needs.

Supporting server virtualization

The Windows Server 2012 Datacenter and Standard editions both include support for Hyper-V, but they vary in the number of VMs permitted by their licenses. Each running instance of the Windows Server 2012 operating system is classified as being in a *physical operating system environment (POSE)* or a *virtual operating system environment (VOSE)*. When you purchase a Windows Server 2012 license, you can perform a POSE installation of the operating system, as always. After installing the Hyper-V role, you can then create VMs and perform VOSE installations on them. The number of VOSE installations permitted by your license depends on the edition you purchased, as shown in Table 1-1.

TABLE 1-1 Physical and virtual instances supported by Windows Server 2012 editions

EDITION	POSE INSTANCES	VOSE INSTANCES
Datacenter	1	Unlimited
Standard	1	2
Foundation	1	0
Essentials	1 (POSE or VOSE)	1 (POSE or VOSE)

NOTE **License restrictions are not software restrictions**

The limitations specified in Table 1-1 are those of the license, not the software. You can, for example, create more than four VMs on a copy of Windows Server 2012 Enterprise, but you must purchase additional licenses to do so.

Server licensing

Microsoft provides several different sales channels for Windows Server 2012 licenses, and not all of the editions are available through all of the channels. Licensing Windows Server 2012 includes purchasing licenses for both servers and clients, and there are many options for each one.

If you are already involved in a licensing agreement with Microsoft, you should be aware of the server editions that are available to you through that agreement. If you are not, you should investigate the licensing options available to you before you select a server edition.

Table 1-2 lists the sales channels through which you can purchase each of the Windows Server 2012 editions.

TABLE 1-2 Windows Server sales channel availability, by edition

	RETAIL	VOLUME LICENSING	ORIGINAL EQUIPMENT MANUFACTURER
Datacenter	No	Yes	Yes
Standard	Yes	Yes	Yes
Foundation	No	No	Yes
Essentials	Yes	Yes	Yes

Installation requirements

If your computer has less than the following hardware specifications, Windows Server 2012 will not install correctly (or possibly at all):

- 1.4 GHz 64-bit processor
- 512 MB RAM
- 32 GB available disk space
- DVD drive
- Super VGA (800 x 600) or higher resolution monitor
- Keyboard and mouse (or other compatible pointing device)
- Internet access

32 GB of available disk space should be considered an absolute minimum. The system partition will need extra space if you install the system over a network or if your computer has more than 16 GB of RAM installed. The additional disk space is required for paging, hibernation, and dump files. In practice, you are unlikely to come across a computer with 32 GB of RAM and only 32 GB of disk space. If you do, free more disk space or invest in additional storage hardware.

As part of Microsoft's increased emphasis on virtualization and cloud computing in its server products, they have increased the maximum hardware configurations significantly for Windows Server 2012. These maximums are listed in Table 1-3.

TABLE 1-3 Maximum hardware configurations in Windows Server versions

	WINDOWS SERVER 2012	WINDOWS SERVER 2008 R2
Logical processors	640	256
RAM	4 terabytes	2 terabytes
Failover cluster nodes	63	16

Choosing installation options

Many enterprise networks today use servers that are dedicated to a particular role. When a server is performing a single role, does it really make sense to have so many other processes running on the server that contribute little to that role?

Many IT administrators today are so accustomed to GUIs that they are unaware that there was ever any other way to operate a computer. When the first version of Windows NT Server appeared in 1993, many complained about wasting server resources on graphical displays and other elements that they deemed unnecessary. Up until that time, server displays were usually minimal, character-based, monochrome affairs. In fact, many servers had no display hardware at all, relying instead on text-based remote administration tools, such as Telnet.

Using Server Core

Windows Server 2012 includes an installation option that addresses those old complaints. When you select the Windows *Server Core* installation option, you get a stripped-down version of the operating system. There is no Start menu, no desktop Explorer shell, no Microsoft Management Console, and virtually no graphical applications. All you see when you start the computer is a single window with a command prompt, as shown in Figure 1-1.

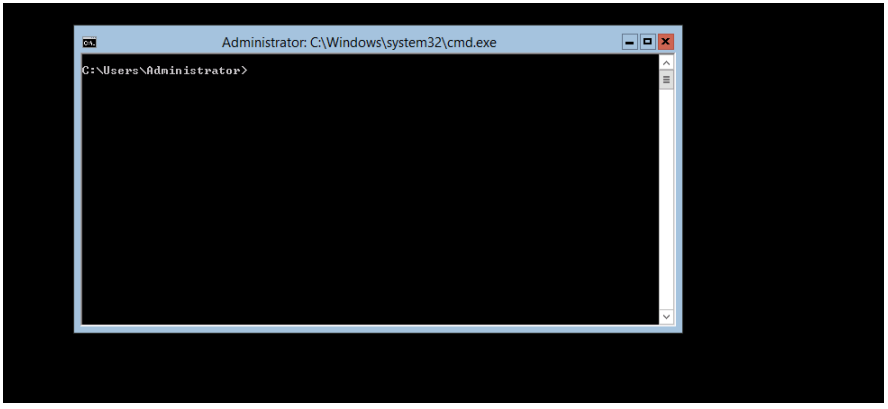

Figure 1-1 The default Server Core interface.

NOTE **What is Server Core?**

Server Core is not a separate product or edition. It is an installation option included with the Windows Server 2012 Standard and Datacenter editions.

The advantages of running servers using Server Core are several:

- **Hardware resource conservation** Server Core eliminates some of the most memory- and processor-intensive elements of the Windows Server 2012 operating system, thus devoting more of the system hardware to running essential services.
- **Reduced disk space** Server Core requires less disk space for the installed operating system elements, as well as less swap space, which maximizes the utilization of the server's storage resources.
- **Reduced patch frequency** The graphical elements of Windows Server 2012 are among the most frequently patched, so running Server Core reduces the number of patches that administrators must apply. Fewer patches also mean fewer server restarts and less downtime.
- **Reduced attack surface** The less software there is running on the computer, the fewer the entrances there are for attackers to exploit. Server Core reduces the potential openings presented by the operating system, increasing its overall security.

When Microsoft first introduced the Server Core installation option in Windows Server 2008, it was an intriguing idea, but few administrators took advantage of it. The main reason for this was that most server administrators were not sufficiently conversant with the command-line interface to manage a Windows server without a GUI.

In Windows Server 2008 and Windows Server 2008 R2, the decision to install the operating system using the Server Core option was irrevocable. Once you installed the operating system using Server Core, there was no way to get the GUI back except to perform a complete reinstallation. That has all changed in Windows Server 2012. You can now switch a server from the Server Core option to the Server with a GUI option, and back again, at will, using PowerShell commands.

MORE INFO **There and back again**

For more information on converting from Server Core to Server with a GUI and back again, see "Objective 1.2: Configure servers," later in this chapter.

This ability means that administrators can install Windows Server 2012 using the Server with a GUI option, if they want to, configure the server using the familiar graphical tools, and then switch the server to Server Core to take advantage of the benefits listed earlier.

SERVER CORE DEFAULTS

In Windows Server 2012, Server Core is the default installation option, and there are reasons

why other than the ability to switch options after installing. In Windows Server 2012, Microsoft is attempting to fundamentally modify the way that administrators work with their servers. Server Core is now the default installation option, because in the new way of managing servers, administrators should rarely, if ever, have to work at the server console, either physically or remotely.

Windows Server has long been capable of remote administration, but this capability has been a piecemeal affair. Some Microsoft Management Console (MMC) snap-ins enabled administrators to connect to remote servers, and Windows PowerShell 2.0 provided some remote capabilities from the command line, but Windows Server 2012, for the first time, includes comprehensive remote administration tools that virtually eliminate the need to work at the server console.

The new Server Manager application in Windows Server 2012 enables administrators to add servers from all over the enterprise and create server groups to facilitate the configuration of multiple systems simultaneously. The new Windows PowerShell 3.0 environment increases the number of available cmdlets from 230 to more than 2,430.

With tools like these, it is possible for administrators to install their servers using the Server Core option, execute a few commands to join each server to an Active Directory Domain Services domain, and then never touch the server console again. They can perform all subsequent administration tasks, including the deployment of roles and features, using Server Manager and PowerShell from a remote workstation.

SERVER CORE CAPABILITIES

In addition to omitting most of the graphical interface, a Server Core installation omits some of the server roles found in a Server with a GUI installation. However, the Server Core option in Windows Server 2012 includes 13 of the 19 roles, plus support for SQL Server 2012, as opposed to only 10 roles in Windows Server 2008 R2 and nine in Windows Server 2008.

Table 1-4 lists the roles and features that are available and not available in a Windows Server 2012 Server Core installation.

TABLE 1-4 Windows Server 2012 Server Core roles

ROLES AVAILABLE IN SERVER CORE INSTALLATION	ROLES NOT AVAILABLE IN SERVER CORE INSTALLATION
Active Directory Certificate Services	Active Directory Federation Services
Active Directory Domain Services	Application Server
Active Directory Lightweight Directory Services	Fax Server
Active Directory Rights Management Services	Network Policy and Access Services
DHCP Server	Remote Desktop Services • Remote Desktop Gateway • Remote Desktop Session Host

	• Remote Desktop Web Access
DNS Server	Volume Activation Services
File and Storage Services	Windows Deployment Services
Hyper-V	
Print and Document Services	
Remote Desktop Services • Remote Desktop Connection Broker • Remote Desktop Desktop Licensing • Remote Desktop Virtualization Host	
Remote Access	
Web Server (IIS)	
Windows Server Update Services	

Using the Minimal Server Interface

If the advantages of Server Core sound tempting, but there are traditional server administration tools you don't want to give up, Windows Server 2012 provides a compromise called the Minimal Server Interface.

The *Minimal Server Interface* is a setting that removes some of the most hardware-intensive elements from the graphical interface. These elements include Internet Explorer and the components that make up the Windows shell, including the desktop, Windows Explorer, and the Windows 8 desktop apps. Also omitted are the Control Panel items implemented as shell extensions, including the following:

- Programs and Features
- Network and Sharing Center
- Devices and Printers Center
- Display
- Firewall
- Windows Update
- Fonts
- Storage Spaces

What's left in the Minimal Server Interface are the Server Manager and MMC applications, as well as Device Manager and the entire PowerShell interface. This provides administrators with most of the tools they need to manage local and remote servers

To configure a Windows Server 2012 Server with a GUI installation to use the Minimal Server Interface, complete the following procedure.

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.

2. Click Manage > Remove Roles And Features. The Remove Roles and Features Wizard opens, showing the Before You Begin page.
3. Click Next to open the Server Selection page.
4. In the Server Pool list, select the server you want to modify and click Next. The Remove Server Roles page opens.
5. Click Next to open the Remove Features page.
6. Scroll down the Features list and expand the User Interfaces And Infrastructure feature, as shown in Figure 1-2.

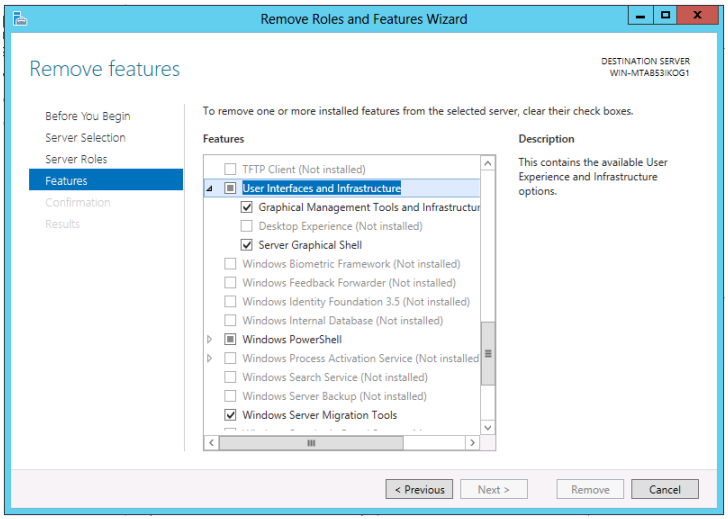

Figure 1-2 The User Interfaces And Infrastructure feature in the Remove Roles and Features Wizard.

7. Clear the Server Graphical Shell check box and click Next. The Confirm Removal Selections page opens.
8. Click Remove to open the Removal Progress page.
9. When the removal is complete, click Close.
10. Restart the server.

Using Features on Demand

During a Windows Server 2012 installation, the Setup program copies the files for all of the operating system components from the installation medium to a directory called *WinSxS*, the side-by-side component store. This enables administrators to activate any of the features included with Windows Server 2012 without having to supply an installation medium.

The drawback of this arrangement is that the WinSxS directory occupies a significant amount of disk space, much of which is, in many cases, devoted to data that will never be used.

With the increasing use of VMs to distribute server roles, enterprise networks often have more copies of the server operating system than ever before, and therefore more wasted disk space. In addition, the advanced storage technologies often used by today's server infrastructures, such as SANs and solid state drives (SSDs), are making that disk space more expensive.

Features on Demand, new to Windows Server 2012, is a third state for operating system features that enables administrators to conserve disk space by removing specific features, not only from operation, but also from the WinSxS directory.

This state is intended for features that administrators have no intention of installing on a particular server. If, for example, you want to disable the Server Graphical Shell feature in Windows Server 2012, to prevent Internet Explorer, Windows Explorer, and the desktop shell from running, and you want to remove the files that provide those features from the disk completely, you can do so with Features on Demand. By removing all the disk files for all of your unused features on all of your VMs, you can achieve substantial savings in disk space.

Features on Demand provides a third installation state for each of the features in Windows Server 2012. In previous versions of the operating system, features could be Enabled or Disabled. Windows Server 2012 provides the following three states:

- Enabled
- Disabled
- Disabled with payload removed

To implement this third state, you must use the Windows PowerShell Uninstall-WindowsFeature cmdlet, which now supports a new –Remove flag. Thus, the PowerShell command to disable the Server Graphical Shell and remove its source files from the WinSxS directory would be as follows:

```
Uninstall-WindowsFeature Server-Gui-Shell -Remove
```

Once you delete the source files for a feature from the WinSxS folder, they are not irretrievably gone. If you attempt to enable that feature again, the system will download it from Windows Update or, alternatively, retrieve it from an image file you specify using the –Source flag with the Install-WindowsFeature cmdlet. This enables you to retrieve the required files from a removable disk or from an image file on the local network. Administrators can also use Group Policy to specify a list of installation sources.

NOTE **Features on Demand**

This ability to retrieve source files for a feature from another location is the actual functionality to which the name Features on Demand is referring. Microsoft often uses this capability to reduce the size of updates downloaded from the Internet. Once the user installs the update, the program downloads the additional files required and completes the installation.

Upgrading servers

An in-place upgrade is the most complicated form of Windows Server 2012 installation. It is also the lengthiest, and the most likely to cause problems during its execution. Whenever possible, Microsoft recommends that administrators perform a clean installation, or migrate required roles, applications, and settings instead.

Although in-place upgrades often proceed smoothly, the complexity of the upgrade process and the large number of variables involved means that there are many things that can potentially go wrong. To minimize the risks involved, it is important for an administrator to take the upgrade process seriously, prepare the system beforehand, and have the ability to troubleshoot any problems that might arise. The following sections discuss these subjects in greater detail.

Upgrade paths

Upgrade paths for Windows Server 2012 are quite limited. In fact, it's easier to specify when you can perform an upgrade than when you can't. If you have a 64-bit computer running Windows Server 2008 or Windows Server 2008 R2, then you can upgrade it to Windows Server 2012, as long as you use the same operating system edition.

Windows Server 2012 does not support the following:

- Upgrades from Windows Server versions prior to Windows Server 2008
- Upgrades from pre-RTM editions of Windows Server 2012
- Upgrades from Windows workstation operating systems
- Cross-edition upgrades, such as Windows Server 2008 Enterprise Edition to Windows Server 2012 Datacenter Edition
- Cross-platform upgrades, such as 32-bit Windows Server 2008 to 64-bit Windows Server 2012
- Upgrades from any Itanium edition
- Cross-language upgrades, such as from Windows Server 2008, U.S. English to Windows Server 2012, French

In any of these cases, the Windows Setup program will not permit the upgrade to proceed.

Preparing to upgrade

Before you begin an in-place upgrade to Windows Server 2012, you should perform a number of preliminary procedures to ensure that the process goes smoothly and that the server data is protected

Consider the following before you perform any upgrade to Windows Server 2012:

- **Check hardware compatibility** Make sure that the server meets the minimum hardware requirements for Windows Server 2012.
- **Check disk space** Make sure that there is sufficient free disk space on the partition

where the old operating system is installed. During the upgrade procedure, sufficient disk space is needed to hold both operating systems simultaneously. After the upgrade is complete, you can remove the old files, freeing up some additional space.

- **Confirm that software is signed** All kernel-mode software on the server, including device drivers, must be digitally signed, or the software will not load. This can result in an aborted upgrade process, hardware failures after the upgrade is completed, or failure of the system to start after the upgrade. If you cannot locate a software update for the application or driver that is signed, then you should uninstall the application or driver before you proceed with the installation.

IMPORTANT **Disabling the driver signature**

If an unsigned driver prevents the computer from starting, you can disable the driver signature requirement by pressing F8 during the startup, selecting Advanced Boot Options, and then selecting Disable Driver Signature Enforcement.

- **Save mass storage drivers on removable media** If a manufacturer has supplied a separate driver for a device in your server, save the driver to a CD, a DVD, or a USB flash drive in either the media root directory or the /amd64 folder. To provide the driver during Setup, click Load Driver or press F6 on the disk selection page. You can browse to locate the driver or have Setup search the media.
- **Check application compatibility** The Setup program displays a Compatibility Report page that can point out possible application compatibility problems. You can sometimes solve these problems by updating or upgrading the applications. Create an inventory of the software products installed on the server and check the manufacturers' websites for updates, availability of upgrades, and announcements regarding support for Windows Server 2012. In an enterprise environment, you should test all applications for Windows Server 2012 compatibility, no matter what the manufacturer says, before you perform any operating system upgrades.
- **Ensure computer functionality** Make sure that Windows Server 2008 or Windows Server 2008 R2 is running properly on the computer before you begin the upgrade process. You must start an in-place upgrade from within the existing operating system, so you cannot count on Windows Server 2012 to correct any problems that prevent the computer from starting or running the Setup program.
- **Perform a full backup** Before you perform any upgrade procedure you should back up the entire system, or at the very least the essential data files. Your backup should include all data and configuration information that is necessary for your target computer to function. When you perform the backup, be sure to include the boot and system partitions and the system state data. Removable hard drives make this a simple process, even if there is not a suitable backup device in the computer.
- **Disable virus protection software** Virus protection software can make installation

much slower by scanning every file that is copied locally to your computer. If installed, you should disable this software before performing the upgrade.

- **Disconnect the UPS device** If you have an uninterruptible power supply (UPS) connected to your target computer, disconnect the cable before performing the upgrade. Setup automatically attempts to detect connected devices, and UPS equipment can cause issues with this process.
- **Purchase Windows Server 2012** Be sure to purchase the appropriate Windows Server 2012 edition for the upgrade, and have the installation disk and product key handy.

During the upgrade process, when the system restarts, the boot menu provides an option to roll back to the previous operating system version. However, once the upgrade is complete, this option is no longer available, and it is not possible to uninstall Windows Server 2012 and revert back to the old operating system version.

Migrating roles

Migration is the preferred method of replacing an existing server with one running Windows Server 2012. Unlike an in-place upgrade, a migration copies vital information from an existing server to a clean Windows Server 2012 installation.

When migrating, virtually all of the restrictions listed earlier in regard to upgrades do not apply. Using the Windows Server Migration Tools and migration guides supplied with Windows Server 2012, you can migrate data between servers under any of the following conditions:

- **Between versions** You can migrate data from any Windows Server version from Windows Server 2003 SP2 to Windows Server 2012. This includes migrations from one server running Windows Server 2012 to another.
- **Between platforms** You can migrate data from an x86- or x64-based server to an x64-based server running Windows Server 2012.
- **Between editions** You can migrate data between servers running different Windows Server editions.
- **Between physical and virtual instances** You can migrate data from a physical server to a virtual one, or the reverse.
- **Between installation options** You can migrate data from a server running Windows Server 2008 R2 to one running Windows Server 2012, even when one server is using the Server Core installation option and the other uses the Server with a GUI option.

Migration at the server level is different from any migrations you might have performed on workstation operating systems. Instead of performing a single migration procedure that copies all of the user data from the source to the destination computer at once, in a server migration you migrate roles or role services individually.

Windows Server 2012 includes a collection of migration guides that provide individualized instructions for each of the roles supported by Windows Server 2012. Some of the roles require the use of Windows Server Migration Tools; others do not.

Installing Windows Server Migration Tools

Windows Server Migration Tools is a Windows Server 2012 feature that consists of PowerShell cmdlets and help files that enable administrators to migrate certain roles between servers.

Before you can use the migration tools, however, you must install the Windows Server Migration Tools feature on the destination server running Windows Server 2012, and then copy the appropriate version of the tools to the source server.

Windows Server Migration Tools is a standard feature that you install on Windows Server 2012 using the Add Roles and Features Wizard in Server Manager, as shown in Figure 1-3, or the Install-WindowsFeature PowerShell cmdlet.

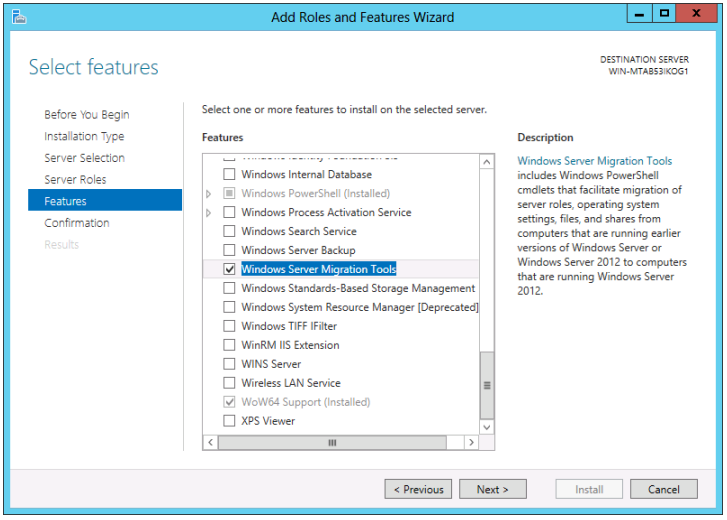

Figure 1-3 The Select Features page of the Add Roles and Features Wizard.

Using migration guides

Once you have installed the Windows Server Migration Tools on both the source and the destination servers, you can proceed to migrate data between the two.

Using the migration tools, administrators can migrate certain roles, features, shares, operating system settings, and other data from the source server to the destination server running Windows Server 2012. Some roles require the use of the migration tools, whereas other do not, having their own internal communication capabilities.

There is no single procedure for migrating all of the Windows Server roles, whether they have their own migration tools or not. Instead, Microsoft provides detailed migration guides for individual roles, and sometimes for individual role services within a role.

MORE INFO **Migration guides**

Up-to-date migration guides are available at the Windows Server Migration Portal at the Windows Server 2012 TechCenter (*http://technet.microsoft.com/en-us/library/jj134039*).

Thought experiment

Converting the Interface

In this thought experiment, apply what you've learned about this objective. You can find answers to these questions in the "Answers" section at the end of this chapter.

Ralph recently took delivery of a new server with Windows Server 2012 Datacenter Edition already installed with the full GUI option. Ralph wants to configure the system as a web server, using the absolute minimum of hardware resources. His first step is to use Server Manager to install the Web Server (IIS) role.

With this in mind, answer the following questions:

1. **What PowerShell command should Ralph use to convert the full GUI installation to Server Core?**
2. **What PowerShell command should Ralph use to remove the GUI installation files completely from the system?**

Objective summary

- Microsoft releases all of its operating systems in multiple editions, which provides consumers with varying price points and feature sets.
- When you select the Windows Server Core installation option, you get a stripped-down version of the operating system.
- The Minimal Server Interface is a setting that removes some of the most hardware-intensive elements from the graphical interface.
- An in-place upgrade is the most complicated form of Windows Server 2012 installation. It is also the lengthiest, and the most likely to cause problems during its execution. Whenever possible, Microsoft recommends that administrators perform a clean installation, or migrate required applications and settings instead.
- Migration is the preferred method of replacing an existing server with one running Windows Server 2012. Unlike an in-place upgrade, a migration copies vital information from an existing server to a clean Windows Server 2012 installation.

Objective review

Answer the following questions to test your knowledge of the information in this objective. You can find the answers to these questions and explanations of why each answer choice is

correct or incorrect in the "Answers" section at the end of this chapter.

1. Which of the following roles implement what can be classified as infrastructure services? (Choose all that apply)
   A. DNS
   B. Web Server (IIS)
   C. DHCP
   D. Remote Desktop Services
2. Which of the following is a valid upgrade path to Windows Server 2012?
   A. Windows Server 2003 Standard to Windows Server 2012 Standard
   B. Windows Server 2008 Standard to Windows Server 2012 Standard
   C. Windows Server 2008 R2 32-bit to Windows Server 2012 64-bit
   D. Windows 7 Ultimate to Windows Server 2012 Essentials
3. Which feature must you add to a Windows Server 2012 Server Core installation to convert it to the Minimal Server Interface?
   A. Graphical Management Tools and Infrastructure
   B. Server Graphical Shell
   C. Windows PowerShell
   D. Microsoft Management Console
4. What is the name of the directory where Windows stores all of the operating system modules it might need to install at a later time?
   A. Windows
   B. System32
   C. bin
   D. WinSxS
5. Which of the following are valid reasons why administrators might want to install their Windows Server 2012 servers using the Server Core option? (Choose all that apply)
   A. A Server Core installation can be converted to the full GUI without reinstalling the operating system.
   B. The PowerShell 3.0 interface in Windows Server 2012 includes more than 10 times as many cmdlets as PowerShell 2.0
   C. The new Server Manager in Windows Server 2012 makes it far easier to administer servers remotely.
   D. A Windows Server 2012 Server Core license costs significantly less than a full GUI license.

Objective 1.2: Configure servers

It seldom happens that a server is ready to perform all the tasks you have planned for it immediately after installation. Typically some postinstallation configuration is required, and further configuration changes might become necessary after the server is in service.

This objective covers how to:

- Configure Server Core
- Delegate administration
- Add and remove features in offline images
- Deploy roles on remote servers
- Convert Server Core to and from full GUI
- Configure services
- Configure NIC teaming

Completing postinstallation tasks

As part of the new emphasis on cloud-based services in Windows networking, Windows Server 2012 contains a variety of tools that have been overhauled to facilitate remote server management capabilities.

The new Server Manager, for example, is designed to enable administrators to fully manage Windows servers without ever having to interact directly with the server console, either physically or remotely. However, there are some tasks that administrators might have to perform immediately after the operating system installation that require direct access to the server console. These tasks might include the following:

- Configure the network connection
- Set the time zone
- Enable Remote Desktop
- Rename the computer
- Join a domain

Using GUI tools

In Windows Server 2012, the Properties tile in Server Manager, as shown in Figure 1-5, provides the same functionality as the Initial Configuration Tasks window in previous Windows Server versions. To complete any or all of the postinstallation configuration tasks on a GUI Windows Server 2012 installation, you can use the tools in the Properties tile, either by working directly at the server console or by using Remote Desktop to access the server from another computer.

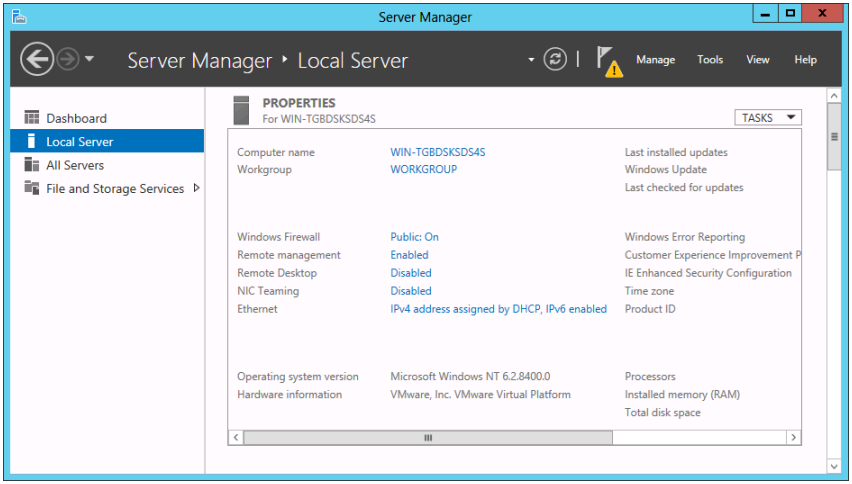

Figure 1-5 The Properties tile of the local server in Server Manager.

The Ethernet entry in the Properties tile specifies the current status of the computer's network interface. If there is an active Dynamic Host Configuration Protocol (DHCP) server on the network, the server will have already retrieved an IP address and other settings and used them to configure the interface. If there is no DHCP server on the network, or if you must configure the computer with a static IP address, click the Ethernet hyperlink to display the Network Connections window from the Control Panel. You can use this to open the Ethernet Properties sheet and the Internet Protocol Version 4 (TCP/IPv4) Properties sheet, where you can configure the TCP/IP client.

Accurate computer clock time is essential for Active Directory Domain Services communication. If the server is located in a time zone other than the default Pacific zone, click the Time Zone hyperlink to open the Date and Time dialog box, where you can correct the setting.

By default, Windows Server 2012 does not allow Remote Desktop connections. To enable them, click the Remote Desktop hyperlink to open the Remote tab of the System Properties sheet.

In a manual operating system installation, the Windows Setup program assigns a unique name beginning with WIN- to the computer. To change the name of the computer and join it to a domain, click the Computer Name hyperlink to open the System Properties sheet and click Change to open the Computer Name/Domain Changes dialog box.

If necessary, because of limited physical access to the server, you can confine this procedure to configuring the network connection and enabling Remote Desktop. Then, you can use Remote Desktop to connect to the server and configure everything else.

Using command-line tools

If you selected the Server Core option when installing Windows Server 2012, you can perform

the same postinstallation tasks from the command line. At the very minimum, you will have to rename the computer and join it to a domain. To do these, you can use the Netdom.exe program.

To rename a computer, run Netdom.exe with the following syntax, as shown in Figure 1-6:

```
netdom renamecomputer %ComputerName% /NewName: <NewComputerName>
```

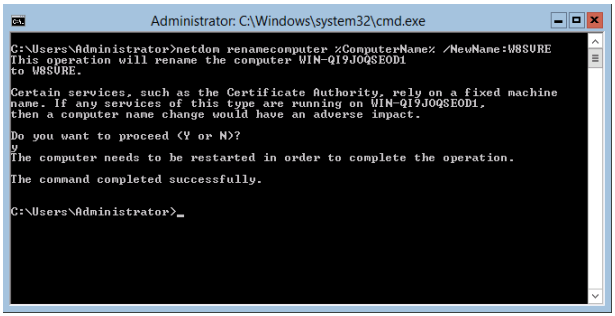

Figure 1-6 Renaming a computer from the command line.

To restart the computer as directed, use the following command:

```
shutdown /r
```

Then, to join the computer to a domain, use the following syntax:

```
netdom join %ComputerName% /domain: <DomainName> /userd: <UserName> /passwordd:*
```

In this command, the asterisk (*) in the /passwordd parameter causes the program to prompt you for the password to the user account you specified.

These commands assume that the computer's TCP/IP client has already been configured by a DHCP server. If this is not the case, you must configure it manually before you can join a domain. To assign a static IP address to a computer using Server Core, you can use the Netsh.exe program or the Windows Management Instrumentation (WMI) access provided by Windows PowerShell.

To enable Remote Desktop connections on the server, use the following cmdlet:

```
Set-RemoteDesktop -Enable
```

Converting between GUI and Server Core

In Windows Server 2012, you can convert a computer installed with the full GUI option to Server Core, and add the full GUI to a Server Core computer. This is a major improvement in the usefulness of Server Core over the version in Windows Server 2008 R2, in which you can only change the interface by reinstalling the entire operating system.

With this capability, administrators can install servers with the full GUI, use the graphical tools to perform the initial setup, and then convert them to Server Core to conserve system resources. If, at a later time, it becomes necessary, it is possible to reinstall the GUI components.

To convert a full GUI installation of Windows Server 2012 to Server Core using Server Manager, use the following procedure.

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.
2. From the Manage menu, select Remove Roles And Features. The Remove Roles and Features Wizard launches, displaying the Before You Begin page.
3. Click Next. The Select Destination Server page opens.
4. Select the server you want to convert to Server Core and click Next to open the Remove Server Roles page.
5. Click Next. The Remove Features page opens.
6. Scroll down in the list and expand the User Interfaces And Infrastructure feature, as shown in Figure 1-7.

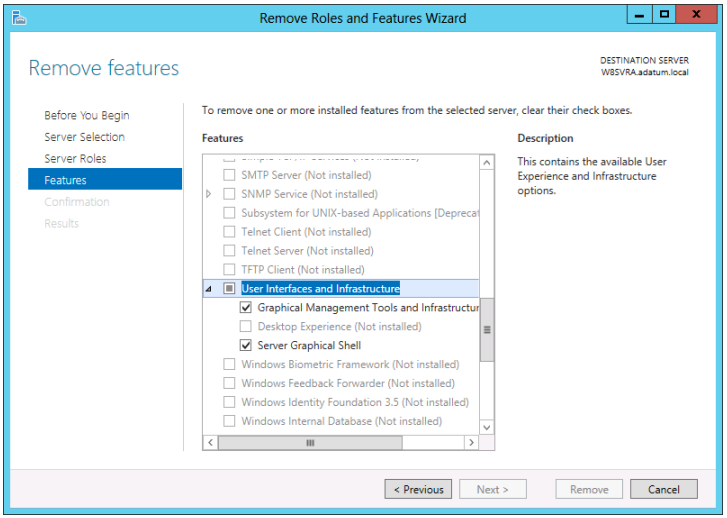

Figure 1-7 The Remove Features page in Server Manager.

7. Clear the check boxes for the following components:
   - Graphical Management Tools And Infrastructure
   - Server Graphical Shell
8. The Remove Features That Require Graphical Management Tools And Infrastructure dialog box opens, with a list of dependent features that must be uninstalled. Click Remove Features.
9. Click Next to open the Confirm Removal Selections page.
10. Select the Restart The Destination Server Automatically If Required check box and click Remove. The Removal Progress page opens as the wizard uninstalls the feature.
11. Click Close. When the removal is completed, the computer restarts.

To add the full GUI to a Server Core computer, you must use PowerShell to install the same features you removed in the previous procedure. To convert a Windows Server 2012 Server Core installation to the full GUI option, use the following PowerShell command:

```
Install-WindowsFeature Server-Gui-Mgmt-Infra,Server-Gui-Shell –Restart
```

To convert a full GUI server installation to Server Core, use the following command:

```
Uninstall-WindowsFeature Server-Gui-Mgmt-Infra,Server-Gui-Shell -Restart
```

Configuring NIC teaming

NIC teaming is a new feature in Windows Server 2012 that enables administrators to combine the bandwidth of multiple network interface adapters, providing increased performance and fault tolerance. Virtualization enables administrators to separate vital network functions on different systems without having to purchase a separate physical computer for each one. However, one of the drawbacks of this practice is that a single server hosting multiple VMs is still a single point of failure for all of them. A single malfunctioning network adapter, a faulty switch, or even an unplugged cable can bring down a host server and all of its VMs with it.

NIC teaming, also called *bonding, balancing, and aggregation*, is a technology that has been available for some time, but it was always tied to specific hardware implementations. The NIC teaming capability in Windows Server 2012 is hardware independent, and enables you to combine multiple physical network adapters into a single interface. The results can include increased performance through the combined throughput of the adapters and protection from adapter failures by dynamically moving all traffic to the functioning NICs.

NIC teaming in Windows Server 2012 supports two modes:

- **Switch Independent Mode** All of the network adapters are connected to different switches, providing alternative routes through the network.
- **Switch Dependent Mode** All of the network adapters are connected to the same switch, providing a single interface with their combined bandwidth.

In Switch Independent Mode, you can choose between two configurations. The active/active configuration leaves all of the network adapters functional, providing increased throughput. If one adapter fails, all of the traffic is shunted to the remaining adapters. In the active/standby configuration, one adapter is left offline to function as a failover in the event the active adapter fails. In active/active mode, an adapter failure causes a performance reduction; in active/standby mode, the performance remains the same before and after an adapter failure.

In Switch Dependent Mode, you can choose static teaming, a generic mode that balances the traffic between the adapters in the team, or you can opt to use the Link Aggregation Control Protocol defined in IEEE 802.3ax, assuming that your equipment supports it.

There is one significant limitation to NIC teaming. If your traffic consists of large TCP sequences, such as a Hyper-V live migration, the system will avoid using multiple adapters for

those sequences to minimize the number of lost and out-of-order TCP segments. You will therefore not realize any performance increase for large file transfers using TCP.

You can create and manage NIC teams using Server Manager or Windows PowerShell. To create a NIC team using Server Manager, follow these steps.

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.
2. In the navigation pane, click Local Server. The Local Server home page appears.
3. In the Properties tile, click NIC Teaming. The NIC Teaming window opens, as shown in Figure 1-8.

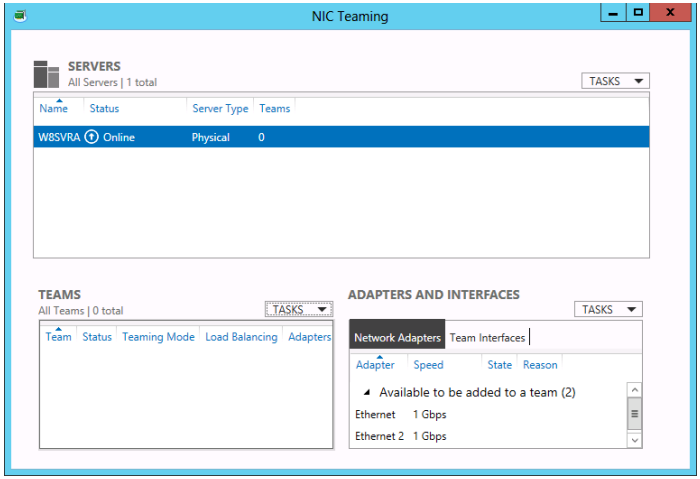

Figure 1-8 The NIC Teaming window in Server Manager.

4. In the Teams tile, click Tasks and select New Team to open the New Team page.
5. Click the Additional Properties arrow to expand the window, as shown in Figure 1-9.

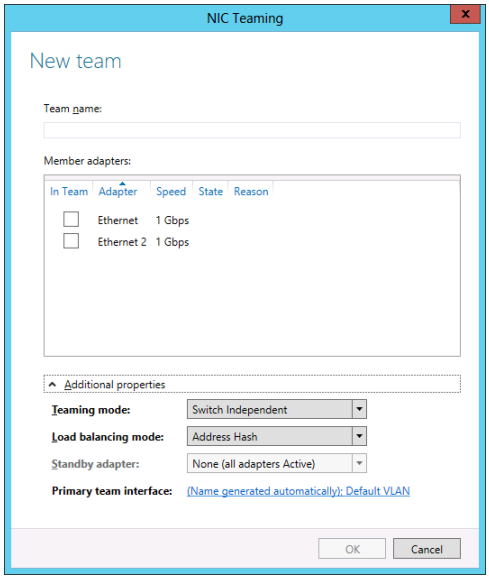

Figure 1-9 The New Team page in Server Manager.

6. In the Team Name text box, type the name you want to assign to the team.
7. In the Member Adapters box, select the network adapters you want to add to the team.
8. In the Teaming Mode drop-down list, select one of the following options:
    - Static Teaming
    - Switch Independent
    - LACP
9. In the Load Balancing Mode drop-down list, select one of the following options:
    - Address Hash
    - Hyper-V Port
10. If you selected Switch Independent for the Teaming Mode value, in the Standby Adapter drop-down list, select one of the adapters you added to the team to function as the offline standby.
11. Click OK. The new team appears in the Teams tile, as shown in Figure 1-10.

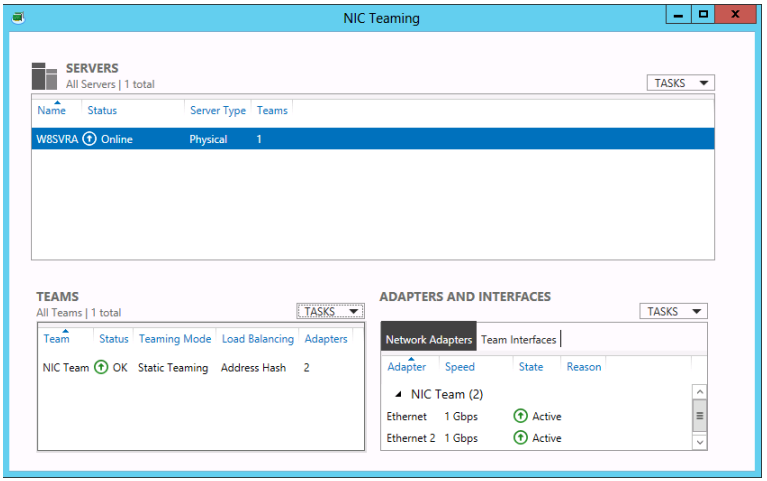

Figure 1-10 A new NIC team in the NIC Teaming window in Server Manager.

Once you have created a NIC team, the NIC Teaming window enables you to monitor the status of the team and the team interface you have created. The team itself and the individual adapters all have status indicators that inform you if an adapter goes offline.

If this does occur, the indicator for the faulty adapter immediately switches to disconnected, as shown in Figure 1-11, and depending on which teaming mode you chose, the status of the other adapter might change as well.

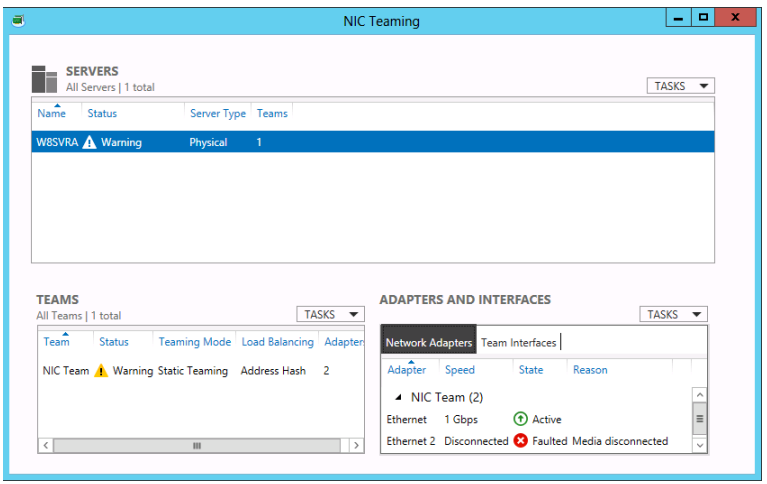

Figure 1-11 A NIC team with a failed adapter.

Using Server Manager

The Server Manager tool in Windows Server 2012 is a completely new application that is the first and most obvious evidence of a major paradigm shift in Windows Server administration.

In previous versions of Windows Server, an administrator wanting to install a role using graphical controls had to work at the server console by either physically sitting at the keyboard or connecting to it using Remote Desktop Services (formerly Terminal Services). By contrast, the Windows Server 2012 Server Manager can install roles and features to any server on the network, and even to multiple servers or groups of servers at once.

Adding servers

The primary difference between the Windows Server 2012 Server Manager and previous versions is the ability to add and manage multiple servers at once. When you log on to a GUI installation of Windows Server 2012 with an administrative account, Server Manager loads automatically, displaying the Welcome tile.

The Server Manager interface consists of a navigation pane on the left containing icons representing various views of server resources. Selecting an icon displays a home page in the right pane, which consists of a number of tiles containing information about the resource. The Dashboard page, which appears by default, contains, in addition to the Welcome tile, thumbnails that summarize the other views available in Server Manager, as shown in Figure 1-12. These other views include a page for the Local Server, one for All Servers, and others for server groups and role groups.

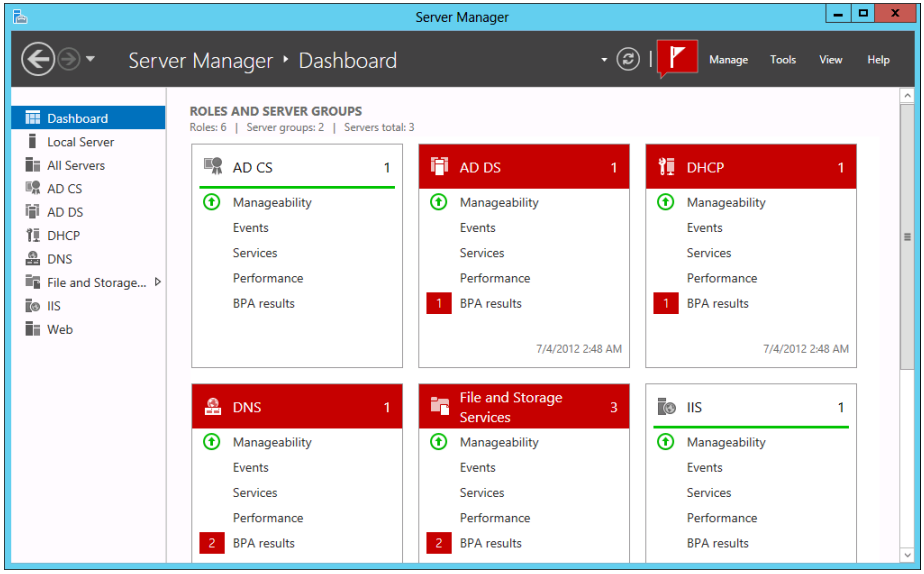

Figure 1-12 Dashboard thumbnails in Server Manager.

Although only the local server appears in Server Manager when you first run it, you can add other servers, enabling you to manage them together. The servers you add can be physical or virtual, and can be running any version of Windows Server since Windows Server 2003. After you add servers to the interface, you can create groups containing collections of servers, such

as the servers at a particular location or those performing a particular function. These groups appear in the navigation pane, enabling you to administer them as a single entity.

To add servers in Server Manager, use the following procedure.

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.
2. In the navigation pane, click All Servers. The All Servers home page appears, as shown in Figure 1-13.

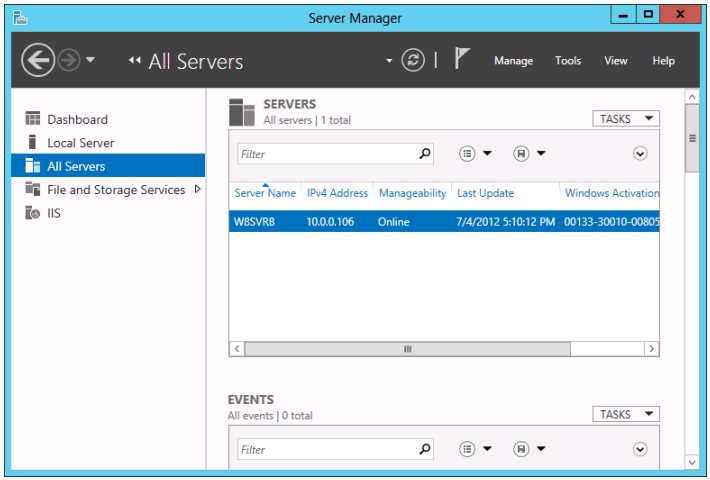

Figure 1-13 The All Servers home page in Server Manager.

3. From the Manage menu, select Add Servers. The Add Servers dialog box opens, as shown in Figure 1-14.

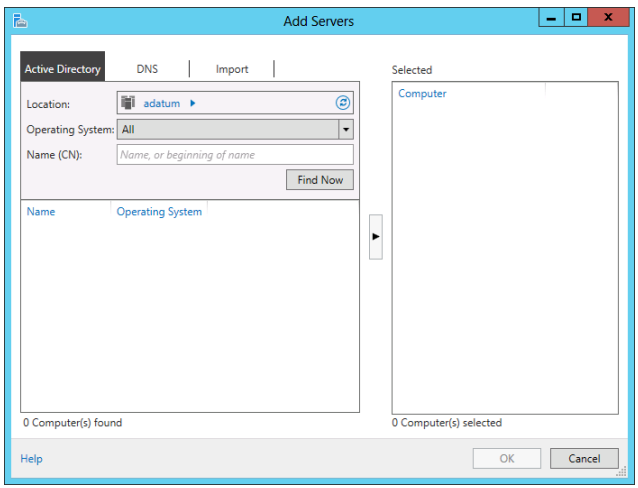

Figure 1-14 The Add Servers dialog box in Server Manager.

4. Select one of the following tabs to specify how you want to locate servers to add:
    - **Active Directory** Enables you to search for computers running specific operating systems in specific locations in an Active Directory Domain Services domain
    - **DNS** Enables you to search for servers in your currently configured Domain Name System (DNS) server
    - **Import** Enables you to supply a text file containing the names of the servers you want to add
5. Initiate a search or upload a text file to display a list of available servers, as shown in Figure 1-15.

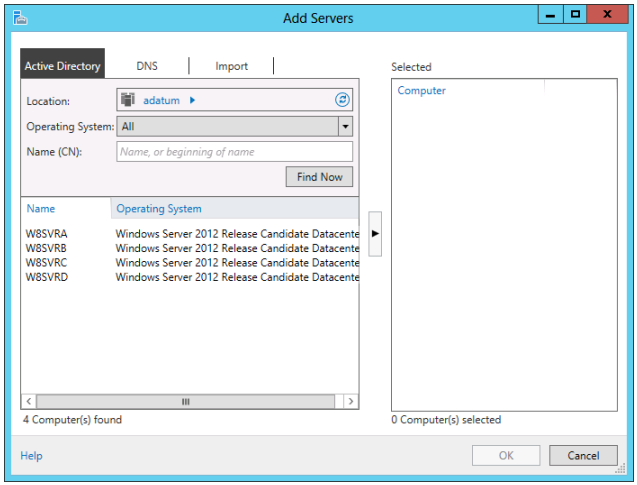

Figure 1-15 Searching for servers in Server Manager.

6. Select the servers you want to add and click the right arrow button to add them to the Selected list.
7. Click OK. The servers you selected are added to the All Servers home page.

Once you have added remote servers to the Server Manager interface, you can access them in a variety of ways, including the standard MMC administrative tools, the Computer Management console, and a remote PowerShell session.

For administrators of enterprise networks, it might be necessary to add a large number of servers to Server Manager. To avoid having to work with a long scrolling list of servers, you can create server groups, based on server locations, functions, or any other organizational paradigm.

Adding roles and features

The Server Manager program in Windows Server 2012 combines what used to be separate wizards for adding roles and features into one, the Add Roles and Features Wizard. Once you

add multiple servers to the Server Manager interface, they are integrated into the Add Roles and Features Wizard, so you can deploy roles and features to any of your servers.

To install roles and features using Server Manager, use the following procedure.

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.
2. From the Manage menu, select Add Roles And Features. The Add Roles and Features Wizard starts, displaying the Before You Begin page.
3. Click Next to open the Select Installation Type page, as shown in Figure 1-16.

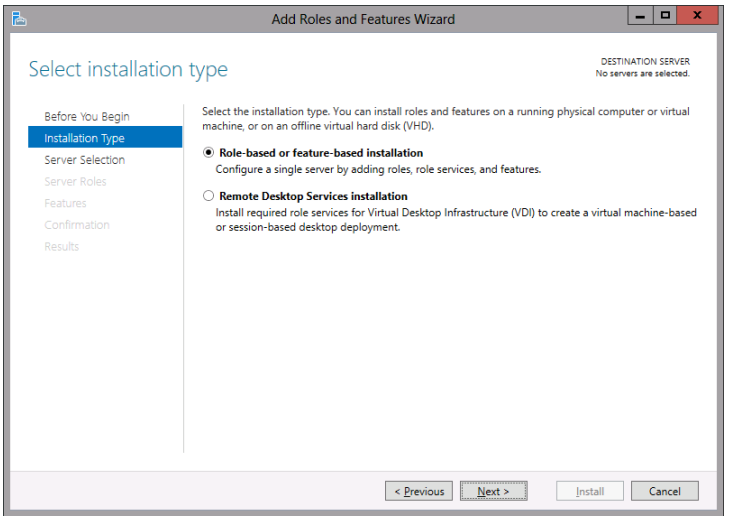

Figure 1-16 The Select Installation Type page in the Add Roles and Features Wizard.

4. Leave the Role-Based Or Feature-Based Installation option selected and click Next. The Select Destination Server page opens, as shown in Figure 1-17.

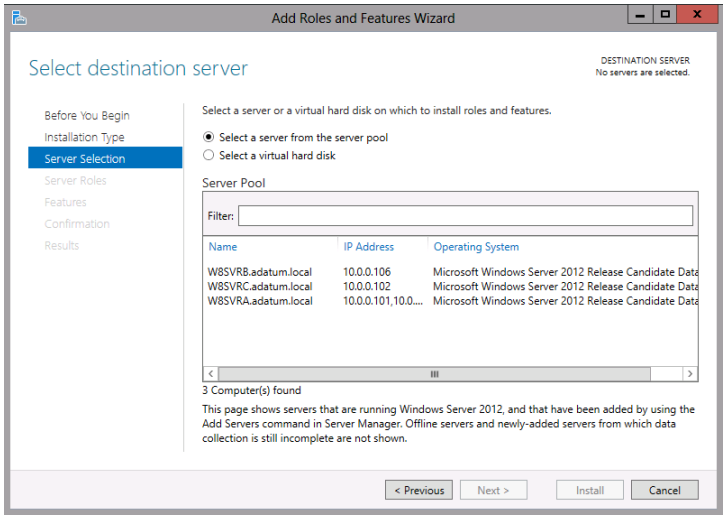

Figure 1-17 The Select Destination Server page in the Add Roles and Features Wizard.

5. Select the server on which you want to install the roles or features. If the server pool contains a large number of servers, you can use the Filter text box to display a subset of the pool based on a text string. When you have selected the server, click Next. The Select Server Roles page opens, as shown in Figure 1-18.

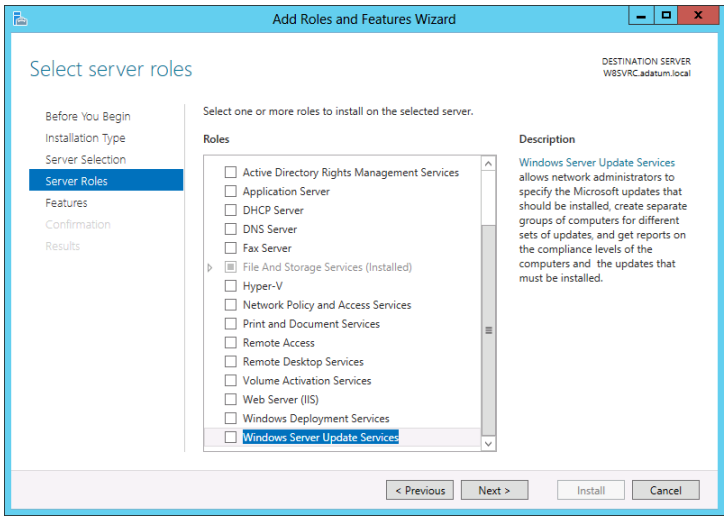

Figure 1-18 The Select Server Roles page in the Add Roles and Features Wizard.

Note **INSTALLING COMPONENTS TO MULTIPLE SERVERS**

Although you can use the Add Roles and Features Wizard to install components to any server you have added to Server Manager, you cannot use it to install components to multiple servers at once. You can, however, do this using Windows PowerShell.

6. Select the role or roles you want to install on the selected server. If the roles you select have other roles or features as dependencies, an Add Features That Are Required dialog box appears.

Note **SELECTING ALL ROLES AND FEATURES**

Unlike previous versions of Server Manager, the Windows Server 2012 version enables you to select all of the roles and features for a particular server configuration at once, rather than making you run the wizard multiple times.

7. Click Add Features to accept the dependencies, and then click Next to open the Select Features page, as shown in Figure 1-19.

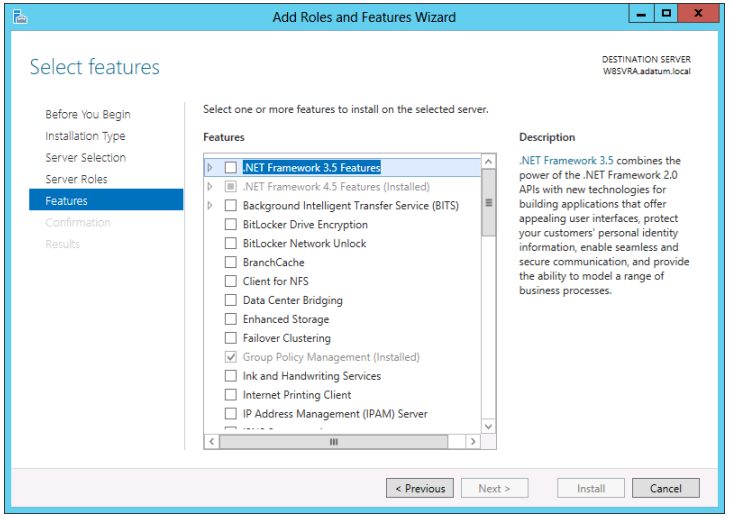

Figure 1-19 The Select Features page in the Add Roles and Features Wizard.

8. Select any features you want to install in the selected server and click Next. Dependencies might appear for your feature selections as well.
9. The wizard then displays pages specific to the roles or features you have chosen. Most roles have a Select Role Services page, on which you can select which elements of the role you want to install. Complete each of the role- or feature-specific pages and click Next. A Confirm Installation Selections page opens.
10. You can select from the following optional functions:
    - **Restart The Destination Server Automatically If Desired** Causes the server to restart automatically when the installation is completed, if the selected roles and features require it
    - **Export Configuration Settings** Creates an XML script documenting the procedures performed by the wizard, which you can use to install the same configuration on another server using Windows PowerShell

- **Specify An Alternate Source Path** Specifies the location of an image file containing the software needed to install the selected roles and features

11. Click Install to open the Installation Progress page. Depending on the roles and features installed, the wizard might display hyperlinks to the tools needed to perform required postinstallation tasks. When the installation is completed, click Close to complete the wizard.

NOTE **Using an exported configuration file**

To use an exported configuration file to install roles and features on another computer running Windows Server 2012, use the following command in a Windows PowerShell session with elevated privileges:

```
Install-WindowsFeature –ConfigrationFilePath <ExportedConfig.xml>
```

Once you install roles on your servers, the roles appear as icons in the navigation pane. These icons actually represent role groups. Each role group contains all the instances of that role found on any of your added servers. You can therefore administer the role across all of the servers on which you have installed it.

Deploying roles to VHDs

In addition to installing roles and features to servers on the network, Server Manager also enables administrators to install them to VMs that are currently in an offline state. For example, you might have an offline web server VM stored on a backup host server, in case the computer hosting your main web server VMs should fail. Server Manager enables you to select a virtual hard disk (VHD) file and install or remove roles and features without having to start the VM.

To install roles or features to an offline VHD file, use the following procedure.

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.
2. From the Manage menu, select Add Roles And Features. The Add Roles and Features Wizard starts, displaying the Before You Begin page.
3. Click Next to open the Select Installation Type page.
4. Leave the Role-Based Or Feature-Based Installation option selected and click Next. The Select Destination Server page opens.
5. Select the Select A Virtual Hard Disk option. A Virtual Hard Disk text box appears at the bottom of the page.
6. In the Virtual Hard Disk text box, type or browse to the location of the VHD file you want to modify.
7. In the Server Pool box, select the server that the wizard should use to mount the VHD file, as shown in Figure 1-20, and click Next. The Select Server Roles page opens.

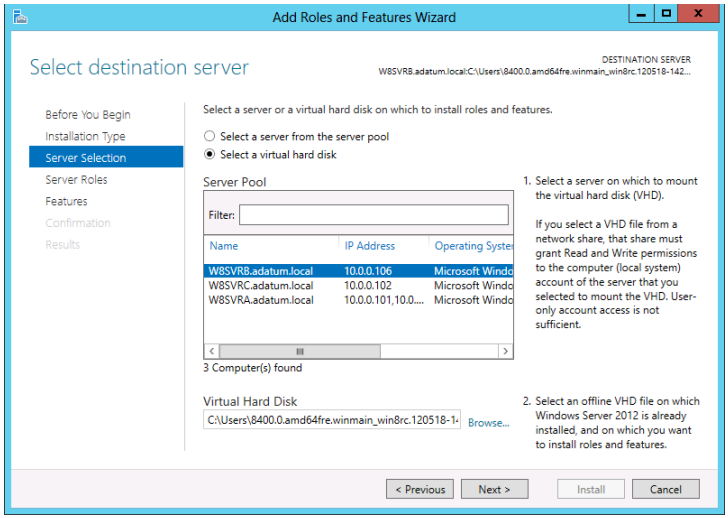

Figure 1-20 The Select Destination Server page in the Add Roles and Features Wizard.

Note **WHAT IT MEANS TO MOUNT THE VHD FILE**

The wizard must mount the VHD file on the server you select to look inside and determine which roles and features are already installed and which are available for installation. Mounting a VHD file only makes it available through the computer's file system; it is not the same as starting the VM using the VHD.

8. Select the role or roles you want to install on the selected server, adding the required dependencies, if necessary, and click Next. The Select Features page opens.
9. Select any features you want to install on the selected server and click Next. Dependencies might appear for your feature selections as well.
10. The wizard then displays pages specific to the roles or features you have chosen, enabling you to select role services and configure other settings. Complete each of the role- or feature-specific pages and click Next. A Confirmation page opens.
11. Click Install. The Installation Progress page opens. When the installation is completed, click Close to dismount the VHD and complete the wizard.

Configuring services

Most Windows Server roles and many of the features include services, which are programs that run continuously in the background, typically waiting for a client process to send a request to them. Server Manager provides access to services running on servers all over the network.

When you first look at the Local Server home page in Server Manager, one of the tiles you find there is the Services tile, shown in Figure 1-21. This tile lists all of the services installed on the server and specifies the operational status and their Start Types. When you right-click a

service, the shortcut menu provides controls that enable you to start, stop, restart, pause, and resume the service.

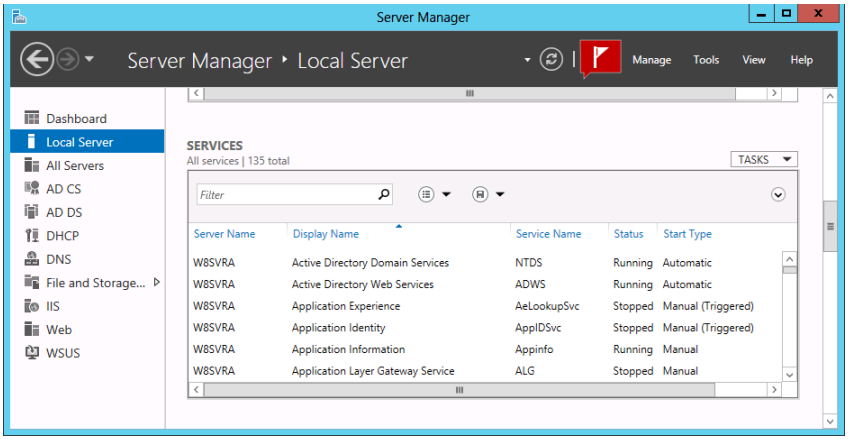

Figure 1-21 The Services tile in Server Manager.

The Services tile in the Server Manager display is not unlike the traditional Services snap-in for MMC found in previous versions of Windows Server. However, although you can start and stop a service in Server Manager, you cannot modify its Start Type, which specifies whether the service should start automatically with the operating system. For that you must use the Services MMC snap-in.

Another difference of the Services tile in Windows Server 2012 Server Manager is that this tile appears in many locations throughout Server Manager, and in each place it displays a list of services for a different context. This is a good example of the organizational principle of the new Server Manager. The same tools, repeated in many places, provide a consistent management interface to different sets of components.

For example, when you select the All Servers icon in the navigation pane, you see first the Servers tile, as usual, containing all of the servers you have added to the Server Manager console. When you select some or all of the servers and scroll down to the Services tile, you see the same display as before, except that it now contains all of the services for all of the computers you selected. This enables you to monitor the services on all of the servers at once.

In the same way, when you select one of the role group icons, you can select from the servers running that role and the Services tile will contain only the services associated with that role for the servers you selected.

To manipulate other server configuration settings, you must use the Services snap-in for MMC as mentioned earlier. However, you can launch that, and many other snap-ins, using Server Manager.

After selecting a server from the Servers pane in any group home page, click the Tools menu to display a list of the server-specific utilities and MMC snap-ins, including the Services snap-in, directed at the selected server.

Delegating server administration

As networks grow larger in size, so do the numbers of administrative tasks there are to perform on a regular basis, and so do the IT staffs that are needed to perform them. Delegating administrative tasks to specific individuals is a natural part of enterprise server management, as is assigning those individuals the permissions they need—and only the permissions they need—to perform those tasks.

NOTE **Delegating Privileges**

For information on delegating printer privileges, see Objective 2.2, "Configure Print and Document Services." For information on delegating administrative control via Active Directory, see Objective 5.3, "Create and Manage Active Directory Groups and Organizational Units." On smaller networks, with small IT staffs, it is not uncommon for task delegation to be informal, and for everyone in the IT department to have full access to the entire network. However, on larger networks, with larger IT staffs, this becomes increasingly impractical. For example, you might want the newly hired junior IT staffers to be able to create new user accounts, but you do not want them to be able to redesign your Active Directory tree or change the CEO's password.

Delegation, therefore, is the practice by which administrators grant other users a subset of the privileges that they themselves possess. As such, delegation is as much a matter of restricting permissions as it is of granting them. You want to provide individuals with the privileges they need, while protecting sensitive information and delicate infrastructure.

Thought experiment

Configuring Server Core

In this thought experiment, apply what you've learned about this objective. You can find answers to these questions in the "Answers" section at the end of this chapter.

Deepak is an IT technician who has been assigned the task of configuring a new server running Windows Server 2012 Server Core, called ServerA, which is to be shipped out to the company's branch office. The server must be configured to function as a file server with support for the Distributed File Sytstem (DFS), a print server with support for Internet and UNIX printing, and a secured, intranet web/FTP server for domain users.

With this in mind, answer the following questions:

1. **What PowerShell command should Deepak use to install the required roles on the servers?**
2. **What PowerShell command can Deepak use to obtain the short names for the roles used by PowerShell?**

3. **List the commands that Deepak must run on the new server to install the required modules.**

Objective summary

- The new Server Manager is designed to enable administrators to fully manage Windows servers without ever having to interact directly with the server console, either physically or remotely.
- There are some tasks that administrators might have to perform immediately after the operating system installation that require direct access to the server console.
- If you selected the Server Core option when installing Windows Server 2012, you can perform postinstallation tasks from the command line.
- In Windows Server 2012, the Properties tile in Server Manager provides the same functionality as the Initial Configuration Tasks window in previous versions.
- In Windows Server 2012, you can convert a computer installed with the full GUI option to Server Core, and add the full GUI to a Server Core computer.
- NIC teaming is a new feature in Windows Server 2012 that enables administrators to combine the bandwidth of multiple network interface adapters, providing increased performance and fault tolerance.
- For administrators of enterprise networks, it might be necessary to add a large number of servers to Server Manager. To avoid having to work with a long scrolling list of servers, you can create server groups, based on server locations, functions, or any other organizational paradigm.
- In addition to installing roles and features to servers on the network, Server Manager also enables administrators to install them to VMs that are currently in an offline state.

Objective review

Answer the following questions to test your knowledge of the information in this objective. You can find the answers to these questions and explanations of why each answer choice is correct or incorrect in the "Answers" section at the end of this chapter.

1. Which features must you remove from a full GUI installation of Windows Server 2012 to convert it to a Server Core installation? (Choose all that apply)
   A. Windows Management Instrumentation
   B. Graphical Management Tools and Infrastructure
   C. Desktop Experience
   D. Server Graphical Shell
2. Which of the following NIC teaming modes provides fault tolerance and bandwidth aggregation?

A. Hyper-V live migration
B. Switch Independent Mode
C. Switch Dependent Mode
D. Link Aggregation Control Protocol

3. Which of the following command-line tools do you use to join a computer to a domain?
   A. Net.exe
   B. Netsh.exe
   C. Netdom.exe
   D. Ipconfig.exe
4. Which of the following statements about Server Manager is not true?
   A. Server Manager can deploy roles to multiple servers at the same time.
   B. Server Manager can deploy roles to VHDs while they are offline.
   C. Server Manager can install roles and features at the same time.
   D. Server Manager can install roles and features to any Windows Server 2012 server on the network.
5. Which of the following operations can you not perform on a service using Server Manager? (Choose all that apply)
   A. Stop a running service
   B. Start a stopped service
   C. Disable a service
   D. Configure a service to start when the computer starts

Objective 1.3: Configure local storage

Although Windows Server 2012 is designed to take advantage of remote storage and cloud computing, the configuration of local storage remains an important consideration.

This objective covers how to:

- Design storage spaces
- Configure basic and dynamic disks
- Configure MBR and GPT disks
- Manage volumes
- Create and mount VHDs
- Configure storage pools and disk pools

Planning server storage

A Windows server can conceivably perform its tasks using the same type of storage as a workstation, that is, one or more standard hard disks connected to a standard drive interface such as Serial ATA (SATA). However, the I/O burdens of a server are quite different from those of a workstation, and a standard storage subsystem can easily be overwhelmed by file requests from dozens or hundreds of users. In addition, standard hard disks offer no fault tolerance and are limited in their scalability.

A variety of storage technologies are better suited for server use, and the process of designing a storage solution for a server depends on several factors, including the following:

- The amount of storage the server needs
- The number of users who will be accessing the server at the same time
- The sensitivity of the data to be stored on the server
- The importance of the data to the organization

The following sections examine these factors and the technologies you can choose when creating a plan for your network storage solutions.

How many servers do I need?

When is one big file server preferable to several smaller ones? One of the most frequently asked questions when planning a server deployment is whether it is better to use one large server or several smaller ones. In the past, you might have considered the advantages and disadvantages of using one server to perform several roles versus distributing the roles among several smaller servers, but today, the emphasis is on virtualization, which means that although you might have many VMs running different roles, they could all be running on a single large physical server.

If you are considering large physical servers, or if your organization's storage requirements are extremely large, you must also consider the inherent storage limitations of Windows Server 2012.

The number of sites your enterprise network encompasses and the technologies you use to provide network communication among those sites can also affect your plans. If, for example, your organization has branch offices scattered around the world and uses relatively expensive wide area network (WAN) links to connect them, it would probably be more economical to install a server at each location than to have all of your users access a single server using the WAN links.

Within each site, the number of servers you need can depend on how often your users work with the same resources and how much fault tolerance and high availability you want to build into the system. For example, if each department in your organization typically works with its own applications and documents and rarely needs access to those of other departments, deploying individual servers to each department might be preferable. If

everyone in your organization works with the same set of resources, centralized servers might be a better choice.

Estimating storage requirements

The amount of storage space you need in a server depends on a variety of factors, not just the initial requirements of your applications and users. In the case of an application server, start by allocating the amount of space needed for the application files themselves, plus any other space the application needs, as recommended by the developer. If users will be storing documents on the server, then allocate a specific amount of space for each user the server will support. Then, factor in the potential growth of your organization and your network, both in terms of additional users and additional space required by each user, and of the application itself, in terms of data files and updates.

Using Storage Spaces

Windows Server 2012 includes a new disk virtualization technology called *Storage Spaces*, which enables a server to concatenate storage space from individual physical disks and allocate that space to create virtual disks of any size supported by the hardware.

This type of virtualization is a feature often found in SAN and NAS technologies, which require a substantial investment in specialized hardware and administrative skill. Storage Spaces provides similar capabilities using standard direct-attached disk drives or simple external "Just a Bunch of Disk" (JBOD) arrays.

Storage Spaces uses unallocated disk space on server drives to create storage pools. A *storage pool* can span multiple drives invisibly, providing an accumulated storage resource that administrators can expand or reduce as needed by adding disks to or removing them from the pool. Using the space in the pool, administrators can create *virtual disks* of any size.

Once created, a virtual disk behaves just like a physical disk, except that the actual bits might be stored on any number of physical drives in the system. Virtual disks can also provide fault tolerance by using the physical disks in the storage pool to hold mirrored or parity data.

After creating a virtual disk, you can create volumes on it, just as you would on a physical disk. Server Manager provides the tools needed to create and manage storage pools and virtual disks, as well as the ability to create volumes and file system shares, with some limitations.

Understanding Windows disk settings

When you install Windows Server 2012 on a computer, the setup program automatically performs all of the preparation tasks for the primary hard disk in the system. However, when you install additional hard disk drives on a server, or when you want to use settings that differ from the system defaults, you must perform the following tasks manually:

- **Select a partitioning style** Windows Server 2012 supports two hard disk partition styles: the master boot record (MBR) partition style and the GUID (globally unique

identifier) partition table (GPT) partition style. You must choose one of these partition styles for a drive; you cannot use both.

- **Select a disk type** Windows Server 2012 supports two disk types: basic and dynamic. You cannot use both types on the same disk drive, but you can mix disk types in the same computer.
- **Divide the disk into partitions or volumes** Although many professionals use the terms partition and volume interchangeably, it is correct to refer to partitions on basic disks, and volumes on dynamic disks.
- **Format the partitions or volumes with a file system** Windows Server 2012 supports the NTFS file system, the FAT file system (including the FAT16, FAT32, and exFAT variants), and the new ReFS file system.

The following sections examine the options for each of these tasks.

Selecting a partition style

The term *partition style* refers to the method that Windows operating systems use to organize partitions on the disk. Servers running Windows Server 2012 computers can use either one of the following two hard disk partition styles:

- **Master Boot Record (MBR)** The MBR partition style has been around since before Windows and is still a common partition style for x86-based and x64-based computers.
- **GUID Partition Table (GPT)** GPT has existed since the late 1990s, but no x86 version of Windows prior to Windows Server 2008 and Windows Vista supports it. Today, most operating systems support GPT, including Windows Server 2012.

Before Windows Server 2008 and Windows Vista, all x86-based Windows computers used only the MBR partition style. Computers based on the x64 platform could use either the MBR or GPT partition style, as long as the GPT disk was not the boot disk.

Unless the computer's architecture provides support for an Extensible Firmware Interface (EFI)–based boot partition, it is not possible to boot from a GPT disk. If this is the case, the system drive must be an MBR disk, and you can use GPT only on separate nonbootable disks used for data storage.

When you use Server Manager to initialize a disk in Windows Server 2012, it uses the GPT partition style, whether it is a physical or a virtual disk. There are no controls in Server Manager supporting MBR, although it does display the partition style in the Disks tile.

Understanding disk types

Most personal computers use basic disks because they are the easiest to manage. Advanced volume types require the use of dynamic disks. A *basic disk* using the MBR partition style uses primary partitions, extended partitions, and logical drives to organize data. A primary partition appears to the operating system as though it is a physically separate disk and can host an

operating system, in which case it is known as the active partition.

When you work with basic MBR disks in Windows Server 2012, you can create three volumes that take the form of primary partitions. When you create the fourth volume, the system creates an extended partition, with a logical drive on it, of the size you specified. If there is still free space left on the disk, the system allocates it to the extended partition, as shown in Figure 1-22, where you can use it to create additional logical drives.

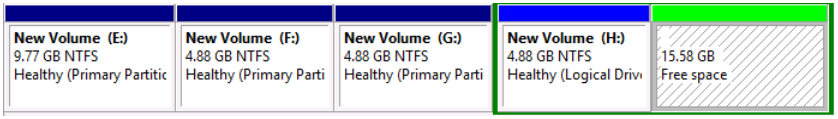

Figure 1-22 Primary and extended partitions on a basic disk using MBR.

When you select the GPT partition style, the disk still appears as a basic disk, but you can create up to 128 volumes, each of which appears as a primary partition, as shown in Figure 1-23. There are no extended partitions or logical drives on GPT disks.

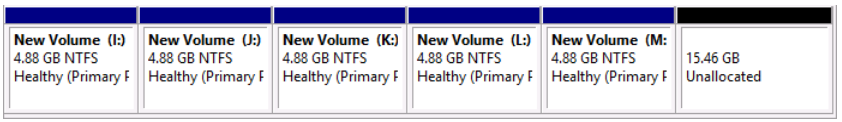

Figure 1-23 Primary partitions on a basic disk using GPT.

The alternative to using a basic disk is to convert it to a *dynamic disk*. The process of converting a basic disk to a dynamic disk creates a single partition that occupies the entire disk. You can then create an unlimited number of volumes out of the space in that partition. Dynamic disks support several different types of volumes, as described in the next section.

Understanding volume types

A dynamic disk can contain an unlimited number of volumes that function much like primary partitions on a basic disk, but you cannot mark an existing dynamic disk as active. When you create a volume on a dynamic disk using the Disk Management snap-in in Windows Server 2012, you choose from the following five volume types:

- **Simple volume** Consists of space from a single disk. After you have created a simple volume, you can extend it to multiple disks to create a spanned or striped volume, as long as it is not a system volume or boot volume. You can also extend a simple volume into any adjacent unallocated space on the same disk or, with some limitations, shrink the volume by deallocating any unused space in the volume.
- **Spanned volume** Consists of space from 2 to 32 physical disks, all of which must be dynamic disks. A spanned volume is essentially a method for combining the space from multiple dynamic disks into a single large volume. Windows Server 2012 writes to the spanned volume by filling all of the space on the first disk and then fills each of the additional disks in turn. You can extend a spanned volume at any time by adding

disk space. Creating a spanned volume does not increase the disk's read/write performance, nor does it provide fault tolerance. In fact, if a single physical disk in the spanned volume fails, all of the data in the entire volume is lost.

- **Striped volume** Consists of space from 2 to 32 physical disks, all of which must be dynamic disks. The difference between a striped volume and a spanned volume is that in a striped volume, the system writes data one stripe at a time to each successive disk in the volume. Striping provides improved performance because each disk drive in the array has time to seek the location of its next stripe while the other drives are writing. Striped volumes do not provide fault tolerance, however, and you cannot extend them after creation. If a single physical disk in the striped volume fails, all of the data in the entire volume is lost.
- **Mirrored volume** Consists of an identical amount of space on two physical disks, both of which must be dynamic disks. The system performs all read and write operations on both disks simultaneously, so they contain duplicate copies of all data stored on the volume. If one of the disks fails, the other continues to provide access to the volume until the failed disk is repaired or replaced.
- **RAID-5 volume** Consists of space on three or more physical disks, all of which must be dynamic. The system stripes data and parity information across all of the disks so that if one physical disk fails, the missing data can be re-created using the parity information on the other disks. RAID-5 volumes provide improved read performance, because of the disk striping, but write performance suffers, due to the need for parity calculations.

Understanding file systems

To organize and store data or programs on a hard drive, you must install a file system. A file system is the underlying disk drive structure that enables you to store information on your computer. You install file systems by formatting a partition or volume on the hard disk.

In Windows Server 2012, five file system options are available: NTFS, FAT32, exFAT, FAT (also known as FAT16), and ReFS. NTFS is the preferred file system for a server; the main benefits are improved support for larger hard drives than FAT and better security in the form of encryption and permissions that restrict access by unauthorized users.

Because the FAT file systems lack the security that NTFS provides, any user who gains access to your computer can read any file without restriction. Additionally, FAT file systems have disk size limitations: FAT32 cannot handle a partition greater than 32 GB, or a file greater than 4 GB. FAT cannot handle a hard disk greater than 4 GB, or a file greater than 2 GB. Because of these limitations, the only viable reason for using FAT16 or FAT32 is the need to dual boot the computer with a non-Windows operating system or a previous version of Windows that does not support NTFS, which is not a likely configuration for a server.

ReFS is a new file system first appearing in Windows Server 2012 that offers practically unlimited file and directory sizes and increased resiliency that eliminates the need for error-

checking tools, such as Chkdsk.exe. However, ReFS does not include support for NTFS features such as file compression, Encrypted File System (EFS), and disk quotas. ReFS disks also cannot be read by any operating systems older than Windows Server 2012 and Windows 8.

Working with disks

Windows Server 2012 includes tools that enable you to manage disks graphically or from the command prompt. All Windows Server 2012 installations include the File and Storage Services role, which causes Server Manager to display a submenu when you click the icon in the navigation pane, as shown in Figure 1-24. This submenu provides access to home pages that enable administrators to manage volumes, disks, storage pools, shares, and iSCSI devices.

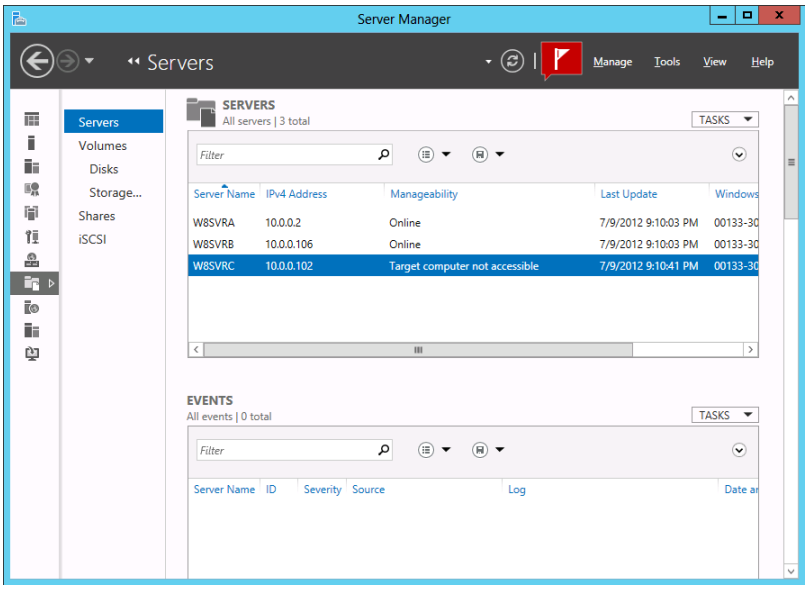

Figure 1-24 The File and Storage Services submenu in Server Manager.

Server Manager is the only graphical tool that can manage storage pools and create virtual disks. It can also perform some—but not all—of the standard disk and volume management operations on physical disks. As with the other Server Manager home pages, the File and Storage Services pages also enable you to perform tasks on any servers you have added to the interface.

Disk Management is an MMC snap-in that is the traditional tool for performing disk-related tasks.

To access the Disk Management snap-in, you must open the Computer Management console and select Disk Management.

You can also manage disks and volumes from the command line using the DiskPart.exe utility.

Adding a new physical disk

When you add a new hard disk to a Windows Server 2012 computer, you must initialize the disk before you can access its storage. To add a new secondary disk, shut down the computer and install or attach the new physical disk per the manufacturer's instructions. A newly added physical disk appears in Server Manager in the Disks tile, as shown in Figure 1-25, with a status of Offline and an unknown partition style.

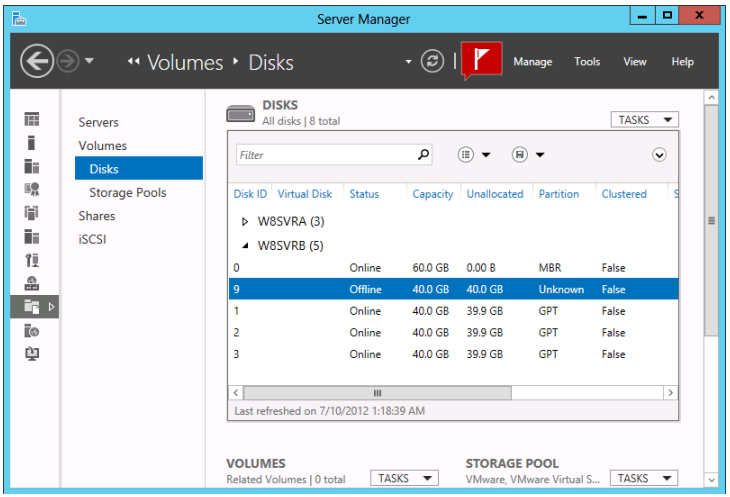

Figure 1-25 A new physical disk in Server Manager.

To make the disk accessible, you must first bring it online by right-clicking it in the Disks tile and, from the shortcut menu, selecting Bring Online. After you confirm your action and the disk status changes to Online, right-click it and select Initialize.

Unlike the Disk Management snap-in, Server Manager gives you no choice of the partition style for the disk. A Task Progress window opens, and when the process is completed, click Close, and the disk appears in the list with a partition style of GPT.

You can convert a disk from one partition style to another at any time, by right-clicking the disk you need to convert and then, from the shortcut menu, selecting Convert To GPT Disk or Convert To MBR Disk. However, be aware that converting the disk partition style is a destructive process. You can perform the conversion only on an unallocated disk, so if the disk you want to convert contains data, you must back up the disk, and then delete all existing partitions or volumes before you begin the conversion.

Creating and mounting VHDs

Hyper-V relies on the *VHD* format to store virtual disk data in files that can easily be transferred from one computer to another. The Disk Management snap-in in Windows Server 2012 enables you to create VHD files and mount them on the computer. Once they are mounted, you can treat them just like physical disks and use them to store data. Dismounting

a VHD packages the stored data in the file, so you can copy or move it as needed.

To create a VHD in Disk Management, use the following procedure.

1. Log on to Windows Server 2012 using an account with Administrator privileges. The Server Manager window opens.
2. Click Tools > Computer Management. The Computer Management console opens.
3. Click Disk Management to open the Disk Management snap-in.
4. From the Action menu, select Create VHD. The Create And Attach Virtual Hard Disk dialog box appears, as shown in Figure 1-27.

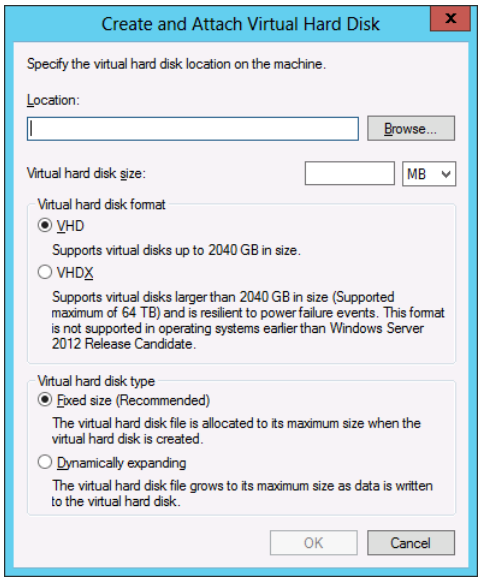

Figure 1-27 The Create And Attach Virtual Hard Disk dialog box.

5. In the Location text box, specify the path and file name for the file you want to create.
6. In the Virtual Hard Disk Size box, specify the maximum size of the disk you want to create.
7. Select one of the following Virtual Hard Disk Format options:
   - **VHD** The original and more compatible format, which supports files up to 2,040 GB
   - **VHDX** A new version of the format that supports files up to 64 TB, but can only be read by computers running Windows Server 2012
8. Select one of the following Virtual Hard Disk Type options:
   - **Fixed Size (Recommended)** Allocates all of the disk space for the VHD file at once
   - **Dynamically Expanding** Allocates disk space to the VHD file as you add

data to the virtual hard disk

9. Click OK. The system creates the VHD file and attaches it, so that it appears as a disk in the snap-in.

Once you have created and attached the VHD, it appears as an uninitialized disk in the Disk Management snap-in and in Server Manager. Using either tool, you can initialize the disk and create volumes on it, just as you would a physical disk. After storing data on the volumes, you can detach the VHD and move it to another location or mount it on a Hyper-V VM.

Creating a storage pool

Once you have installed your physical disks, you can concatenate their space into a storage pool, from which you can create virtual disks of any size.

To create a storage pool using Server Manager, use follow this procedure.

1. Log on to Windows Server 2012 using an account with Administrator privileges. The Server Manager window opens.
2. Click the File And Storage Services icon and, in the submenu that appears, click Storage Pools. The Storage Pools home page appears, as shown in Figure 1-28.

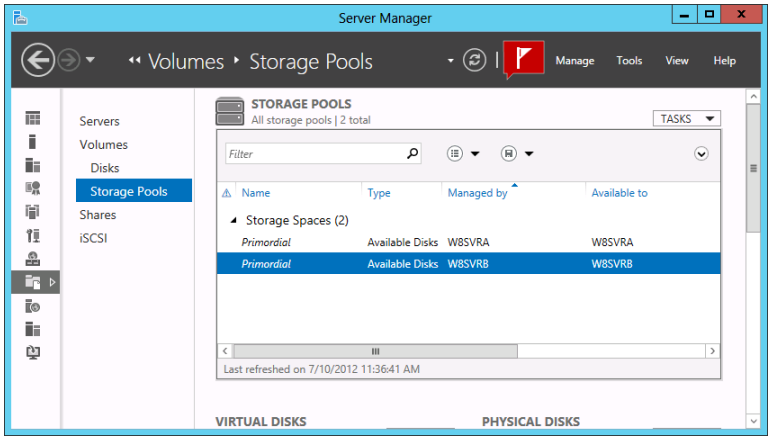

Figure 1-28 The Storage Pools home page.

3. In the Storage Pools tile, select the primordial space on the server where you want to create the pool, and, from the Tasks menu, select New Storage Pool. The New Storage Pool Wizard starts, displaying the Before You Begin page.
4. Click Next. The Specify A Storage Pool Name And Subsystem page appears, as shown in Figure 1-29.

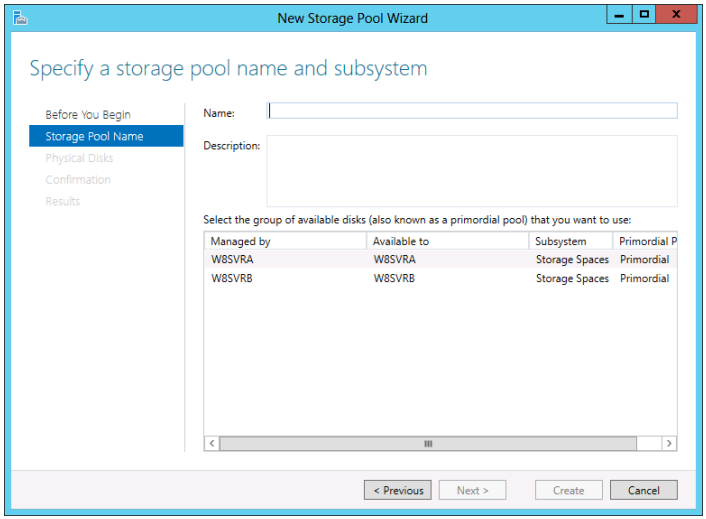

Figure 1-29 The Specify A Storage Pool Name And Subsystem page.

5. In the Name text box, type the name you want to assign to the storage pool. Then, select the server on which you want to create the pool and click Next. The Select Physical Disks For The Storage Pool page opens, as shown in Figure 1-30.

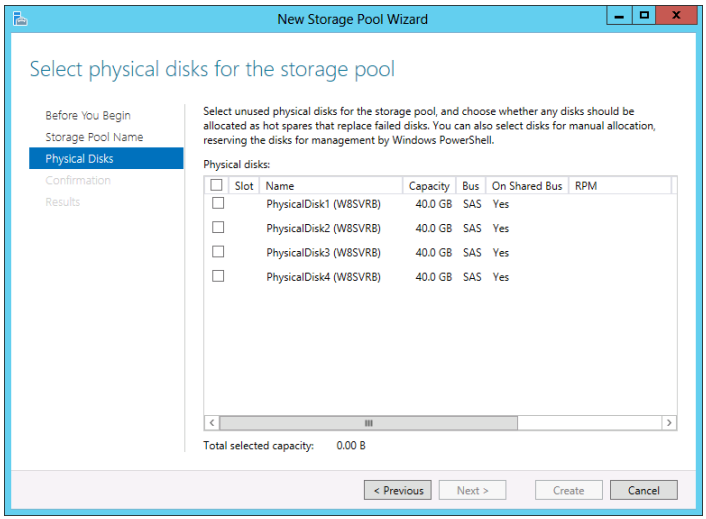

Figure 1-30 The Select Physical Disks For The Storage Pool page.

Note **THE WIZARD DISPLAYS ELIGIBLE DISKS ONLY**

The wizard displays only the disks that are eligible for addition to the pool. Disks that already have partitions or volumes on them do not appear.

6. Select the check boxes for the disks you want to add to the pool and click Next to

open the Confirm Selections page.

7. Click Create. The wizard creates the new storage pool and the View Results page opens.
8. Click Close. The wizard closes and the new pool appears on the Storage Pools home page, as shown in Figure 1-31.

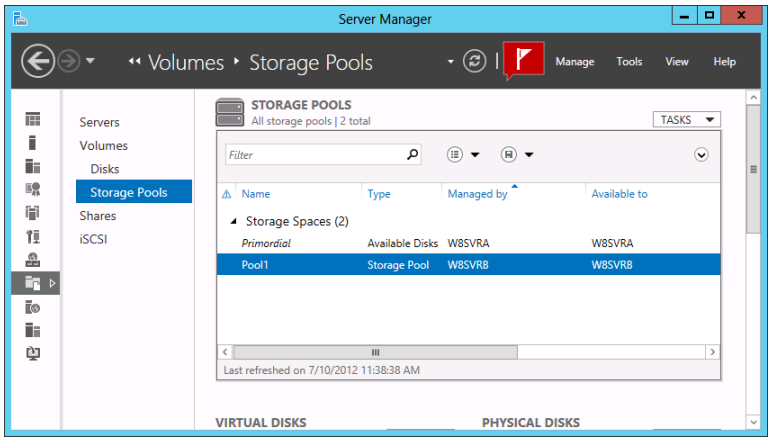

Figure 1-31 A new pool on the Storage Pools home page.

9. Close the Server Manager window.

After you have created a storage pool, you can modify its capacity by adding or removing physical disks. The Tasks menu in the Physical Disks tile on the Storage Pools home page contains the following options:

- **Add Physical Disk** Enables you to add a physical disk to the pool, as long as it is initialized and does not contain any volumes.
- **Evict Disk** Prepares a physical disk for removal from the storage pool by moving all of the data it contains to the other physical disks in the pool. This can cause the status of virtual disks using mirror or parity fault tolerance to revert to Warning, if the eviction causes the number of physical disks in the pool to fall below the minimum required.
- **Remove Disk** Removes the space provided by a physical disk from the storage pool. This option only appears if all data has already been evicted from the disk.

To create a new storage pool using Windows PowerShell, you use the New-StoragePool cmdlet with the following basic syntax:

```
New-StoragePool –FriendlyName <pool name> -StorageSubSystemFriendlyName <subsystem name>
-PhysicalDisks <disk names>
```

To obtain the correct designations for the storage subsystem and the physical disks, use the Get-StorageSubsystem and Get-PhysicalDisk cmdlets.

Creating virtual disks

After you have created a storage pool, you can use the space to create as many virtual disks as you need.

To create a virtual disk using Server Manager, use the following procedure.

1. Log on to Windows Server 2012 using an account with Administrator privileges. The Server Manager window opens.
2. Click the File And Storage Services icon and, in the submenu that appears, click Storage Pools. The Storage Pools home page appears.
3. Scroll down (if necessary) to expose the Virtual Disks tile and, from the Tasks menu, select New Virtual Disk. The New Virtual Disk menu appears, displaying the Before You Begin page.
4. Click Next to open the Select The Server And Storage Pool page.
5. Select the pool in which you want to create a virtual disk and click Next. The Specify The Virtual Disk Name page opens.
6. In the Name text box, type a name for the virtual disk and click Next. The Select The Storage Layout page opens, as shown in Figure 1-32.

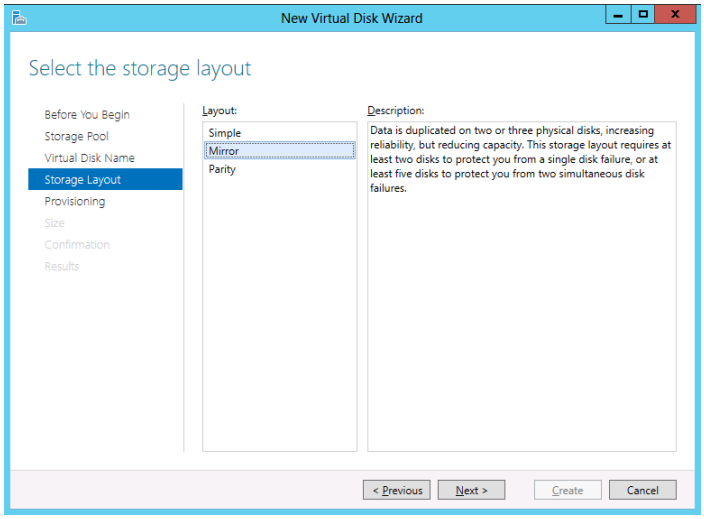

Figure 1-32 The Select The Storage Layout page.

7. Select one of the following layout options and click Next.
   - **Simple** Requires the pool to contain at least one physical disk and provides no fault tolerance. When more than one physical disk is available, the system stripes data across the disks.
   - **Mirror** Requires the pool to contain at least two physical disks and provides fault tolerance by storing identical copies of every file. Two physical disks

provide protection against a single disk failure; five physical disks provide protection against two disk failures.

- **Parity** Requires the pool to contain at least three physical disks and provides fault tolerance by striping parity information along with data.

Important **DISK LEVEL FAULT TOLERANCE**

The fault tolerance built into Storage Spaces is provided at the disk level, not the volume level, as in the Disk Management snap-in. Theoretically, one can use Disk Management to create mirrored or RAID-5 volumes out of virtual disks, but this would defeat the purpose of creating them in the first place, because the virtual disks might very well be located on the same physical disk.

8. The Specify The Provisioning Type page opens, as shown in Figure 1-33.

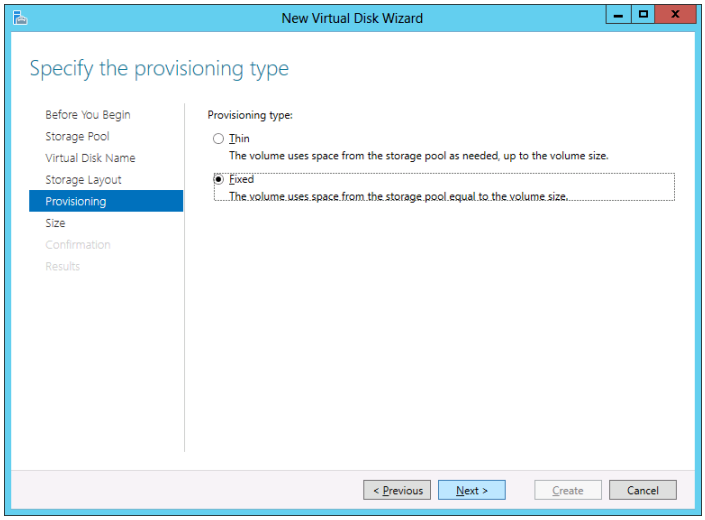

Figure 1-33 The Specify The Provisioning Type page.

9. Select one of the following Provisioning Type options and click Next.
   - **Thin** The system allocates space from the storage pool to the disk as needed, up to the maximum specified size.
   - **Fixed** The system allocates the maximum specified amount of space to the disk immediately on creating it.

The Specify The Size Of The Virtual Disk page opens, as shown in Figure 1-34.

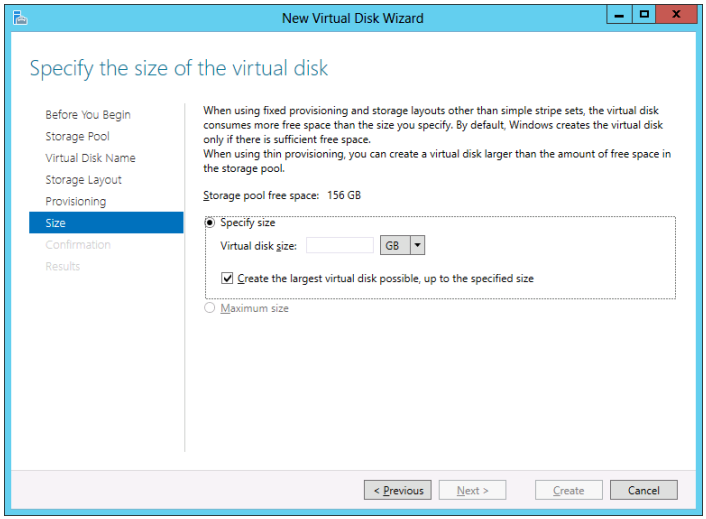

Figure 1-34 The Specify The Size Of The Virtual Disk page.

10. In the Virtual Disk Size text box, specify the size of the disk you want to create and click Next. The Confirm Selections page opens.
11. Click Create. The View Results page appears as the wizard creates the disk.
12. Click Close. The wizard closes and the new disk appears in the Virtual Disks tile, as shown in Figure 1-35.

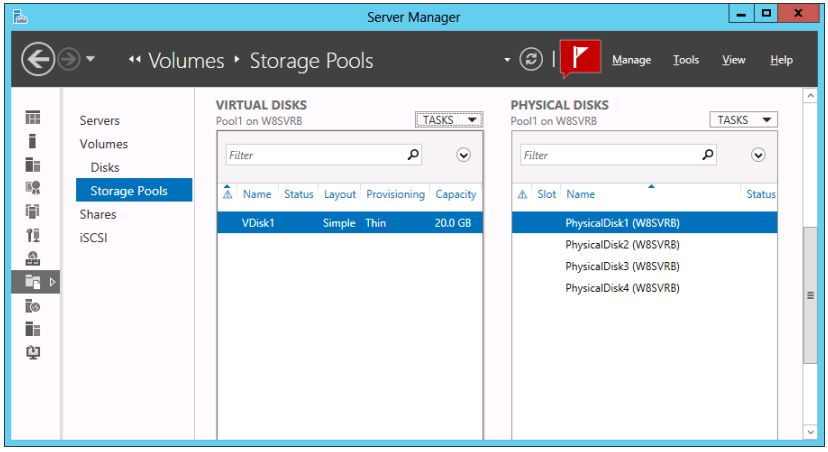

Figure 1-35 A new disk in the Virtual Disks tile in Server Manager.

13. Close the Server Manager window.

By default, the New Volume Wizard launches when you create a new virtual disk. At this point, the disk is a virtual equivalent to a newly installed physical disk. It contains nothing but unallocated space, and you must create at least one volume before you can store data on it.

Creating a simple volume

Technically speaking, you create partitions on basic disks and volumes on dynamic disks. This is not just an arbitrary change in nomenclature. Converting a basic disk to a dynamic disk actually creates one big partition, occupying all of the space on the disk. The volumes you create on the dynamic disk are logical divisions within that single partition.

Windows versions prior to 2008 use the correct terminology in the Disk Management snap-in. The menus enable you to create partitions on basic disks and volumes on dynamic disks. Windows Server 2012 uses the term volume for both disk types, and enables you to create any of the available volume types, whether the disk is basic or dynamic. If the volume type you select is not supported on a basic disk, the wizard converts it to a dynamic disk as part of the volume creation process.

Despite the menus that refer to basic partitions as volumes, the traditional rules for basic disks remain in effect. The New Simple Volume menu option on a basic disk creates up to three primary partitions. When you create a fourth volume, the wizard actually creates an extended partition and a logical drive of the size you specify. If there is any remaining space on the disk, you can create additional logical drives in the extended partition.

> ***IMPORTANT*** **Be careful if using the DiskPart.exe utility**
>
> **When you use DiskPart.exe, a command-line utility included with Windows Server 2012, to manage basic disks, you can create four primary partitions or three primary partitions and one extended partition. The DiskPart.exe utility contains a superset of the commands supported by the Disk Management snap-in. In other words, DiskPart can do everything Disk Management can do, and more. However, whereas the Disk Management snap-in prevents you from unintentionally performing actions that might result in data loss, DiskPart has no safeties, and so does not prohibit you from performing such actions. For this reason, Microsoft recommends that only advanced users use DiskPart and that they use it with due caution.**

To create a new simple volume on a basic or dynamic disk using the Disk Management snap-in, use the following procedure.

1. Log on to Windows Server 2012 using an account with Administrator privileges. The Server Manager window opens.
2. Click Tools > Computer Management. The Computer Management console opens.
3. Click Disk Management to launch the Disk Management snap-in.
4. In the Graphical View, right-click an unallocated area in the disk on which you want to create a volume and, from the shortcut menu, select New Simple Volume. The New Simple Volume Wizard starts.
5. Click Next to bypass the Welcome page. The Specify Volume Size page opens, as shown in Figure 1-36.

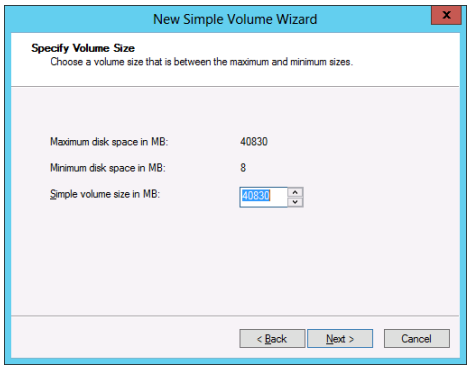

Figure 1-36 The Specify Volume Size page.

6. Select the size for the new partition or volume, within the maximum and minimum limits stated on the page, using the Simple Volume Size In MB spin box, and then click Next. The Assign Drive Letter Or Path page opens, as shown in Figure 1-37.

Figure 1-37 The Assign Drive Letter Or Path page.

7. Configure one of the following three options:
   - **Assign The Following Drive Letter** If you select this option, click the associated drop-down list for a list of available drive letters and select the letter you want to assign to the drive.
   - **Mount In The Following Empty NTFS Folder** If you select this option, either type the path to an existing NTFS folder or click Browse to search for or create a new folder. The entire contents of the new drive will appear in the folder you specify.
   - **Do Not Assign A Drive Letter Or Drive Path** Select this option if you want to create the partition, but are not yet ready to use it. When you do not assign a volume a drive letter or path, the drive is left unmounted and inaccessible. When you want to mount the drive for use, assign a drive letter or path to it.
8. Click Next to open the Format Partition page, as shown in Figure 1-38.

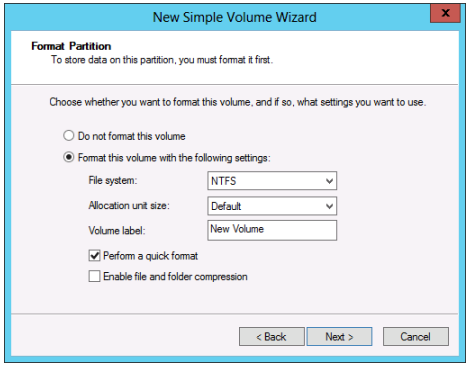

Figure 1-38 The Format Partition page.

9. Specify whether the wizard should format the volume, and if so, how. If you do not want to format the volume at this time, select the Do Not Format This Volume option. If you do want to format the volume, select the Format This Volume With The Following Settings option, and then configure the associated options, as follows.
    - **File System** Select the desired file system. The options available depend on the size of the volume, and can include ReFS, NTFS, exFAT, FAT32, or FAT.
    - **Allocation Unit Size** Specify the file system's cluster size. The cluster size signifies the basic unit of bytes in which the system allocates disk space. The system calculates the default allocation unit size based on the size of the volume. You can override this value by clicking the associated drop-down list and then selecting one of the values. For example, if your client uses consistently small files, you might want to set the allocation unit size to a smaller cluster size.
    - **Volume Label** Specify a name for the partition or volume. The default name is New Volume, but you can change the name to anything you want.
    - **Perform A Quick Format** When this check box is selected, Windows formats the disk without checking for errors. This is a faster method to format the drive, but Microsoft does not recommend it. When you check for errors, the system looks for and marks bad sectors on the disk so that your clients will not use those portions of the disk.
    - **Enable File And Folder Compression** Selecting this check box turns on folder compression for the disk. This option is available only for volumes being formatted with the NTFS file system.
10. Click Next. The Completing The New Simple Volume Wizard page opens.
11. Review the settings to confirm your options, and then click Finish. The wizard creates the volume according to your specifications.
12. Close the console containing the Disk Management snap-in.

After you create a simple volume, you can use the Disk Management snap-in to modify its properties by extending it or shrinking it, as described later in this lesson.

This procedure can create volumes on physical or virtual disks. You can also create simple volumes using a similar wizard in Server Manager.

When you launch the New Volume Wizard in Server Manager, which you can do from the Volumes or Disks home page, the options the wizard presents are virtually identical to those in the New Simple Volume Wizard in Disk Management.

The primary difference is that, like all Server Manager wizards, the New Volume Wizard includes a page that enables you to select the server and the disk on which you want to create volume, as shown in Figure 1-39. You can therefore use this wizard to create volumes on any disk, on any of your servers.

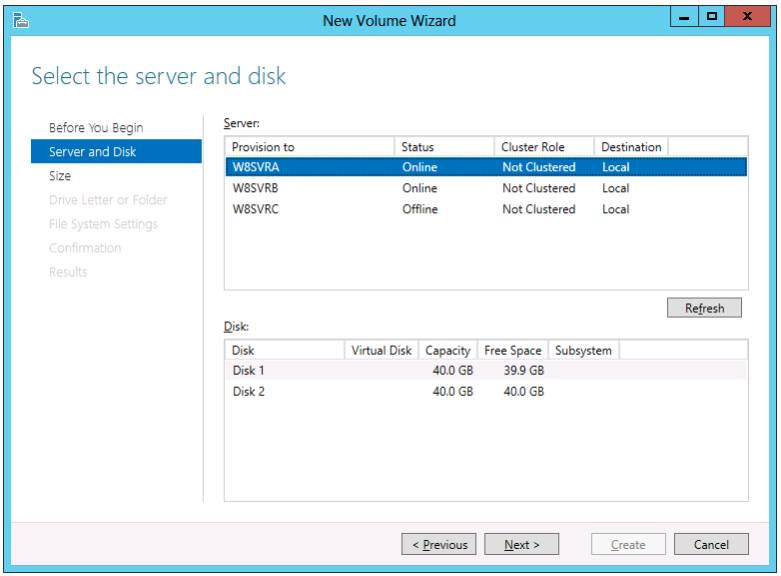

Figure 1-39 The Select The Server And Disk page in the New Volume Wizard in Server Manager.

Creating a striped, spanned, mirrored, or RAID-5 volume

The procedure for creating a striped, spanned, mirrored, or RAID-5 volume is almost the same as that for creating a simple volume, except that the Specify Volume Size page is replaced by the Select Disks page.

To create a striped, spanned, mirrored, or RAID-5 volume, use the following procedure.

1. Log on to Windows Server 2012 using an account with Administrator privileges. The Server Manager window appears.
2. Click Tools > Computer Management. The Computer Management console opens.
3. Click Disk Management to open the Disk Management snap-in.
4. Right-click an unallocated area on a disk and then, from the shortcut menu, select the command for the type of volume you want to create. A New Volume Wizard appears, named for your selected volume type.

5. Click Next to bypass the Welcome page. The Select Disks page appears, as shown in Figure 1-40.

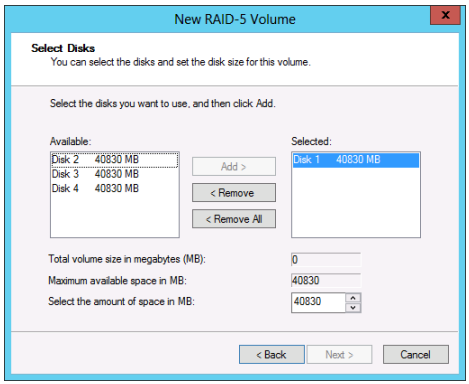

Figure 1-40 The Select Disks page.

6. On the Select Disks page, select the disks you want to use for the new volume from the Available list box, and then click Add. The disks you chose are moved to the Selected list box, joining the original disk you selected when launching the wizard. For a striped, spanned, or mirrored volume, you must have at least two disks in the Selected list; for a RAID-5 volume, you must have at least three.
7. Specify the amount of space you want to use on each disk, using the Select The Amount Of Space In MB spin box. Then click Next. The Assign Drive Letter Or Path page opens.
    - If you are creating a spanned volume, you must click each disk in the Selected list and specify the amount of space to use on that disk. The default value for each disk is the size of the unallocated space on that disk.
    - If you are creating a striped, mirrored, or RAID-5 volume, you only specify one value, because these volumes require the same amount of space on each disk. The default value is the size of the unallocated space on the disk with the least amount of space free.
8. Specify whether you want to assign a drive letter or path, and then click Next. The Format Partition page opens.
9. Specify if or how you want to format the volume, and then click Next to open the Completing The New Simple Volume Wizard page.
10. Review the settings to confirm your options, and then click Finish. If any of the disks you selected to create the volume are basic disks, a Disk Management message box appears, warning you that the volume creation process will convert the basic disks to dynamic disks.
11. Click Yes. The wizard creates the volume according to your specifications.

More Info **ADDITIONAL OPTIONS**

See "Creating a simple volume" earlier for more information about the options on the Assign Drive Letter or Path And Format Partition pages.

12. Close the Disk Management snap-in.

The commands that appear in a disk's shortcut menu depend on the number of disks installed in the computer and the presence of unallocated space on them. For example, at least two disks with unallocated space must be available to create a striped, spanned, or mirrored volume, and at least three disks must be available to create a RAID-5 volume.

Thought experiment

Planning Storage

In this thought experiment, apply what you've learned about this objective. You can find answers to these questions in the "Answers" section at the end of this chapter.

On a new server running Windows Server 2012, Morris created a storage pool that consists of two physical drives holding 1 TB each. Then he created three simple virtual disks out of the space in the storage pool. Using the Disk Management snap-in, Morris then created a RAID-5 volume out of the three virtual disks.

With this in mind, answer the following questions:

1. **In what way is Morris's storage plan ineffectual at providing fault tolerance?**
2. **Why will adding a third disk to the storage pool fail to improve the fault tolerance of the storage plan?**
3. **How can Morris modify the storage plan to make it fault tolerant?**

Objective summary

- Windows Server 2012 supports two hard disk partition types: MBR and GPT; two disk types: basic and dynamic; five volume types: simple, striped, spanned, mirrored, and RAID-5; and three file systems: ReFS, NTFS, and FAT.
- The Disk Management snap-in can initialize, partition, and format disks on the local machine. Server Manager can perform many of the same tasks for servers all over the network.
- A Windows server can conceivably perform its tasks using the same type of storage as a workstation. However, the I/O burdens of a server are quite different from those of a workstation, and a standard storage subsystem can easily be overwhelmed by file requests from dozens or hundreds of users. In addition, standard hard disks offer no fault tolerance and are limited in their scalability.
- Windows Server 2012 includes a new disk virtualization technology called Storage

Spaces, which enables a server to concatenate storage space from individual physical disks and allocate that space to create virtual disks of any size supported by the hardware.

- All Windows Server 2012 installations include the File and Storage Services role, which causes Server Manager to display a submenu when you click the icon in the navigation pane. This submenu provides access to home pages that enable administrators to manage volumes, disks, storage pools, shares, and iSCSI devices.
- The Disk Management snap-in in Windows Server 2012 enables you to create VHD files and mount them on the computer.
- Once you have installed your physical disks, you can concatenate their space into a storage pool, from which you can create virtual disks of any size. Once you have created a storage pool, you can use the space to create as many virtual disks as you need.

Objective review

Answer the following questions to test your knowledge of the information in this objective. You can find the answers to these questions and explanations of why each answer choice is correct or incorrect in the "Answers" section at the end of this chapter.

1. Which of the following statements are true of striped volumes? (Choose all that apply)
   A. Striped volumes provide enhanced performance over simple volumes.
   B. Striped volumes provide greater fault tolerance than simple volumes.
   C. You can extend striped volumes after creation.
   D. If a single physical disk in the striped volume fails, all of the data in the entire volume is lost.
2. Which of the following are requirements for extending a volume on a dynamic disk? (Choose all that apply)
   A. If you want to extend a simple volume, you can use only the available space on the same disk, if the volume is to remain simple.
   B. The volume must have a file system (a raw volume) before you can extend a simple or spanned volume.
   C. You can extend a simple or spanned volume if you formatted it using the FAT or FAT32 file systems.
   D. You can extend a simple volume across additional disks if it is not a system volume or a boot volume.
3. Which of the following are *not* true about differences between network attached storage (NAS) devices and storage area network (SAN) devices?
   A. NAS devices provide a file system implementation; SAN devices do not.
   B. NAS devices must have their own processor and memory hardware; SAN devices

do not require these components.

C. NAS devices must run their own operating system and typically provide a web interface for administrative access; SAN devices do not have to have either one.

D. NAS devices require a specialized protocol, such as Fibre Channel or iSCSI; SAN devices use standard networking protocols.

4. Which of the following volume types supported by Windows Server 2012 do not provide fault tolerance? (Choose all that apply.)

   A. Striped

   B. Spanned

   C. Mirrored

   D. RAID-5

5. A JBOD drive array is an alternative to which of the following?

   A. SAN

   B. SCSI

   C. RAID

   D. iSCSI

Chapter summary

- When you select the Windows Server Core installation option, you get a stripped-down version of the operating system.
- The Minimal Server Interface is a setting that removes some of the most hardware intensive elements from the graphical interface.
- Migration is the preferred method of replacing an existing server with one running Windows Server 2012. Unlike an in-place upgrade, a migration copies vital information from an existing server to a clean Windows Server 2012 installation.
- In Windows Server 2012, you can convert a computer installed with the full GUI option to Server Core, and add the full GUI to a Server Core computer.
- NIC teaming is a new feature in Windows Server 2012 that enables administrators to combine the bandwidth of multiple network interface adapters, providing increased performance and fault tolerance.
- Windows Server 2012 supports two hard disk partition types: MBR and GPT; two disk types: basic and dynamic; five volume types: simple, striped, spanned, mirrored, and RAID-5; and three file systems: ReFS, NTFS, and FAT.

- Once you have installed your physical disks, you can concatenate their space into a storage pool, from which you can create virtual disks of any size. Once you have created a storage pool, you can use the space to create as many virtual disks as you need.

Answers

This section contains the solutions to the thought experiments and answers to the lesson review questions in this chapter.

Objective 1.1: Thought experiment

1. Uninstall-WindowsFeature Server-Gui-Mgmt-Infra,Server-Gui-Shell –Restart
2. Uninstall-WindowsFeature Server-Gui-Mgmt-Infra,Server-Gui-Shell –Remove

Objective 1.1: Review

1. **Correct answers:** A and C
   A. **Correct:** DNS is an infrastructure service.
   B. **Incorrect:** Web Server (IIS) is an application service, not an infrastructure service.
   C. **Correct:** DHCP is an infrastructure service.
   D. **Incorrect:** Remote Desktop Services is an application service, not an infrastructure service.
2. **Correct answer:** B
   A. **Incorrect:** You cannot upgrade any version of Windows Server 2003 Standard to Windows Server 2012 Standard.
   B. **Correct:** You can upgrade Windows Server 2008 Standard to Windows Server 2012 Standard.
   C. **Incorrect:** You cannot upgrade Windows Server 2008 R2 32-bit, or any 32-bit version, to Windows Server 2012 64-bit.
   D. **Incorrect:** You cannot upgrade Windows 7 Ultimate, or any workstation operating system, to Windows Server 2012 Essentials.
3. **Correct answer:** A
   A. **Correct:** Installing the Graphical Management Tools and Infrastructure module—and only that module—on a Server Core installation results in the Minimal Server Interface.
   B. **Incorrect:** Installing the Server Graphical Shell with the Graphical Management Tools and Infrastructure converts a Server Core installation to the full GUI.
   C. **Incorrect:** Windows PowerShell is a command-line interface that has no effect on

the Minimal Server Installation.

D. **Incorrect:** MMC is one of the graphical applications available in the Minimal Server Installation, but you do not install it individually.

4. **Correct answer:** D

A. **Incorrect:** The Windows directory contains live operating system files, not the installation files.

B. **Incorrect:** The System32 directory contains live operating system files, not the installation files.

C. **Incorrect:** There is no bin directory associated with the Windows operating system.

D. **Correct:** Windows stores all of the operating system installation modules in the WinSxS directory.

5. **Correct answers:** A, B, and C

A. **Correct:** It is possible to convert a computer running Windows Server 2012 between the Server Core and the Full GUI interface as needed.

B. **Correct:** The PowerShell 3.0 interface in Windows Server 2012 has many more cmdlets than PowerShell 2.0.

C. **Correct:** Server Manager incorporates a server selection interface into many of its wizards.

D. **Incorrect:** There are no different licenses for Server Core and Full GUI versions of Windows Server 2012.

Objective 1.2: Thought experiment

1. Install-WindowsFeature
2. Get-WindowsFeature
3. Deepak must run the following commands:

```
Install-WindowsFeature FS-FileServer
Install-WindowsFeature FS-DFS-Namespace
Install-WindowsFeature FS-DFS-Replication
Install-WindowsFeature FS-NFS-Service
Install-WindowsFeature Print-Services –allsubfeatures
Install-WindowsFeature Web-Server
Install-WindowsFeature Web-Windows-Auth
Install-WindowsFeature Web-Ftp-Service
```

Objective 1.2: Review

1. **Correct answers:** B and D

A. **Incorrect:** Windows Management Instrumentation is a set of driver extensions often used with Windows PowerShell. You do not have to remove it to convert to Server Core.

B. **Correct:** Removing the Graphical Management Tools and Infrastructure feature is required to convert to a Server Core installation.

C. **Incorrect:** Desktop Experience is not installed by default on a full GUI or a Server Core installation.

D. **Correct:** Server Graphical Shell provides support for MMC, Server Manager, and part of Control Panel. You must remove it to convert to a Server Core installation.

2. **Correct answer:** B

A. **Incorrect:** Hyper-V live migration is not a NIC teaming mode.

B. **Correct:** In Switch Independent Mode, the NICs in the team are connected to different switches, providing alternate paths through the network.

C. **Incorrect:** In Switch Dependent Mode, the NICs in the team are connected to the same switches, providing link aggregation, but no fault tolerance.

D. **Incorrect:** Link Aggregation Control Protocol is not a NIC teaming mode.

3. **Correct answer:** C

A. **Incorrect:** Net.exe is a Windows command-line tool that provides many different functions, but it cannot join a computer to a domain.

B. **Incorrect:** Netsh.exe is a network shell program that you can use to configure the network interface, but it cannot join a computer to a domain.

C. **Correct:** Netdom.exe is the Windows command-line domain manager application.

D. **Incorrect:** Ipconfig.exe can display network configuration settings and reset DHCP settings, but it cannot join a computer to a domain.

4. **Correct answer:** A

A. **Correct:** Server Manager cannot deploy roles to multiple servers at the same time.

B. **Incorrect:** Server Manager can mount offline VHD files and install roles and features to them.

C. **Incorrect:** Server Manager combines the role and feature installation processes into a single wizard.

D. **Incorrect:** Server Manager can install roles and features to any Windows Server 2012 server on the network.

5. **Correct answers:** C and D

A. **Incorrect:** You can stop a running service using Server Manager.

B. **Incorrect:** You can start a stopped service using Server Manager

C. **Correct:** You cannot disable a service using Server Manager.

D. **Correct:** You cannot configure a service to start when the computer starts using

Server Manager.

Objective 1.3: Thought experiment

1. Morris has created a RAID-5 volume out of virtual disks created out of a storage pool that has only two physical disks in it. A RAID-5 volume can only provide fault tolerance by storing data on three physical disks.
2. Adding a third disk will not guarantee fault tolerance because there is no assurance that each of the three virtual disks exists on a separate individual disk.
3. To make the plan fault tolerant, Morris should delete the three simple virtual disks and create one new virtual disk using either the mirror or parity layout option.

Objective 1.3: Review

1. Correct answers: A and D

A. **Correct:** Striping provides improved performance because each disk drive in the array has time to seek the location of its next stripe while the other drives are writing.

B. **Incorrect:** Striped volumes do not contain redundant data, and therefore do not provide fault tolerance.

C. **Incorrect:** Striped volumes cannot be extended after creation without destroying the data stored on them in the process.

D. **Correct:** If a single physical disk in the striped volume fails, all of the data in the entire volume is lost.

2. **Correct answers:** A and D

A. **Correct:** When extending a simple volume, you can use only the available space on the same disk. If you extend the volume to another disk, it is no longer simple.

B. **Incorrect:** You can extend a simple or spanned volume, even if it does not have a file system (a raw volume).

C. **Incorrect:** You can extend a volume if you formatted it using the NTFS file system. You cannot extend volumes using the FAT or FAT32 file systems.

D. **Correct:** You can extend a simple volume across additional disks if it is not a system volume or a boot volume.

3. **Correct Answer:** D

A. **Incorrect:** A SAN provides block-based storage services to the computers connected to it, just as if the storage devices were installed in the computer. The storage hardware on a SAN might provide additional capabilities, such as RAID, but the file system used to store and protect data on the SAN devices is implemented by the computer.

B. **Incorrect:** A NAS array connects to a standard LAN, and does not require a

computer to implement the file system or function as a file server. It has its own processor and memory array.

C. **Incorrect:** NAS devices are essentially dedicated file servers with their own operating systems, which provide file-based storage services directly to clients on the network.

D. **Correct:** A storage area network (SAN) is a separate network dedicated solely to storage devices. SANs use a high-speed networking technology, such as SCSI, iSCSI, or Fibre Channel to enable them to transmit large amounts of file data very quickly.

4. **Correct answers:** C and D

A. **Incorrect:** A striped volume spreads data among multiple disks, but it writes the data only once. Therefore, it does not provide fault tolerance.

B. **Incorrect:** A spanned volume uses space on multiple drives, but it writes the data only once. Therefore, it does not provide fault tolerance.

C. **Correct:** A mirrored volume writes duplicate copies of all data to two or more disks, thereby providing fault tolerance.

D. **Correct:** A RAID-5 volume writes data and parity information on multiple disks, providing fault tolerance.

5. **Correct answer:** C

A. **Incorrect:** A SAN is a separate network dedicated to storage, and a JBOD is a drive array that can be installed on a SAN or on a standard network.

B. **Incorrect:** SCSI is disk interface, not a type of drive array.

C. **Correct:** A JBOD array is an alternative to a RAID array that treats each disk as an independent volume.

D. **Incorrect:** A JBOD array is not an alternative to iSCSI, which is a protocol used for SAN communications.

CHAPTER 2

Configure server roles and features

This chapter covers some of the fundamental services that most Windows servers perform. In the business world, file and printer sharing were the reasons why computers were networked in the first place, and with Windows Server 2012, remote management has become a critical element of server administration.

Objectives in this chapter:

- Objective 2.1: Configuring file and share access
- Objective 2.2: Configure print and document services
- Objective 2.3: Configure servers for remote management

Objective 2.1: Configuring file and share access

One of the critical daily functions of server administrators is to decide where users should store their files and who should be permitted to access them.

This objective covers how to:

- Create and configure shares
- Configure share permissions
- Configure offline files
- Configure NTFS permissions
- Configure access-based enumeration (ABE)
- Configure Volume Shadow Copy Service (VSS)
- Configure NTFS quotas

Creating folder shares

Sharing folders makes them accessible to network users. After you have configured the disks on a file server, you must create shares for network users to be able to access those disks. As noted in the planning discussions earlier in this chapter, you should have a sharing strategy in place by the time you are ready to actually create your shares. This strategy should consist of

the following information:

- What folders you will share
- What names you will assign to the shares
- What permissions you will grant users to the shares
- What Offline Files settings you will use for the shares

If you are the Creator Owner of a folder, you can share it on a Windows Server 2012 computer by right-clicking the folder in any File Explorer window, selecting Share With > Specific People from the shortcut menu, and following the instructions in the File Sharing dialog box, as shown in Figure 2-1.

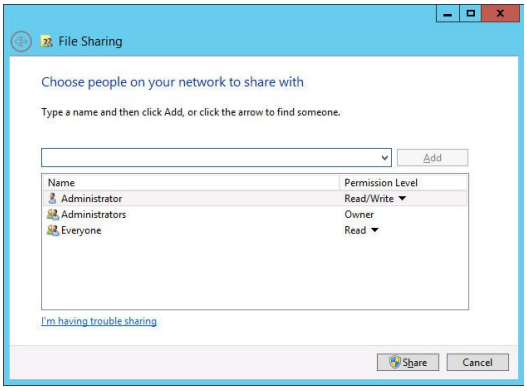

FIGURE 2-1 The File Sharing dialog box.

This method of creating shares provides a simplified interface that contains only limited control over elements such as share permissions. You can specify only that the share users receive Read or Read/Write permissions to the share. If you are not the Creator Owner of the folder, you can access the Sharing tab of the folder's Properties sheet instead. Clicking the Share button launches the same dialog box, and the Advanced Sharing button displays the Advanced Sharing dialog box shown in Figure 2-2, which provides greater control over share permissions.

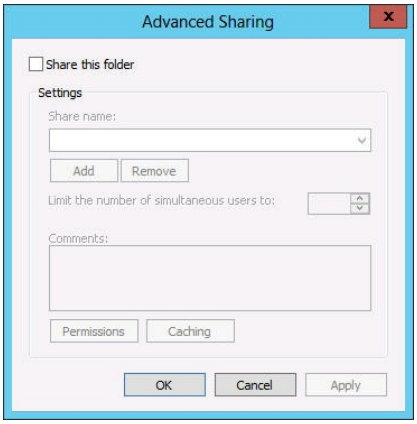

FIGURE 2-2 The Advanced Sharing dialog box.

NOTE **Network Discovery**

For the users on the network to be able to see the shares you create on the file server, you must make sure that the Network Discovery and File Sharing settings are turned on in the Network and Sharing Center control panel.

However, to take control of the shares on all of your disks on all of your servers, and exercise granular control over their properties, you can use the File and Storage Services home page in Server Manager.

Windows Server 2012 supports two types of folder shares:

- **Server Message Blocks (SMB)** SMB is the standard file sharing protocol used by all versions of Windows.
- **Network File System (NFS)** NFS is the standard file sharing protocol used by most UNIX and Linux distributions.

When you install Windows Server 2012, the setup program installs the Storage Services role service in the File and Storage Services role by default. However, before you can create and manage SMB shares using Server Manager, you must install the File Server role service, and to create NFS shares, you must install the Server for NFS role service.

To create a folder share using Server Manager, use the following procedure.

1. Log on to Windows Server 2012 using an account with Administrator privileges. The Server Manager window opens.
2. Click the File and Storage Services icon and, in the submenu that appears, click Shares to open the Shares home page.
3. From the Tasks menu, select New Share. The New Share Wizard starts, displaying the Select The Profile For This Share page, as shown in Figure 2-3.

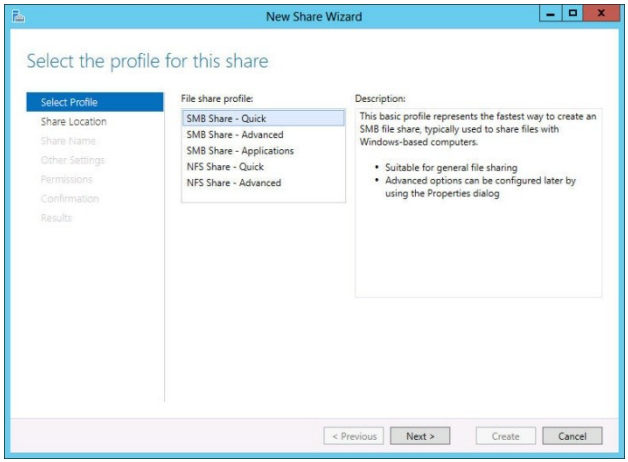

FIGURE 2-3 The Select The Profile For This Share page in the New Share Wizard.

4. From the File Share Profile list, select one of the following options:
   - **SMB Share–Quick** Provides basic SMB sharing with full share and NTFS permissions
   - **SMB Share–Advanced** Provides SMB sharing with full share and NTFS permissions and access to services provided by File Server Resource Manager
   - **SMB Share–Applications** Provides SMB sharing with settings suitable for Hyper-V and other applications
   - **NFS Share–Quick** Provides basic NFS sharing with authentication and permissions.
   - **NFS Share–Advanced** Provides NFS sharing with authentication and permissions, plus access to services provided by File Server Resource Manager
5. Click Next. The Select The Server And Path For This Share page appears.
6. Select the server on which you want to create the share and either select a volume on the server or specify a path to the folder you want to share. Click Next. The Specify Share Name page appears.

   MORE INFO **NFS SHARING**

   Selecting one of the NFS share profiles adds two pages to the wizard: Specify Authentication Methods and Specify The Share Permissions. Both of these pages provide access to functions implemented by the Server for NFS role service, as covered in Objective 2.1, "Configure Advanced File Services," in Exam 70-412, "Configuring Advanced Windows Server 2012 Services."

7. In the Share Name text box, specify the name you want to assign to the share and click Next. The Configure Share Settings page appears, as shown in Figure 2-4.

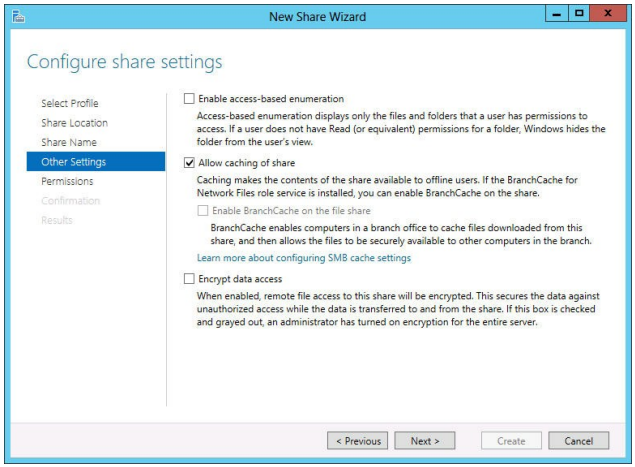

FIGURE 2-4 The Configure Share Settings page of the New Share Wizard.

8. Select any or all of the following options:

- **Enable Access-Based Enumeration** Prevents users from seeing files and folders they do not have permission to access
- **Allow Caching Of Share** Enables offline users to access the contents of the share
- **Enable BranchCache On The File Share** Enables BranchCache servers to cache files accessed from this share
- **Encrypt Data Access** Causes the server to encrypt remote file access to this share

Note ACCESS-BASED ENUMERATION

Access-based enumeration (ABE), a feature first introduced in Windows Server 2003 R2, applies filters to shared folders based on the individual user's permissions to the files and subfolders in the share. Simply put, users who cannot access a particular shared resource are unable to see that resource on the network. This feature prevents users from searching through files and folders they cannot access. You can enable or disable ABE for a share at any time by opening the share's Properties sheet in the Sharing and Storage Management console and clicking Advanced, to display the same Advanced dialog box displayed by the Provision a Shared Folder Wizard.

Note OFFLINE FILES

Offline Files, also known as client-side caching, is a Windows feature that enables client systems to maintain local copies of files they access from server shares. When a client selects the Always Available Offline option for a server-based file, folder, or share, the client system copies the selected data to the local drive, and updates it regularly, so that the client user can always access it, even if the server is offline. To enable clients to use the Offline Files feature, the share must have the Allow Caching Of Share check box selected. Windows Server 2012 and Windows 8 also have a new Always Offline mode for the Offline Files feature that causes clients to always use the cached copy of server

files, providing better performance. To implement this mode, you must set the Configure slow-link mode Group Policy setting on the client to a value of 1 millisecond.

9. Click Next to move to the Specify Permissions To Control Access page.
10. Modify the default share and NTFS permissions as needed and click Next. The Confirm Selections page appears.

Note **ADVANCED SHARE PROFILES**

Selecting one of the Advanced share profiles adds two pages to the wizard: Specify Folder Management Properties and Apply A Quota To A Folder Or Volume. Both of these pages provide access to functions of the File Server Resource Manager application, as covered in Objective 2.2, "Configure File Server Resource Manager (FSRM)," in Exam 70-411, "Administering Windows Server 2012."

11. Click Create. The View Results page appears as the wizard creates the share.
12. Close the New Share Wizard.

After you create a share with the wizard, the new share appears in the Shares tile on the Shares home page in Server Manager. You can now use the tile to manage a share by right-clicking it and opening its Properties sheet, or by clicking Stop Sharing.

Assigning permissions

Earlier in this chapter, you learned about controlling access to a file server, to provide network users with the access they need, while protecting other files against possible intrusion and damage, whether deliberate or not. To implement this access control, Windows Server 2012 uses permissions.

Permissions are privileges granted to specific system entities, such as users, groups, or computers, enabling them to perform a task or access a resource. For example, you can grant a specific user permission to read a file, while denying that same user the permissions needed to modify or delete the file.

Windows Server 2012 has several sets of permissions, which operate independently of each other. For the purpose of file sharing, you should be familiar with the operation of the following permission systems:

- **Share permissions** Control access to folders over a network. To access a file over a network, a user must have appropriate share permissions (and appropriate NTFS permissions, if the shared folder is on an NTFS volume).
- **NTFS permissions** Control access to the files and folders stored on disk volumes formatted with the NTFS file system. To access a file, whether on the local system or over a network, a user must have the appropriate NTFS permissions.

All of these permission systems operate independently of each other, and sometimes combine to provide increased protection to a specific resource. For network users to be able to access a shared folder on an NTFS drive, you must grant them both share permissions and

NTFS permissions. As you saw earlier, you can grant these permissions as part of the share creation process, but you can also modify the permissions at any time afterward.

Understanding the Windows permission architecture

To store the permissions, each of these elements has an access control list (ACL). An ACL is a collection of individual permissions, in the form of access control entries (ACEs). Each ACE consists of a security principal (that is, the name of the user, group, or computer granted the permissions) and the specific permissions assigned to that security principal. When you manage permissions in any of the Windows Server 2012 permission systems, you are actually creating and modifying the ACEs in an ACL.

To manage permissions in Windows Server 2012, you can use a tab in the protected element's Properties sheet, like the one shown in Figure 2-5, with the security principals listed at the top and the permissions associated with them at the bottom. Share permissions are typically found on a Share Permissions tab, and NTFS permissions are located on a Security tab. All of the Windows permission systems use the same basic interface, although the permissions themselves differ. Server Manager also provides access to NTFS and share permissions using a slightly different interface.

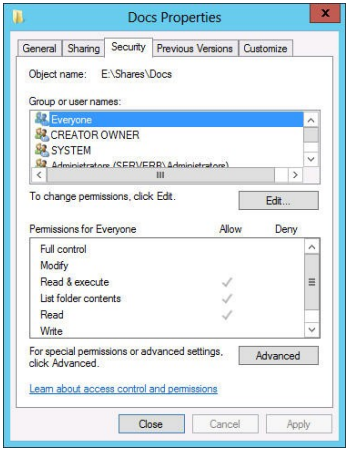

FIGURE 2-5 The Security tab of a Properties dialog box.

Understanding basic and advanced permissions

The permissions protecting a particular system element are not like the keys to a lock, which provide either full access or no access at all. Permissions are designed to be granular, enabling you to grant specific degrees of access to security principals.

To provide this granularity, each of the Windows permission systems has an assortment of permissions that you can assign to a security principal in any combination. Depending on the permission system you are working with, you might have dozens of different permissions available for a single system element.

Windows provides preconfigured permission combinations suitable for most common access control chores. When you open the Properties sheet for a system element and look at its Security tab, the NTFS permissions you see are called basic permissions. Basic permissions are actually combinations of advanced permissions, which provide the most granular control over the element.

EXAM TIP

Prior to Windows Server 2012, basic permissions were known as standard permissions and advanced permissions were known as special permissions. Candidates for certification exams should be aware of these alternative terms.

For example, the NTFS permission system has 14 advanced permissions that you can assign to a folder or file. However, there are also six basic permissions, which are various combinations of the 14 advanced permissions. In most cases, administrators work only with basic permissions. Many administrators rarely, if ever, work directly with advanced permissions.

If you do find it necessary to work with advanced permissions directly, Windows makes it possible. When you click the Advanced button on the Security tab of any Properties sheet, an Advanced Security Settings dialog box appears, as shown in Figure 2-6, which enables you to access the ACEs for the selected system element directly. System Manager provides access to the same dialog box through a share's Properties sheet.

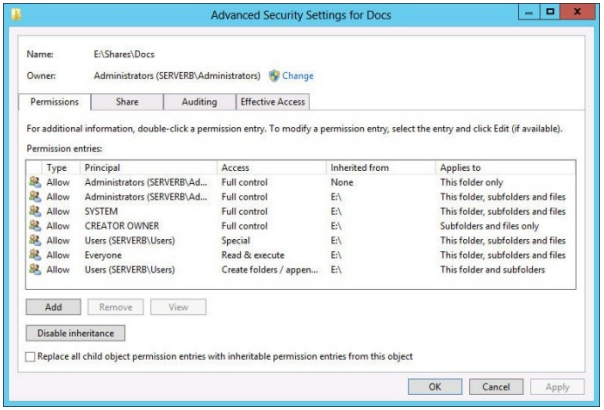

FIGURE 2-6 The Advanced Security Settings dialog box.

Allowing and denying permissions

When you assign permissions to a system element, you are, in effect, creating a new ACE in the element's ACL. There are two basic types of ACE: Allow and Deny. This makes it possible to approach permission management tasks from two directions:

- **Additive** Start with no permissions and then grant Allow permissions to individual security principals to provide them with the access they need.

- **Subtractive** Start by granting all possible Allow permissions to individual security principals, providing them with full control over the system element, and then grant them Deny permissions for the access you don't want them to have.

Most administrators prefer the additive approach, because Windows, by default, attempts to limit access to important system elements. In a properly designed permission hierarchy, the use of Deny permissions is often not needed at all. Many administrators frown on their use, because combining Allow and Deny permissions in the same hierarchy can make it difficult to determine the effective permissions for a specific system element.

Inheriting permissions

The most important principle in permission management is that permissions tend to run downward through a hierarchy. This is called permission inheritance. Permission inheritance means that parent elements pass their permissions down to their subordinate elements. For example, when you grant Alice Allow permissions to access the root of the D drive, all of the folders and subfolders on the D drive inherit those permissions, and Alice can access them.

The principle of inheritance greatly simplifies the permission assignment process. Without it, you would have to grant security principals individual Allow permissions for every file, folder, share, object, and key they need to access. With inheritance, you can grant access to an entire file system by creating one set of Allow permissions.

In most cases, whether consciously or not, system administrators take inheritance into account when they design their file systems and Active Directory Domain Services trees. The location of a system element in a hierarchy is often based on how the administrators plan to assign permissions.

In some situations, an administrator might want to prevent subordinate elements from inheriting permissions from their parents. There are two ways to do this:

- **Turn off inheritance** When you assign advanced permissions, you can configure an ACE not to pass its permissions down to its subordinate elements. This effectively blocks the inheritance process.
- **Deny permissions** When you assign a Deny permission to a system element, it overrides any Allow permissions that the element might have inherited from its parent objects.

Understanding effective access

A security principal can receive permissions in many ways, and it is important for an administrator to understand how these permissions interact. The combination of Allow permissions and Deny permissions that a security principal receives for a given system element, whether explicitly assigned, inherited, or received through a group membership, is called the effective access for that element. Because a security principal can receive permissions from so many sources, it is not unusual for those permissions to conflict, so the following rules define how the permissions combine to form the effective access.

- **Allow permissions are cumulative** When a security principal receives Allow permissions from more than one source, the permissions are combined to form the effective access permissions.
- **Deny permissions override Allow permissions** When a security principal receives Allow permissions, whether explicitly, by inheritance, or from a group, you can override those permissions by granting the principal Deny permissions of the same type.
- **Explicit permissions take precedence over inherited permissions** When a security principal receives permissions by inheriting them from a parent or from group memberships, you can override those permissions by explicitly assigning contradicting permissions to the security principal itself.

Of course, instead of examining and evaluating all of the possible permission sources, you can just open the Advanced Security Settings dialog box and click the Effective Access tab. On this tab, you can select a user, group, or device and view its effective access, with or without the influence provided by specific groups.

Setting share permissions

On Windows Server 2012, shared folders have their own permission system, which is completely independent from the other Windows permission systems. For network users to access shares on a file server, you must grant them the appropriate share permissions. By default, the Everyone special identity receives the Allow Full Control share permission to any new shares you create.

To modify the share permissions for an existing share using File Explorer, you open the Properties sheet for the shared folder, select the Sharing tab, and then click Advanced Sharing and Permissions to open the Share Permissions tab, as shown in Figure 2-7.

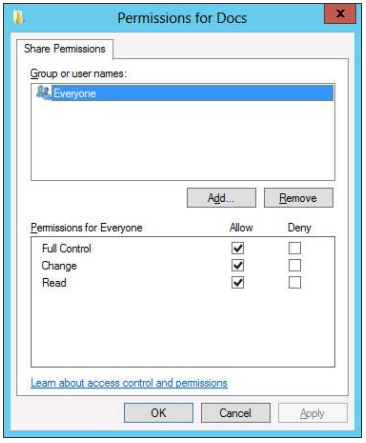

FIGURE 2-7 The Share Permissions tab for a shared folder.

Using this interface, you can add security principals and allow or deny them the three share permissions. To set share permissions using Server Manager, either while creating a share or modifying an existing one, use the following procedure.

1. Log on to Windows Server 2012 and launch Server Manager.
2. Click the File and Storage Services icon and, in the submenu that appears, click Shares to open the Shares home page.
3. In the Shares tile, right-click a share and, from the shortcut menu, select Properties. The Properties sheet for the share opens.
4. Click Permissions. The Permissions page opens.
5. Click Customize Permissions. The Advanced Security Settings dialog box for the share opens.
6. Click the Share tab to display the interface shown in Figure 2-8.

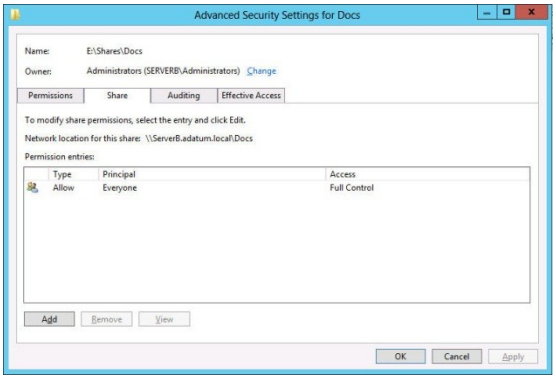

FIGURE 2-8 The Share tab of the Advanced Security Settings dialog box for a share in Server Manager.

7. Click Add to open a Permission Entry dialog box for the share.
8. Click the Select A Principal link to display the Select User, Computer, Service Account, Or Group dialog box.
9. Type the name of or search for the security principal to which you want to assign share permissions and click OK. The security principal you specified appears in the Permission Entry dialog box.
10. Select the type of permissions you want to assign (Allow or Deny).
11. Select the check boxes for the permissions you want to assign and click OK.
12. The new ACE you just created appears in the Advanced Security Settings dialog box.

Note **BYPASSING SHARE PERMISSIONS**

As discussed later in this lesson, many file server administrators simply leave the Allow Full Control share permission to the Everyone special identity in place, essentially

bypassing the share permission system, and rely solely on NTFS permissions for granular file system protection.

13. Click OK to close the Advanced Security Settings dialog box.
14. Click OK to close the share's Properties sheet.
15. Close the Server Manager window.

Understanding NTFS authorization

The majority of Windows installations today use the NTFS and ReFS file systems, as opposed to FAT32. One of the main advantages of NTFS and ReFS is that they support permissions, which FAT32 does not. As described earlier in this chapter, every file and folder on an NTFS or ReFS drive has an ACL that consists of ACEs, each of which contains a security principal and the permissions assigned to that principal.

In the NTFS permission system, which ReFS also supports, the security principals involved are users and groups, which Windows refers to using security identifiers (SIDs). When a user attempts to access an NTFS file or folder, the system reads the user's security access token, which contains the SIDs for the user's account and all of the groups to which the user belongs. The system then compares these SIDs to those stored in the file or folder's ACEs, to determine what access the user should have. This process is called authorization.

Assigning basic NTFS permissions

Most file server administrators work with basic NTFS permissions almost exclusively because there is no need to work directly with advanced permissions for most common access control tasks.

To assign basic NTFS permissions to a shared folder, the options are essentially the same as with share permissions. You can open the folder's Properties sheet in File Explorer and select the Security tab, or you can open a share's Properties sheet in Server Manager, as in the following procedure.

1. Log on to Windows Server 2012 and launch Server Manager.
2. Open the Shares home page.

   Note **NTFS PERMISSIONS**

   NTFS permissions are not limited to shared folders. Every file and folder on an NTFS volume has permissions. Although this procedure describes the process of assigning permissions to a shared folder, you can open the Properties sheet for any folder in a File Explorer window, click the Security tab, and work with its NTFS permissions in the same way.

3. Open the Properties sheet for a share and click Permissions to open the Permissions page.

Note **NEW SHARE WIZARD**

The New Share Wizard displays this same Permissions interface on its Specify permissions to control access page. The rest of this procedure applies equally well to that page and its subsequent dialog boxes.

4. Click Customize Permissions to open the Advanced Security Settings dialog box for the share, displaying the Permissions tab, as shown in Figure 2-9. This dialog box is as close as the Windows graphical interface can come to displaying the contents of an ACL.

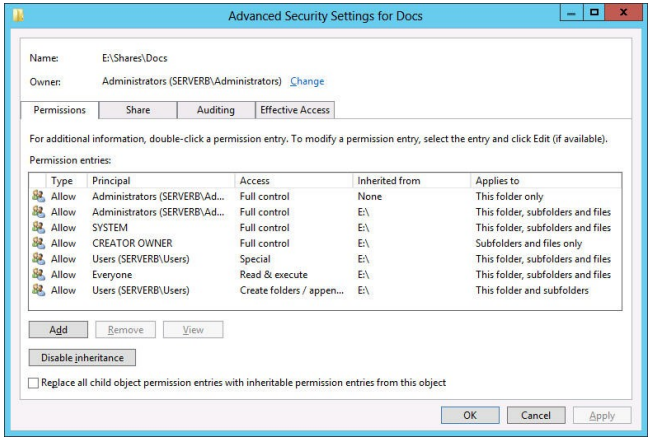

FIGURE 2-9 The Advanced Security Settings dialog box for a share in Server Manager.

5. Click Add. This opens the Permission Entry dialog box for the share.
6. Click the Select A Principal link to display the Select User, Computer, Service Account, or Group dialog box.
7. Type the name of or search for the security principal to which you want to assign share permissions and click OK. The security principal you specified appears in the Permission Entry dialog box.
8. In the Type drop-down list, select the type of permissions you want to assign (Allow or Deny).
9. In the Applies To drop-down list, specify which subfolders and files should inherit the permissions you are assigning.
10. Select the check boxes for the basic permissions you want to assign and click OK. The new ACE you just created appears in the Advanced Security Settings dialog box.
11. Click OK twice to close the Advanced Security Settings dialog box and the Properties sheet.
12. Close the Server Manager window.

Assigning advanced NTFS permissions

In Windows Server 2012, the ability to manage advanced permissions is integrated into the same interface you use to manage basic permissions.

In the Permission Entry dialog box, clicking the Show Advanced Permissions link changes the list of basic permissions to a list of advanced permissions. You can then assign advanced permissions in any combination, just as you would basic permissions.

Combining share and NTFS permissions

It is important for file server administrators to understand that the NTFS and share permission systems are completely separate from each other, and that for network users to access files on a shared NTFS drive, they must have both the correct NTFS and the correct share permissions.

The share and NTFS permissions assigned to a file or folder can conflict. For example, if a user has the NTFS Write and Modify permissions for a folder and lacks the share Change permission, that user will not be able to modify a file in that folder.

The share permission system is the simplest of the Windows permission systems, and it provides only basic protection for shared network resources. Share permissions provide only three levels of access, compared to the far more complex system of NTFS permissions. Generally speaking, network administrators prefer to use either NTFS or share permissions, but not both.

Share permissions provide limited protection, but this might be sufficient on some small networks. Share permissions might also be the only alternative on a computer with FAT32 drives, because the FAT file system does not have its own permission system.

On networks already possessing a well-planned system of NTFS permissions, share permissions are not really necessary. In this case, you can safely leave the Full Control share permission to Everyone, overriding the default Read permission, and allow the NTFS permissions to provide security. Adding share permissions to the mix would only complicate the administration process without providing any additional security.

Configuring Volume Shadow Copies

Volume Shadow Copies is a Windows Server 2012 feature that enables you to maintain previous versions of files on a server, so that if users accidentally delete or overwrite a file, they can access a copy. You can only implement Shadow Copies for an entire volume; you cannot select specific shares, folders, or files.

To configure a Windows Server 2012 volume to create Shadow Copies, use the following procedure.

1. Log on to Windows Server 2012 using an account with administrative privileges.
2. Open File Explorer. The File Explorer window appears.
3. In the Folders list, expand the Computer container, right-click a volume and, from the shortcut menu, select Configure Shadow Copies. The Shadow Copies dialog box

appears, as shown in Figure 2-10.

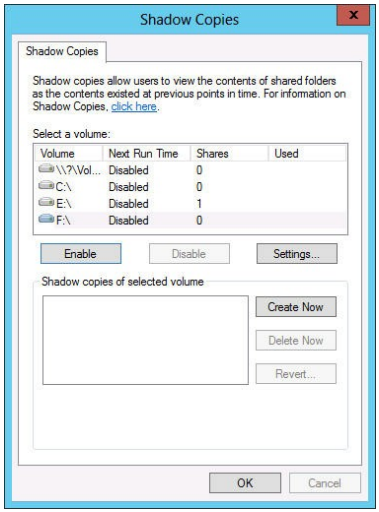

FIGURE 2-10 The Shadow Copies dialog box.

4. In the Select A Volume box, choose the volume for which you want to enable Shadow Copies. By default, when you enable Shadow Copies for a volume, the system uses the following settings:
   - The system stores the shadow copies on the selected volume.
   - The system reserves a minimum of 300 MB of disk space for the shadow copies.
   - The system creates shadow copies at 7:00 AM and 12:00 PM every weekday.
5. To modify the default parameters, click Settings to open the Settings dialog box.
6. In the Storage Area box, specify the volume where you want to store the shadow copies.
7. Specify the Maximum Size for the storage area, or choose the No Limit option. If the storage area becomes filled, the system begins deleting the oldest shadow copies.
8. Click Schedule to open the Schedule dialog box. Using the controls provided, you can modify the existing Shadow Copies tasks, delete them, or create new ones, based on the needs of your users.
9. Click OK twice to close the Schedule and Settings dialog boxes.
10. Click Enable. The system enables the Shadow Copies feature for the selected volume and creates the first copy in the designated storage area.
11. Close File Explorer.

After you complete this procedure, users can restore previous versions of files on the selected volumes from the Previous Versions tab on any file or folder's Properties sheet.

Configuring NTFS quotas

Managing disk space is a constant concern for server administrators, and one way to prevent users from monopolizing large amount of storage is to implement quotas. Windows Server 2012 supports two types of storage quotas. The more elaborate of the two is implemented as part of File Server Resource Manager. The second, simpler option is NTFS quotas.

NTFS quotas enable administrators to set a storage limit for users of a particular volume. Depending on how you configure the quota, users exceeding the limit can either be denied disk space or just receive a warning. The space consumed by individual users is measured by the size of the files they own or create.

NTFS quotas are relatively limited in that you can set only a single limit for all of the users of a volume. The feature is also limited in the actions it can take in response to a user exceeding the limit. The quotas in File Server Resource Manager, by contrast, are much more flexible in the nature of the limits you can set and the responses of the program, which can send email notifications, execute commands, and generate reports, as well as log events.

To configure NTFS quotas for a volume, use the following procedure.

1. Log on to Windows Server 2012 using an account with administrative privileges.
2. Open File Explorer. The File Explorer window appears.
3. In the Folders list, expand the Computer container, right-click a volume and, from the shortcut menu, select Properties. The Properties sheet for the volume appears.
4. Click the Quota tab to display the interface shown in Figure 2-11.

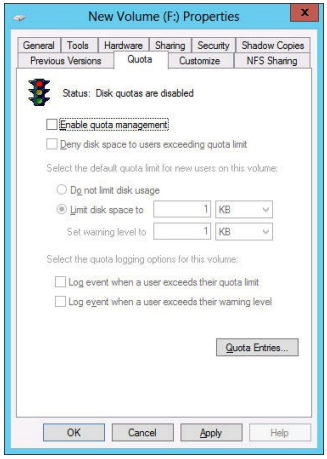

FIGURE 2-11 The Quota tab of a volume's Properties sheet.

5. Select the Enable Quota Management check box to activate the rest of the controls.
6. If you want to prevent users from consuming more than their quota of disk space, select the Deny Disk Space To Users Exceeding Quota Limit check box.
7. Select the Limit Disk Space To option and specify amounts for the quota limit and the

warning level.

8. Select the Log Event check boxes to control whether users exceeding the specified limits should trigger log entries.
9. Click OK to create the quota and close the Properties sheet.
10. Close File Explorer.

Objective Summary

- Creating folder shares makes the data stored on a file server's disks accessible to network users.
- NTFS permissions enable you to control access to files and folders by specifying the tasks individual users can perform on them. Share permissions provide rudimentary access control for all of the files on a network share. Network users must have the proper share and NTFS permissions to access file server shares.
- ABE applies filters to shared folders based on an individual user's permissions to the files and subfolders in the share. Simply put, users who cannot access a particular shared resource are unable to see that resource on the network.
- Offline Files is a Windows feature that enables client systems to maintain local copies of files they access from server shares.
- Volume Shadow Copies is a Windows Server 2012 feature that enables you to maintain previous versions of files on a server, so that if users accidentally delete or overwrite a file, they can access a copy.
- NTFS quotas enable administrators to set a storage limit for users of a particular volume.

Objective review

Answer the following questions to test your knowledge of the information in this objective. You can find the answers to these questions and explanations of why each answer choice is correct or incorrect in the "Answers" section at the end of this chapter.

1. What is the maximum number of shadow copies that a Windows Server 2012 system can maintain for each volume?
   A. 8
   B. 16
   C. 64
   D. 128
2. Which of the following terms describes the process of granting users access to file server shares by reading their permissions?
   A. authentication

B. authorization

C. enumeration

D. assignment

3. Which of the following are tasks that you can perform using the quotas in File Server Resource Manager but you can't perform with NTFS quotas?

   A. Send an email message to an administrator when users exceed their limits.

   B. Specify different storage limits for each user.

   C. Prevent users from consuming any storage space on a volume beyond their allotted limit.

   D. Generate warnings to users when they approach their allotted storage limit.

4. In the NTFS permission system, combinations of advanced permissions are also known as _____ permissions.

   A. special

   B. basic

   C. share

   D. standard

5. Which of the following best defines the role of the security principal in file system permission assignments?

   A. The only person who can access a file that has no permissions assigned to it

   B. The person responsible for creating permission policies

   C. The person assigning the permissions

   D. The person to whom the permissions are assigned

Thought experiment

In the following thought experiment, apply what you've learned about the objective to predict what steps you need to take. You can find answers to these questions in the "Answers" section at the end of this chapter.

You are working the help desk for a corporate network and you receive a call from a user named Leo, who is requesting access to the files for a new classified project called Contoso. The Contoso files are stored in a shared folder on a file server, which is locked in a secured underground data storage facility. After verifying that the user has the appropriate security clearance for the project, you create a new group on the file server called CONTOSO_USERS and add Leo's user account to that group. Then, you add the CONTOSO_USER group to the access control list for the Trinity folder on the file server, and assign the group the following NTFS permissions:

- **Allow Modify**

- **Allow Read & Execute**
- **Allow List Folder Contents**
- **Allow Read**
- **Allow Write**

Sometime later, Leo calls you back to tell you that although he is able to access the Contoso folder and read the files stored there, he has been unable to save changes back to the server.

With this in mind, what is the most likely cause of the problem?

Objective 2.2: Configure print and document services

Like the file-sharing functions discussed in the previous section, print device sharing is one of the most basic applications for which local area networks were designed.

This objective covers how to:

- Configure the Easy Print print driver
- Configure Enterprise Print Management
- Configure drivers
- Configure printer pooling
- Configure print priorities
- Configure printer permissions

Deploying a print server

Installing, sharing, monitoring, and managing a single network print device is relatively simple, but when you are responsible for dozens or even hundreds of print devices on a large enterprise network, these tasks can be overwhelming.

Understanding the Windows print architecture

It is important to understand the terms that Microsoft uses when referring to the various components of the network printing architecture. Printing in Microsoft Windows typically involves the following four components:

- **Print device** A print device is the actual hardware that produces hard-copy documents on paper or other print media. Windows Server 2012 supports both local print devices, which are directly attached to computer ports, and network interface print devices, which are connected to the network, either directly or through another

computer.

- **Printer** In Windows, a printer is the software interface through which a computer communicates with a print device. Windows Server 2012 supports numerous physical interfaces, including Universal Serial Bus (USB), IEEE 1394 (FireWire), parallel (LPT), serial (COM), Infrared Data Access (IrDA), Bluetooth ports, and network printing services such as lpr, Internet Printing Protocol (IPP), and standard TCP/IP ports.
- **Print server** A print server is a computer (or standalone device) that receives print jobs from clients and sends them to print devices that are either locally attached or connected to the network.
- **Printer driver** A printer driver is a device driver that converts the print jobs generated by applications into an appropriate string of commands for a specific print device. Printer drivers are designed for a specific print device and provide applications with access to all of the print device's features.

NOTE **Printing Nomenclature**

"Printer" and "print device" are the most commonly misused terms in the Windows printing vocabulary. Obviously, many sources use "printer" to refer to the printing hardware. However, in Windows, printer and print device are not equivalents. For example, you can add a printer to a Windows Server 2012 computer without a physical print device being present. The computer can then host the printer, print server, and printer driver. These three components enable the computer to process the print jobs and store them in a print queue until the print device is available.

Understanding Windows printing

These four components work together to process the print jobs produced by Windows applications and turn them into hard-copy documents, as shown in Figure 2-12.

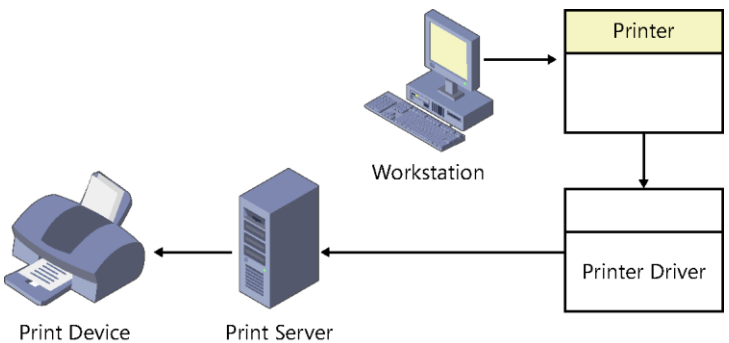

FIGURE 2-12 The Windows print architecture.

Before you can print documents in Windows, you must install at least one printer. To install a printer in Windows, you must do the following:

- Select the print device's specific manufacturer and model.
- Specify the port (or other interface) the computer will use to access the print device.
- Supply a printer driver specifically created for that print device.

When you print a document in an application, you select the printer that will be the destination for the print job.

The printer is associated with a printer driver that takes the commands generated by the application and converts them into a printer control language (PCL), a language understood by the printer. PCLs can be standardized, like the PostScript language, or they can be proprietary languages developed by the print device manufacturer.

The printer driver enables you to configure the print job to use the various capabilities of the print device. These capabilities are typically incorporated into the printer's Properties sheet. For example, your word-processing application does not know if your print device is color, monochrome, or supports duplex printing. It is the printer driver that provides support for print device features such as these.

After the printer processes a print job, it stores the job in a print queue, known as a spooler. Depending on the arrangement of the printing components, the spooled jobs might be in PCL format, ready to go to the print device, or in an interim format, in which case the printer driver must process the spooled jobs into the PCL format before sending them to the device. If other jobs are waiting to be printed, a new job might wait in the spooler for some time. When the server finally sends the job to the print device, the device reads the PCL commands and produces the hard-copy document.

Windows printing flexibility

The flexibility of the Windows print architecture is manifested in the different ways that you can deploy the four printing components. A single computer can perform all of the roles (except for the print device, of course), or you can distribute them across the network. The following sections describe four fundamental configurations that are the basis of most Windows printer deployments. You can scale these configurations up to accommodate a network of virtually any size.

DIRECT PRINTING

The simplest print architecture consists of one print device connected to one computer, also known as a locally attached print device, as shown in Figure 2-13. When you connect a print device directly to a Windows Server 2012 computer and print from an application running on that system, the computer supplies the printer, printer driver, and print server functions.

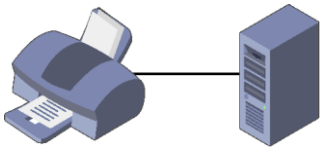

FIGURE 2-13 A locally attached print device.

LOCALLY ATTACHED PRINTER SHARING

In addition to printing from an application running on that computer, you can also share the printer (and the print device) with other users on the same network. In this arrangement, the computer with the locally attached print device functions as a print server. Figure 2-14 shows the other computers on the network, the print clients.

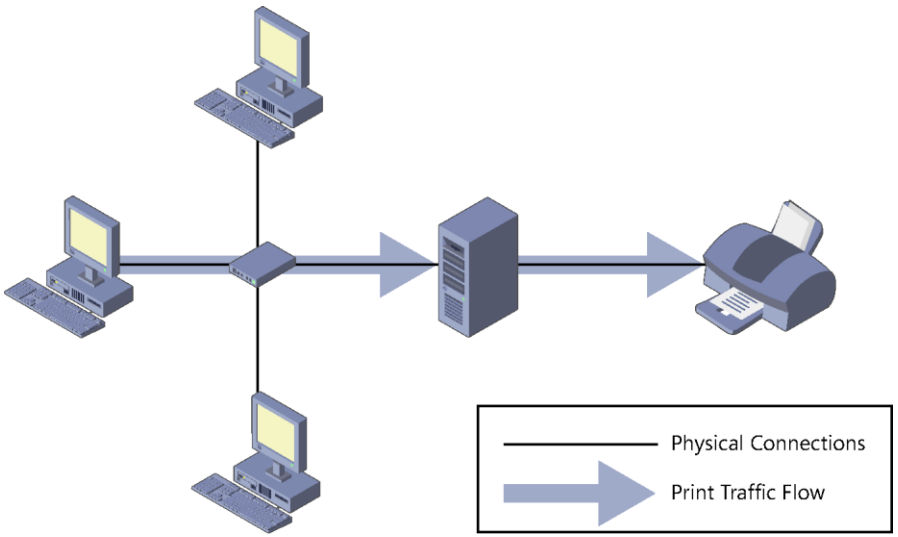

FIGURE 2-14 Sharing a locally attached printer.

In the default Windows Server 2012 printer-sharing configuration, each client uses its own printer and printer driver. As before, the application running on the client computer sends the print job to the printer and the printer driver renders the job, based on the capabilities of the print device.

The main advantage of this printing arrangement is that multiple users, located anywhere on the network, can send jobs to a single print device, connected to a computer functioning as

a print server. The downside is that processing the print jobs for many users can impose a significant burden on the print server. Although any Windows computer can function as a print server, you should use a workstation for this purpose only when you have no more than a handful of print clients to support or a very light printing volume.

NETWORK-ATTACHED PRINTING

The printing solutions discussed thus far involve print devices connected directly to a computer using a USB or other port. Print devices do not necessarily have to be attached to computers, however. You can connect a print device directly to the network, instead. Many print device models are equipped with network interface adapters, enabling you to attach a standard network cable. Some print devices have expansion slots into which you can install a network printing adapter purchased separately. Finally, for print devices with no networking capabilities, standalone network print servers are available, which connect to the network and enable you to attach one or more print devices. Print devices so equipped have their own IP addresses and typically an embedded Web-based configuration interface.

With network-attached print devices, the primary deployment decision that the administrator must make is to decide which computer will function as the print server. One simple, but often less than practical, option is to let each print client function as its own print server, as shown in Figure 2-15. Each client processes and spools its own print jobs, connects to the print device using a TCP (Transmission Control Protocol) port, and sends the jobs directly to the device for printing.

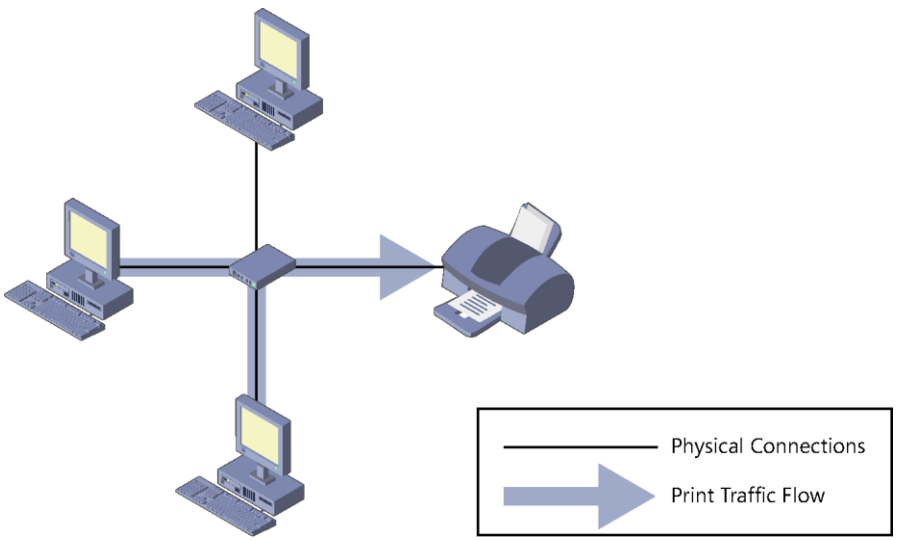

FIGURE 2-15 A network-attached print device with multiple print servers.

Even individual end users with no administrative assistance will find this arrangement simple to set up. However, the disadvantages are many, including the following:

- Users examining the print queue see only their own jobs.
- Users are oblivious of the other users accessing the print device. They have no way of knowing what other jobs have been sent to the print device, or how long it will be until the print device completes their jobs.
- Administrators have no way of centrally managing the print queue because each client has its own print queue.
- Administrators cannot implement advanced printing features, such as printer pools or remote administration.
- Error messages appear only on the computer that originated the job the print device is currently processing.
- All print job processing is performed by the client computer, rather than being partially offloaded to an external print server.

For these reasons, this arrangement is only suitable for small workgroup networks that do not have dedicated administrators supporting them.

NETWORK-ATTACHED PRINTER SHARING

The other, far more popular option for network-attached printing is to designate one computer as a print server and use it to service all of the print clients on the network. To do this, you install a printer on one computer, the print server, and configure it to access the print device directly through a TCP port. You then share the printer, just as you would a locally attached print device, and configure the clients to access the print share.

As you can see in Figure 2-16, the physical configuration is exactly the same as in the previous arrangement, but the logical path the print jobs take on the way to the print device is different. Instead of going straight to the print device, the jobs go to the print server, which spools them and sends them to the print device in order.

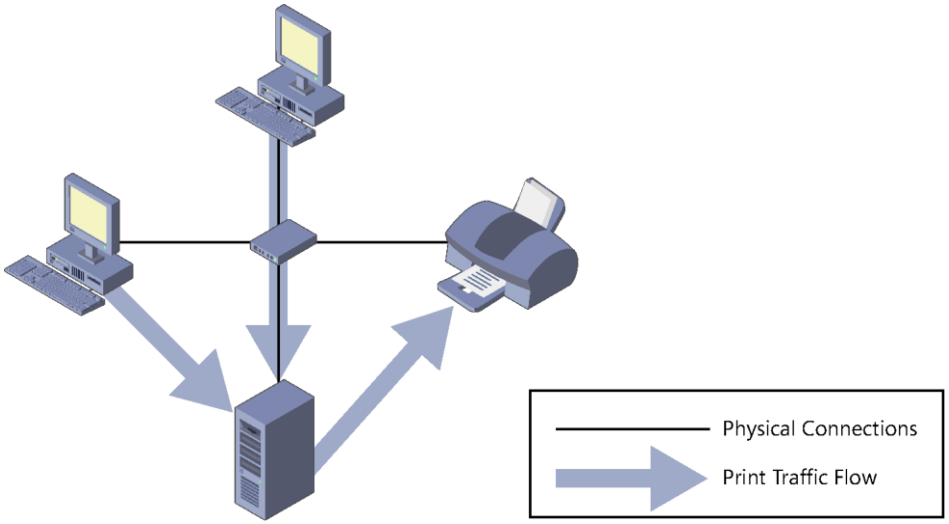

FIGURE 2-16 A network-attached print device with a single, shared print server.

With this arrangement, virtually all of the disadvantages of the multiple print server arrangement become advantages, as follows:

- All of the client jobs are stored in a single print queue, so that users and administrators can see a complete list of the jobs waiting to be printed.
- Part of the job rendering burden is shifted to the print server, returning control of the client computer to the user more quickly.
- Administrators can manage all of the queued jobs from a remote location.
- Print error messages appear on all client computers.
- Administrators can implement printer pools and other advanced printing features.
- Administrators can manage security, auditing, monitoring, and logging functions from a central location.

ADVANCED PRINTING CONFIGURATIONS

Administrators can use the four configurations described in the previous sections as building blocks to create printing solutions for their networks. Many possible variations can be used to create a network printing architecture that supports your organization's needs. Some of the more advanced possibilities are as follows:

- You can connect a single printer to multiple print devices, creating what is called a printer pool. On a busy network with many print clients, the print server can distribute large numbers of incoming jobs among several identical print devices to provide more timely service and fault tolerance.

- You can connect multiple print devices that support different forms and paper sizes to a single print server, which will distribute jobs with different requirements to the appropriate print devices.
- You can connect multiple print servers to a single print device. By creating multiple print servers, you can configure different priorities, security settings, auditing, and monitoring parameters for different users. For example, you can create a high-priority print server for company executives, while junior users send their jobs to a lower priority server. This ensures that the executives' jobs get printed first, even if the servers are both connected to the same print device.

Sharing a printer

Using Windows Server 2012 as a print server can be simple or complex, depending on how many clients the server has to support and how much printing they do. For a home or small business network, in which a handful of users need occasional access to the printer, no special preparation is necessary. However, if the computer must support heavy printer use, hardware upgrades, such as additional disk space or system memory, might be needed.

You might also consider making the computer a dedicated print server. In addition to memory and disk space, using Windows Server 2012 as a print server requires processor clock cycles, just like any other application. On a server handling heavy print traffic, other roles and applications are likely to experience substantial performance degradation. If you need a print server to handle heavy traffic, consider dedicating the computer to print server tasks only and deploying other roles and applications elsewhere.

On a Windows Server 2012 computer, you can share a printer as you are installing it, or at any time afterward. On older printers, initiate the installation process by launching the Add Printer Wizard from the Printers control panel. However, most of the print devices on the market today use either a USB connection to a computer or an Ethernet connection to a network.

In the case of a USB-connected printer, you plug the print device into a USB port on the computer and turn the device on to initiate the installation process. Manual intervention is only required when Windows Server 2012 does not have a driver for the print device.

For network-attached print devices, an installation program supplied with the product locates the print device on the network, installs the correct drivers, creates a printer on the computer, and configures the printer with the proper IP address and other settings.

After the printer is installed on the Windows Server 2012 computer that will function as your print server, you can share it with your network clients, using the following procedure.

1. Log on to Windows Server.
2. Open the Devices and Printers control panel. The Devices and Printers window appears.

3. Right-click the icon for the printer you want to share and, from the shortcut menu, select Printer Properties. The printer's Properties sheet appears.

   Note **PROPERTIES**

   The shortcut menu for every printer provides access to two Properties sheets. The Printer Properties menu item opens the Properties sheet for the printer and the Properties menu item opens the Properties sheet for the print device.

4. Click the Sharing tab,
5. Select the Share This Printer check box. The printer name appears in the Share Name text box. You can accept the default name or supply one of your own.
6. Select one or both of the following optional check boxes:
   - **Render Print Jobs On Client Computers** Minimizes the resource utilization on the print server by forcing the print clients to perform the bulk of the print processing.
   - **List In The Directory** Creates a new printer object in the Active Directory Domain Services (AD DS) database, enabling domain users to locate the printer by searching the directory. This option only appears when the computer is a member of an AD DS domain.
7. Click Additional Drivers to open the Additional Drivers dialog box. This dialog box enables you to load printer drivers for other Windows platforms, such as Itanium and x86. When you install the alternate drivers, the print server automatically supplies them to clients running those operating system versions.
8. Select any combination of the available check boxes and click OK. For each check box you select, Windows Server 2012 displays a Printer Drivers dialog box.
9. In each Printer Drivers dialog box, type in or browse to the location of the printer drivers for the selected operating system, and then click OK.
10. Click OK to close the Additional Drivers dialog box.
11. Click OK to close the Properties sheet for the printer. The printer icon in the Printers control panel now includes a symbol indicating that it has been shared.
12. Close the control panel.

At this point, the printer is available to clients on the network.

Managing printer drivers

Printer drivers are the components that enable your computers to manage the capabilities of your print devices. When you install a printer on a server running Windows Server 2012, you install a driver that other Windows computers can use as well.

Point and Print is the Windows function that enables clients to access the printers installed on print servers. A user on a workstation can select a printer on a server and Windows will

automatically install the driver that the client needs to process its own print jobs and send them to that printer.

The printer drivers you install on Windows Server 2012 are the same drivers that Windows workstations and other server versions use, with one stipulation. As a 64-bit platform, Windows Server 2012 uses 64-bit device drivers, which are suitable for other computers running 64-bit versions of Windows. If you have 32-bit Windows systems on your network, however, you must install a 32-bit driver on the server for those systems to use.

The Additional Drivers dialog box, accessible from the Sharing tab of a printer's Properties sheet, enables you to install drivers for other processor platforms. However, you must install those drivers from a computer running on the alternative platform. In other words, to install a 32-bit driver for a printer on a server running Windows Server 2012, you must access the printer's Properties sheet from a computer running 32-bit version of Windows. You can do this by accessing the printer directly through the network using File Explorer, or by running the Print Management snap-in on the 32-bit system and using it to manage your Windows Server 2012 print server.

NOTE **Installing drivers**

For the server to provide drivers supporting different platforms to client computers, you must make sure when installing the drivers for the same print device that they have the exact same name. For example, Windows Server 2012 will treat "HP LaserJet 5200 PCL6" and "HP LaserJet 5200 PCL 6" as two completely different drivers. The names must be identical for the server to apply them properly.

Using remote access Easy Print

When a Remote Desktop Services client connects to a server, it runs applications using the server's processor(s) and memory. However, if that client wants to print a document from one of those applications, it wants the print job to go to the print device connected to the client computer.

The component that enables Remote Desktop clients to print to their local print devices is called Easy Print. Easy Print takes the form of a printer driver that is installed on the server along with the Remote Desktop Session Host role service.

The Remote Desktop Easy Print driver appears in the Print Management snap-in automatically, but it is not associated with a particular print device. Instead, the driver functions as a redirector, enabling the server to access the printers on the connected clients.

Easy Print requires no configuration other than the installation of the Remote Desktop Services role. However, once it is operational, it provides the server administrator with additional access to the printers on the Remote Desktop clients.

When a Remote Desktop client connects to a server, using the Remote Desktop Connection program or the RD Web Access site, the printers installed on the client system are redirected to

the server and appear in the Print Management snap-in as redirected server printers, as shown in Figure 2-17.

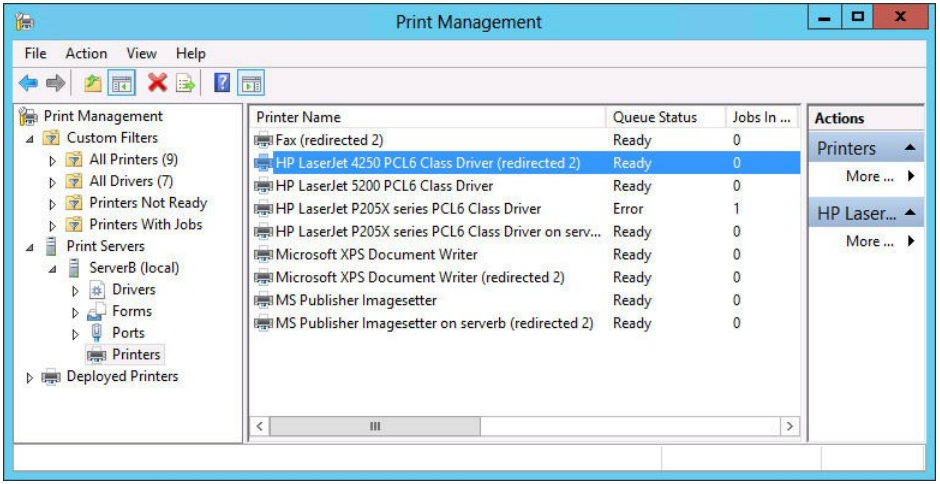

FIGURE 2-17 Printers redirected by Easy Print on a Remote Desktop server.

A client running an application on the server can therefore print to a local print device using the redirected printer. Administrators can also open the Properties sheet for the redirected printer in the usual manager and manipulate its settings.

Configuring printer security

Like folder shares, clients must have the proper permissions to access a shared printer. Printer permissions are much simpler than NTFS permissions; they basically dictate whether users are allowed to merely use the printer, manage documents submitted to the printer, or manage the properties of the printer itself. To assign permissions for a printer, use the following procedure.

1. Log on to Windows Server 2012 using a domain account with Administrator privileges.
2. Open Control Panel and select Hardware> Devices and Printers. The Devices and Printers window appears.
3. Right-click one of the printer icons in the window and, from the shortcut menu, select Printer Properties. The printer's Properties sheet appears.
4. Click the Security tab. The top half of the display lists all of the security principals currently possessing permissions to the selected printer. The bottom half lists the permissions held by the selected security principal.
5. Click Add. The Select Users, Computers, Or Groups dialog box appears.
6. In the Enter The Object Names To Select text box, type a user or group name, and then click OK. The user or group appears in the Group Or User Names list.
7. Select the security principal you added, and select or clear the check boxes in the bottom half of the display to Allow or Deny the user any of the basic permissions.

8. Click OK to close the Properties sheet.
9. Close Control Panel.

Like NTFS permissions, there are two types of printer permissions: basic and advanced. Each of the three basic permissions consists of a combination of advanced permissions.

Managing documents

By default, all printers assign the Allow Print permission to the Everyone special identity, which enables all users to access the printer and manage their own documents. Users who possess the Allow Manage Documents permission can manage any users' documents.

Managing documents refers to pausing, resuming, restarting, and cancelling documents that are currently waiting in a print queue. Windows Server 2012 provides a print queue window for every printer, which enables users to view the jobs that are currently waiting to be printed. To manage documents, use the following procedure.

1. Log on to Windows Server 2012.
2. Open Control Panel and select Hardware> Devices and Printers. The Devices and Printers window appears.
3. Right-click one of the printer icons and, from the shortcut menu, select See What's Printing. A print queue window named for the printer appears, as shown in Figure 2-18.

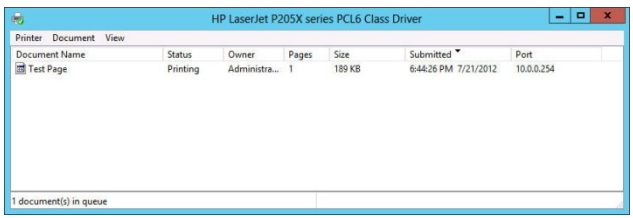

FIGURE 2-18 A Windows Server 2012 print queue window.

4. Select one of the menu items to perform the associated function.
5. Close the print queue window.
6. Close Control Panel.

Managing printers

Users with the Allow Manage This Printer permission can go beyond manipulating queued documents; they can reconfigure the printer itself. Managing a printer refers to altering the operational parameters that affect all users and controlling access to the printer.

Generally, most of the software-based tasks that fall under the category of managing a printer are those you perform once while setting up the printer for the first time. Day-to-day printer management is more likely to involve physical maintenance, such as clearing print jams,

reloading paper, and changing toner or ink cartridges. However, the following sections examine some of the printer manager's typical configuration tasks.

Setting printer priorities

In some cases, you might want to give certain users in your organization priority access to a print device so that when print traffic is heavy, their jobs are processed before those of other users. To do this, you must create multiple printers, associate them with the same print device, and then modify their priorities, as described in the following procedure.

1. Log on to Windows Server 2012 using an account with the Manage This Printer permission.
2. Open Control Panel and select Hardware> Devices and Printers. The Devices and Printers window opens.
3. Right-click one of the printer icons and then, from the shortcut menu, select Printer Properties. The Properties sheet for the printer appears.
4. Click the Advanced tab, as shown in Figure 2-19.

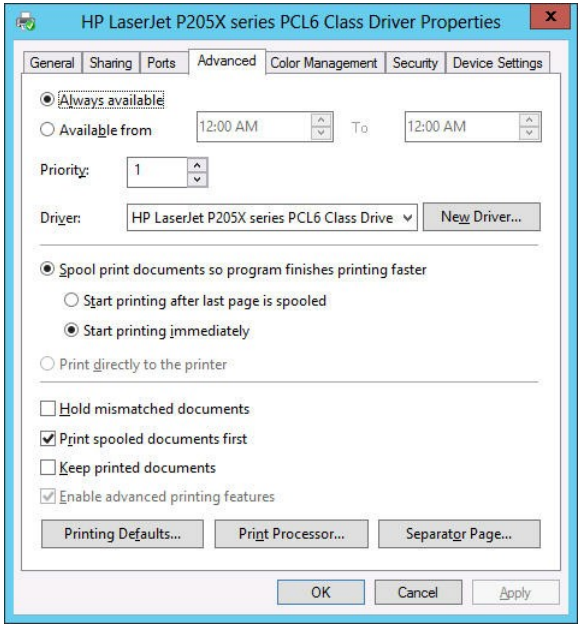

FIGURE 2-19 The Advanced tab of a printer's Properties sheet.

5. Set the Priority spin box to a number representing the highest priority you want to set for the printer. Higher numbers represent higher priorities. The highest possible priority is 99.

Note **PRINTER PRIORITIES**

The values of the Priority spin box do not have any absolute significance; they are pertinent only in relation to each other. As long as one printer has a higher priority value than another, the server will process its print jobs first. In other words, it doesn't matter if the higher priority value is 9 or 99, as long as the lower priority value is less.

6. Click the Security tab.
7. Add the users or groups that you want to provide with high-priority access to the printer and assign the Allow Print permission to them.
8. Revoke the Allow Print permission from the Everyone special identity.
9. Click OK to close the Properties sheet.
10. Create an identical printer using the same printer driver and pointing to the same print device. Leave the Priority setting at its default value of 1 and leave the default permissions in place.
11. Rename the printers, specifying the priority assigned to each one.
12. Close Control Panel.

Inform the privileged users that they should send their jobs to the high-priority printer. All jobs sent to that printer will be processed before those sent to the other, lower priority printer.

Creating a printer pool

As mentioned earlier, a printer pool increases the production capability of a single printer by connecting it to multiple print devices. When you create a printer pool, the print server sends each incoming job to the first print device it finds that is not busy. This effectively distributes the jobs among the available print devices, providing users with more rapid service.

To configure a printer pool, use the following procedure.

1. Log on to Windows Server 2012 using an account with the Manage Printer permission.
2. Open Control Panel and select Hardware> Devices and Printers. The Devices and Printers window opens.
3. Right-click one of the printer icons and then, from the shortcut menu, select Printer Properties. The Properties sheet for the printer appears.
4. Click the Ports tab.
5. Select all of the ports to which the print devices are connected.
6. Select the Enable Printer Pooling check box, and then click OK.
7. Close Control Panel.

To create a printer pool, you must have at least two identical print devices, or at least print devices that use the same printer driver. The print devices must be in the same location, because there is no way to tell which print device will process a given document. You must

also connect all of the print devices in the pool to the same print server. If the print server is a Windows Server 2012 computer, you can connect the print devices to any viable ports.

Using the Print and Document Services role

All of the printer sharing and management capabilities discussed in the previous sections are available on any Windows Server 2012 computer in its default installation configuration. However, installing the Print and Document Services role on the computer provides additional tools that are particularly useful to administrators involved with network printing on an enterprise scale.

When you install the Print and Document Services role using Server Manager's Add Roles and Features Wizard, a Select Role Services page appears, enabling you to select from the following options:

- **Print Server** Installs the Print Management console for Microsoft Management Console (MMC), which enables administrators to deploy, monitor, and manage printers throughout the enterprise.
- **Distributed Scan Server** Enables the computer to receive documents from network-based scanners and forward them to the appropriate users.
- **Internet Printing** Creates a website that enables users on the Internet to send print jobs to shared Windows printers.
- **LPD Service** Enables UNIX clients running the line printer remote (LPR) program to send their print jobs to Windows printers.

As always, Windows Server 2012 adds a new icon to the Server Manager navigation pane when you install a role. The Print Services home page contains a filtered view of print-related event log entries, a status display for the role-related system services and role services, and performance counters.

The Print Management snap-in for MMC, an administrative tool, consolidates the controls for the printing components throughout the enterprise into a single console. With this tool, you can access the print queues and Properties sheets for all of the network printers in the enterprise, deploy printers to client computers using Group Policy, and create custom views that simplify the process of detecting print devices that need attention due to errors or depleted consumables.

Windows Server 2012 installs the Print Management console when you add the Print and Document Services role to the computer. You can also install the console without the role by adding the Print and Document Services Tools feature, found under Remote Server Administration Tools > Role Administration Tools in the Add Roles and Features Wizard.

The following sections demonstrate some of the administration tasks you can perform with the Print Management console.

Adding print servers

By default, the Print Management console displays only the local machine in its list of print servers. Each print server has four nodes beneath it, as shown in Figure 2-20, listing the drivers, forms, ports, and printers associated with that server.

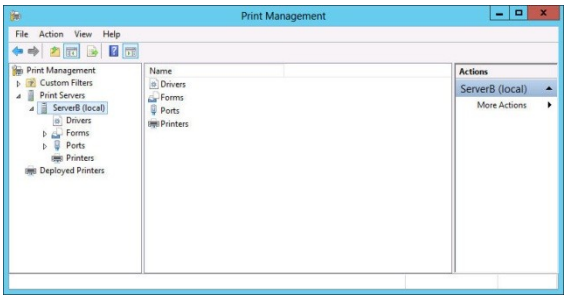

FIGURE 2-20 A print server displayed in the Print Management console.

To manage other print servers and their printers, you must add them to the console, using the following procedure.

1. Log on to Windows Server 2012 and launch Server Manager.
2. Click Tools > Print Management to open the Print Management console.
3. Right-click the Print Servers node and, from the shortcut menu, click Add/Remove Servers to open the Add/Remove Servers dialog box.
4. In the Specify Print Server box, click Browse. The Select Print Server dialog box opens.
5. Select the print server you want to add to the console and click Select Server. The server you selected appears in the Add Server text box in the Add/Remove Servers dialog box.
6. Click Add To List. The server you selected appears in the Print Servers list.
7. Click OK. The server appears under the Print Servers node.
8. Close Control Panel.

You can now manage the printers associated with the server you have added to the console.

Viewing printers

One of the major problems for printing administrators on large enterprise networks is keeping track of dozens or hundreds of print devices, all in frequent use, and all needing attention on a regular basis. Whether the maintenance required is a major repair, replenishing ink or toner, or just filling the paper trays, print devices will not get the attention they need until an administrator is aware of the problem.

The Print Management console provides a multitude of ways to view the printing components associated with the print servers on the network. To create views, the console

takes the complete list of printers and applies various filters to it, selecting which printers to display. Under the Custom Filters node, there are four default filters, as follows:

- **All Printers** Contains a list of all the printers hosted by all of the print servers added to the console
- **All Drivers** Contains a list of all the printer drivers installed on all of the print servers added to the console
- **Printers Not Ready** Contains a list of all printers that are not reporting a Ready status
- **Printers With Jobs** Contains a list of all the printers that currently have jobs waiting in the print queue

Views such as Printer Not Ready are a useful way for administrators to identify printers that need attention, without having to browse individual print servers or search through a long list of every printer on the network. In addition to these defaults, you can create your own custom filters.

Managing printers and print servers

After you have used filtered views to isolate the printers you want to examine, selecting a printer displays its status, the number of jobs currently in its print queue, and the name of the print server hosting it. If you right-click the filter in the scope pane and select Show Extended View from the shortcut menu, an additional pane appears containing the contents of the selected printer's queue. You can manipulate the queued jobs just as you would from the Print Queue window in the Print Server console.

The Print Management console also enables administrators to access the configuration interface for any printer or print server appearing in any of its displays. Right-clicking a printer or print server anywhere in the console interface and then selecting Properties from the shortcut menu displays the same Properties sheet you would see on the print server computer itself. Administrators can then configure printers and print servers without having to travel to the site of the print server or establish a Remote Desktop connection to the print server.

Deploying printers with Group Policy

Configuring a print client to access a shared printer is a simple matter of browsing the network or the AD DS tree and selecting the printer. However, when you have to configure hundreds or thousands of print clients, the task becomes more complicated. AD DS helps simplify the process of deploying printers to large numbers of clients.

Publishing printers in the AD DS database enables users and administrators to search for printers by name, location, or model (if you populate the Location and Model fields in the printer object). To create a printer object in the AD DS database, you can either select the List In The Directory check box while sharing the printer or right-click a printer in the Print Management console and, from the shortcut menu, select List In Directory.

To use AD DS to deploy printers to clients, you must configure the appropriate policies in a Group Policy object (GPO). You can link a GPO to any domain, site, or organizational unit (OU) in the AD DS tree. When you configure a GPO to deploy a printer, all of the users or computers in that domain, site, or OU will receive the printer connection when they log on, by default.

To deploy printers with Group Policy, use the following procedure.

1. Log on to Windows Server 2012 using a domain account with Administrator privileges. The Server Manager window opens.
2. Open the Print Management console.
3. Right-click a printer in the console's scope pane and, from the shortcut menu, select Deploy With Group Policy. The Deploy With Group Policy dialog box appears, as shown in Figure 2-21.

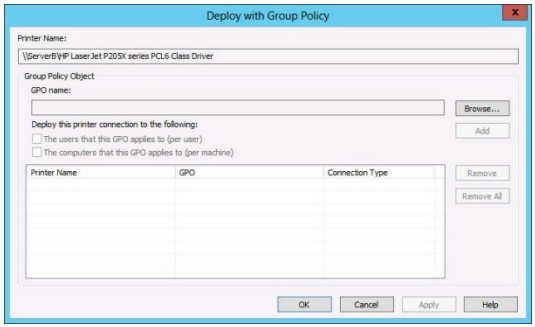

FIGURE 2-21 The Deploy With Group Policy dialog box.

4. Click Browse to open the Browse For A Group Policy Object dialog box.
5. Select the GPO you want to use to deploy the printer and click OK. The GPO you selected appears in the GPO Name field.
6. Select the appropriate check box to select whether to deploy the printer to the users associated with the GPO, the computers, or both, and then click Add. The new printer/GPO associations appear in the table.

   Deploying the printer to the users means that all of the users associated with the GPO will receive the printer connection, no matter what computer they use to log on. Deploying the printer to the computers means that all of the computers associated with the GPO will receive the printer connection, no matter who logs on to them.
7. Click OK. A Print Management message box appears, informing you that the operation has succeeded.
8. Click OK, then click OK again to close the Deploy With Group Policy dialog box.
9. Close Control Panel.

The next time the users running Windows Server 2008 or later and Windows Vista or later who are associated with the GPO refresh their policies or restart, they will receive the new settings and the printer will appear in the Printers control panel.

NOTE **PushPrinterConnections.exe**

Clients running earlier versions of Windows, including Windows XP and Windows Server 2003, do not support automatic policy-based printer deployments. To enable the GPO to deploy printers on these computers, you must configure the systems to run a utility called PushPrinterConnections.exe. The most convenient way to do this is to configure the same GPO you used for the printer deployment to run the program from a user logon script or machine script.

Objective summary

- Printing in Microsoft Windows typically involves the following four components: print device, printer, print server, and print driver.
- The simplest form of print architecture consists of one print device connected to one computer, known as a locally attached print device. You can share this printer (and the print device) with other users on the same network.
- With network-attached print devices, the administrator's primary deployment decision is which computer will function as the print server.
- Remote Desktop Easy Print is a driver that enables Remote Desktop clients running applications on a server to redirect their print jobs back to their local print devices.
- Printer permissions are much simpler than NTFS permissions; they basically dictate whether users are allowed to merely use the printer, manage documents submitted to the printer, or manage the properties of the printer itself.
- The Print Management snap-in for MMC is an administrative tool that consolidates the controls for the printing components throughout the enterprise into a single console.

Objective review

Answer the following questions to test your knowledge of the information in this objective. You can find the answers to these questions and explanations of why each answer choice is correct or incorrect in the "Answers" section at the end of this chapter.

1. Which of the following terms describes the software interface through which a computer communicates with a print device?

   A. Printer

   B. Print server

   C. Printer driver

   D. Print Management snap-in

2. You are setting up a printer pool on a computer running Windows Server 2012. The printer pool contains three print devices, all identical. You open the Properties dialog box for the printer and select the Enable Printer Pooling option on the Ports tab. What must you do next?
   A. Configure the LPT1 port to support three printers.
   B. Select or create the ports mapped to the three printers.
   C. On the Device Settings tab, configure the installable options to support two additional print devices.
   D. On the Advanced tab, configure the priority for each print device so that printing is distributed among the three print devices.
3. One of your print devices is not working properly, and you want to temporarily prevent users from sending jobs to the printer serving that device. What should you do?
   A. Stop sharing the printer
   B. Remove the printer from Active Directory
   C. Change the printer port
   D. Rename the share
4. You are administering a computer running Windows Server 2012 configured as a print server. Users in the Marketing group complain that they cannot print documents using a printer on the server. You view the permissions in the printer's properties. The Marketing group is allowed Manage Documents permission. Why can't the users print to the printer?
   A. The Everyone group must be granted the Manage Documents permission.
   B. The Administrators group must be granted the Manage Printers permission.
   C. The Marketing group must be granted the Print permission.
   D. The Marketing group must be granted the Manage Printers permission.
5. You are administering a print server running Windows Server 2012. You want to perform maintenance on a print device physically connected to the print server. There are several documents in the print queue. You want to prevent the documents from being printed to the printer, but you don't want users to have to resubmit the documents to the printer. What is the best way to do this?
   A. Open the printer's Properties dialog box, select the Sharing tab, and then select the Do Not Share This Printer option.
   B. Open the printer's Properties dialog box and select a port that is not associated with a print device.

C. Open the printer's queue window, select the first document, and then select Pause from the Document window.

D. Open the printer's queue window, and select the Pause Printing option from the Printer menu.

Thought experiment

In the following thought experiment, apply what you've learned about the objective to predict what steps you need to take. You can find answers to these questions in the "Answers" section at the end of this chapter.

You are a desktop support technician for a law firm with a group of ten legal secretaries who provide administrative support to the attorneys. All of the secretaries use a single, shared, high-speed laser printer that is connected to a dedicated Windows print server. The secretaries print multiple copies of large documents on a regular basis, and although the laser printer is fast, it is kept running almost constantly. Sometimes the secretaries have to wait 20 minutes or more after submitting a print job for their documents to reach the top of the queue. The office manager has offered to purchase additional printers for the department. However, the secretaries are accustomed to simply clicking Print, and don't like the idea of having to examine multiple print queues to determine which one has the fewest jobs before submitting a document.

With this in mind, answer the following question:

What can you do to provide the department with a printing solution that will enable the secretaries to utilize additional printers most efficiently?

Objective 2.3: Configure servers for remote management

Windows Server 2012 is designed to facilitate remote server management, so that administrators rarely if ever have to work directly at the server console. This conserves server resources that can better be devoted to applications.

This objective covers how to:

- Configure WinRM
- Configure down-level server management
- Configure servers for day-to-day management tasks
- Configure multiserver management

- Configure Server Core
- Configure Windows Firewall

Using Server Manager for remote management

Server Manager has been the primary server administration tool for Windows Server ever since Windows Server 2003. The most obvious improvement to the Server Manager tool in Windows Server 2012 is the ability to perform administrative tasks on remote servers, as well as on the local system.

When you log on to a GUI installation of Windows Server 2012 with an administrative account, Server Manager loads automatically, displaying the Welcome tile. The Server Manager interface consists of a navigation pane on the left containing icons representing various views of server resources. Selecting an icon displays a home page in the right pane, which consists of a number of tiles containing information about the resource. The Dashboard page, which appears by default, contains, in addition to the Welcome tile, thumbnails that summarize the other views available in Server Manager. These other views include a page for the Local Server, one for All Servers, and others for server groups and role groups.

Adding servers

The primary difference between the Windows Server 2012 Server Manager and previous versions is the ability to add and manage multiple servers at once. Although only the local server appears in Server Manager when you first run it, you can add other servers, enabling you to manage them together. The servers you add can be physical or virtual, and can be running any version of Windows Server since Windows Server 2003. After you add servers to the interface, you can create groups containing collections of servers, such as the servers at a particular location or those performing a particular function. These groups appear in the navigation pane, enabling you to administer them as a single entity.

To add servers in Server Manager, use the following procedure.

1. Log on to the server running Windows Server 2012 using an account with administrative privileges. The Server Manager window appears.
2. In the navigation pane, click the All Servers icon to open the All Servers home page.
3. From the Manage menu, select Add Servers to open the Add Servers dialog box.
4. Select one of the following tabs to specify how you want to locate servers to add:
   - **Active Directory** Enables you to search for computers running specific operating systems in specific locations in the local AD DS domain.
   - **DNS** Enables you to search for servers in your currently configured Domain Name System (DNS) server.
   - **Import** Enables you to supply a text file containing the names or IP addresses of the servers you want to add.
5. Initiate a search or upload a text file to display a list of available servers.

6. Select the servers you want to add and click the right arrow button to add them to the Selected list, as shown in Figure 2-22.

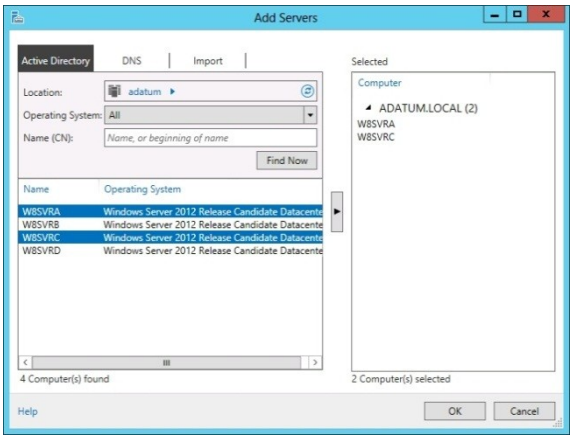

FIGURE 2-22 Selecting servers in Server Manager.

7. Click OK. The servers you selected are added to the All Servers home page.
8. Close the Server Manger console.

Once you have added remote servers to the Server Manager interface, they appear on the All Servers home page. You can then access them in a variety of ways, depending on the version of Windows the remote server is running.

Managing Windows Server 2012 servers

When you add servers running Windows Server 2012 to Server Manager, you can immediately begin using the Add Roles and Features Wizard to install roles and features on any of the servers you have added.

You can also perform other administrative tasks, such as configuring network interface card (NIC) teaming and restarting the server, because Windows Remote Management (WinRM) is enabled by default on Windows Server 2012.

CONFIGURING WINRM

WinRM enables administrators to manage a computer from a remote location using tools based on Windows Management Instrumentation (WMI) and Windows PowerShell. If the default WinRM setting has been modified, or if you want to change it manually, you can do so through the Server Manager interface.

On the Local Server home page, the Properties tile contains a Remote Management indicator that specifies the server's current WinRM status. To change the WinRM state, click the Remote Management hyperlink to open the Configure Remote Management dialog box.

Clearing the Enable Remote Management Of This Server From Other Computers check box disables WinRM, and selecting the check box enables it.

NOTE **Using PowerShell**

To manage WinRM from a PowerShell session, as in the case of a computer with a Server Core installation, use the following command:

```
Configure-SMRemoting.exe –Get|–Enable|–Disable
```

- **-Get** **Displays the current WinRM status**
- **-Enable** **Enables WinRM**
- **-Disable** **Disables WinRM**

CONFIGURING WINDOWS FIREWALL

However, if you attempt to launch MMC snap-ins targeting a remote server, such as the Computer Management console, you will receive an error because of the default Windows Firewall settings in Windows Server 2012. MMC uses the Distributed Component Object Model (DCOM) for remote management, instead of WinRM, and these settings are not enabled by default.

To address this problem, you must enable the following inbound Windows Firewall rules on the remote server you want to manage:

- COM+ Network Access (DCOM-In)
- Remote Event Log Management (NP-In)
- Remote Event Log Management (RPC)
- Remote Event Log Management (RPC-EPMAP)

To modify the firewall rules on the remote system, you can use any one of the following methods:

- Open the Windows Firewall with Advanced Security MMC snap-in on the remote server (if it is a Full GUI installation).
- Run the Netsh AdvFirewall command from an administrative command prompt.
- Use the NetSecurity module in Windows PowerShell.
- Create a GPO containing the appropriate settings and apply it to the remote server.

NOTE **Using PowerShell**

To configure the Windows Firewall rules required for remote server management using DCOM on a Server Core installation, you can use the following Windows PowerShell syntax:

```
Set-NetFirewallRule –name <rule name> –enabled True
```

To obtain the PowerShell names for the preconfigured rules in Windows Firewall, you use the Get-NetFirewallRule command. The resulting commands to enable the four rules listed earlier are therefore as follows:

```
Set-NetFirewallRule –name

 ComPlusNetworkAccess-DCOM-In –enabled True

Set-NetFirewallRule –name

 RemoteEventLogSvc-In-TCP –enabled True

Set-NetFirewallRule –name RemoteEventLogSvc-NP-In-TCP

–enabled True

Set-NetFirewallRule –name

 RemoteEventLogSvc-RPCSS-In-TCP –enabled True
```

For the administrator interested in remote management solutions, the Group Policy method provides distinct advantages. Not only does it enable you to configure the firewall on the remote system without accessing the server console directly, but it also can also configure the firewall on Server Core installations without having to work from the command line. Finally, and possibly most important for large networks, you can use Group Policy to configure the firewall on all of the servers you want to manage at once.

To configure Windows Firewall settings using Group Policy, use the following procedure. This procedure assumes that the server is a member of an AD DS domain and has the Group Policy Management feature installed.

1. Log on to the server running Windows Server 2012 using an account with administrative privileges. The Server Manager window appears.
2. Open the Group Policy Management console and create a new GPO, giving it a name like Server Firewall Configuration.
3. Open the GPO you created using the Group Policy Management Editor.

MORE INFO **GPOs**

For more detailed information on creating GPOs and linking them to other objects, see Objective 6.1, "Create Group Policy Objects (GPOs)."

4. Browse to the Computer Configuration\Policies\Windows Settings\Security Settings\Windows Firewall with Advanced Security\Inbound Rules node.
5. Right-click Inbound Rules and, from the shortcut menu, select New Rule. The New Inbound Rule Wizard appears, displaying the Rule Type page.
6. Select the Predefined option and, in the drop-down list, select COM+ Network Access and click Next. The Predefined Rules page opens.

7. Click Next to open the Action page.
8. Leave the Allow The Connection option selected and click Finish. The rule appears in the Group Policy Management Editor console.
9. Open the New Inbound Rule Wizard again.
10. Select the Predefined option and, in the drop-down list, select Remote Event Log Management. Click Next. The Predefined Rules page opens, displaying the three rules in the Remote Event Log Management group.
11. Leave the three rules selected and click Next to open the Action page.
12. Leave the Allow The Connection option selected and click Finish. The three rules appears in the Group Policy Management Editor console.
13. Close the Group Policy Management Editor.
14. In the Group Policy Management console, link the Server Firewall Configuration GPO you just created to your domain.
15. Close the Group Policy Management console.

The settings in the GPO you created will be deployed to your remote servers the next time they recycle or restart, and you will be able to use MMC snap-ins, such as Computer Management and Disk Management, on them.

Managing downlevel servers

The Windows Firewall rules you have to enable for remote servers running Windows Server 2012 are also disabled by default on computers running earlier versions of Windows Server, so you have to enable them there as well.

Unlike Windows Server 2012, however, earlier versions of the operating system also lack the WinRM support needed for them to be managed using the new Server Manager.

By default, when you add servers running Windows Server 2008 or Windows Server 2008 R2 to the Windows Server 2012 Server Manager, they appear with a manageability status that reads “Online - Verify WinRM 3.0 service is installed, running, and required firewall ports are open.”

To add WinRM support to servers running Windows Server 2008 or Windows Server 2008 R2, you must download and install the following updates:

- .NET Framework 4.0
- Windows Management Framework 3.0

These updates are available from the Microsoft Download Center at the following respective URLs:

- *http://www.microsoft.com/en-us/download/details.aspx?id=17718*
- *http://www.microsoft.com/en-us/download/details.aspx?id=34595*

After you install the updates, the system automatically starts the Windows Remote Management service, but there are still tasks you must complete on the remote server, as follows:

- Enable the Windows Remote Management (HTTP-In) rules in Windows Firewall, as shown in Figure 2-23.

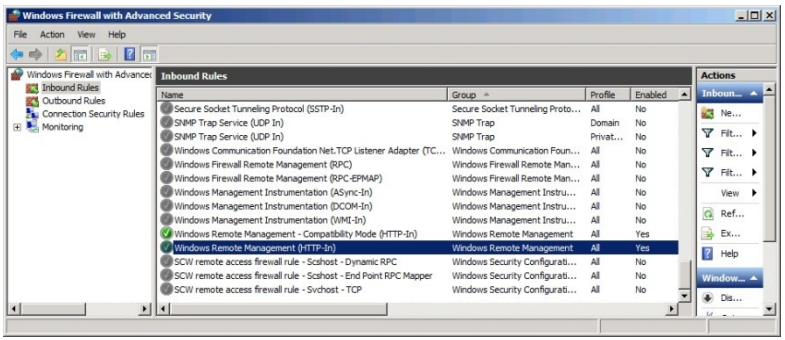

FIGURE 2-23 The Windows Remote Management rules in the Windows Firewall with Advanced Security console.

- Create a WinRM listener by running the winrm quickconfig command at a command prompt with administrative privileges.
- Enable the COM+ Network Access and Remote Event Log Management rules in Windows Firewall, as described in the previous section.

Even after installing the updates listed here, there are still limitations to the management tasks you can perform on downlevel servers from a remote location. For example, you cannot use the Add Roles and Features Wizard in Server Manager to install roles and features on downlevel servers. These servers do not appear in the server pool on the Select Destination Server page.

However, you can use Windows PowerShell to install roles and features on servers running Windows Server 2008 and Windows Server 2008 R2 remotely, as in the following procedure.

1. Log on to the server running Windows Server 2012 using an account with administrative privileges. The Server Manager window appears.
2. Open a PowerShell session with administrative privileges.
3. Establish a PowerShell session with the remote computer using the following command:

```
Enter-PSSession <remote server name> -credential <user name>
```

4. Type the password associated with the user name you specified and press Enter.
5. Display a list of the roles and features on the remote server using the following command:

```
Get-WindowsFeature
```

6. Using the short name of the role or service as it appears in the Get-WindowsFeature display, install the component using the following command.

```
Add-WindowsFeature <feature name>
```

7. Close the session with the remote server using the following command:

```
Exit-PSSession
```

8. Close the Windows PowerShell window.

NOTE **PowerShell**

When you install a role or feature on a remote server using Windows PowerShell, the installation does not include the role's management tools, as a wizard-based installation does. However, you can install the tools along with the role or feature if you include the IncludeManagementTools parameter in the Install-WindowsFeature command line. Be aware, however, that in the case of a Server Core installation, adding the IncludeManagementTools parameter will not install any MMC snap-ins or other graphical tools.

Creating server groups

For administrators of enterprise networks, it might be necessary to add a large number of servers to Server Manager. To avoid having to work with a long scrolling list of servers, you can create server groups, based on server locations, functions, or any other organizational paradigm.

When you create a server group, it appears as an icon in the navigation pane, and you can manage the servers in the group just as you would those in the All Servers group.

To create a server group, use the following procedure.

1. Log on to Windows Server 2012 and launch Server Manager.
2. In the navigation pane, click the All Servers icon. The All Servers home page appears.
3. From the Manage menu, select Create Server Group to open the Create Server Group dialog box, as shown in Figure 2-24.

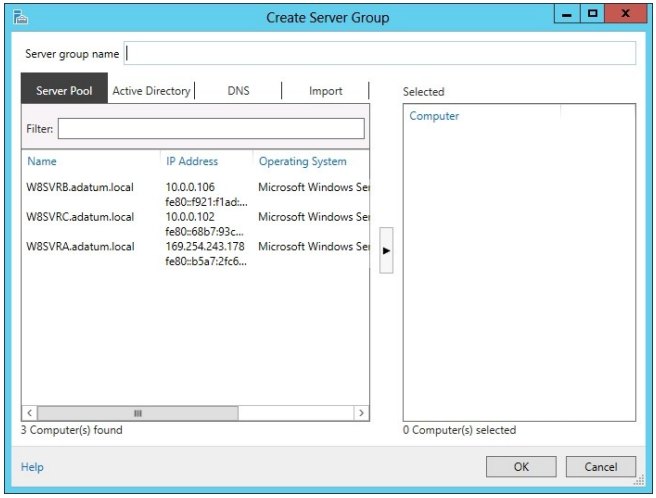

FIGURE 2-24 The Create Server Group dialog box in Server Manager.

4. In the Server Group Name text box, type the name you want to assign to the server group.
5. Select one of the four tabs to choose a method for selecting servers.
6. Select the servers you want to add to the group and click the right arrow button to add them to the Selected box.
7. Click OK. A new server group icon with the name you specified appears in the navigational pane.
8. Close the Server Manager console.

Creating server groups does not affect the functions you can perform on them. You cannot, for example, perform actions on entire groups of servers. The groupings are simply a means to keep a large number of servers organized and easily locatable.

Using Remote Server Administration Tools

You can manage remote servers from any computer running Windows Server 2012; all of the required tools are installed by default. However, the new administrative method that Microsoft is promoting urges administrators to keep servers locked away and use a workstation to manage servers from a remote location.

To manage Windows servers from a workstation, you must download and install the Remote Server Administration Tools package for the version of Windows running on your workstation from the Microsoft Download Center at *http://www.microsoft.com/download*.

Remote Server Administration Tools is packaged as a Microsoft Update file with an .msu extension, enabling you to deploy it easily from File Explorer, from the command prompt, or by using Software Distribution in a GPO. When you install Remote Server Administration Tools on a workstation running Windows 8, all of the tools are activated by default, unlike previous

versions that required you to turn them on using the Windows Features control panel. You can still use the control panel to turn selected features off, however.

When you launch Server Manager on a Windows workstation, there is no local server, and there are no remote servers to manage, until you add some. You add servers using the same process described earlier in this objective.

Your access to the servers you add depends on the account you use to log on to the workstation. If an "Access denied" message appears, you can connect to the server using another account by right-clicking it and, from the shortcut menu, selecting Manage As to display a standard Windows Security dialog box, in which you can supply alternative credentials.

Working with remote servers

Once you have added remote servers to Server Manager, you can access them using a variety of remote administration tools.

Server Manager provides three basic methods for addressing remote servers, as follows:

- **Contextual tasks** When you right-click a server in a Servers tile, anywhere in Server Manager, you see a shortcut menu that provides access to tools and commands pointed at the selected server. Some of these are commands that Server Manager executes on the remote server, such as Restart Server and Windows PowerShell. Others launch tools on the local system and direct them at the remote server, such as MMC snap-ins and the Install Roles and Features Wizard. Still others modify Server Manager itself, by removing servers from the interface. Other contextual tasks sometimes appear in the Tasks menus for specific panes.
- **Noncontextual tasks** The menu bar at the top of the Server Manager console provides access to internal tasks, such as launching the Add Server and Install Roles and Features Wizards, as well as the Server Manager Properties dialog box, in which you can specify the console's refresh interval.
- **Noncontextual tools** The console's Tools menu provides access to external programs, such as MMC snap-ins and the Windows PowerShell interface, that are directed at the local system.

Objective summary

- Windows Server 2012 is designed to facilitate remote server management, so that administrators rarely if ever have to work directly at the server console. This conserves server resources that can better be devoted to applications.
- When you add servers running Windows Server 2012 to Server Manager, you can immediately begin using the Add Roles and Features Wizard to install roles and features on any of the servers you have added.

- The Windows Firewall rules you have to enable for remote servers running Windows Server 2012 are also disabled by default on computers running earlier versions of Windows Server, so you have to enable them there as well.
- For administrators of enterprise networks, it might be necessary to add a large number of servers to Server Manager. To avoid having to work with a long scrolling list of servers, you can create server groups, based on server locations, functions, or any other organizational paradigm.
- You can manage remote servers from any computer running Windows Server 2012; all of the required tools are installed by default. However, the new administrative method that Microsoft is promoting urges administrators to keep servers locked away and use a workstation to manage servers from a remote location.

Objective review

Answer the following questions to test your knowledge of the information in this objective. You can find the answers to these questions and explanations of why each answer choice is correct or incorrect in the "Answers" section at the end of this chapter.

1. Which of the following tasks must you perform before you can manage a remote server running Windows Server 2012 using the Computer Management snap-in?
   A. Enable WinRM on the remote server.
   B. Enable the COM+ Network Access rule on the remote server.
   C. Enable the Remote Event Log Management rules on the remote server.
   D. Install Remote Server Administration Tools on the remote server.
2. Which of the following PowerShell cmdlets can you use to list the existing Windows Firewall rules on a computer running Windows Server 2012?
   A. Get-NetFirewallRule
   B. Set-NetFirewallRule
   C. Show-NetFirewallRule
   D. New-NetFirewallRule
3. Which of the following tasks can you not perform remotely on a server running Windows Server 2008?
   A. Install roles using Server Manager.
   B. Install roles using Windows PowerShell.
   C. Connect to the remote server using the Computer Management snap-in.
   D. Monitor event log entries.
4. Which of the following updates must you install on a server running Windows Server 2008 before you can connect to it using Windows Server 2012 Server Manager?
   A.. NET Framework 3.5

B. . NET Framework 4.0

C. Windows Management Framework 3.0

D. Windows Server 2008 R2

5. When you run Server Manager from a Windows 8 workstation using Remote Server Administration Tools, which of the following elements do not appear in the default display?

A. The Dashboard

B. The Local Server home page

C. The All Servers home page

D. The Welcome tile

Thought experiment

In the following thought experiment, apply what you've learned about the objective to predict what steps you need to take. You can find answers to these questions in the "Answers" section at the end of this chapter.

Ralph is responsible for the 24 servers running a particular application, which are scattered all over his company's enterprise network. Ralph wants to use Server Manager on his Windows 8 workstation to manage those servers and monitor the events that occur on them. To do this, he must enable the incoming COM+ Network Access and Remote Event Log Management rules in Windows Firewall on the servers.

Because he can't travel to the locations of all the servers, and many of the sites do not have trustworthy IT personnel, Ralph has decided to use Group Policy to configure Windows Firewall on all of the servers. The company's Active Directory Domain Services tree is organized geographically, which means that Ralph's servers are located in many different OUs, all under one domain.

With this in mind, answer the following question:

How can Ralph use Group Policy to deploy the required Windows Firewall rule settings to his 24 servers, and only those servers?

Answers

This section contains the answers to the Objective Reviews and the Thought Experiments.

Objective 2.1: Review

1. **Correct Answer:** C
   A. **Incorrect:** Windows Server 2012 can maintain more than 8 volume shadow copies.
   B. **Incorrect:** Windows Server 2012 can maintain more than 16 volume shadow copies.
   C. **Correct:** Windows Server 2012 can maintain up to 64 volume shadow copies before it begins deleting the oldest data.
   D. **Incorrect:** Windows Server 2012 cannot maintain 128 volume shadow copies.
2. **Correct Answer:** B
   A. **Incorrect:** Authentication is the process of verifying the user's identity.
   B. **Correct:** Authorization is the process by which a user is granted access to specific resources based on the permissions he or she possesses.
   C. **Incorrect:** Access-based enumeration is a Windows feature that prevents users from seeing resources to which they do not have permissions.
   D. **Incorrect:** Assignment describes the process of granting permissions, but not reading them.
3. **Correct Answers:** A and B
   A. **Correct:** Using File Server Resource Manager, you can notify administrators with email messages when users exceed their allotment of storage.
   B. **Correct:** Using File Server Resource Manager, you can create quotas for individual users that specify different storage limits.
   C. **Incorrect:** You can use NTFS quotas to prevent users from consuming any storage space on a volume beyond their allotted limit.
   D. **Incorrect:** You can use NTFS quotas to generate warnings to users when they approach their allotted storage limit.
4. **Correct Answers:** B and D
   A. **Incorrect:** In Windows Server versions prior to Windows Server 2012, special permissions are combined to form standard permissions.
   B. **Correct:** Basic permissions are formed by creating various combinations of advanced permissions.
   C. **Incorrect:** Share permissions are a system that is completely separate from the NTFS permission system.
   D. **Correct:** In Windows Server versions prior to Windows Server 2012, standard permissions are formed by creating various combinations of special permissions.
5. **Correct Answer:** D
   A. **Incorrect:** It is the owner who is the only person who can access a file that has no permissions assigned to it.

B. **Incorrect:** The security principal is not the person responsible for creating an organization's permission policies.

C. **Incorrect:** The security principal receives permissions; the security principal does not create them.

D. **Correct:** The security principal is the user or computer to which permissions are assigned.

Objective 2.1: Thought experiment

The most likely cause of the problem is that Leo does not have sufficient share permissions for read/write access to the Contoso files. By granting the CONTOSO_USER group the Allow Full Control share permission, Leo should be able to save his changes to the Contoso files.

Objective 2.2: Review

1. **Correct Answer:** A

   A. **Correct:** In Windows, a printer is the software interface through which a computer communicates with a print device.

   B. **Incorrect:** A print server is a device that receives print jobs from clients and sends them to print devices that are either locally attached or connected to the network.

   C. **Incorrect:** A printer driver is a device driver that converts the print jobs generated by applications into an appropriate string of commands for a specific print device.

   D. **Incorrect:** The Print Management snap-in is a tool that administrators can use to manage printers all over the network.

2. **Correct Answer:** B

   A. **Incorrect:** Whether the printers are pooled or not, each one must be connected to a separate port.

   B. **Correct:** To set up printer pooling, select the Enable Printer Pooling check box, and then select or create the ports corresponding to printers that will be part of the pool.

   C. **Incorrect:** You do not use the installable options settings to create a printer pool.

   D. **Incorrect:** Priorities have nothing to do with printer pooling.

3. **Correct Answer:** A

   A. **Correct:** If you stop sharing the printer, users will no longer be able to use the print device.

   B. **Incorrect:** Removing the printer from Active Directory will prevent users from finding the printer using a search, but they can still access it.

   C. **Incorrect:** Changing the printer port will prevent the printer from sending jobs to the print device, but it will not prevent users from sending jobs to the printer.

D. **Incorrect:** Renaming the share can make it difficult for users to find the printer, but they can still use it when they do find it.

4. **Correct Answer:** C
   A. **Incorrect:** The Manage Documents permission does not allow users to send jobs to the printer.
   B. **Incorrect:** The Manage Printers permission does not allow users to send jobs to the printer.
   C. **Correct:** The Print permission enables users to send documents to the printer; the Manage Documents permission does not.
   D. **Incorrect:** The Manage Documents permission does not allow users to send jobs to the printer.
5. **Correct Answer:** D
   A. **Incorrect:** A printer that is not shared will continue to process jobs that are already in the queue.
   B. **Incorrect:** Changing the port will require the users to resubmit the jobs that were in the queue.
   C. **Incorrect:** Pausing the first document in the queue will not prevent the other queued jobs from printing.
   D. **Correct:** When you select the Pause Printing option, the documents will remain in the print queue until you resume printing. This option applies to all documents in the queue.

Objective 2.2: Thought experiment

Install additional, identical printers, connecting them to the same Windows Vista print server, and create a printer pool by selecting the appropriate check box on the Ports tab of the printer's Properties sheet.

Objective 2.3: Review

1. **Correct Answer:** B
   A. **Incorrect:** WinRM is enabled by default on Windows Server 2012.
   B. **Correct:** The COM+ Network Access rule must be enabled on the remote server for MMC snap-ins to connect.
   C. **Incorrect:** The Remote Event Log Management rules are not necessary to connect to a remote server using an MMC snap-in.
   D. **Incorrect:** PTR records contain the information needed for the server to perform reverse name lookups.

2. **Correct Answers:** A and C
   A. **Correct:** The Get-NetFirewallRule cmdlet displays a list of all the rules on a system running Windows Firewall.
   B. **Incorrect:** The Set-NetFireWall rule is for managing specific rules, not listing them.
   C. **Correct:** The Show-NetFirewallRule cmdlet displays a list of all the rules on a system running Windows Firewall.
   D. **Incorrect:** The New-NetFireWall rule is for creating rules, not listing them.
3. **Correct Answer:** A
   A. **Correct:** You cannot install roles on a remote server running Windows Server 2008 using Server Manager.
   B. **Incorrect:** You can install roles on a remote server running Windows Server 2008 using Windows PowerShell.
   C. **Incorrect:** You can connect to a remote server running Windows Server 2008 using the Computer Management console, as long as you enable the COM+ Network Access rule.
   D. **Incorrect:** You can monitor event log entries on a remote server running Windows Server 2008, as long as you enable the Remote Event Log Management rules.
4. **Correct Answers:** B and C
   A. **Incorrect:** .NET Framework 3.5 is not needed for Server Manager to connect to Windows Server 2008.
   B. **Correct:** .NET Framework 4.0 is needed for Server Manager to connect to Windows Server 2008.
   C. **Correct:** Windows Management Framework 3.0 is needed for Server Manager to connect to Windows Server 2008.
   D. **Incorrect:** It is not necessary to upgrade to Windows Server 2008 R2 for Server Manager to connect to Windows Server 2008.
5. **Correct Answer:** B
   A. **Incorrect:** The Dashboard does appear in the default Server Manager display.
   B. **Correct:** The Local Server home page does not appear, because the local system is a workstation, not a server.
   C. **Incorrect:** The All Servers home page does appear in the default Server Manager display.
   D. **Incorrect:** The Welcome tile does appear in the default Server Manager display.

Objective 2.3: Thought experiment

After creating a GPO containing the required Windows Firewall settings, Ralph should create a security group containing all of the 24 computer objects representing his servers. Then, he

should link the GPO to the company domain and use security filtering to limit the scope of the GPO to the group he created.

CHAPTER 4

Deploying and configuring core network services

This chapter discusses the vital infrastructure services that virtually every network must implement. Every computer on a TCP/IP network must have at least one IP address, and most networks today use the Dynamic Host Configuration Protocol (DHCP) to assign those addresses. To access resources on the Internet and to locate Active Directory Domain Services (AD DS) domain controllers, TCP/IP computers must have access to a Domain Name System (DNS) server. Windows Server 2012 includes all of these services and provides the tools to manage them.

Objectives in this chapter:

- Objective 4.1: Configure IPv4 and IPv6 addressing
- Objective 4.2: Deploy and configure Dynamic Host Configuration Protocol (DHCP) service
- Objective 4.3: Deploy and configure DNS service

Objective 4.1: Configure IPv4 and IPv6 addressing

Server administrators must be familiar with the basic principles of the IPv4 and IPv6 address spaces. This section reviews those principles, and describes the usual process for designing IPv4 and IPv6 addressing strategies.

This objective covers how to:

- Configure IP address options
- Configure subnetting
- Configure supernetting
- Configure interoperability between IPv4 and IPv6
- Configure ISATAP
- Configure Teredo

IPv4 addressing

The IPv4 address space, as you probably know, consists of 32-bit addresses, notated as four 8-bit decimal values from 0 to 255, separated by periods, as in the example 192.168.43.100. This is known as dotted-decimal notation, and the individual 8-bit decimal values are called octets or bytes.

Each address consists of network bits, which identify a network, and host bits, which identify a particular device on that network. To differentiate the network bits from the host bits, each address must have a subnet mask.

A subnet mask is another 32-bit value consisting of binary 1 bits and 0 bits. When compared to an IP address, the bits corresponding to the 1s in the mask are the network bits, and the bits corresponding to the 0s are the host bits. Thus, if the 192.168.43.100 address mentioned earlier has a subnet mask of 255.255.255.0 (which in binary form is 11111111.11111111.11111111.00000000), the first three octets (192.168.43) identify the network and the last octet (100) identifies the host.

IPv4 classful addressing

Because the subnet mask associated with IP addresses can vary, so can the number of bits used to identify the network and the host.

The original Internet Protocol (IP) standard defines three classes of IP addresses, which provide support for networks of different sizes, as shown in Figure 4-1.

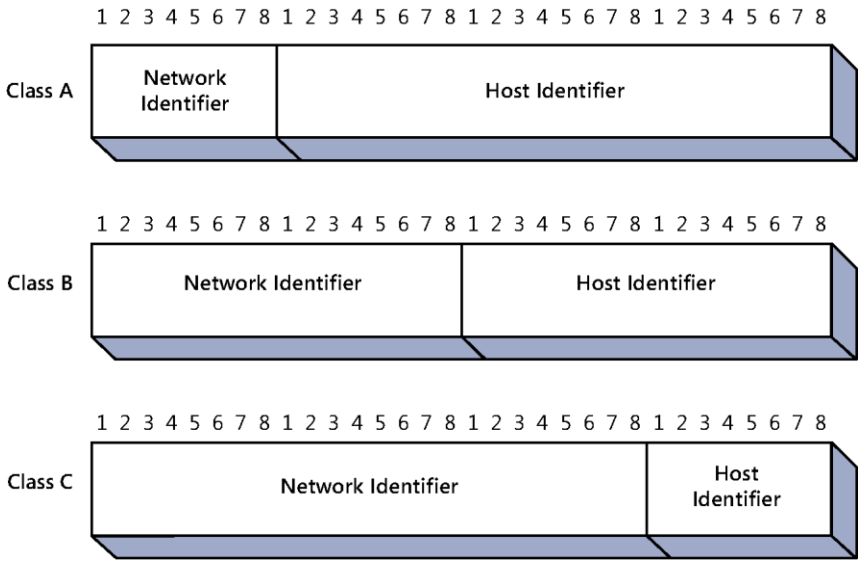

FIGURE 4-1 The three IPv4 address classes.

The number of networks and hosts supported by each of the address classes are listed in Table 4-1.

TABLE 4-1 IPv4 address classes

IP ADDRESS CLASS	CLASS A	CLASS B	CLASS C
First bit values (binary)	0	10	110
First byte value (decimal)	0–127	128–191	192–223
Number of network identifier bits	8	16	24
Number of host identifier bits	24	16	8
Number of possible networks	126	16,384	2,097,152
Number of possible hosts	16,777,214	65,534	254

NOTE **Additional classes**

In addition to Classes A, B, and C, the IP standard also defines two additional address classes, Class D and Class E. Class D addresses begin with the bit values 1110, and Class E addresses begin with the values 11110. The Internet Assigned Numbers Authority (IANA) has allocated Class D addresses for use as multicast identifiers. A multicast address identifies a group of computers on a network, all of which possess a similar trait. Multicast addresses enable TCP/IP applications to send traffic to computers that perform specific functions (such as all the routers on the network), even if they're located on different subnets. Class E addresses are defined as experimental and are as yet unused.

The "First bit value" row in the table specifies the values that the first one, two, or three bits of an address in each class must have. Early TCP/IP implementations used these bit values instead of a subnet mask to determine the class of an address. The binary values of the first bits of each address class limit the possible decimal values for the first byte of the address. For example, because the first bit of Class A addresses must be 0, the possible binary values of the first byte in a Class A address range from 00000000 to 01111111, which in decimal form are values ranging from 1 to 127. Thus, when you see an IP address in which the first byte is a number from 1 to 127, you know that this is a Class A address.

In a Class A address, the network identifier is the first 8 bits of the address and the host identifier is the remaining 24 bits. Thus, there are only 126 possible Class A networks (network identifier 127 is reserved for diagnostic purposes), but each network can have up to 16,777,214 network interface adapters on it. Class B and Class C addresses devote more bits to the network identifier, which means that they support a greater number of networks, but at the cost of having fewer host identifier bits. This trade-off reduces the number of hosts that can be created on each network.

The values Table 4-1 for the number of hosts supported by each address class might appear low. For example, an 8-bit binary number can have 256 (that is, 28) possible values, not 254, as shown in the table for the number of hosts on a Class C address. The value 254 is used because the original IP addressing standard states that you can't assign the "all zeros" or "all ones" addresses to individual hosts. The "all zeros" address identifies the network, not a specific host,

and the "all ones" identifier always signifies a broadcast address. You cannot assign either value to an individual host. Therefore, to calculate the number of possible network or host addresses you can create with a given number of bits, you use the formula $2^x - 2$, where x is the number of bits.

Classless Inter-Domain Routing

At the time when IP was developed, no one imagined that the 32-bit address space would ever be exhausted. In the early 1980s, there were no networks that had 65,536 computers, never mind 16 million, and no one worried about the wastefulness of assigning IP addresses based on these classes.

Because of that wastefulness, classful addressing was gradually obsolesced by a series of subnetting methods, including variable length subnet masking (VLSM) and eventually Classless Inter-Domain Routing (CIDR). CIDR is a subnetting method that enables administrators to place the division between the network bits and the host bits anywhere in the address, not just between octets. This makes it possible to create networks of almost any size.

CIDR also introduces a new notation for network addresses. A standard dotted-decimal address representing the network is followed by a forward slash and a numeral specifying the size of the network identifying prefix. For example, 192.168.43.0/24 represents a single Class C address that uses a 24-bit network identifier, leaving the other 8 bits for up to 254 host identifiers. Each of those hosts would receive an address from 192.168.43.1 to 192.168.43.254, using the subnet mask 255.255.255.0.

However, using CIDR, an administrator can subnet this address further, by allocating some of the host bits to create subnets. To create subnets for four offices, for example, the administrator can take two of the host identifier bits, changing the network address in CIDR notation to 192.168.43.0/26. Because the network identifier is now 26 bits, the subnet masks for all four networks will now be 11111111.11111111.11111111.11000000, in binary form, or 255.255.255.192 in standard decimal form. Each of the four networks will have up to 62 hosts, using the IP address ranges shown in Table 4-2.

TABLE 4-2 Sample CIDR 192.168.43.0/26 networks

NETWORK ADDRESS	STARTING IP ADDRESS	ENDING IP ADDRESS	SUBNET MASK
192.168.43.0	192.168.43.1	192.168.43.62	255.255.255.192
192.168.43.64	192.168.43.65	192.168.43.126	255.255.255.192
192.168.43.128	192.168.43.129	192.168.43.190	255.255.255.192
192.168.43.192	192.168.43.193	192.168.43.254	255.255.255.192

If the administrator needs more than four subnets, changing the address to 192.168.43.0/28 adds two more bits to the network address, for a maximum of 16 subnets, each of which can support up to 14 hosts. The subnet mask for these networks would therefore be 255.255.255.240.

Public and private IPv4 addressing

For a computer to be accessible from the Internet, it must have an IP address that is both registered and unique. All of the web servers on the Internet have registered addresses, as do all of the other types of Internet servers.

The IANA is the ultimate source for all registered addresses; managed by the Internet Corporation for Assigned Names and Numbers (ICANN), this organization allocates blocks of addresses to regional Internet registries (RIR), which allocate smaller blocks in turn to Internet service providers (ISPs). An organization that wants to host a server on the Internet typically obtains a registered address from an ISP.

Registered IP addresses are not necessary for workstations that merely access resources on the Internet. If organizations used registered addresses for all of their workstations, the IPv4 address space would have been depleted long ago. Instead, organizations typically use private IP addresses for their workstations. Private IP addresses are blocks of addresses that are allocated specifically for private network use. Anyone can use these addresses without registering them, but they cannot make computers using private addresses accessible from the Internet.

The three blocks of addresses allocated for private use are as follows:

- 10.0.0.0/8
- 172.16.0.0/12
- 192.168.0.0/16

Most enterprise networks use addresses from these blocks for their workstations. It doesn't matter if other organizations use the same addresses also, because the workstations are never directly connected to the same network.

IPv4 subnetting

In most cases, enterprise administrators use addresses in one of the private IP address ranges to create the subnets they need. If you are building a new enterprise network from scratch, you can choose any one of the private address blocks and make things easy on yourself by subnetting along the octet boundaries.

For example, you can take the 10.0.0.0/8 private IP address range and use the entire second octet as a subnet ID. This enables you to create up to 256 subnets with as many as 65,536 hosts on each one. The subnet masks for all of the addresses on the subnets will be 255.255.0.0 and the network addresses will proceed as follows:

- 10.0.0.0/16
- 10.1.0.0/16
- 10.2.0.0/16
- 10.3.0.0/16
- 10.255.0.0/16

Of course, when you are working on an existing network, the subnetting process is likely to be more difficult. You might, for example, be given a relatively small range of addresses and be asked to create a certain number of subnets out of them. To do this, you use the following procedure.

1. Determine how many subnet identifier bits you need to create the required number of subnets.
2. Subtract the subnet bits you need from the host bits and add them to the network bits.
3. Calculate the subnet mask by adding the network and subnet bits in binary form and converting the binary value to decimal.
4. Take the least significant subnet bit and the host bits, in binary form, and convert them to a decimal value.
5. Increment the network identifier (including the subnet bits) by the decimal value you calculated to determine the network addresses of your new subnets.

Using the same example from earlier in this chapter, if you take the 192.168.43.0/24 address and allocate two extra bits for the subnet ID, you end up with a binary subnet mask value of 11111111.11111111.11111111.11000000 (255.255.255.192 in decimal form, as noted earlier).

The least significant subnet bit plus the host bits gives you a binary value of 1000000, which converts to a decimal value of 64. Therefore, if we know that the network address of your first subnet is 192.168.43.0, the second subnet must be 192.168.43.64, the third 192.168.43.128, and the fourth 192.168.43.192, as shown in Table 4-2.

Supernetting

In addition to simplifying network notation, CIDR also makes a technique called IP address aggregation or supernetting possible, which can help to reduce the size of Internet routing tables. A supernet is a combination of contiguous networks that all contain a common CIDR prefix. When an organization possesses multiple contiguous networks that can be expressed as a supernet, it becomes possible to list those networks in a routing table using only one entry instead of many.

For example, if an organization has the following five subnets, standard practice would be to create a separate routing table entry for each one.

- 172.16.43.0/24
- 172.16.44.0/24
- 172.16.45.0/24
- 172.16.46.0/24
- 172.16.47.0/24

To create a supernet encompassing all five of these networks, you must isolate the bits they have in common. When you convert the network addresses from decimal to binary, you get the following values:

```
172.16.43.0     10101100.00010000.00101011.00000000

172.16.44.0     10101100.00010000.00101100.00000000

172.16.45.0     10101100.00010000.00101101.00000000

172.16.46.0     10101100.00010000.00101110.00000000

172.16.47.0     10101100.00010000.00101111.00000000
```

In binary form, you can see that all five addresses have the same first 21 bits. Those 21 bits become the network identifier of the supernet address, as follows:

```
10101100.00010000.00101
```

After zeroing out the host bits to form the network address and converting the binary number back to decimal form, as follows, the resulting supernet address is 172.16.40.0/21.

```
10101100.00010000.00101000.00000000          172.16.40.0/21
```

This one network address can replace the original five in routing tables duplicated throughout the Internet. Obviously, this is just an example of a technique that administrators can use to combine dozens or even hundreds of subnets into single routing table entries.

Assigning IPv4 addresses

In addition to understanding how IP addressing works, a network administrator must be familiar with the methods for deploying IP addresses to the computers on a network.

To assign IPv4 addresses, there are three basic alternatives:

- Manual configuration
- Dynamic Host Configuration Protocol (DHCP)
- Automatic Private IP Addressing (APIPA)

The advantages and disadvantages of these methods are discussed in the following sections.

MANUAL IPV4 ADDRESS CONFIGURATION

Configuring a TCP/IP client manually is not terribly difficult, nor is it very time-consuming. Most operating systems provide a graphical interface that enables you to enter an IPv4 address, a subnet mask, and various other TCP/IP configuration parameters. To configure IP address settings in Windows Server 2012, you use the Internet Protocol Version 4 (TCP/IPv4) Properties sheet, as shown in Figure 4-2.

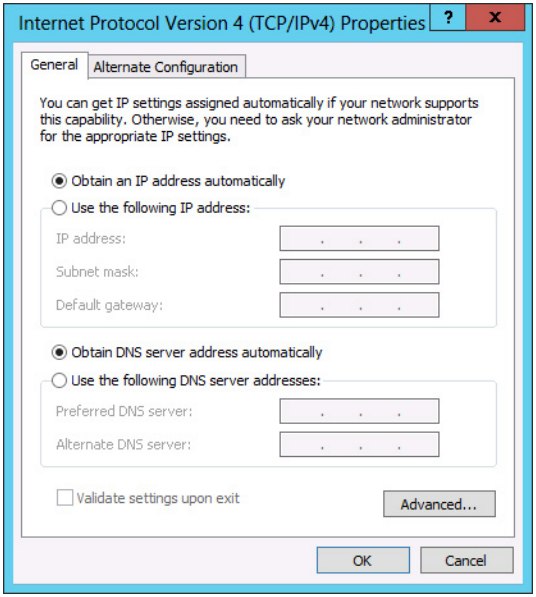

FIGURE 4-2 The Internet Protocol Version 4 (TCP/IPv4) Properties sheet.

When you select the Use The Following IP Address option, you can configure the following IP address options:

- **IP Address** Specifies the IP address on the local subnet that will identify the network interface in the computer
- **Subnet Mask** Specifies the mask associated with the local subnet
- **Default Gateway** Specifies the IP address of a router on the local subnet, which the system will use to access destinations on other networks
- **Preferred DNS Server** Specifies the IP address of the DNS server the system will use to resolve host names into IP addresses

The primary problem with manual configuration is that a task requiring two minutes for one workstation requires several hours for 100 workstations and several days for 1,000. Manually configuring all but the smallest networks is highly impractical, and not just for reasons of time. There is also the matter of tracking the IPv4 addresses you assign and making sure each system has an address that is unique. This can end up being a logistical nightmare, which is why few network administrators choose this option.

DYNAMIC HOST CONFIGURATION PROTOCOL

DHCP is an application and an application-layer protocol that together enable administrators to dynamically allocate IP addresses from a pool. Computers equipped with DHCP clients automatically contact a DHCP server when they start, and the server assigns them unique addresses and all of the other configuration parameters the TCP/IP client requires.

The DHCP server provides addresses to clients on a leased basis, and after a predetermined interval, each client either renews its address or releases it back to the server for reallocation. DHCP not only automates the address assignment process; it also keeps track of the addresses it assigns, preventing address duplication on the network.

AUTOMATIC PRIVATE IP ADDRESSING

Automatic Private IP Addressing (APIPA) is the name assigned by Microsoft to a DHCP failover mechanism used by all of the current Microsoft Windows operating systems. On Windows computers, the DHCP client is enabled by default. If, after several attempts, a system fails to locate a DHCP server on the network, APIPA takes over and automatically assigns an address on the 169.254.0.0/16 network to the computer.

For a small network that consists of only a single local area network (LAN), APIPA is a simple and effective alternative to installing a DHCP server. However, for installations consisting of multiple LANs, with routers connecting them, administrators must take more positive control over the IP address assignment process. This usually means deploying one or more DHCP servers in some form.

IPv6 addressing

As most administrators know, IPv6 is designed to increase the size of the IP address space, thus providing addresses for many more devices than IPv4. The 128-bit address size of IPv6 allows for 2^{128} possible addresses, an enormous number that works out to over 54 million addresses for each square meter of the Earth's surface.

In addition to providing more addresses, IPv6 will also reduce the size of the routing tables in the routers scattered around the Internet. This is because the size of the addresses provides for more than the two levels of subnetting currently possible with IPv4.

Introducing IPv6

IPv6 addresses are different from IPv4 addresses in many ways other than length. Instead of the four 8-bit decimal numbers separated by periods that IPv4 uses, IPv6 addresses use a notation called colon-hexadecimal format, which consists of eight 16-bit hexadecimal numbers, separated by colons, as follows:

```
XX:XX:XX:XX:XX:XX:XX:XX
```

Each X represents eight bits (or one byte), which in hexadecimal notation is represented by two characters, as in the following example:

```
21cd:0053:0000:0000:e8bb:04f2:003c:c394
```

CONTRACTING IPV6 ADDRESSES

When an IPv6 address has two or more consecutive 8-bit blocks of zeros, you can replace them with a double colon, as follows (but you can only use one double colon in any IPv6 address):

```
21cd:0053::e8bb:04f2:003c:c394
```

You can also remove the leading zeros in any block where they appear, as follows:

```
21cd:53::e8bb:4f2:3c:c394
```

EXPRESSING IPV6 NETWORK ADDRESSES

There are no subnet masks in IPv6. Network addresses use the same slash notation as CIDR to identify the network bits. In the example specified here, the network address is notated as follows:

```
21cd:53::/64
```

This is the contracted form for the following network address:

```
21cd:0053:0000:0000/64
```

IPv6 address types

There are no broadcast transmissions in IPv6, and therefore no broadcast addresses, as in IPv4. IPv6 supports three types of transmissions, as follows:

- **Unicast** Provides one-to-one transmission service to individual interfaces, including server farms sharing a single address
- **Multicast** Provides one-to-many transmission service to groups of interfaces identified by a single multicast address
- **Anycast** Provides one-to-one-of-many transmission service to groups of interfaces, only the nearest of which (measured by the number of intermediate routers) receives the transmission

NOTE **IPv6 scopes**

In IPv6, the scope of an address refers to the size of its functional area. For example, the scope of a global unicast is unlimited, the entire Internet. The scope of a link-local unicast is the immediate link; that is, the local network. The scope of a unique local unicast consists of all the subnets within an organization.

IPv6 also supports several address types, as described in the following sections.

GLOBAL UNICAST ADDRESSES

A global unicast address is the equivalent of a registered IPv4 address, routable worldwide and unique on the Internet.

LINK-LOCAL UNICAST ADDRESSES

In IPv6, systems that assign themselves an address automatically create a link-local unicast address, which is essentially the equivalent of an APIPA address in IPv4. All link-local addresses

have the same network identifier: a 10-bit prefix of 11111110 010 followed by 54 zeros, resulting in the following network address:

```
fe80:0000:0000:0000/64
```

In its more compact form, the link-local network address is as follows:

```
fe80::/64
```

Because all link-local addresses are on the same network, they are not routable and systems possessing them can only communicate with other systems on the same link.

UNIQUE LOCAL UNICAST ADDRESSES

Unique local unicast addresses are the IPv6 equivalent of the 10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16 private network addresses in IPv4. Like the IPv4 private addresses, unique local addresses are routable within an organization. Administrators can also subnet them as needed to support an organization of any size.

> ***NOTE*** **Deprecated IPv6 addresses**
>
> **Many sources of IPv6 information continue to list site-local unicast addresses as a valid type of unicast, with a function similar to that of the private IPv4 network addresses. For various reasons, site-local unicast addresses have been deprecated, and although their use is not forbidden, their functionality has been replaced by unique local unicast addresses.**

MULTICAST ADDRESSES

Multicast addresses always begin with an FP value of 11111111, in binary, or ff in hexadecimal.

ANYCAST ADDRESSES

The function of an anycast address is to identify the routers within a given address scope and send traffic to the nearest router, as determined by the local routing protocols. Organizations can use anycast addresses to identify a particular set of routers in the enterprise, such as those that provide access to the Internet. To use anycasts, the routers must be configured to recognize the anycast addresses as such.

Assigning IPv6 addresses

The processes by which administrators assign IPv6 addresses to network computers are basically similar to those in IPv4. As with IPv4, a Windows computer can obtain an IPv6 address by three possible methods:

- **Manual allocation** A user or administrator manually supplies an address and other information for each network interface.
- **Self-allocation** The computer creates its own address using a process called stateless address autoconfiguration.

- **Dynamic allocation** The computer solicits and receives an address from a DHCPv6 server on the network.

MANUAL IPV6 ADDRESS ALLOCATION

For the enterprise administrator, manual allocation of IPv6 addresses is even more impractical than in IPv4, because of the length of the addresses involved. However, it is possible, and the procedure for doing so in Windows Server 2012 is the same as that for IPv4, except that you open the Internet Protocol Version 6 (TCP/IPv6) Properties sheet, as shown in Figure 4-3.

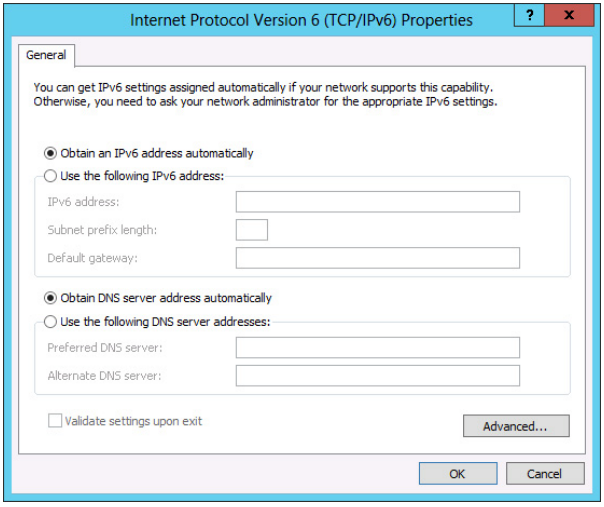

FIGURE 4-3 The Internet Protocol Version 6 (TCP/IPv6) Properties sheet.

Because of the difficulties of working with IPv6 addresses manually, the following two options are far more prevalent.

STATELESS IPV6 ADDRESS AUTOCONFIGURATION

When a Windows computer starts, it initiates the stateless address autoconfiguration process, during which it assigns each interface a link-local unicast address. This assignment always occurs, even when the interface is to receive a global unicast address later. The link-local address enables the system to communicate with the router on the link, which provides additional instructions.

The steps of the stateless address autoconfiguration process are as follows.

1. **Link-local address creation** The IPv6 implementation on the system creates a link-local address for each interface by using the fe80::/64 network address and generating an interface ID, either using the interface's media access control (MAC) address or a pseudorandom generator.
2. **Duplicate address detection** Using the IPv6 Neighbor Discovery (ND) protocol, the system transmits a Neighbor Solicitation message to determine if any other computer

on the link is using the same address and listens for a Neighbor Advertisement message sent in reply. If there is no reply, the system considers the address to be unique on the link. If there is a reply, the system must generate a new address and repeat the procedure.

3. **Link-local address assignment** When the system determines that the link-local address is unique, it configures the interface to use that address. On a small network consisting of a single segment or link, this might be the interface's permanent address assignment. On a network with multiple subnets, the primary function of the link-local address assignment is to enable the system to communicate with a router on the link.
4. **Router advertisement solicitation** The system uses the ND protocol to transmit Router Solicitation messages to the all routers multicast address. These messages compel routers to transmit the Router Advertisement messages more frequently.
5. **Router advertisement** The router on the link uses the ND protocol to transmit Router Advertisement messages to the system, which contain information on how the autoconfiguration process should proceed. The Router Advertisement messages typically supply a network prefix, which the system will use with its existing interface ID to create a global or unique local unicast address. The messages might also instruct the system to initiate a stateful autoconfiguration process by contacting a specific DHCPv6 server. If there is no router on the link, as determined by the system's failure to receive Router Advertisement messages, then the system must attempt to initiate a stateless autoconfiguration process.
6. **Global or unique local address configuration** Using the information it receives from the router, the system generates a suitable address that is routable, either globally or within the enterprise, and configures the interface to use it. If so instructed, the system might also initiate a stateful autoconfiguration process by contacting the DHCPv6 server specified by the router and obtaining a global or unique local address from that server, along with other configuration settings.

DYNAMIC HOST CONFIGURATION PROTOCOL V6

For the enterprise administrator with a multisegment network, it will be necessary to use unique local or global addresses for internetwork communication, so you will either need routers that advertise the appropriate network prefixes or DHCPv6 servers that can supply addresses with the correct prefixes.

The Remote Access role in Windows Server 2012 supports IPv6 routing and advertising, and the DHCP Server role supports IPv6 address allocation.

Planning an IP transition

Many enterprise administrators are so comfortable working with IPv4 addresses that they are hesitant to change. Network Address Translation (NAT) and CIDR have been excellent stopgaps to the depletion of the 32-bit IP address space for years, and many would like to see

them continue as such. However, the IPv6 transition, long a specter on the distant horizon, is now suddenly approaching at frightening speed, and it is time for administrators not familiar with the new technologies to catch up or be left behind.

The networking industry, and particularly the Internet, has made huge investments in IPv4 technologies, and replacing them with IPv6 has been a gradual process. In fact, it is a gradual process that was supposed to have begun in earnest over ten years ago. However, many people treat their IPv4 equipment like household appliances: Unless it stops working, there is no need to replace it. Unfortunately, the day when that equipment stops working is approaching rapidly. So, although it might not yet be time to embrace IPv6 exclusively, administrators should have the transition in mind as they design their networks and make their purchasing decisions.

NOTE **IPv4 address exhaustion**

The exhaustion of the IANA unallocated address pool occurred on January 31, 2011. One of the RIRs, the Asia Pacific Network Information Center (APNIC), was depleted on April 15, 2011, and the other RIRs are expected to follow suit before long.

Enterprise administrators can do as they wish within the enterprise itself. If all of the network devices in the organization support IPv6, they can begin to use IPv6 addresses at any time. However, the Internet is still firmly based on IPv4, and will continue to be so for several years. Therefore, a transition from IPv4 to IPv6 must be a gradual project that includes some period of support for both IP versions.

At the present time, and for the immediate future, administrators must work under the assumption that the rest of the world is using IPv4, and you must implement a mechanism for transmitting your IPv6 traffic over an IPv4 connection. Eventually, the situation will be reversed. Most of the world will be running IPv6, and the remaining IPv4 technologies will have to transmit their older traffic over new links.

Using a dual IP stack

The simplest and most obvious method for transitioning from IPv4 to IPv6 is to run both, and this is what all current versions of Windows do, going back as far as Windows Server 2008 and Windows Vista.

By default, these operating systems install both IP versions and use them simultaneously. In fact, even if you have never heard of IPv6 until today, your computers are likely already using it, and have IPv6 link-local addresses that you can see by running the ipconfig /all command.

The network layer implementations in Windows are separate, so you configure them separately. For both IPv4 and IPv6, you can choose to configure the address and other settings manually, or use autoconfiguration.

Because Windows supports both IP versions, the computers can communicate with TCP/IP resources running either IPv4 or IPv6. However, an enterprise network includes other devices

also, most particularly routers, that might not yet support IPv6. The Internet also is nearly all still based on IPv4.

Beginning immediately, administrators should make sure that any network layer equipment they purchase includes support for IPv6. Failure to do so will almost certainly cost them later.

Tunneling

Right now, there are many network services that are IPv4-only, and comparatively few that require IPv6. Those IPv6 services are coming, however.

The DirectAccess remote networking feature in Windows Server 2012 and Windows 8 is an example of an IPv6-only technology, and much of its complexity is due to the need to establish IPv6 connections over the IPv4 Internet.

The primary method for transmitting IPv6 traffic over an IPv4 network is called tunneling. Tunneling, in this case, is the process by which a system encapsulates an IPv6 datagram within an IPv4 packet, as shown in Figure 4-4. The system then transmits the IPv4 packet to its destination, with none of the intermediate systems aware of the packet's contents.

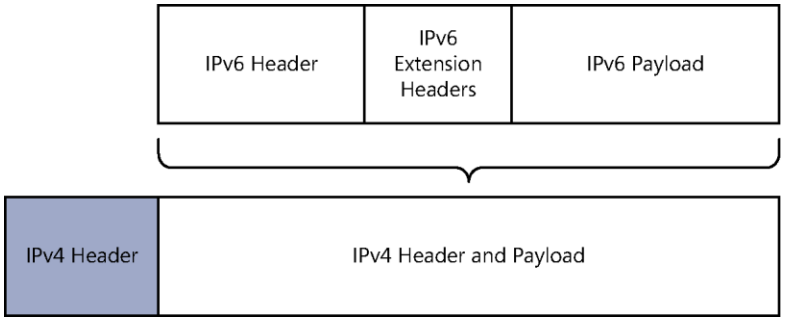

FIGURE 4-4 IPv6 traffic encapsulated inside an IPv4 datagram.

Tunneling can work in a variety of configurations, depending on the network infrastructure, including router-to-router, host-to-host, router-to-host, and host-to-router. However, the most common configuration is router-to-router, as in the case of an IPv4-only connection between an IPv6 branch office and an IPv6 home office, as shown in Figure 4-5.

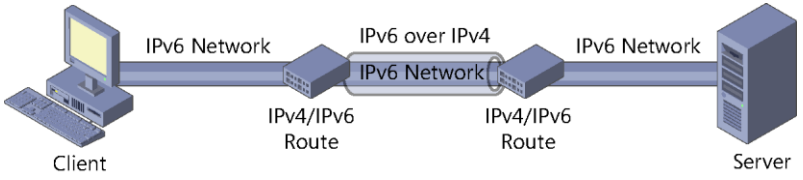

FIGURE 4-5 Two IPv6 networks connected by an IPv4 tunnel.

The two routers support both IPv4 and IPv6, and the local networks at each site use IPv6. However, the link connecting the two sites is IPv4-only. By creating a tunnel between the routers in the two offices, using their IPv4 interfaces, they can exchange IPv6 traffic as needed.

Computers at either site can send IPv6 traffic to the other site, and the routers are responsible for encapsulating the IPv6 data in IPv4 packets for the trip through the tunnel.

Windows supports several different tunneling methods, both manual and automatic, as described in the following sections.

CONFIGURING TUNNELS MANUALLY

It is possible to manually create semipermanent tunnels that carry IPv6 traffic through an IPv4-only network. When a computer running Windows Server 2012 or Windows 8 is functioning as one end of the tunnel, you can use the following command:

```
netsh interface ipv6 add v6v4tunnel "interface" localaddress remoteaddress
```

In this command, interface is a friendly name you want to assign to the tunnel you are creating and localaddress and remoteaddress are the IPv4 addresses forming the two ends of the tunnel. An example of an actual command would be as follows:

```
netsh interface ipv6 add v6v4tunnel "tunnel" 206.73.118.19 157.54.206.43
```

CONFIGURING TUNNELS AUTOMATICALLY

There are also a number of mechanisms that automatically create tunnels over IPv4 connections. These are technologies designed to be temporary solutions during the transition from IPv4 to IPv6. All of them include a mechanism for expressing an IPv4 address in the IPv6 format. The IPv4-to-IPv6 transition technologies that Windows supports are described in the following sections.

6TO4

The 6to4 mechanism essentially incorporates the IPv4 connections in a network into the IPv6 infrastructure by defining a method for expressing IPv4 addresses in IPv6 format and encapsulating IPv6 traffic into IPv4 packets.

ISATAP

Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) is an automatic tunneling protocol used by the Windows workstation operating systems that emulates an IPv6 link using an IPv4 network.

ISATAP also converts IPv4 addresses into IPv6 link-layer address format, but it uses a different method than 6to4. ISATAP does not support multicasting, so it cannot locate routers in the usual manner, using the Neighbor Discovery protocol. Instead, the system compiles a potential routers list (PRL) using DNS queries and sends Router Discovery messages to them on a regular basis, using Internet Control Message Protocol version 6 (ICMPv6).

TEREDO

To use 6to4 tunneling, both endpoints of the tunnel must have registered IPv4 addresses. However, on many networks, the system that would function as the endpoint is located behind

a NAT router, and therefore has an unregistered address. In such a case, the only registered address available is assigned to the NAT router itself, and unless the router supports 6to4 (which many don't), it is impossible to establish the tunnel.

Teredo is a mechanism that addresses this shortcoming by enabling devices behind non-IPv6 NAT routers to function as tunnel endpoints. To do this, Teredo encapsulates IPv6 packets within transport-layer User Datagram Protocol (UDP) datagrams, rather than network-layer IPv4 datagrams, as 6to4 does.

For a Teredo client to function as a tunnel endpoint, it must have access to a Teredo server, with which it exchanges Router Solicitation and Router Advertisement messages to determine whether the client is located behind a NAT router

To initiate communications, a Teredo client exchanges null packets called bubbles with the desired destination, using the Teredo servers at each end as intermediaries. The function of the bubble messages is to create mappings for both computers in each other's NAT routers.

Objective summary

- The IPv4 address space consists of 32-bit addresses, notated as four 8-bit decimal values from 0 to 255, separated by periods, as in the example 192.168.43.100. This is known as dotted-decimal notation, and the individual 8-bit decimal values are called octets or bytes.
- Because the subnet mask associated with IP addresses can vary, so can the number of bits used to identify the network and the host. The original IP standard defines three address classes for assignment to networks, which support different numbers of networks and hosts.
- Because of its wastefulness, classful addressing was gradually made obsolete by a series of subnetting methods, including VLSM and eventually CIDR.
- When a Windows computer starts, it initiates the IPv6 stateless address autoconfiguration process, during which it assigns each interface a link-local unicast address.
- The simplest and most obvious method for transitioning from IPv4 to IPv6 is to run both, and this is what all current versions of Windows do.
- The primary method for transmitting IPv6 traffic over an IPv4 network is called tunneling. Tunneling is the process by which a system encapsulates an IPv6 datagram within an IPv4 packet.

Objective review

Answer the following questions to test your knowledge of the information in this objective. You can find the answers to these questions and explanations of why each answer choice is correct or incorrect in the "Answers" section at the end of this chapter.

1. Which of the following is the primary method for transmitting IPv6 traffic over an IPv4 network?
   A. Subnetting
   B. Tunneling
   C. Supernetting
   D. Contracting
2. Which of the following is the IPv6 equivalent to a private IPv4 address?
   A. Link-local unicast address
   B. Global unique unicast address
   C. Unique local unicast address
   D. Anycast address
3. Which of the following is an automatic tunneling protocol used by Windows operating systems that are located behind NAT routers?
   A. Teredo
   B. 6to4
   C. ISATAP
   D. APIPA
4. What kind of IP address must a system have to be visible from the Internet?
   A. Registered
   B. Binary
   C. Class B
   D. Subnetted
5. Which of the following subnet mask values would you use when configuring a TCP/IP client with an IPv4 address on the 172.16.32.0/19 network?
   A. 255.224.0.0
   B. 255.240.0.0
   C. 255.255.224.0
   D. 255.255.240.0
   E. 255.255.255.240

Thought experiment

In the following thought experiment, apply what you've learned about this objective to predict what steps you need to take. You can find answers to these questions in the "Answers" section at the end of this chapter.

The enterprise administrator has assigned Arthur the network address 172.16.8.0/25 for the branch office network that he is constructing. Arthur calculates that this gives him 126 (27) IP addresses, which is enough for his network, but he has determined that he needs six subnets with at least 10 hosts on each one.

With this in mind, answer the following questions:

1. **How can Arthur subnet the address he has been given to satisfy his needs?**
2. **What IP addresses and subnet masks will the computers on his branch office network use?**

Objective 4.2: Configure servers

It seldom happens that a server is ready to perform all the tasks you have planned for it immediately after installation. Typically some postinstallation configuration is required, and further configuration changes might become necessary after the server is in service.

This objective covers how to:

- Create and configure scopes
- Configure a DHCP reservation
- Configure DHCP options
- Configure client and server for PXE boot
- Configure DHCP relay agent
- Authorize DHCP server

Understanding DHCP

DHCP is a service that automatically configures the IP address and other TCP/IP settings on network computers by assigning addresses from a pool (called a scope) and reclaiming them when they are no longer in use.

Aside from being a time-consuming chore, manually configuring TCP/IP clients can result in typographical errors that cause addressing conflicts that interrupt network communications. DHCP prevents these errors and provides many other advantages, including automatic

assignment of new addresses when computers are moved from one subnet to another and automatic reclamation of addresses that are no longer in use.

DHCP consists of three components, as follows:

- A DHCP server application, which responds to client requests for TCP/IP configuration settings
- A DHCP client, which issues requests to servers and applies the TCP/IP configuration settings it receives to the local computer
- A DHCP communications protocol, which defines the formats and sequences of the messages exchanged by DHCP clients and servers

All of the Microsoft Windows operating systems include DHCP client capabilities, and all of the server operating systems (including Windows Server 2012) include the Microsoft DHCP Server.

The DHCP standards define three different IP address allocation methods, which are as follows:

- **Dynamic allocation** The DHCP server assigns an IP address to a client computer from a scope, for a specified length of time. Each client must periodically renew the lease to continue using the address. If the client allows the lease to expire, the address is returned to the scope for reassignment to another client.
- **Automatic allocation** The DHCP server permanently assigns an IP address to a client computer from a scope. Once the DHCP server assigns the address to the client, the only way to change it is to manually reconfigure the computer.
- **Manual allocation** The DHCP server permanently assigns a specific IP address to a specific computer on the network. In the Windows Server 2012 DHCP server, manually allocated addresses are called reservations.

In addition to IP addresses, DHCP also can provide clients with values for the other parameters needed to configure a TCP/IP client, including a subnet mask, default gateway, and DNS server addresses. The object is to eliminate the need for any manual TCP/IP configuration on a client system. For example, the Microsoft DHCP server includes more than 50 configuration parameters, which it can deliver along with the IP address, even though Windows clients can only use a subset of those parameters.

DHCP communications use eight different types of messages, all of which use the same basic packet format. DHCP traffic is carried within standard UDP/IP datagrams, using port 67 at the server and port 68 at the client.

DHCP options

The DHCP options field is a catch-all area designed to carry the various parameters (other than the IP address) used to configure the client system's TCP/IP stack. Because you can configure a DHCP server to deliver many options to clients, defining separate fields for each one would be impractical.

THE DHCP MESSAGE TYPE OPTION

The DHCP Message Type option identifies the overall function of the DHCP message and is required in all DHCP packets. The DHCP communication protocol defines eight different message types, as follows:

- **DHCPDISCOVER** Used by clients to request configuration parameters from a DHCP server
- **DHCPOFFER** Used by servers to offer IP addresses to requesting clients
- **DHCPREQUEST** Used by clients to accept or renew an IP address assignment
- **DHCPDECLINE** Used by clients to reject an offered IP address
- **DHCPACK** Used by servers to acknowledge a client's acceptance of an offered IP address
- **DHCPNAK** Used by servers to reject a client's acceptance of an offered IP address
- **DHCPRELEASE** Used by clients to terminate an IP address lease
- **DHCPINFORM** Used by clients to obtain additional TCP/IP configuration parameters from a server

BOOTP VENDOR INFORMATION EXTENSIONS

These options include many of the basic TCP/IP configuration parameters used by most client systems, such as the following:

- **Subnet Mask** Specifies which bits of the IP address identify the host system and which bits identify the network where the host system resides
- **Router** Specifies the IP address of the router (or default gateway) on the local network segment the client should use to transmit to systems on other network segments
- **Domain Name Server** Specifies the IP addresses of the servers the client will use for DNS name resolution
- **Host Name** Specifies the DNS host name the client system will use
- **Domain Name** Specifies the name of the DNS domain on which the system will reside

DHCP EXTENSIONS

These options are used to provide parameters that govern the DHCP lease negotiation and renewal processes.

- **Requested IP Address** Used by the client to request a particular IP address from the server
- **IP Address Lease Time** Specifies the duration of a dynamically allocated IP address lease

- **Server Identifier** Specifies the IP address of the server involved in a DHCP transaction; used by the client to address unicasts to the server
- **Parameter Request List** Used by the client to send a list of requested configuration options (identified by their code numbers) to the server
- **Message** Used to carry an error message from the server to the client in a DHCPNAK message
- **Renewal (T1) time value** Specifies the time period that must elapse before an IP address lease enters the renewing state
- **Rebinding (T2) time value** Specifies the time period that must elapse before an IP address lease enters the rebinding state

DHCP communications

To design a DHCP strategy for an enterprise network and deploy it properly requires an understanding of the communications that occur between DHCP clients and servers. In Windows computers, the DHCP client is enabled by default, although it is not mentioned by name in the interface. The Obtain An IP Address Automatically option in the Internet Protocol Version 4 (TCP/IPv4) Properties sheet and the Obtain An IPv6 Address Automatically option in the Internet Protocol Version 6 (TCP/IPv6) Properties sheet control the activation of the client for IPv4 and IPv6, respectively.

DHCP LEASE NEGOTIATION

DHCP communication is always initiated by the client, as shown in Figure 4-6, and proceeds as follows:

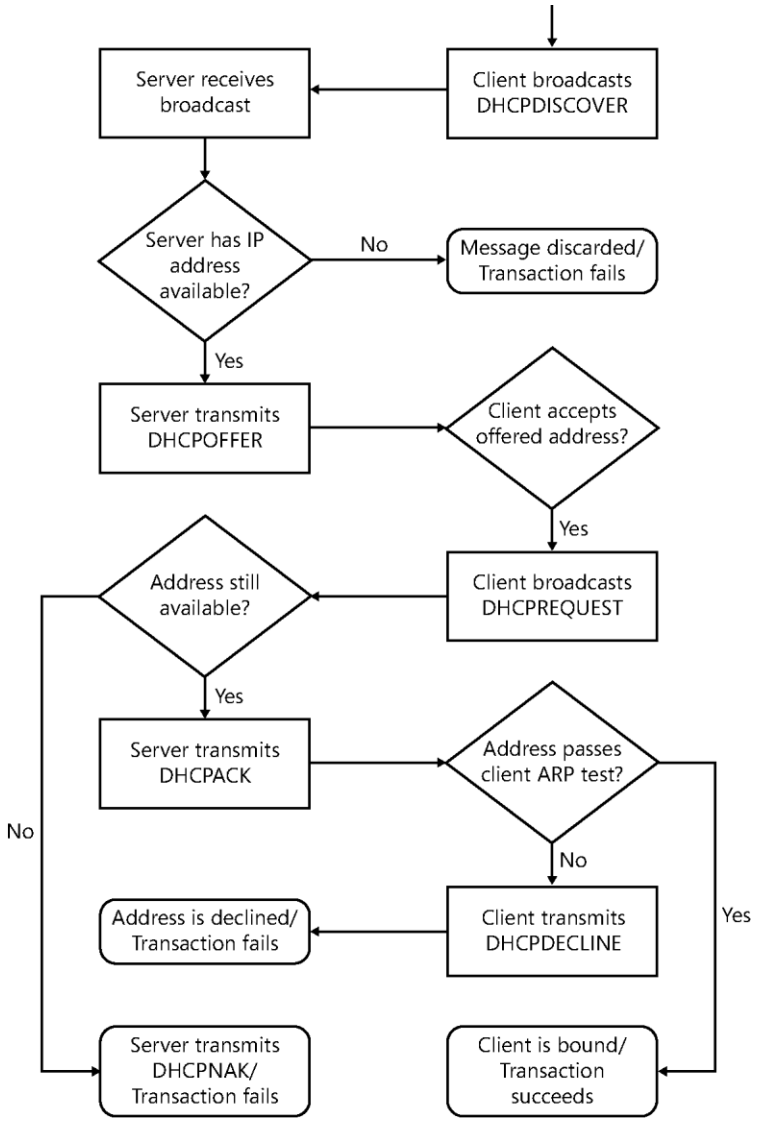

FIGURE 4-6 The DHCP IP address assignment process.

1. When a computer boots for the first time with the DHCP client active, the client generates a series of DHCPDISCOVER messages to solicit an IP address assignment from a DHCP server and broadcasts them on the local network.
2. All DHCP servers receiving the DHCPDISCOVER broadcast messages generate DHCPOFFER messages containing an IP address and other TCP/IP configuration parameters, and transmit them to the client.
3. After a specified period, the client stops broadcasting and signals its acceptance of

one of the offered addresses by generating a DHCPREQUEST message containing the address of the server from which it is accepting the offer, and broadcasting it on the local network.

4. When the server offering the accepted IP address receives the DHCPREQUEST message, it adds the offered IP address and other settings to its database.
5. The server then transmits a DHCPACK message to the client, acknowledging the completion of the process. If the server cannot complete the assignment, it transmits a DHCPNAK message to the client and the whole process begins again.
6. As a final test, the client transmits the offered IP address in a broadcast using the Address Resolution Protocol (ARP), to ensure that no other system on the network is using the same address. If the client receives no response to the ARP broadcast, the DHCP transaction is completed. If another system does respond to the ARP message, the client discards the IP address and transmits a DHCPDECLINE message to the server, nullifying the transaction. The client then restarts the entire process.

DHCP LEASE RENEWAL

By default, the DHCP Server service in Windows Server 2012 uses dynamic allocation, leasing IP addresses to clients for eight-day periods. At periodic intervals during the course of the lease, the client attempts to contact the server to renew the lease, as shown in Figure 4-7, using the following procedure:

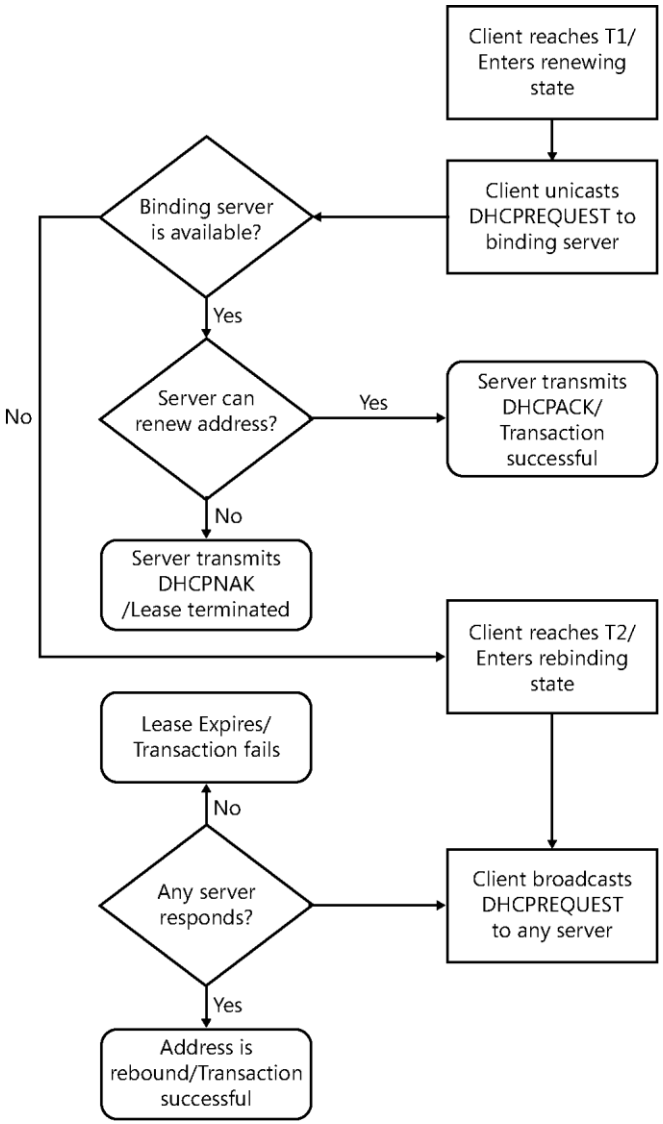

FIGURE 4-7 The DHCP IP address renewal process.

1. When the DHCP client reaches the 50 percent point of the lease's duration (called the renewal time value or T1 value), the client begins generating DHCPREQUEST messages and transmitting them to the DHCP server holding the lease as unicasts.
2. If the server does not respond by the time the client reaches the 87.5 percent point of the lease's duration (called the rebinding time value or T2 value), the client begins transmitting its DHCPREQUEST messages as broadcasts in an attempt to solicit an IP address assignment from any DHCP server on the network.

3. If the server receives the DHCPREQUEST message from the client, it responds with either a DHCPACK message, approving the lease renewal request, or a DHCPNAK message, which terminates the lease. If the client receives no responses to its DHCPREQUEST messages by the time the lease expires, or if it receives a DHCPNAK message, the client releases its IP address. All TCP/IP communication then ceases, except for the transmission of DHCPDISCOVER broadcasts.

Deploying a DHCP server

DHCP servers operate independently, so you must install the service and configure scopes on every computer that will function as a DHCP server. The DHCP Server service is packaged as a role in Windows Server 2012, which you can install using the Add Roles and Features Wizard, accessible from the Server Manager console.

When you install the DHCP Server role on a computer that is a member of an Active Directory Domain Services domain, the DHCP Server is automatically authorized to allocate IP addresses to clients that are also members of the same domain. If the server is not a domain member when you install the role, and you join it to a domain later, you must manually authorize the DHCP server in the domain by right-clicking the server node in the DHCP console and, from the shortcut menu, selecting Authorize.

After installing the DHCP Server role, you must configure the service by creating a scope before it can serve clients.

Creating a scope

A scope is a range of IP addresses on a particular subnet that are selected for allocation by a DHCP server. In Windows Server versions prior to 2012, you can create a scope as you install the DHCP Server role. However, in Windows Server 2012, the procedures are separated. To create a scope using the DHCP snap-in for Microsoft Management Console (MMC), use the following procedure.

1. Log on to Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.
2. Click Tools > DHCP. The DHCP console opens.
3. Expand the server node and the IPv4 node.
4. Right-click the IPv4 node and, from the shortcut menu, select New Scope. The New Scope Wizard opens, displaying the Welcome page.
5. Click Next. The Scope Name page appears.
6. Type a name for the scope into the Name text box and click Next. The IP Address Range page opens, as shown in Figure 4-8.

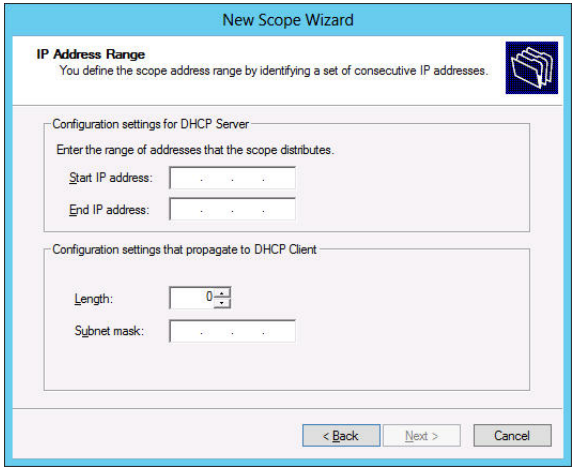

FIGURE 4-8 The Address Range page in the DHCP console.

7. In the Start IP Address text box, type the first in the range of addresses you want to assign. In the End IP Address box, type the last address in the range.
8. In the Subnet Mask text box, type the mask value for the subnet on which the scope will operate and click Next. The Add Exclusions And Delay page appears.
9. In the Start IP Address and End IP Address text boxes, specify a range of addresses you want to exclude from the scope. You can also specify a delay interval between the server's receipt of DHCPDISCOVER messages and its transmission of DHCPOFFER messages. Then click Next to open the Lease Duration page.
10. Specify the length of the leases for the addresses in the scope and click Next. The Configure DHCP Options page opens.
11. Select Yes, I Want To Configure These Options Now and click Next. The Router (Default Gateway) page opens, as shown in Figure 4-9.

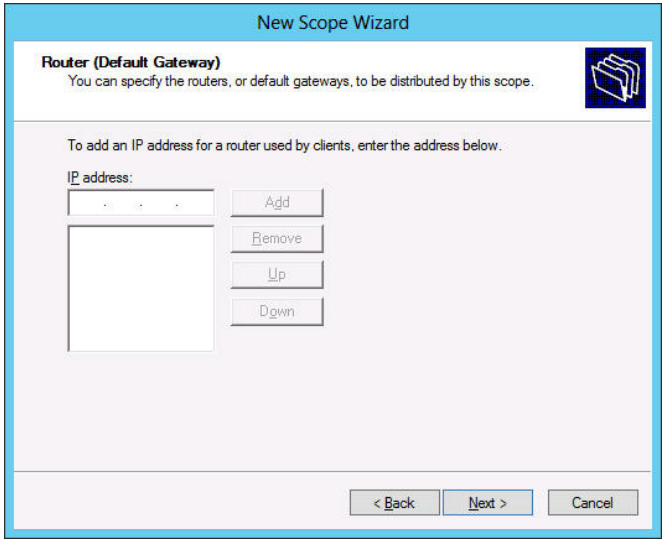

FIGURE 4-9 The Router (Default Gateway) page in the DHCP console.

12. In the IP Address text box, specify the address of a router on the subnet served by the scope and click Add. Then click Next. The Domain Name And DNS Servers page opens.
13. In the Server Name text box, type the name of a DNS server on the network and click Resolve, or type the address of a DNS server in the IP Address text box and click Add. Then click Next. The WINS Servers page opens.
14. Click Next to open the Activate Scope page.
15. Select Yes, I Want To Activate This Scope Now and click Next. The Completing The New Scope Wizard page opens.
16. Click Finish to close the wizard.
17. Close the DHCP console.

Once the role installation is completed, all of the DHCP clients on the subnet identified in the scope you created can obtain their IP addresses and other TCP/IP configuration settings via DHCP. You can also use the DHCP console to create additional scopes for other subnets.

Configuring DHCP options

The New Scope Wizard enables you to configure a few of the most commonly used DHCP options as you create a new scope, but you can always configure the many other options at a later time.

The Windows DHCP server supports two kinds of options:

- **Scope options** Options supplied only to DHCP clients receiving addresses from a particular scope
- **Server options** Options supplied to all DHCP clients receiving addresses from the server

The Router option is a typical example of a scope option, because a DHCP client's default gateway address must be on the same subnet as its IP address. The DNS Servers option is typically a scope option, because DNS servers do not have to be on the same subnet, and networks often use the same DNS servers for all of their clients.

All of the options supported by the Windows DHCP server can be either scope or server options, and the process of configuring them is basically the same. To configure a scope option, right-click the Scope Options node and, from the shortcut menu, select Configure Options. The Scope Options dialog box, which provides appropriate controls for each of the available options, opens (Figure 4-10).

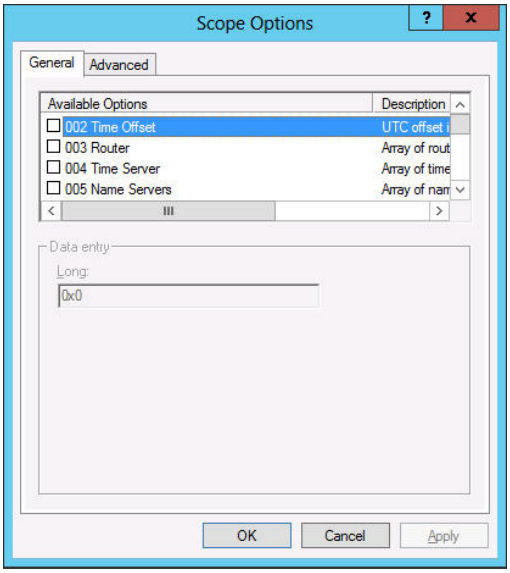

FIGURE 4-10 The Scope Options dialog box.

Right-clicking the Server Options node enables you to open the Server Options dialog box, which behaves in exactly the same way.

Creating a reservation

Although DHCP is an excellent TCP/IP configuration solution for most of the computers on a network, there are a few for which it is not. Domain controllers, Internet web servers, and DHCP servers themselves need static IP addresses.

Because the DHCP dynamic allocation method allows for the possibility that a computer's IP address could change, it is not appropriate for these particular roles. However, it is still possible to assign addresses to these computers with DHCP, using manual, instead of dynamic, allocation.

In a Windows DHCP server, a manually allocated address is called a reservation. You create a reservation by expanding the scope node, right-clicking the Reservations node, and, from the shortcut menu, selecting New Reservation. The New Reservation dialog box opens, as shown in Figure 4-11.

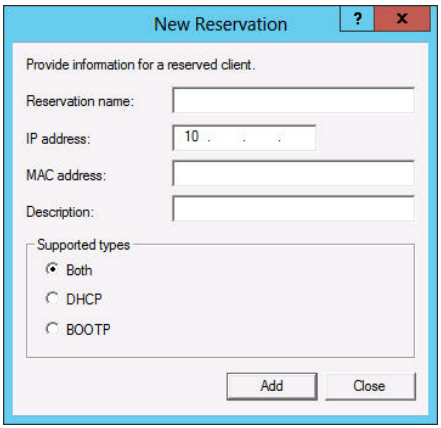

FIGURE 4-11 A DHCP server's New Reservation dialog box.

In this dialog box, you specify the IP address you want to assign and associate it with the client computer's MAC address, which is hard-coded into its network interface adapter.

Of course, it is also possible to manually configure the computer's TCP/IP client, but creating a DHCP reservation ensures that all of your IP addresses are managed by your DHCP servers. In a large enterprise, where various administrators might be dealing with DHCP and TCP/IP configuration issues, the IP address that one technician manually assigns to a computer might be included in a DHCP scope by another technician, resulting in potential addressing conflicts. Reservations create a permanent record of the IP address assignment on the DHCP server.

Using PXE

The Windows operating systems include a DHCP client that can configure the IP address and other TCP/IP settings of computers with an operating system already installed. However, it is also possible for a bare metal computer—that is, a computer with no operating system installed—to use DHCP.

The Preboot Execution Environment (PXE) is a feature built into many network interface adapters that enables them to connect to a DHCP server over the network and obtain TCP/IP client settings, even when there is no operating system on the computer. Administrators

typically use this capability to automate the operating system deployment process on large fleets of workstations.

In addition to configuring the IP address and other TCP/IP client settings on the computer, the DHCP server can also supply the workstation with an option specifying the location of a boot file that the system can download and use to start the computer and initiate a Windows operating system installation. A PXE-equipped system downloads boot files using the Trivial File Transfer Protocol (TFTP), a simplified version of the FTP protocol that requires no authentication.

Windows Server 2012 includes a role called Windows Deployment Services (WDS), which enables administrators to manage image files that remote workstations can use to start up and install Windows. For a PXE adapter to access WDS images, the DHCP server on the network must have a custom PXEClient option (option 60) configured with the location of the WDS server on the network.

The PXE client on the workstation typically needs no configuration, except possibly for an alteration of the boot device order, so that the computer attempts a network boot before using the local devices.

In a properly configured WDS installation of Windows 8, the client operating system deployment process proceeds as follows:

1. The client computer starts and, finding no local boot device, attempts to perform a network boot.
2. The client computer connects to a DHCP server on the network, from which it obtains a DHCPOFFER message containing an IP address and other TCP/IP configuration parameters, plus the 060 PXEClient option, containing the name of a WDS server.
3. The client connects to the WDS server and is supplied with a boot image file, which it downloads using TFTP.
4. The client loads Windows PE and the WDS client from the boot image file onto a RAM disk (a virtual disk created out of system memory) and displays a boot menu containing a list of the install images available from the WDS server.
5. The user on the client computer selects an install image from the boot menu, and the operating system installation process begins. From this point, the setup process proceeds just like a manual installation.

MORE INFO **Windows Deployment Services**

For more information on using WDS, see Objective 1.1, "Deploy and Manage Server Images," in Exam 70-411, "Administering Windows Server 2012."

Deploying a DHCP relay agent

If you opt to create a centralized or hybrid DHCP infrastructure, you will need a DHCP relay agent on every subnet that does not have a DHCP server on it. Many routers are capable of

functioning as DHCP relay agents, but in situations where they are not, you can configure a Windows Server 2012 computer to function as a relay agent, using the following procedure.

1. Log on to Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.
2. Using the Add Roles and Features Wizard, install the Remote Access role, including the Routing role service.
3. Click Open The Getting Started Wizard. The Configure Remote Access Getting Started Wizard opens.
4. Click Deploy VPN Only. The Routing And Remote Access console appears.
5. Right-click the server node and, on the shortcut menu, select Configure And Enable Routing And Remote Access. The Routing and Remote Access Server Setup Wizard appears.
6. Click Next to bypass the Welcome page. The Configuration page opens, as shown in Figure 4-12.

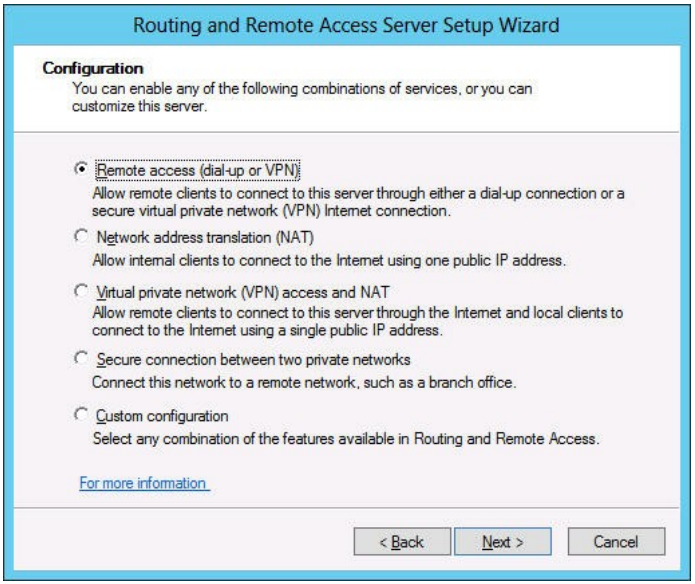

FIGURE 4-12 The Configuration page of the Routing and Remote Access Server Setup Wizard.

7. Select Custom Configuration and click Next. The Custom Configuration page appears.
8. Select the LAN Routing check box and click Next. The Completing The Routing And Remote Access Server Setup Wizard page opens.
9. Click Finish. A Routing and Remote Access message box appears, prompting you to start the service.
10. Click Start Service.

11. Expand the IPv4 node. Then, right-click the General node and, in the shortcut menu, select New Routing Protocol. The New Routing Protocol dialog box appears.
12. Select DHCP Relay Agent and click OK. A DHCP Relay Agent node appears, subordinate to the IPv4 node.
13. Right-click the DHCP Relay Agent node and, on the shortcut menu, select New Interface. The New Interface For DHCP Relay Agent dialog box appears.
14. Select the interface to the subnet on which you want to install the relay agent and click OK. The DHCP Relay Properties sheet for the interface appears.
15. Leave the Relay DHCP Packets check box selected, and configure the following settings, if needed.
    - **Hop-count threshold** Specifies the maximum number of relay agents that DHCP messages can pass through before being discarded. The default value is 4 and the maximum value is 16. This setting prevents DHCP messages from being relayed endlessly around the network.
    - **Boot threshold** Specifies the time interval (in seconds) that the relay agent should wait before forwarding each DHCP message it receives. The default value is 4 seconds. This setting enables you to control which DHCP server processes the clients for a particular subnet.
16. Click OK.
17. Right-click the DHCP Relay Agent node and, on the shortcut menu, select Properties. The DHCP Relay Agent Properties sheet appears, as shown in Figure 4-13.

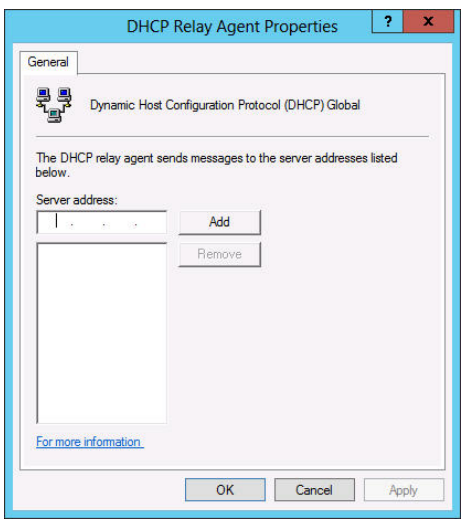

FIGURE 4-13 The DHCP Relay Agent Properties sheet.

18. Type the IP address of the DHCP server to which you want the agent to relay messages and click Add. Repeat this step to add additional servers, if necessary.

19. Click OK.
20. Close the Routing And Remote Access console.

At this point, the server is configured to relay DHCP messages to the server addresses you specified.

Objective summary

- DHCP is a service that automatically configures the IP address and other TCP/IP settings on network computers by assigning addresses from a pool (called a scope) and reclaiming them when they are no longer in use
- DHCP consists of three components: a DHCP server application, a DHCP client, and a DHCP communications protocol.
- The DHCP standards define three different IP address allocation methods: dynamic allocation, automatic allocation, and manual allocation.

Objective review

Answer the following questions to test your knowledge of the information in this objective. You can find the answers to these questions and explanations of why each answer choice is correct or incorrect in the "Answers" section at the end of this chapter.

1. Which of the following is the term for the component that enables DHCP clients to communicate with DHCP servers on other subnets?
   A. Forwarder
   B. Resolver
   C. Scope
   D. Relay agent
2. Which of the following message types is not used during a successful DHCP address assignment?
   A. DHCPDISCOVER
   B. DHCPREQUEST
   C. DHCPACK
   D. DHCPINFORM
3. Which of the following types of DHCP address allocation is the equivalent of a reservation in Windows Server 2012?
   A. Dynamic allocation
   B. Automatic allocation
   C. Manual allocation
   D. Hybrid allocation

4. Which of the following network components are typically capable of functioning as DHCP relay agents?
   A. Windows 8 computers
   B. Routers
   C. Switches
   D. Windows Server 2012 computers
5. Which of the following TCP/IP parameters is typically deployed as a scope option in DHCP?
   A. DNS Server
   B. Subnet Mask
   C. Lease Duration
   D. Default Gateway

Thought experiment

In the following thought experiment, apply what you've learned about this objective to predict what steps you need to take. You can find answers to these questions in the "Answers" section at the end of this chapter.

After deploying a large number of wireless laptop computers on the network, Ralph, the IT director at Contoso, Ltd., decides to use DHCP to enable the laptop users to move from one subnet to another without having to manually reconfigure their IP addresses. Soon after the DHCP deployment, however, Ralph notices that some of the IP address scopes are being depleted, resulting in some computers being unable to connect to a new subnet.

With this in mind, answer the following question:

What can Ralph do to resolve this problem without altering the network's subnetting?

Objective 4.3: Deploy and configure the DNS service

DNS is a crucial element of both Internet and Active Directory communications. All TCP/IP communication is based on IP addresses. Each computer on a network has at least one network interface, which is called a host, in TCP/IP parlance, and each host has an IP address that is unique on that network. Every datagram transmitted by a TCP/IP system contains the IP address of the sending computer and the IP address of the intended recipient. However, when users access a shared folder on the network or a website on the Internet, they do so by specifying or selecting a host name, not an IP address. This is because names are far easier to

remember and use than IP addresses.

This objective covers how to:

- Configure Active Directory integration of primary zones
- Configure forwarders
- Configure root hints
- Manage DNS cache
- Create A and PTR resource records

Understanding the DNS architecture

For TCP/IP systems to use these friendly host names, they must have some way to discover the IP address associated with a specific name. In the early days of TCP/IP networking, each computer had a list of names and their equivalent IP addresses, called a host table. At that time, there were few enough computers on the fledgling Internet for the maintenance and distribution of a single host table to be practical.

Today, there are many millions of computers on the Internet, and the idea of maintaining and distributing a single file containing names for all of them is absurd. Instead of a host table stored on every computer, TCP/IP networks today use DNS servers to convert host names into IP addresses. This conversion process is referred to as name resolution.

At its core, the DNS is still a list of names and their equivalent IP addresses, but the methods for creating, storing, and retrieving those names is very different from those in a host table. The DNS consists of three elements:

- **The DNS namespace** The DNS standards define a tree-structured namespace in which each branch of the tree identifies a domain. Each domain contains a collection of resource records that contain host names, IP addresses, and other information. Query operations are attempts to retrieve specific resource records from a particular domain.
- **Name servers** A DNS server is an application running on a server computer that maintains information about the domain tree structure and (usually) contains authoritative information about one or more specific domains in that structure. The application is capable of responding to queries for information about the domains for which it is the authority, and also of forwarding queries about other domains to other name servers. This enables any DNS server to access information about any domain in the tree.
- **Resolvers** A resolver is a client program that generates DNS queries and sends them to a DNS server for fulfillment. A resolver has direct access to at least one DNS server and can also process referrals to direct its queries to other servers when necessary.

In its most basic form, the DNS name resolution process consists of a resolver submitting a name resolution request to its designated DNS server. When the server does not possess

information about the requested name, it forwards the request to another DNS server on the network. The second server generates a response containing the IP address of the requested name and returns it to the first server, which relays the information in turn to the resolver, as shown in Figure 4-14. In practice, however, the DNS name resolution process can be considerably more complex, as you will learn in the following sections.

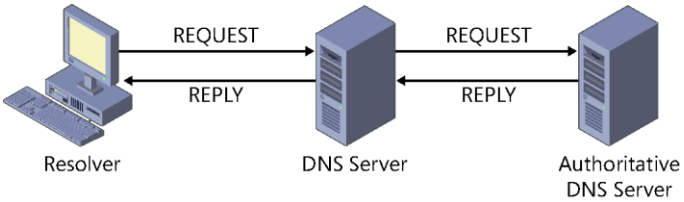

FIGURE 4-14 DNS servers relay requests and replies to other DNS servers.

DNS communications

Although all Internet applications use DNS to resolve host names into IP addresses, this name resolution process is easiest to see when you're using a web browser to access an Internet site. When you type a URL containing a DNS name (for example, www.microsoft.com) into the browser's Address box and press the Enter key, if you look quickly enough, you might be able to see a message that says something like "Finding Site: www.microsoft.com." Then, a few seconds later, you might see a message that says "Connecting to," followed by an IP address. It is during this interval that the DNS name resolution process occurs.

From the client's perspective, the procedure that occurs during these few seconds consists of the application sending a query message to its designated DNS server that contains the name to be resolved. The server then replies with a message containing the IP address corresponding to that name. Using the supplied address, the application can then transmit a message to the intended destination. It is only when you examine the DNS server's role in the process that you see how complex the procedure really is.

To better explain the relationship of the DNS servers for various domains in the namespace, the following procedure diagrams the Internet name resolution process.

1. A user on a client system specifies the DNS name of an Internet server in an application such as a web browser. The application generates an application programming interface (API) call to the resolver on the client system, and the resolver creates a DNS recursive query message containing the server name, which it transmits to the DNS server identified in computer's TCP/IP configuration, as shown in Figure 4-15.

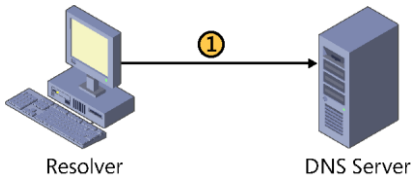

FIGURE 4-15 The client resolver sends a name resolution request to its DNS server.

2. The client's DNS server, after receiving the query, checks its resource records to see if it is the authoritative source for the zone containing the requested server name. If it is not, which is typical, the DNS server generates an iterative query and submits it to one of the root name servers, as shown in Figure 4-16. The root name server examines the name requested by the client's DNS server and consults its resource records to identify the authoritative servers for the name's top-level domain. The root name server then transmits a reply to the client's DNS server that contains a referral to the top-level domain server addresses.

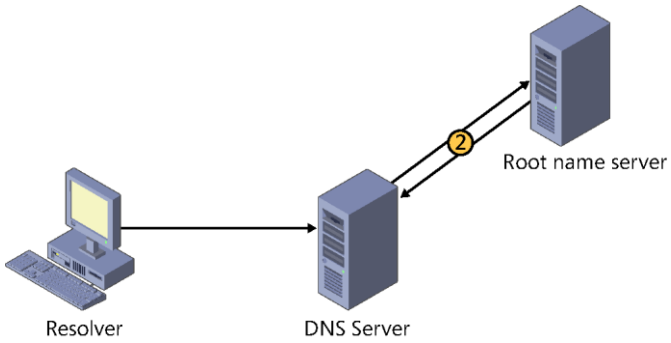

FIGURE 4-16 The client's DNS server forwards the request to a root name server.

3. The client's DNS server, now in possession of the top-level domain server address for the requested name, generates a new iterative query and transmits it to the top-level domain server, as shown in Figure 4-17. The top-level domain server examines the second-level domain in the requested name and transmits a referral containing the addresses of authoritative servers for that second-level domain back to the client's DNS server.

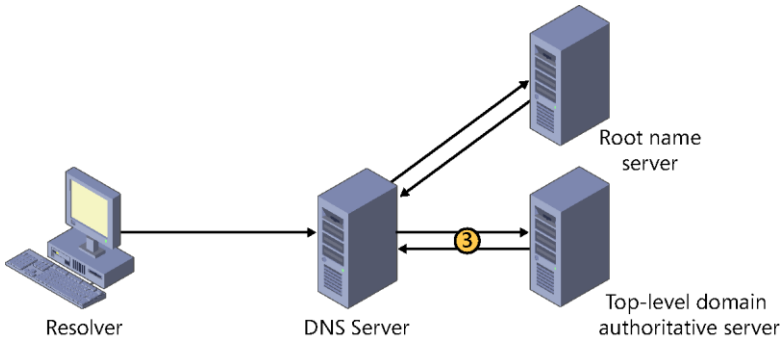

FIGURE 4-17 The client's DNS server forwards the request to a top-level domain server.

Note **COMBINING STEPS**

In the DNS name resolution process just described, the process of resolving the top-level and second-level domain names is portrayed in separate steps, but this is often not the case. The most commonly used top-level domains, such as com, net, and org, are actually hosted by the root name servers, which eliminates one entire referral from the name resolution process.

4. The client's DNS server generates yet another iterative query and transmits it to the second-level domain server, as shown in Figure 4-18. If the second-level domain server is the authority for the zone containing the requested name, it consults its resource records to determine the IP address of the requested system and transmits it in a reply message back to that client's DNS server.

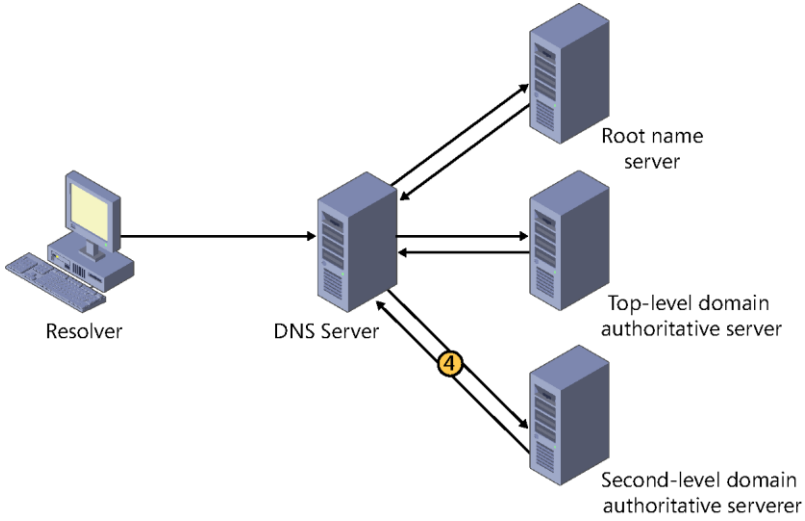

FIGURE 4-18 The client's DNS server forwards the request to a second-level domain server.

5. The client's DNS server receives the reply from the authoritative server and transmits the IP address back to the resolver on the client system, as shown in Figure 4-19. The resolver relays the address to the application, which can then initiate IP communications with the system specified by the user.

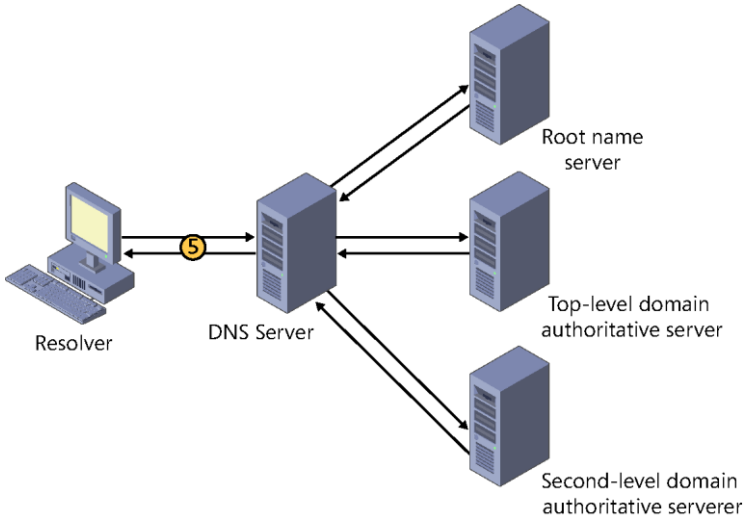

FIGURE 4-19 The client's DNS server responds to the client resolver.

Depending on the name the client is trying to resolve, this process can be simpler or considerably more complex than the one shown here. If, for example, the client's DNS server is the authority for the domain in which the requested name is located, no other servers or iterative requests are necessary. On the other hand, if the requested name contains three or more levels of domains, additional iterative queries might be necessary.

This procedure also assumes a successful completion of the name resolution procedure. If any of the authoritative DNS servers queried returns an error message to the client's DNS server stating, for example, that one of the domains in the name does not exist, then this error message is relayed back to the client and the name resolution process is said to have failed.

DNS server caching

The DNS name resolution process might seem long and complex, but in many cases, it isn't necessary for the client's DNS server to send queries to the servers for each domain specified in the requested DNS name. This is because DNS servers are capable of retaining the information they learn about the DNS namespace in the course of their name resolution procedures and storing it in a cache on the local drive.

A DNS server that receives requests from clients, for example, caches the addresses of the requested systems, as well as the addresses for authoritative servers of particular domains. The next time that a client requests the resolution of a previously resolved name, the server can respond immediately with the cached information. In addition, if a client requests another

name in one of the same domains, the server can send a query directly to an authoritative server for that domain, and not to a root name server. Thus, the names in commonly accessed domains generally resolve more quickly, because one of the servers along the line has information about the domain in its cache, while names in obscure domains take longer, because the entire request/referral process is needed.

Caching is a vital element of the DNS architecture, because it reduces the number of requests sent to the root name and top-level domain servers, which, being at the top of the DNS tree, are the most likely to act as a bottleneck for the whole system. However, caches must be purged eventually, and there is a fine line between effective and ineffective caching.

Because DNS servers retain resource records in their caches, it can take hours or even days for changes made in an authoritative server to be propagated around the Internet. During this period, users might receive incorrect information in response to a query. If information remains in server caches too long, then the changes that administrators make to the data in their DNS servers take too long to propagate around the Internet. If caches are purged too quickly, then the number of requests sent to the root name and top-level domain servers increases precipitously.

The amount of time that DNS data remains cached on a server is called its time to live (TTL). Unlike most data caches, the TTL is not specified by the administrator of the server where the cache is stored. Instead, the administrators of each authoritative DNS server specify how long the data for the resource records in their domains or zones should be retained in the servers where it is cached. This enables administrators to specify a TTL value based on the volatility of their server data. On a network where changes in IP addresses or the addition of new resource records is frequent, a lower TTL value increases the likelihood that clients will receive current data. On a network that rarely changes, you can use a longer TTL value, and minimize the number of requests sent to the parent servers of your domain or zone.

To modify the TTL value for a zone on a Windows Server 2012 DNS server, right-click the zone, open the Properties sheet, and click the Start Of Authority (SOA) tab, as shown in Figure 4-20. On this tab, you can modify the TTL for this record setting from its default value of one hour.

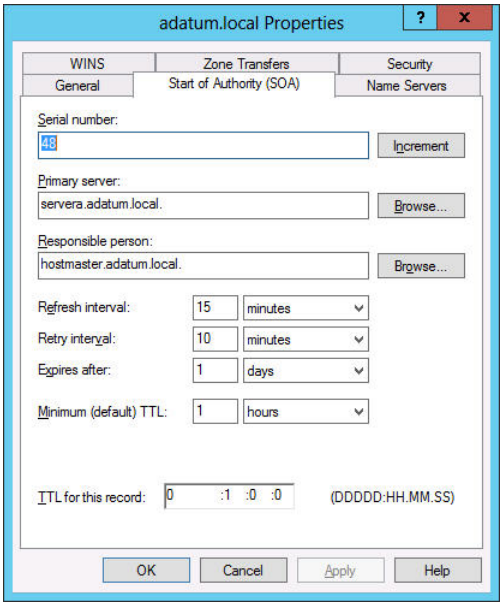

FIGURE 4-20 The Start Of Authority (SOA) tab on a DNS server's Properties sheet.

DNS referrals and queries

The process by which one DNS server sends a name resolution request to another DNS server is called a referral. Referrals are essential to the DNS name resolution process.

As you noticed in the process described earlier, the DNS client is not involved in the name resolution process at all, except for sending one query and receiving one reply. The client's DNS server might have to send referrals to several servers before it reaches the one that has the information it needs.

DNS servers recognize two types of name resolution requests, as follows:

- **Recursive query** In a recursive query, the DNS server receiving the name resolution request takes full responsibility for resolving the name. If the server possesses information about the requested name, it replies immediately to the requestor. If the server has no information about the name, it sends referrals to other DNS servers until it obtains the information it needs. TCP/IP client resolvers always send recursive queries to their designated DNS servers.
- **Iterative query** In an iterative query, the server that receives the name resolution request immediately responds with the best information it possesses at the time. DNS servers use iterative queries when communicating with each other. In most cases, it would be improper to configure one DNS server to send a recursive query to another DNS server. The only time a DNS server does send iterative queries to another server is in the case of a special type of server called a forwarder, which is specifically configured to interact with other servers in this way.

DNS forwarders

One of the scenarios in which DNS servers do send recursive queries to other servers is when you configure a server to function as a forwarder. On a network running several DNS servers, you might not want all of the servers sending queries to other DNS servers on the Internet. If the network has a relatively slow connection to the Internet, for example, several servers transmitting repeated queries might use too much of the available bandwidth.

To prevent this, most DNS implementations enable you to configure one server to function as the forwarder for all Internet queries generated by the other servers on the network. Any time that a server has to resolve the DNS name of an Internet system and fails to find the needed information in its cache, it transmits a recursive query to the forwarder, which is then responsible for sending its own iterative queries over the Internet connection. Once the forwarder resolves the name, it sends a reply back to the original DNS server, which relays it to the client.

To configure forwarders on a Windows Server 2012 DNS server, right-click the server node, open the Properties sheet, and click the Forwarders tab, as shown in Figure 4-21. On this tab, you can add the names and addresses of the servers that you want your server to use as forwarders.

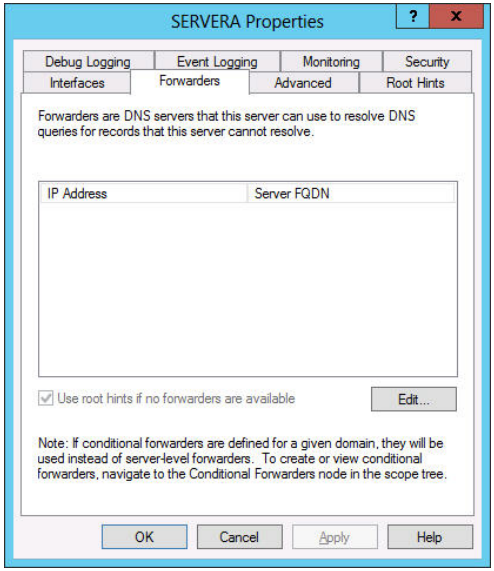

FIGURE 4-21 The Forwarders tab on a DNS server's Properties sheet.

Reverse name resolution

The name resolution process described earlier is designed to convert DNS names into IP addresses. However, there are occasions when it is necessary for a computer to convert an IP address into a DNS name. This is called a reverse name resolution.

Because the domain hierarchy is broken down by domain names, there is no apparent way to resolve an IP address into a name using iterative queries, except by forwarding the reverse name resolution request to every DNS server on the Internet in search of the requested address, which is obviously impractical.

To overcome this problem, the developers of the DNS created a special domain called in-addr.arpa, specifically designed for reverse name resolution. The in-addr.arpa second-level domain contains four additional levels of subdomains. Each of the four levels consists of subdomains that are named using the numerals 0 to 255. For example, beneath in-addr.arpa, there are 256 third-level domains, which have names ranging from 0.in-addr.arpa to 255.in-addr.arpa. Each of those 256 third-level domains has 256 fourth-level domains beneath it, also numbered from 0 to 255, and each fourth-level domain has 256 fifth-level domains, as shown in Figure 4-22. Each of those fifth-level domains can then have up to 256 hosts in it, also numbered from 0 to 255.

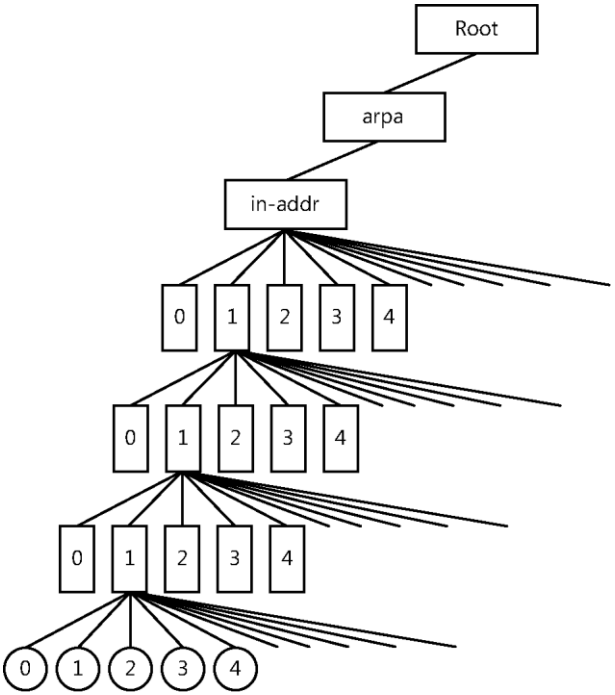

FIGURE 4-22 The DNS reverse lookup domain.

Using this hierarchy of subdomains, it is possible to express the first three bytes of an IP address as a DNS domain name, and to create a resource record named for the fourth byte in the appropriate fifth-level domain. For example, to resolve the IP address 192.168.89.34 into a name, a DNS server would locate a domain called 89.168.192.in-addr.arpa in the usual manner and read the contents of a resource record named 34 in that domain.

NOTE **Reverse lookup addresses**

In the in-addr.arpa domain, the IP address is reversed in the domain name because IP addresses have the least pertinent bit (that is, the host identifier) on the right and in DNS fully qualified domain names (FQDNs), the host name is on the left.

Deploying a DNS server

The process of actually deploying a DNS server on a Windows Server 2012 computer is simply a matter of installing the DNS Server role, using the Add Roles and Features Wizard in Server Manager. The actual installation requires no additional input; there are no additional pages in the wizard and no role services to select.

Once you install the DNS Server role, the computer is ready to perform caching-only name resolution services for any clients that have access to it. The role also installs the DNS Manager console, which you use to configure the DNS server's other capabilities. To configure the server to perform other services, consult the following sections.

Creating zones

A zone is an administrative entity you create on a DNS server to represent a discrete portion of the DNS namespace. Administrators typically divide the DNS namespace into zones to store them on different servers and to delegate their administration to different people. Zones always consist of entire domains or subdomains. You can create a zone that contains multiple domains, as long as those domains are contiguous in the DNS namespace. For example, you can create a zone containing a parent domain and its child, because they are directly connected, but you cannot create a zone containing two child domains without their common parent, because the two children are not directly connected, as shown in Figure 4-23.

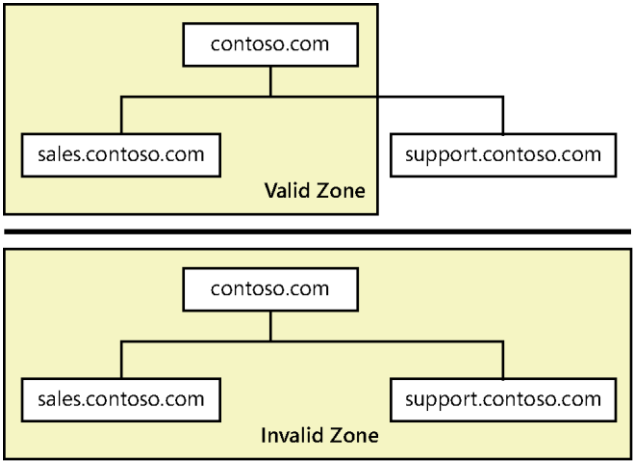

FIGURE 4-23 Valid zones must consist of contiguous domains.

You can divide the DNS namespace into multiple zones and host them on a single DNS server if you want to, although there is usually no persuasive reason to do so. The DNS server in Windows Server 2012 can support as many as 200,000 zones on a single server, although it is hard to imagine a scenario that would require that many. In most cases, an administrator creates multiple zones on a server and then delegates most of them to other servers, which then become responsible for hosting them.

Every zone consists of a zone database, which contains the resource records for the domains in that zone. The DNS server in Windows Server 2012 supports three zone types, which specify where the server stores the zone database and what kind of information it contains. These zone types are as follows:

- **Primary zone** Creates a primary zone that contains the master copy of the zone database, where administrators make all changes to the zone's resource records. If the Store The Zone In Active Directory (Available Only If DNS Server Is A Domain Controller) check box is cleared, the server creates a primary master zone database file on the local drive. This is a simple text file that is compliant with most non-Windows DNS server implementations.
- **Secondary zone** Creates a duplicate of a primary zone on another server. The secondary zone contains a backup copy of the primary master zone database file, stored as an identical text file on the server's local drive. You can only update the resource records in a secondary zone by replicating the primary master zone database file, using a process called a zone transfer.
- **Stub zone** Creates a copy of a primary zone that contains the key resource records that identify the authoritative servers for the zone. The stub zone forwards or refers requests. When you create a stub zone, you configure it with the IP address of the server that hosts the zone from which you created the stub. When the server hosting the stub zone receives a query for a name in that zone, it either forwards the request to the host of the zone or replies with a referral to that host, depending on whether the query is recursive or iterative.

DNS was designed long before Active Directory, so most of the Internet relies on primary and secondary zones using text-based database files. The most common DNS server implementation on the Internet is a UNIX program called bind that uses these databases.

However, for DNS servers supporting internal domains, and especially AD DS domains, using the Windows DNS server to create a primary zone and store it in Active Directory is the recommended procedure. When you store the zone in the AD DS database, you do not have to create secondary zones or perform zone transfers, because AD DS takes the responsibility for replicating the data, and whatever backup solution you use to protect Active Directory protects the DNS data as well.

EXAM TIP

Exam 70-410 covers only the process of creating a primary zone stored in Active Directory. The procedures for creating text-based primary and secondary zones and configuring zone transfers are covered on Exam 70-411, "Administering Windows Server 2012," in Objective 3.1, "Configure DNS zones."

USING ACTIVE DIRECTORY–INTEGRATED ZONES

When you are running the DNS server service on a computer that is an Active Directory Domain Services domain controller and you select the Store The Zone In Active Directory (Available Only If DNS Server Is A Domain Controller) check box while creating a zone in the New Zone Wizard, the server does not create a zone database file. Instead, the server stores the DNS resource records for the zone in the AD DS database. Storing the DNS database in Active Directory provides a number of advantages, including ease of administration, conservation of network bandwidth, and increased security.

In Active Directory–integrated zones, the zone database is replicated automatically to other domain controllers, along with all other Active Directory data. Active Directory uses a multiple master replication system so that copies of the database are updated on all domain controllers in the domain. You can modify the DNS resource records on any domain controller hosting a copy of the zone database, and Active Directory will update all of the other domain controllers automatically. You don't have to create secondary zones or manually configure zone transfers, because Active Directory performs all database replication activities.

By default, Windows Server 2012 replicates the database for a primary zone stored in Active Directory to all the other domain controllers running the DNS server in the AD DS domain where the primary domain controller is located. You can also modify the scope of zone database replication to keep copies on all domain controllers throughout the enterprise, or on all domain controllers in the AD DS domain, whether or not they are running the DNS server. You can also create a custom replication scope that copies the zone database to the domain controllers you specify.

Active Directory conserves network bandwidth by replicating only the DNS data that has changed since the last replication, and by compressing the data before transmitting it over the network. The zone replications also use the full security capabilities of Active Directory, which are considerably more robust than those of file-based zone transfers.

CREATING AN ACTIVE DIRECTORY ZONE

To create a new primary zone and store it in Active Directory, use the following procedure.

1. Log on to the Windows Server 2012 domain controller using an account with Administrative privileges. The Server Manager window opens.
2. Click Tools > DNS to open the DNS Manager console.
3. Expand the server node and select the Forward Lookup Zones folder.

4. Right-click the Forward Lookup Zones folder and, from the shortcut menu, select New Zone. The New Zone Wizard starts.
5. Click Next to bypass the Welcome page and open the Zone Type page.
6. Leave the Primary Zone option and the Store The Zone In Active Directory (Available Only If DNS Server Is A Domain Controller) check box selected and click Next. The Active Directory Zone Replication Scope page opens.
7. Click Next. The Zone Name page opens.
8. Specify the name you want to assign to the zone in the Zone Name text box and click Next. The Dynamic Update page opens.
9. Select one of the following options:
    - Allow Only Secure Dynamic Updates
    - Allow Both Nonsecure And Secure Dynamic Updates
    - Do Not Allow Dynamic Updates
10. Click Next. The Completing the New Zone Wizard page opens.
11. Click Finish. The wizard creates the zone.
12. Close the DNS Manager console.

Once you have created a primary zone, you can now proceed to create resource records that specify the names of the hosts on the network and their equivalent IP addresses.

Creating resource records

When you run your own DNS server, you create a resource record for each host name that you want to be accessible by the rest of the network.

There are several different types of resource records used by DNS servers, the most important of which are as follows:

- **SOA (Start of Authority)** Indicates that the server is the best authoritative source for data concerning the zone. Each zone must have an SOA record, and only one SOA record can be in a zone.
- **NS (Name Server)** Identifies a DNS server functioning as an authority for the zone. Each DNS server in the zone (whether primary master or secondary) must be represented by an NS record.
- **A (Address)** Provides a name-to-address mapping that supplies an IPv4 address for a specific DNS name. This record type performs the primary function of the DNS, converting names to addresses.
- **AAAA (Address)** Provides a name-to-address mapping that supplies an IPv6 address for a specific DNS name. This record type performs the primary function of the DNS, converting names to addresses.

- **PTR (Pointer)** Provides an address-to-name mapping that supplies a DNS name for a specific address in the in-addr.arpa domain. This is the functional opposite of an A record, used for reverse lookups only.
- **CNAME (Canonical Name)** Creates an alias that points to the canonical name (that is, the "real" name) of a host identified by an A record. Administrators use CNAME records to provide alternative names by which systems can be identified.
- **MX (Mail Exchanger)** Identifies a system that will direct e-mail traffic sent to an address in the domain to the individual recipient, a mail gateway, or another mail server.

EXAM TIP

Exam 70-410 covers only the process of creating A and PTR resource records. The procedures for creating other resource record types are covered on Exam 70-411, "Administering Windows Server 2012," in Objective 3.2, "Configure DNS records."

To create a new Address resource record, use the following procedure.

1. Log on to Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.
2. Click Tools > DNS to open the DNS Manager console.
3. Expand the server node and select the Forward Lookup Zones folder.
4. Right-click the zone in which you want to create the record and, from the shortcut menu, select New Host (A or AAAA). The New Host dialog box appears, as shown in Figure 4-24.

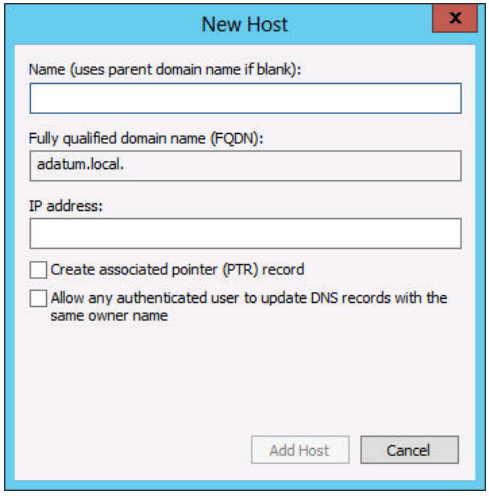

FIGURE 4-24 The New Host dialog box.

5. In the Name text box, type the host name for the new record. The FQDN for the record appears.
6. In the IP Address text box, type the IPv4 or IPv6 address associated with the host name.
7. Select the following check boxes, if necessary:
   - **Create Associated Pointer (PTR) Record** Creates a reverse name lookup record for the host in the in-addr.arpa domain
   - **Allow Any Authenticated User To Update DNS Records With The Same Owner Name** Enables users to modify their own resource records
8. Click Add Host. The new resource record is created in the zone you selected.
9. Close the DNS Manager console.

To create a PTR record for a new host, you can select the Create Associated Pointer (PTR) Record check box in the New Host dialog box, but that will only be effective if a reverse lookup zone already exists on the server. To create the zone, you follow the same procedure described earlier, this time selecting the Reverse Lookup Zones folder.

When you elect to create an IPv4 reverse lookup zone, a Reverse Lookup Zone Name page appears, like the one shown in Figure 4-25, in which you supply the Network ID that the wizard will use to create the zone.

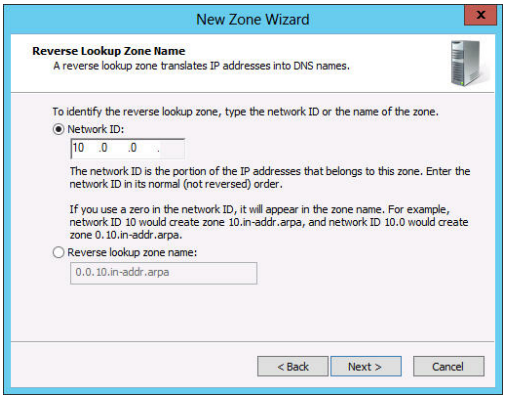

FIGURE 4-25 A Reverse Lookup Zone Name page in the New Zone Wizard.

Once the zone is created, you can either create PTR records along with A or AAAA records, or you create a new PTR record, using the New Resource Record dialog box.

Configuring DNS server settings

Once you have installed a DNS server and created zones and resource records on it, there are many settings you can alter to modify its behavior. The following sections describe some of these settings.

CONFIGURING ACTIVE DIRECTORY DNS REPLICATION

To modify the replication scope for an Active Directory–integrated zone, open the zone's Properties sheet in the DNS Manager console, and on the General tab, click Change for Replication: All DNS Servers In The Active Directory Domain to display the Change Zone Replication Scope dialog box, shown in Figure 4-26. The options are the same as those in the New Zone Wizard.

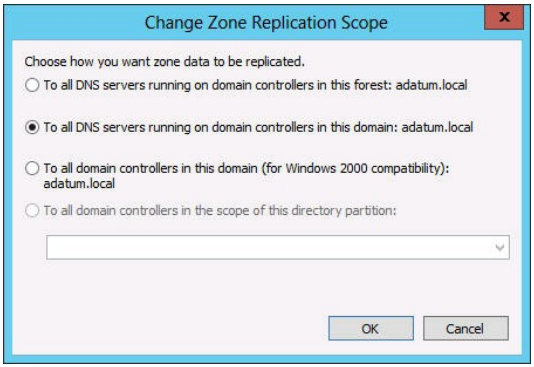

FIGURE 4-26 The Change Zone Replication Scope dialog box.

CONFIGURING ROOT HINTS

Every DNS server must be able to contact the root name servers to initiate name resolution processes. Most server implementations, including Microsoft DNS Server, are preconfigured with the names and addresses of multiple root name servers. These are called root hints.

The 13 root name server names are located in a domain called root-servers.net, and are named using letters of the alphabet. The servers are scattered around the world on different subnets to provide fault tolerance.

To modify the root hints on a Windows Server 2012 DNS server, right-click the server node, open the Properties sheet, and click the Root Hints tab, as shown in Figure 4-27. On this tab, you can add, edit, or remove root hints from the list provided.

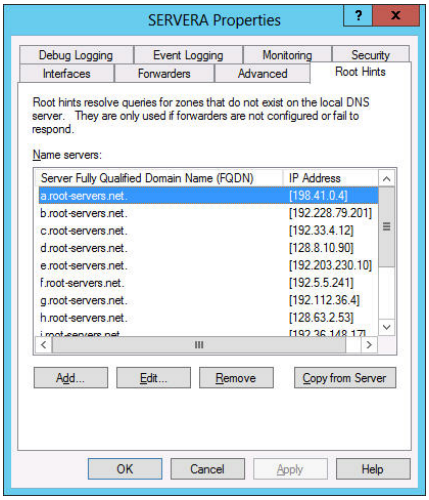

FIGURE 4-27 The Root Hints tab on a DNS server's Properties sheet.

Objective summary

- DHCP is a service that automatically configures the IP address and other TCP/IP settings on network computers by assigning addresses from a pool (called a scope) and reclaiming them when they are no longer in use.
- TCP/IP networks today use DNS servers to convert host names into IP addresses. This conversion process is referred to as name resolution.
- DNS consists of three elements: the DNS namespace, name servers, and resolvers.
- The hierarchical nature of the DNS namespace is designed to make it possible for any DNS server on the Internet to locate the authoritative source for any domain name, using a minimum number of queries.
- In a recursive query, the DNS server receiving the name resolution request takes full responsibility for resolving the name. In an iterative query, the server that receives the name resolution request immediately responds with the best information it possesses at the time.
- For Internet name resolution purposes, the only functions required of the DNS server are the ability to process incoming queries from resolvers and send its own queries to other DNS servers on the Internet.

Objective review

Answer the following questions to test your knowledge of the information in this objective. You can find the answers to these questions and explanations of why each answer choice is correct or incorrect in the "Answers" section at the end of this chapter.

1. Which of the following resource record types contains the information a DNS server needs to perform reverse name lookups?
   A. A
   B. CNAME
   C. SOA
   D. PTR
2. Which of the following would be the correct FQDN for a resource record in a reverse lookup zone if the computer's IP address is 10.75.143.88?
   A. 88.143.75.10.in-addr.arpa
   B. 10.75.143.88.in-addr.arpa
   C. in-addr.arpa.88.143.75.10
   D. arpa.in-addr.10.75.143.88
3. Which of the following is not one of the elements of DNS?
   A. Resolvers
   B. Relay agents
   C. Name servers
   D. Namespace
4. In which of the following DNS transactions does the querying system generate a recursive query?
   A. A DNS client sends the server name www.adatum.com from a URL to its designated DNS server for resolution.
   B. A client's DNS server sends a request to a root domain server to find the authoritative server for the com top-level domain.
   C. A client's DNS server sends a request to the com top-level domain server to find the authoritative server for the adatum.com domain.
   D. A client's DNS server sends a request to the adatum.com domain server to find the IP address associated with the server name www.
5. Which of the following contains the controls used to modify DNS name caching?
   A. The Forwarders tab of a server's Properties sheet
   B. The Start of Authority's (SOA) tab of a zone's Properties sheet
   C. The Root Hints tab of a server's Properties sheet
   D. The New Zone Wizard

Thought experiment

In the following thought experiment, apply what you've learned about this objective to predict what steps you need to take. You can find answers to these questions in the "Answers" section at the end of this chapter.

Alice is an enterprise administrator for Wingtip Toys, which has recently expanded its Customer Service division by adding 100 workstations. All of the workstations on the company network are configured to use a server on the perimeter network as their primary DNS server and a server on their ISP's network as a secondary server. As a result of the expansion, Internet performance has slowed down noticeably, and a Network Monitor trace indicates that there is a disproportionate amount of DNS traffic on the link between the perimeter network and the ISP's network.

With this in mind, answer the following question:

What are two ways that Alice can reduce the amount of DNS traffic passing over the Internet connection?

Answers

Objective 4.1: Review

1. **Correct Answer:** B
   A. **Incorrect:** Subnetting is a technique for creating administrative divisions on a network; it does not transmit IPv6 traffic over an IPv4 network.
   B. **Correct:** Tunneling is a method for encapsulating IPv6 traffic within IPv4 datagrams.
   C. **Incorrect:** Supernetting is a method for combining consecutive subnets into a single entity.
   D. **Incorrect:** Contracting is a method for shortening IPv6 addresses.
2. **Correct Answer:** C
   A. **Incorrect:** Link-local unicast addresses are self-assigned by IPv6 systems. They are therefore the equivalent of APIPA addresses on IPv4.
   B. **Incorrect:** A global unicast address is the equivalent of a registered IPv4 address, routable worldwide and unique on the Internet.
   C. **Correct:** Unique local unicast addresses are the IPv6 equivalent of the 10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16 private network addresses in IPv4.
   D. **Incorrect:** The function of an anycast address is to identify the routers within a given address scope and send traffic to the nearest router.

3. **Correct Answer:** A
   A. **Correct:** Teredo is a mechanism that enables devices behind non-IPv6 NAT routers to function as tunnel endpoints.
   B. **Incorrect:** 6to4 incorporates the IPv4 connections in a network into the IPv6 infrastructure by defining a method for expressing IPv4 addresses in IPv6 format and encapsulating IPv6 traffic into IPv4 packets.
   C. **Incorrect:** Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) is an automatic tunneling protocol used by the Windows workstation operating systems that emulates an IPv6 link using an IPv4 network.
   D. **Incorrect:** APIPA is an automatic IPv4 address self-assignment process. It has nothing to do with tunneling.
4. **Correct Answer:** A
   A. **Correct:** For an address to be visible from the Internet, it must be registered with the IANA.
   B. **Incorrect:** Binary is a system of numbering that can be used to express any IP address.
   C. **Incorrect:** All address classes can be visible or invisible to the Internet.
   D. **Incorrect:** Subnetted addresses can be visible or invisible to the Internet.
5. **Correct Answer:** C
   A. **Incorrect:** In binary form, the mask 255.224.0.0 is 11111111.11100000.00000000.00000000, which contains only 11 network identifier bits.
   B. **Incorrect:** In binary form, the mask 255.240.0.0 is 11111111.11110000.00000000.00000000, which contains only 12 network identifier bits.
   C. **Correct:** In binary form, the mask 255.255.224.0 is 11111111.11111111.11100000.00000000, which contains 19 network identifier bits.
   D. **Incorrect:** In binary form, the mask 255.255.240.0 is 11111111.11111111.11110000.00000000, which contains 20 network identifier bits.
   E. **Incorrect:** In binary form, the mask 255.255.255.240 is 11111111.11111111.11111111.11110000, which contains 28 network identifier bits.

Objective 4.1: Thought experiment

Arthur can subnet the address he has been given by using three host bits to give him eight subnets with up to 16 hosts on each one. The computers will use a subnet mask of 255.255.255.240 and IP address ranges as follows:

`192.16.8.1 – 192.16.8.14`
`192.16.8.17 – 192.16.8.30`

```
192.16.8.33 - 192.16.8.46
192.16.8.49 - 192.16.8.62
192.16.8.65 - 192.16.8.78
192.16.8.81 - 192.16.8.94
192.16.8.97 - 192.16.8.110
192.16.8.113 - 192.16.8.126
192.16.8.129 - 192.16.8.142
192.16.8.145 - 192.16.8.158
192.16.8.161 - 192.16.8.174
192.16.8.177 - 192.16.8.190
192.16.8.193 - 192.16.8.206
192.16.8.209 - 192.16.8.222
192.16.8.225 - 192.16.8.238
192.16.8.241 - 192.16.8.254
```

Objective 4.2: Review

1. **Correct Answer:** D
   A. **Incorrect:** A forwarder is a DNS server that accepts recursive queries from other servers.
   B. **Incorrect:** A resolver is a DNS client component.
   C. **Incorrect:** A scope is a range of IP addresses that a DHCP server is configured to allocate.
   D. **Correct:** A relay agent is a software module that receives DHCP broadcast messages and forwards them to a DHCP server on another subnet.
2. **Correct Answer:** D
   A. **Incorrect:** The DHCP address assignment process begins when the DHCP client generates DHCPDISCOVER messages and broadcasts them on the local network.
   B. **Incorrect:** The client eventually stops broadcasting and signals its acceptance of one of the offered addresses by generating a DHCPREQUEST message.
   C. **Incorrect:** When the server offering the accepted IP address receives the DHCPREQUEST message, it then transmits a DHCPACK message to the client, acknowledging the completion of the process.
   D. **Correct:** The DHCPINFORM message type is not used during an IP address assignment.
3. **Correct Answer:** C
   A. **Incorrect:** Dynamic allocation is when the DHCP server assigns an IP address to a client computer from a scope, for a specified length of time.
   B. **Incorrect:** Automatic allocation is when the DHCP server permanently assigns an IP address to a client computer from a scope.
   C. **Correct:** Manual allocation is when the DHCP server permanently assigns a specific

IP address to a specific computer on the network. In the Windows Server 2012 DHCP server, manually allocated addresses are called reservations.

D. **Incorrect:** Hybrid is a DHCP infrastructure type, not a type of address allocation.

4. **Correct Answers:** B and D

A. **Incorrect:** Windows 8 cannot function as a LAN router, and it therefore does not have the ability to function as a DHCP relay agent.

B. **Correct:** Most IP routers have DHCP relay agent capabilities built into them. If the routers connecting your subnets are so equipped, you can use them as relay agents, eliminating the need for a DHCP server on each subnet.

C. **Incorrect:** Switches are data-link layer devices and are designed to communicate with devices on the same subnet. A DHCP relay agent requires access to two subnets.

D. **Correct:** If your routers cannot function as DHCP relay agents, you can use the relay agent capability built into the Windows server operating systems. In Windows Server 2012, the DHCP relay agent capability is built into the Remote Access role.

5. **Correct Answer:** D

A. **Incorrect:** In most cases, all of the computers on a network will use the same DNS server, so it is more convenient to deploy its address just once, using a server option, rather than having to deploy it as a scope option on every scope.

B. **Incorrect:** The subnet mask is automatically included with every address lease, and therefore does not have to be deployed as a scope option or a server option.

C. **Incorrect:** The lease duration option is automatically included with every address lease, and therefore does not have to be deployed as a scope option or a server option.

D. **Correct:** The default gateway must be a router on the same subnet as the IP addresses the DHCP server is allocating. Therefore, the gateway address is different for every scope, and must be deployed as a scope option.

Objective 4.2: Thought experiment

Roger can reduce the duration of the IP address leases in his scopes, so that abandoned addresses will be available to clients more quickly than before.

Objective 4.3: Review

1. **Correct Answer:** D

A. **Incorrect:** A resource record contains information for forward name lookups, not reverse name lookups.

B. **Incorrect:** CNAME resource records contain alias information for A records. They are not used for reverse name lookups.

C. **Incorrect:** SOA records specify that a server is the authoritative source for a zone. They are not used for reverse name lookups.

D. **Correct:** PTR records contain the information needed for the server to perform reverse name lookups.

2. **Correct Answer:** A

A. **Correct:** To resolve the IP address 10.75.143.88 into a name, a DNS server would locate a domain called 143.75.10.in-addr.arpa in the usual manner and read the contents of a resource record named 88 in that domain.

B. **Incorrect:** Striped volumes do not contain redundant data, and therefore do not provide fault tolerance.

C. **Incorrect:** The top-level domain used for reverse lookups is arpa. Therefore, arpa must be the last and most significant name in a reverse lookup FQDN.

D. **Incorrect:** The top-level domain used for reverse lookups is arpa. Therefore, arpa must be the last and most significant name in a reverse lookup FQDN.

3. **Correct Answer:** B

A. **Incorrect:** Resolvers are client programs that generate DNS queries and send them to a DNS server for fulfillment.

B. **Correct:** Relay agents are router devices that enable DHCP clients to communicate with servers on other networks.

C. **Incorrect:** Name servers are applications running on server computers that maintain information about the domain tree structure.

D. **Incorrect:** DNS consists of a tree-structured namespace in which each branch of the tree identifies a domain.

4. **Correct Answer:** A

A. **Correct:** When a client sends a name resolution query to its DNS server, it uses a recursive request so that the server will take on the responsibility for resolving the name.

B. **Incorrect:** A DNS server seeking the server for a top-level domain uses iterative, not recursive, queries.

C. **Incorrect:** A DNS server seeking the server for a second-level domain uses iterative, not recursive, queries.

D. **Incorrect:** A DNS server requesting a server name resolution from an authoritative server uses iterative, not recursive, queries.

5. **Correct Answer:** B

A. **Incorrect:** The Forwarders tab is where you specify the addresses of servers that will have your server's recursive queries.

B. **Correct:** The Start of Authority (SOA) tab of a zone's Properties sheet contains the Minimum (Default) TTL setting that controls DNS name caching for the zone.

C. **Incorrect:** The Root Hints tab is where you specify the addresses of the root name servers on the Internet.

D. **Incorrect:** The New Zone Wizard does not enable you to modify name caching settings.

Objective 4.3: Thought experiment

1. Alice can configure the DNS server on the perimeter network to use the ISP's DNS server as a forwarder.
2. Alice can configure the workstations to use the ISP's DNS server as their primary DNS server.

CHAPTER 5

Install and administer Active Directory

A directory service is a repository of information about the resources—hardware, software, and human—that are connected to a network. Users, computers, and applications throughout the network can access the repository for a variety of purposes, including user authentication, storage of configuration data, and even simple white pages–style information lookups. Active Directory Domain Services (AD DS) is the directory service that Microsoft first introduced in Windows 2000 Server, and they have upgraded it in each successive server operating system release, including Windows Server 2012.

This chapter covers some of the fundamental tasks that administrators perform to install and manage AD DS.

Objectives in this chapter:

- Objective 5.1: Install domain controllers
- Objective 5.2: Create and manage Active Directory users and computers
- Objective 5.3: Create and Manage Active Directory groups and organizational units (OUs)

Objective 5.1: Install domain controllers

AD DS is a directory service that enables administrators to create organizational divisions called domains. A domain is a logical container of network components, hosted by at least one server designated as a domain controller. The domain controllers for each domain replicate their data among themselves, for fault tolerance and load balancing purposes.

Deploying Active Directory Domain Services

Once you have created an Active Directory design, it is time to think about the actual deployment process. As with most major network technologies, it is a good idea to install AD DS on a test network first, before you put it into actual production.

There are a great many variables that can affect the performance of an Active Directory installation, including the hardware you select for your domain controllers, the capabilities of your network, and the types of wide area network (WAN) links connecting your remote sites. In

many cases, an Active Directory design that looks good on paper will not function well in your environment, and you might want to modify the design before you proceed with the live deployment.

Active Directory is one of the more difficult technologies to test, because an isolated lab environment usually cannot emulate many of the factors that can affect the performance of a directory service. Most test labs cannot duplicate the network traffic patterns of the production environment, and few have the WAN links necessary to simulate an actual multisite network. Wherever possible, you should try to test your design under real-life conditions, using your network's actual local area network (LAN) and WAN technologies, but limiting the domain controllers and AD DS clients to laboratory computers.

To create a new forest or a new domain, or to add a domain controller to an existing domain, you must install the Active Directory Domain Services role on a Windows Server 2012 computer, and then run the Active Directory Domain Services Configuration Wizard.

To use a Windows Server 2012 computer as a domain controller, you must configure it to use static IP addresses, not addresses supplied by a Dynamic Host Configuration Protocol (DHCP) server. In addition, if you are creating a domain in an existing forest, or adding a domain controller to an existing domain, you must configure the computer to use the Domain Name System (DNS) server that hosts the existing forest or domain, at least during the Active Directory installation.

Installing the Active Directory Domain Services role

Although it does not actually convert the computer into a domain controller, installing the Active Directory Domain Services role prepares the computer for the conversion process.

To install the role, use the following procedure.

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.
2. From the Manage menu, select Add Roles And Features. The Add Roles and Features Wizard starts, displaying the Before You Begin page.
3. Click Next. The Select Installation Type page appears.
4. Leave the Role-Based Or Feature-Based Installation option selected and click Next to open the Select Destination Server page.
5. Select the server that you want to promote to a domain controller, and click Next. The Select Server Roles page opens.
6. Select the Active Directory Domain Service role. The Add Features That Are Required For Active Directory Domain Services dialog box opens.
7. Click Add Features to accept the dependencies, and then click Next. The Select Features page opens.
8. Click Next. The Active Directory Domain Services page opens, displaying information about the role.

9. Click Next. A Confirm Installation Selections page appears.
10. Select from the following optional functions, if desired:
    - **Restart The Destination Server Automatically If Desired** Causes the server to restart automatically when the installation is completed, if the selected roles and features require it
    - **Export Configuration Settings** Creates an XML script documenting the procedures performed by the wizard, which you can use to install the same configuration on another server using Windows PowerShell
    - **Specify An Alternate Source Path** Specifies the location of an image file containing the software needed to install the selected roles and features
11. Click Install, which displays the Installation Progress page. Once the role has been installed, a Promote This Server To A Domain Controller link appears.
12. Leave the wizard open.

NOTE **Dcpromo.exe**

The Dcpromo.exe program from previous version of Windows Server has been deprecated in favor of the Server Manager domain controller installation process documented in the following sections. However, it is still possible to automate AD DS installations by running Dcpromo.exe with an answer file.

Once you have installed the role, you can proceed to run the Active Directory Domain Services Installation Wizard. The wizard procedure varies, depending on what the function of the new domain controller will be. The following sections describe the procedures for the most common types of domain controller installations.

Creating a new forest

When beginning a new AD DS installation, the first step is to create a new forest, which you do by creating the first domain in the forest, the forest root domain.

To create a new forest, use the following procedure.

1. Log on to the server running Windows Server 2012 using an account with administrative privileges and install the Active Directory Domain Services role, as described earlier in this lesson.
2. On the Installation Progress page that appears at the end of the Active Directory Domain Services role installation procedure, click the Promote This Server To A Domain Controller hyperlink. The Active Directory Domain Services Configuration Wizard opens, displaying the Deployment Configuration page.
3. Select the Add A New Forest option, as shown in Figure 5-1, and, in the Root Domain Name text box, type the name of the domain you want to create.

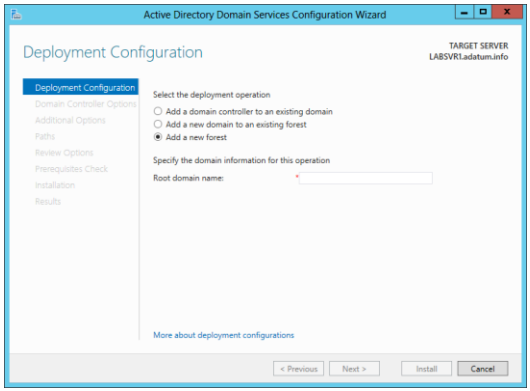

FIGURE 5-1 The Deployment Configuration page of the Active Directory Domain Services Configuration Wizard .

4. Click Next. The Domain Controller Options page open, as shown in Figure 5-2.

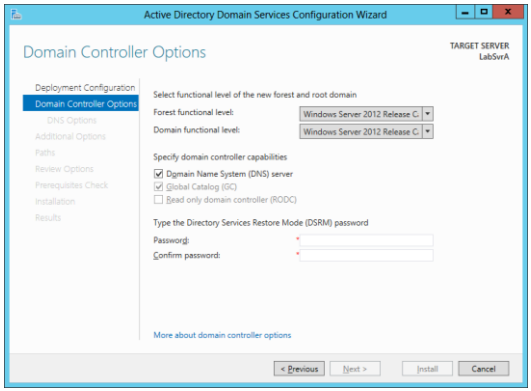

FIGURE 5-2 The Domain Controller Options page of the Active Directory Domain Services Configuration Wizard.

5. If you plan to add domain controllers running earlier versions of Windows Server to this forest, select the earliest Windows version you plan to install from the Forest Functional Level drop-down list.
6. If you plan to add domain controllers running earlier versions of Windows Server to this domain, select the earliest Windows version you plan to install from the Domain Functional Level drop-down list.
7. If you do not already have a DNS server on your network, leave the Domain Name System (DNS) Server check box selected. If you have a DNS server on the network, and the domain controller is configured to use that server for DNS services, then clear the check box.

Note **DOMAIN CONTROLLER OPTIONS**

The Global Catalog (GC) and Read Only Domain Controller (RODC) options are unavailable because the first domain controller in a new forest must be a Global Catalog server, and it cannot be a read-only domain controller.

8. In the Password and Confirm Password text boxes, type the password you want to use for Directory Services Restore Mode (DSRM) and click Next. The DNS Options page appears, with a warning that a delegation for the DNS server cannot be created, because the DNS Server service is not installed yet.
9. Click Next to open the Additional Options page, which displays the NetBIOS equivalent of the domain name you specified.
10. Modify the name, if desired, and click Next to open the Paths page.
11. Modify the default locations for the AD DS files, if desired, and click Next. The Review Options page opens.
12. Click Next to open the Prerequisites Check page, as shown in Figure 5-3.

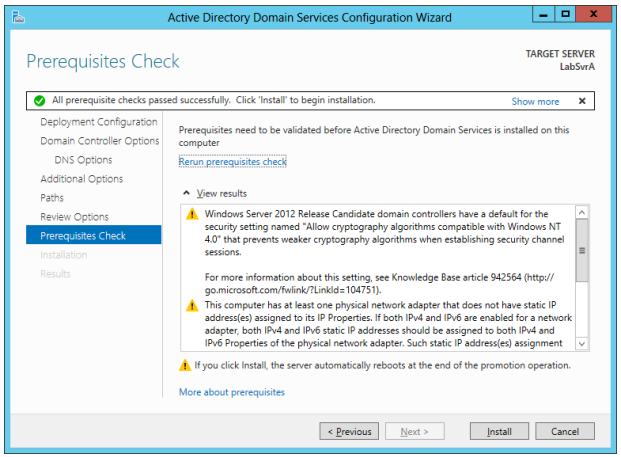

FIGURE 5-3 The Prerequisites Check page of the Active Directory Domain Services Configuration Wizard.

13. The wizard performs a number of environment tests, to determine if the system can function as a domain controller. The results can appear as cautions, which enable the procedure to continue, or warnings, which require you to perform certain actions before the server can be promoted. Once the system has passed all of the prerequisite checks, click Install. The wizard creates the new forest and configures the server to function as a domain controller.
14. Restart the computer.

With the forest root domain in place, you can now proceed to create additional domain controllers in that domain, or add new domains to the forest.

Adding a domain controller to an existing domain

Every Active Directory domain should have a minimum of two domain controllers.

To add a domain controller to an existing Windows Server 2012 domain, use the following procedure.

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges and install the Active Directory Domain Services role, as described earlier in this objective.
2. On the Installation Progress page that appears at the end of the Active Directory Domain Services role installation procedure, click the Promote This Server To A Domain Controller hyperlink. The Active Directory Domain Services Configuration Wizard starts, displaying the Deployment Configuration page.
3. Select the Add A Domain Controller To An Existing Domain option and click Select.
4. If you are not logged on to an existing domain in the forest, a Credentials For Deployment Operation dialog box appears, in which you must supply administrative credentials for the domain to proceed. After you are authenticated, the Select A Domain From The Forest dialog box opens.
5. Select the domain to which you want to add a domain controller and click OK. The selected domain name appears in the Domain field.
6. Click Next. The Domain Controller Options page, shown in Figure 5-4, opens.

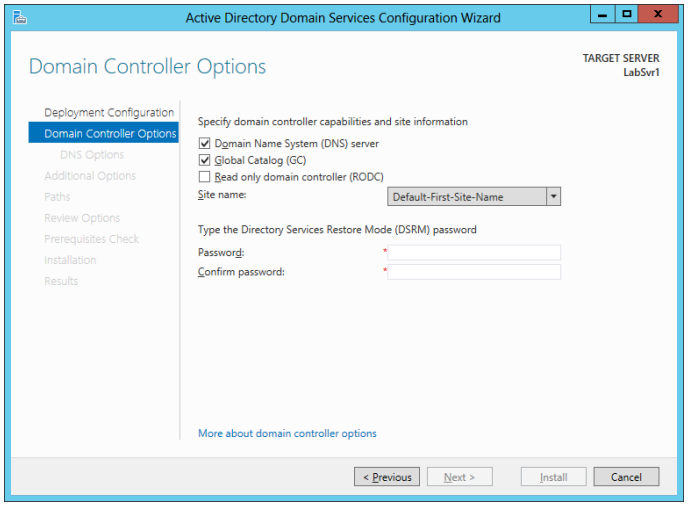

FIGURE 5-4 The Domain Controller Options page of the Active Directory Domain Services Configuration Wizard.

7. If you want to install the DNS Server service on the computer, leave the Domain Name System (DNS) Server check box selected. Otherwise, the domain will be hosted on the DNS server the computer is configured to use.

8. Leave the Global Catalog (GC) check box selected if you want the computer to function as a global catalog server. This is essential if you will be deploying the new domain controller at a site that does not already have a GC server.
9. Select the Read Only Domain Controller (RODC) check box to create a domain controller that administrators cannot use to modify AD DS objects.
10. In the Site Name drop-down list, select the site where the domain controller will be located.
11. In the Password and Confirm Password text boxes, type the password you want to use for Directory Services Restore Mode (DSRM) and click Next to move to the Additional Options page, shown in Figure 5-5.

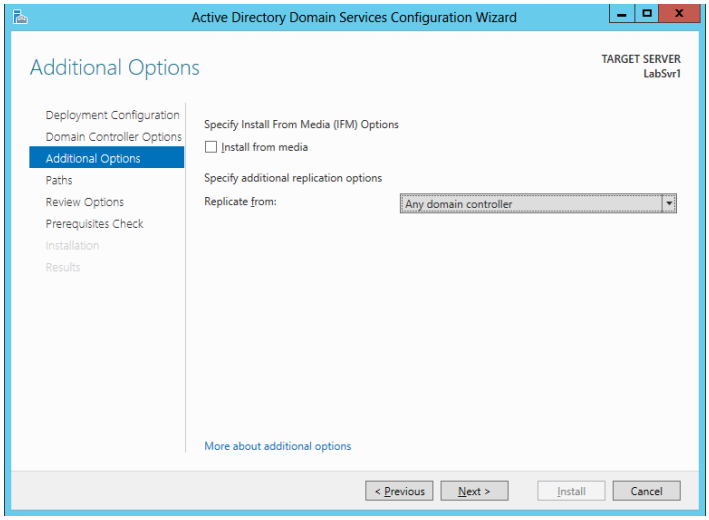

FIGURE 5-5 The Additional Options page of the Active Directory Domain Services Configuration Wizard.

12. To use the Install From Media option, select the Install From Media check box.
13. In the Replicate From drop-down list, select the existing domain controller that the server should use as a data source. Then click Next to open the Paths page.
14. Modify the default locations for the AD DS files, if desired, and click Next. The Review Options page opens.
15. Click Next to move to the Prerequisites Check page.
16. Once the system has passed all of the prerequisite checks, click Install. The wizard configures the server to function as a domain controller.
17. Restart the computer.

The domain controller is now configured to service the existing domain. If the new domain controller is located in the same site as another, then AD DS replication between the two will begin automatically.

Creating a new child domain in a forest

Once you have a forest with at least one domain, you can add a child domain beneath any existing domain. The process of creating a new child domain is roughly similar to that of creating a new forest, except that the Deployment Configuration page of the Active Directory Domain Services Configuration Wizard requires you to specify the parent domain beneath which you want to create a child, as shown in Figure 5-6.

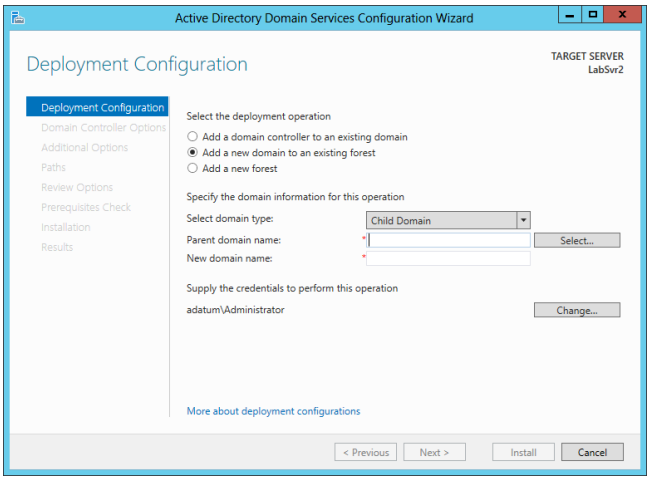

FIGURE 5-6 The Deployment Configuration page of the Active Directory Domain Services Configuration Wizard.

> ***NOTE*** **Tree domains**
>
> **The wizard also supplies the option to create a tree domain, which is a new domain that is not subordinate to an existing domain in the forest.**

Installing AD DS on Server Core

In Windows Server 2012, it is now possible to install Active Directory Domain Services on a computer running the Server Core installation option, and promote the system to a domain controller, all using Windows PowerShell.

In Windows Server 2008 and Windows Server 2008 R2, the accepted method for installing AD DS on a computer using the Server Core installation option is to create an answer file and load it from the command prompt using the Dcpromo.exe program with the /unattend parameter.

In Windows Server 2012, running Dcpromo.exe with no parameters no longer launches the Active Directory Domain Services Configuration Wizard, but administrators who have already invested considerable time in developing answer files for unattended domain controller installations can continue to execute them from the command prompt, although doing so also

produces this warning: “The dcpromo unattended operation is replaced by the ADDSDeployment module for Windows PowerShell.”

For AD DS installations on Server Core, Windows PowerShell is now the preferred method. As with the wizard-based installation, the PowerShell procedure occurs in two phases; first you must install the Active Directory Domain Services role; then, you must promote the server to a domain controller.

Installing the Active Directory Domain Services role using PowerShell is no different from installing any other role. In an elevated PowerShell session, use the following command:

```
Install-WindowsFeature –name AD-Domain-Services
-IncludeManagementTools
```

As with other PowerShell role installations, the Install-WindowsFeature cmdlet does not install the management tools for the role, such as Active Directory Administrative Center and Active Directory Users and Computers, unless you include the –IncludeManagementTools parameter in the command.

Once you have installed the role, promoting the server to a domain controller is somewhat more complicated. The ADDSDeployment PowerShell module includes separate cmdlets for the three deployment configurations covered in the previous sections:

- Install-AddsForest
- Install-AddsDomainController
- Install-AddsDomain

Each of these cmdlets has a great many possible parameters to support the many configuration options you find in the Active Directory Domain Services Configuration Wizard. In its simplest form, the following command would install a domain controller for a new forest called adatum.com:

```
Install-AddsForest  -DomainName "adatum.com"
```

The defaults for all of the cmdlet’s other parameters are the same as those in the Active Directory Domain Services Configuration Wizard. Running the cmdlet with no parameters steps through the options, prompting you for values. You can also display basic syntax information using the Get-Help command, as shown in Figure 5-7.

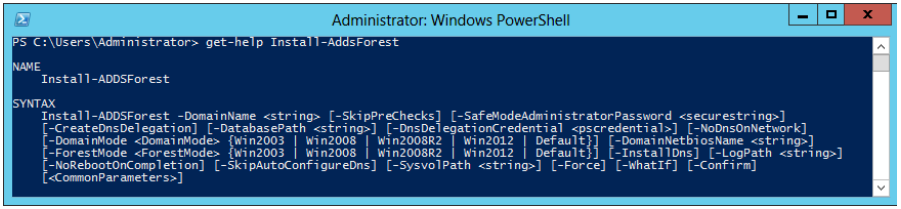

FIGURE 5-7 Syntax for the Install-AddsForest cmdlet in Windows PowerShell.

Another way to perform a complex installation using PowerShell is to use a computer running Windows Server 2012 with the full GUI option to generate a script. Begin by running

the Active Directory Domain Services Configuration Wizard, configuring all of the options with your desired settings. When you reach the Review Option page, click View Script to display the PowerShell code for the appropriate cmdlet, as shown in Figure 5-8.

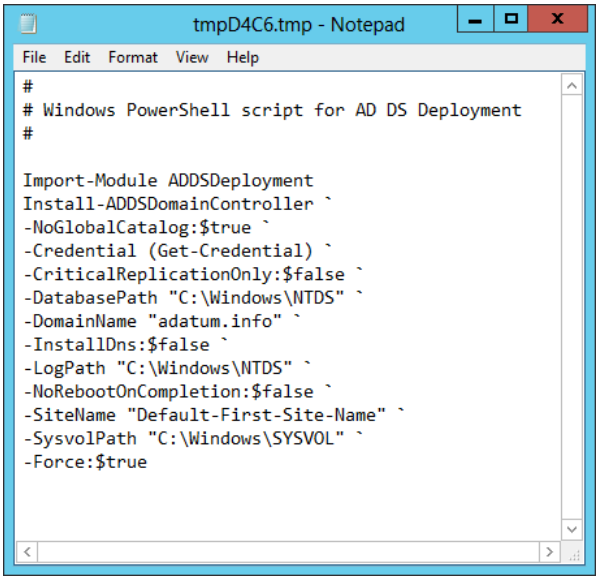

FIGURE 5-8 An installation script generated by the Active Directory Domain Services Configuration Wizard.

This feature works as it does because Server Manager is actually based on PowerShell, so the script contains the cmdlets and parameters that are actually running when the wizard performs an installation. You can also use this scripting capability with the Install-AddsDomainController cmdlet to deploy multiple domain controllers for the same domain.

Using Install from Media (IFM)

Earlier in this objective, in the procedure for installing a replica domain controller, the Additional Options page of the Active Directory Domain Services Configuration Wizard included an Install From Media check box. This is an option that enables administrators to streamline the process of deploying replica domain controllers to remote sites.

Normally, installing a domain controller to an existing domain creates the AD DS database structure, but there is no data in it until the server is able to receive replication traffic from the other domain controllers. When the domain controllers for a particular domain are well-connected, such as by LAN, replication occurs almost immediately after the new domain controller is installed, and is entirely automatic.

When installing a domain controller at a remote location, however, the connection to the other domain controllers is most likely a WAN link, which is typically slower and more expensive than a LAN connection. In this case, the initial replication with the other domain controllers can be much more of a problem. The slow speed of the WAN link might cause the

replication to take a long time, and it might also flood the connection, delaying regular traffic. If the domain controllers are located in different AD DS sites, no replication will occur at all until an administrator creates and configures the required site links.

NOTE **Replication**

The first replication that occurs after the installation of a new domain controller is the only one that requires the servers to exchange a complete copy of the AD DS database. In subsequent replications, the domain controllers only exchange information about the objects and attributes that have changed since the last replication.

Using a command-line tool called Ndtsutil.exe, administrators can avoid these problems by creating domain controller installation media that includes a copy of the AD DS database. By using this media when installing a remote domain controller, the data is installed along with the data base structure, and no initial replication is necessary.

To create IFM media, you must run the Ntdsutil.exe program on a domain controller running the same version of Windows that you intend to deploy. The program is interactive, requiring you to enter a sequence of commands like the following:

- **Ntdsutil** Launches the program
- **Activate instance ntds** Focuses the program on the installed AD DS instance
- **Ifm** Switches the program into IFM mode
- **Create Full|RODC <*path name*>** Creates media for either a full read/write domain controller or a read-only domain controller and saves it to the folder specified by the path name variable

NOTE **Ntdsutil.exe parameters**

The Ntdsutil.exe create command also supports parameters that include the contents of the SYSVOL volume with the AD DS data. The Windows Server 2012 version of the program adds a nodefrag parameter that speeds up the media creation process by skipping the defragmentation.

When you execute these commands, the Ntdsutil.exe program creates a snapshot of the AD DS database, mounts it as a volume to defragment it, and then saves it to the specified folder, along with a copy of the Windows Registry, as shown in Figure 5-9.

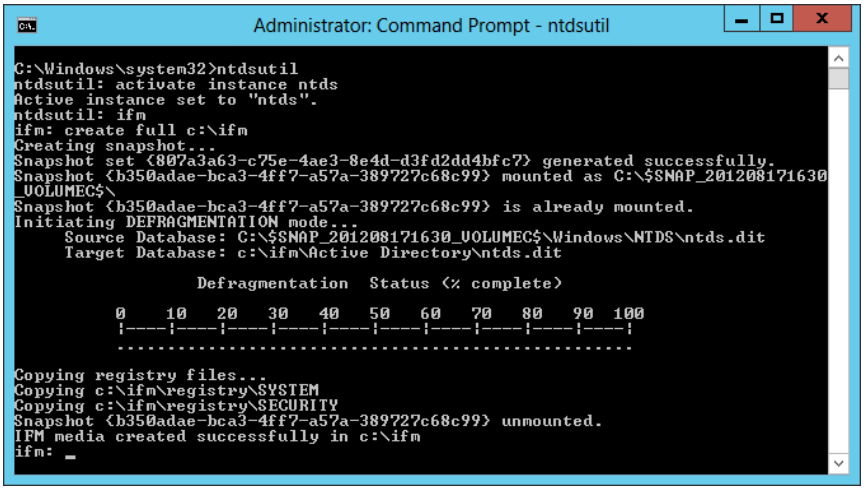

FIGURE 5-9 An Ntdsutil.exe command sequence.

Once you have created the IFM media, you can transport it to the servers you intend to deploy as domain controllers by any convenient means. To use the media, you run the Active Directory Domain Services Configuration Wizard in the usual way, select the Install From Media check box, and specify the path to the location of the folder.

Upgrading Active Directory Domain Services

Introducing Windows Server 2012 onto an existing AD DS installation is easier than it has ever been in previous versions of the operating system.

There are two ways to upgrade an AD DS infrastructure. You can upgrade the existing downlevel domain controllers to Windows Server 2012, or you can add a new Windows Server 2012 domain controller to your existing installation.

The upgrade paths to Windows Server 2012 are few. You can upgrade a Windows Server 2008 or Windows Server 2008 R2 domain controller to Windows Server 2012, but no earlier versions are upgradable.

In the past, if you wanted to add a new domain controller to an existing AD DS installation based on previous Windows versions, you had to run a program called Adprep.exe to upgrade the domains and forest. Depending on the complexity of the installation, this could involve logging on to various domain controllers using different credentials, locating different versions of Adprep.exe, and running the program several times using the /domainprep parameter for each domain and the /forestprep parameter for the forest.

In Windows Server 2012, the Adprep.exe functionality has been fully incorporated into Server Manager, in the Active Directory Domain Services Configuration Wizard. When you install a new Windows Server 2012 domain controller, you only have to supply appropriate credentials, and the wizard takes care of the rest.

NOTE **Group memberships**

To install the first Windows Server 2012 domain controller into a downlevel AD DS installation, you must supply credentials for a user that is a member of the Enterprise Admins and Schema Admins groups, and a member of the Domain Admins group in the domain that hosts the schema master.

Adprep.exe is still included with the operating system, and supports the old preparation method, if you prefer it, but there is no compelling reason to do so.

Removing a domain controller

With the loss of Dcpromo.exe, the process of demoting a domain controller has changed, and is not immediately intuitive.

To remove a domain controller from an AD DS installation, you must begin by running the Remove Roles and Features Wizard, as shown in the following procedure.

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges. The Server Manager console opens.
2. Launch the Remove Roles and Features Wizard and remove the Active Directory Domain Services role and its accompanying features. A Validation Results dialog box appears, as shown in Figure 5-10.

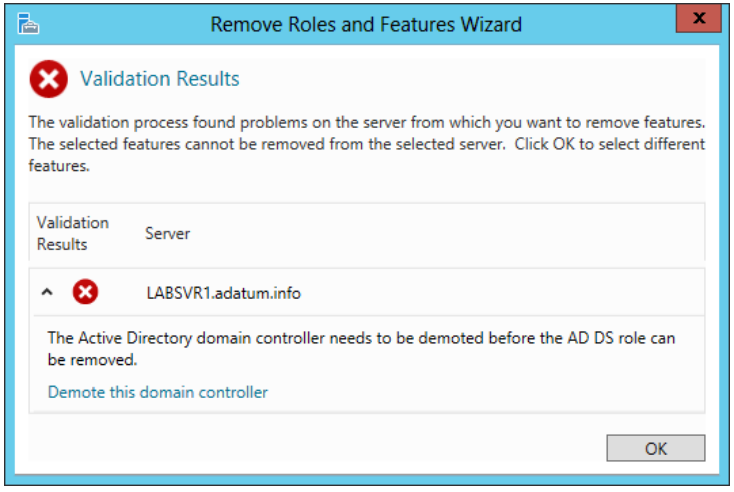

FIGURE 5-10 The Validation Results dialog box of the Remove Roles and Features Wizard.

3. Click the Demote This Domain Controller hyperlink. The Active Directory Domain Services Configuration Wizard starts, displaying the Credentials page.
4. Select the Force The Removal Of This Domain Controller check box and click Next to open the New Administrator Password page.

5. In the Password and Confirm Password text boxes, type the password you want the server to use for the local Administrator account after the demotion. Then click Next. The Review Options page appears.
6. Click Demote. The wizard demotes the domain controller and restarts the system.
7. Log on using the local Administrator password you specified earlier.
8. Launch the Remove Roles and Features Wizard again and repeat the process of removing the Active Directory Domain Services role and its accompanying features.
9. Close the wizard and restart the server.

NOTE **Using PowerShell**

To demote a domain controller with Windows PowerShell, use the following command:

```
Uninstall-ADDSDomainController –ForceRemoval
–LocalAdminisdtratorPassword <password> –Force
```

Configuring the global catalog

The global catalog is an index of all the AD DS objects in a forest that prevents systems from having to perform searches among multiple domain controllers. The importance of the global catalog varies depending on the size of your network and its site configuration.

For example, if your network consists of a single domain, with domain controllers all located at the same site and well-connected, the global catalog serves little purpose other than universal group searches. You can make all of your domain controllers global catalog servers if you wish. The searches will be load balanced and the replication traffic will likely not overwhelm the network.

However, if your network consists of multiple domains, with domain controllers located at multiple sites connected by WAN links, then the global catalog configuration is critical. If at all possible, you do not want users performing AD DS searches that must reach across slow, expensive WAN links to contact domain controllers at other sites. Placing a global catalog server at each site is recommended in this case. The initial replication might generate a lot of traffic, but the savings in the long run should be significant.

When you promote a server to a domain controller, you have the option of making the domain controller a global catalog server. If you decline to do so, however, you can make any domain controller a global catalog server using the following procedure.

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges. The Server Manager console opens.
2. From the Tools menu, select Active Directory Sites And Services. The Active Directory Sites and Services console opens.
3. Expand the site where the domain controller you want to function as a global catalog server is located. Then expand the Servers folder and select the server you want to configure.

4. Right-click the NTDS Settings node for the server and, from the shortcut menu, select Properties to open the NTDS Settings Properties sheet.
5. Select the Global Catalog check box and click OK.
6. Close the Active Directory Sites and Services console.

Troubleshooting DNS SRV registration failure

DNS is essential to the operating of Active Directory Domain Services. To accommodate directory services such as AD DS, a special DNS resource record was created that enables clients to locate domain controllers and other vital AD DS services.

When you create a new domain controller, one of the most important parts of the process is the registration of the server in the DNS. This automatic registration is the reason why an AD DS network must have access to a DNS server that supports the Dynamic Updates standard defined in Request for Comments (RFC) 2136.

If the DNS registration process should fail, then computers on the network will not be able to locate that domain controller, the consequences of which can be serious. Computers will be unable to use that domain controller to join the domain; existing domain members will be unable to log on; and other domain controllers will be unable to replicate with it.

DNS problems are, in most cases, due to general networking faults or DNS client configuration error. The first steps you should take are to try pinging the DNS server, and make sure that the TCP/IP client configuration has the correct addresses for the DNS servers it should be using.

To confirm that a domain controller has been registered in the DNS, open a command prompt window with administrative privileges and enter the following command:

```
dcdiag /test:registerindns /dnsdomain:<domain name> /v
```

Objective summary

- A directory service is a repository of information about the resources—hardware, software, and human—that are connected to a network. Active Directory is the directory service that Microsoft first introduced in Windows 2000 Server, which has been upgraded in each successive server operating system release, including Windows Server 2012.
- When you create your first domain on an Active Directory network, you are, in essence, creating the root of a domain tree. You can populate the tree with additional domains, as long as they are part of the same contiguous namespace.
- When beginning a new AD DS installation, the first step is to create a new forest, which you do by creating the first domain in the forest, the forest root domain.
- In Windows Server 2012, it is now possible to install Active Directory Domain Services on a computer running the Server Core installation option, and promote the system to a domain controller, all using Windows PowerShell.

- Install from Media (IFM) is a feature that enables administrators to streamline the process of deploying replica domain controllers to remote sites.
- There are two ways to upgrade an AD DS infrastructure. You can upgrade the existing downlevel domain controllers to Windows Server 2012, or you can add a new Windows Server 2012 domain controller to your existing installation.
- The global catalog is an index of all the AD DS objects in a forest that prevents systems from having to perform searches among multiple domain controllers.
- DNS is essential to the operation of Active Directory Domain Services. To accommodate directory services such as AD DS, a special DNS resource record was created that enables clients to locate domain controllers and other vital AD DS services.

Objective review

Answer the following questions to test your knowledge of the information in this objective. You can find the answers to these questions and explanations of why each answer choice is correct or incorrect in the "Answers" section at the end of this chapter.

1. Which of the following cannot contain multiple Active Directory domains?
   A. organizational units
   B. sites
   C. trees
   D. forests
2. What are the two basic classes of Active Directory objects?
   A. Resource
   B. Leaf
   C. Domain
   D. Container
3. Which of the following is not true about an object's attributes?
   A. Administrators must manually supply information for certain attributes.
   B. Every container object has, as an attribute, a list of all the other objects it contains.
   C. Leaf objects do not contain attributes.
   D. Active Directory automatically creates the globally unique identifier (GUID).
4. Which of the following is not a reason why you should try to create as few domains as possible when designing an Active Directory infrastructure?
   A. Creating additional domains increases the administrative burden of the installation.
   B. Each additional domain you create increases the hardware costs of the Active Directory deployment.

C. Some applications might have problems working in a forest with multiple domains.

D. You must purchase a license from Microsoft for each domain you create.

5. Which of the following does an Active Directory client use to locate objects in another domain?

A. DNS

B. Global Catalog

C. DHCP

D. Site Link

Thought experiment

In the following thought experiment, apply what you've learned about this objective to predict what steps you need to take. You can find answers to these questions in the "Answers" section at the end of this chapter.

Robert is designing a new Active Directory Domain Services infrastructure for a company called Litware, Inc., which has its headquarters in New York and two additional offices in London and Tokyo. The London office consists only of sales and marketing staff; it does not have its own IT department. The Tokyo office is larger, with representatives from all of the company departments, including a full IT staff. The Tokyo office is connected to the headquarters using a 64 Kbps demand-dial link, and the London office has a 512-Kbps frame relay connection. The company has registered the litware.com domain name, and Robert has created a subdomain called inside.litware.com for use by Active Directory.

Based on this information, design an Active Directory infrastructure for Litware, Inc. that is as economical as possible, specifying how many domains to create, what to name them, how many domain controllers to install, and where. Explain each of your decisions.

Objective 5.2: Create and manage Active Directory users and computers

Users and computers are the basic leaf objects that populate the branches of the Active Directory Domain Services tree. Creating and managing these objects are everyday tasks for most AD DS administrators.

Creating user objects

The user account is the primary means by which people using an Active Directory Domain

Services network access resources. Resource access for individuals takes place through their individual user accounts. To gain access to the network, prospective network users must authenticate to a network with a specific user account.

Authentication is the process of confirming a user's identity using a known value such as a password, a smart card, or a fingerprint. When a user supplies a name and password, the authentication process validates the credentials supplied in the logon against information that has been stored within the AD DS database. Do not confuse authentication with authorization, which is the process of confirming that an authenticated user has the correct permissions to access one or more network resources.

There are two types of user accounts on systems running Windows Server 2012, as follows:

- **Local users** These accounts can only access resources on the local computer and are stored in the local Security Account Manager (SAM) database on the computer where they reside. Local accounts are never replicated to other computers, nor do these accounts provide domain access. This means that a local account configured on one server cannot be used to access resources on a second server; you would need to configure a second local account in that case.
- **Domain users** These accounts can access AD DS or network-based resources, such as shared folders and printers. Account information for these users is stored in the AD DS database and replicated to all domain controllers within the same domain. A subset of the domain user account information is replicated to the global catalog, which is then replicated to other global catalog servers throughout the forest.

By default, two built-in user accounts are created on a computer running Windows Server 2012: the Administrator account and the Guest account. Built-in user accounts can be local accounts or domain accounts, depending on whether the server is configured as a standalone server or a domain controller. In the case of a standalone server, the built-in accounts are local accounts on the server itself. On a domain controller, the built-in accounts are domain accounts that are replicated to each domain controller.

On a member server or standalone server, the built-in local Administrator account has full control of all files as well as complete management permissions for the local computer. On a domain controller, the built-in Administrator account created in Active Directory has full control of the domain in which it was created. By default, there is only one built-in administrator account per domain. Neither the local Administrator account on a member server or standalone server nor a domain Administrator account can be deleted; however, they can be renamed.

The following list summarizes several security guidelines you should consider regarding the Administrator account:

- **Rename the Administrator account** This will stave off attacks that are targeted specifically at the Administrator username on a server or domain. This will only protect against fairly unsophisticated attacks, though, and you should not rely on this as the only means of protecting the accounts on your network.

- **Set a strong password** Make sure that the password is at least seven characters in length and contains uppercase and lowercase letters, numbers, and alphanumeric characters.
- **Limit knowledge of administrator passwords to only a few individuals** Limiting the distribution of administrator passwords limits the risk of security breaches using this account.
- **Do not use the Administrator account for daily nonadministrative tasks** Microsoft recommends using a nonadministrative user account for normal work and using the Run As command when administrative tasks need to be performed.

The built-in Guest account is used to provide temporary access to the network for a user such as a vendor representative or a temporary employee. Like the Administrator account, this account cannot be deleted, but it can and should be renamed. In addition, the Guest account is disabled by default and is not assigned a default password. In most environments, you should consider creating unique accounts for temporary user access rather than relying on the Guest account. In this way, you can be sure the account follows corporate security guidelines defined for temporary users. However, if you decide to use the Guest account, review the following guidelines:

- **Rename the Guest account after enabling it for use** As we discussed with the Administrator account, this will deny intruders a username, which is half of the information necessary to gain access to your domain.
- **Set a strong password** The Guest account, by default, is configured with a blank password. For security reasons, you should not allow a blank password. Make sure that the password is at least seven characters in length and contains uppercase and lowercase letters, numbers, and alphanumeric characters.

User creation tools

One of the most common tasks for administrators is the creation of Active Directory user objects. Windows Server 2012 includes several tools you can use to create objects. The specific tool you use depends on how many objects you need to create, the time frame available for the creation of these groups, and any special circumstances, such as importing users from an existing database.

When creating a single user, administrators can use Active Directory Administrative Center or the Active Directory Users and Computers console. However, when you need to create multiple users in a short time frame or you have an existing database from which to import these objects, you will want to choose a more efficient tool. Windows Server 2012 provides a number of tools you can choose according to what you want to accomplish. The following list describes the most commonly used methods for creating multiple users and groups. These tools are detailed in the upcoming sections.

- **Dsadd.exe** The standard command-line tool for creating AD DS leaf objects, which you can use with batch files to create AD DS objects in bulk

- **Windows PowerShell** The currently approved Windows maintenance tool, with which you can create object creation scripts of nearly unlimited complexity
- **Comma-Separated Value Directory Exchange (CSVDE.exe)** A command-line utility that can create new AD DS objects by importing information from a comma-separated value (.csv) file.
- **LDAP Data Interchange Format Directory Exchange (LDIFDE.exe)** Like CSVDE, a utility that can import AD DS information and use it to add, delete, or modify objects, in addition to modifying the schema, if necessary.

These tools all have their roles in network administration; it is up to the administrator to select the best tool to suit his or her skill set and a particular situation. For example, you might have two tools that can accomplish a job, but your first choice might be the tool with which you are most familiar or the one that can accomplish the task in a shorter amount of time.

The following sections examine various scenarios for using these tools to create user objects.

Creating single users

For some administrators, creating individual user accounts is a daily task, and there are many ways to go about it. Windows Server 2012 has redesigned the Active Directory Administrative Center (ADAC) application, first introduced in Windows Server 2008 R2, to fully incorporate new features such as the Active Directory Recycle Bin and fine-grained password policies. You can also use the tool to create and manage AD DS user accounts

To create a single user account with the Active Directory Administrative Center, use the following procedure.

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.
2. From the Tools menu, select Active Directory Administrative Center. The Active Directory Administrative Center console appears.
3. In the left pane, find the domain in which you want to create the user object and select a container in that domain.
4. In the Tasks pane, under the container name, click New > User to open the Create User window, as shown in Figure 5-11.

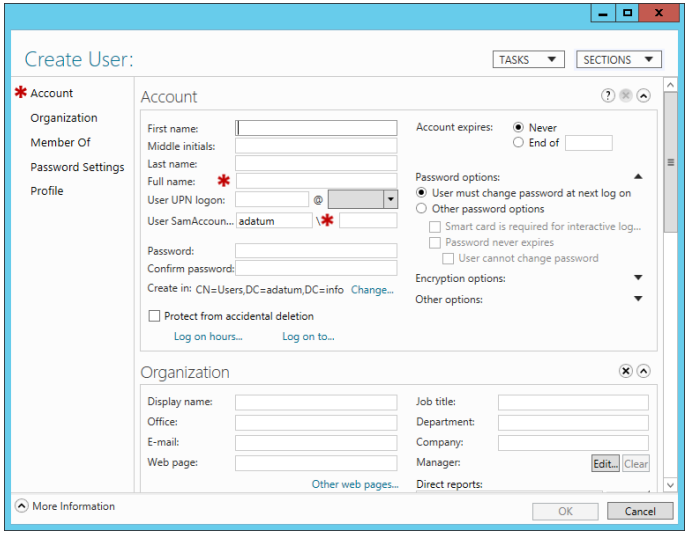

FIGURE 5-11 The Create User window in the Active Directory Administrative Center console.

5. Type the user's name in the Full Name field and an account name in the User SamAccountName Logon field.
6. Type an initial password for the user in the Password and Confirm password fields.
7. Supply information for any of the optional fields on the page you wish.
8. Click OK. The user object appears in the container.
9. Close the Active Directory Administrative Center console.

Administrators who are more comfortable with the familiar Active Directory Users and Computers console can still create user objects with that, using the New Object – User Wizard, as shown in Figure 5-12.

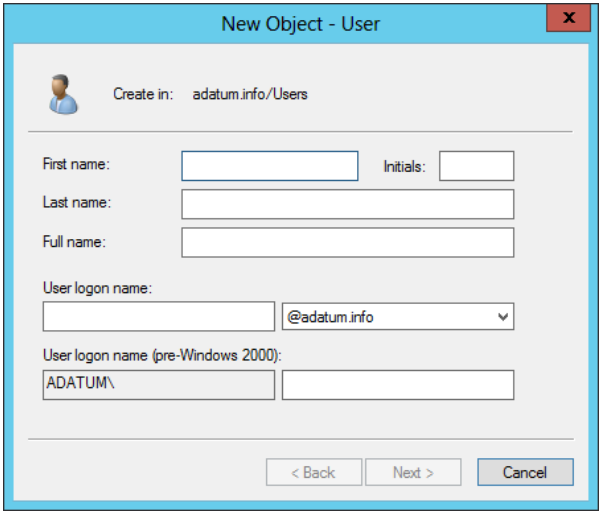

FIGURE 5-12 The New Object - User Wizard in the Active Directory Users and Computers console.

For administrators working on Server Core installations, or for those who are just more comfortable with the command line, it is also possible to create user objects without a graphical interface.

USING DSADD.EXE

For administrators more comfortable with the traditional command prompt, the Dsadd.exe program can create new user objects, using the syntax shown in Figure 5-13.

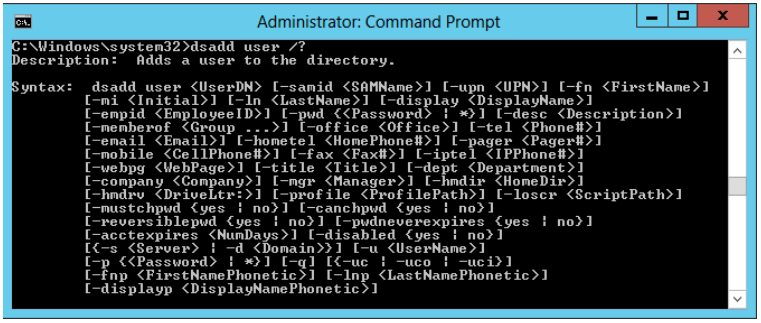

FIGURE 5-13 Syntax of the Dsadd.exe program.

To create a user using the Dsadd.exe utility, you must know the distinguished name (DN) for the user and the user's login ID, also known as the SAM account name attribute within AD DS. The distinguished name of an object signifies its relative location within the Active Directory structure. For example, in the distinguished name cn=Elizabeth Andersen,ou=Research,dc=adatum,dc=com, the cn refers to the common name for Elizabeth

Andersen's user account, which resides in the Research OU, which resides in the adatum.com domain.

Each object has a unique DN, but this DN can change if you move the object to different locations within the Active Directory structure. For example, If you create an additional layer of OUs representing offices in different cities, the previous DN might change to cn=Elizabeth Andersen,ou=Research,ou=Baltimore,dc=adatum,dc=com, even though it is the same user object with the same rights and permissions.

The SAM account name refers to each user's login name—the portion to the left of the '@' within a User Principal Name—which is eander in eander@adatum.com. The SAM account name must be unique across a domain.

When you have both of these items, you can create a user with the Dsadd.exe utility using the following syntax:

```
dsadd user <distinguished name> -samid <SAM account name>
```

For example, in its simplest form, you can create the account for Elizabeth Andersen referenced earlier as follows:

```
dsadd user
cn="Elizabeth Andersen,ou=Research,dc=adatum,dc=com"
-samid eander
```

You can also add attribute values with the Dsadd.exe tool. The following command adds some of the most common attributes to the user object:

```
Dsadd.exe User
"CN=Elizabeth Andersen,OU=Research,DC=adatum,DC=local"
-samid "eander"
-fn "Elizabeth"
-ln "Andersen"
-disabled no
-mustchpwd yes
-pwd "Pa$$w0rd"
```

USING WINDOWS POWERSHELL

Microsoft is placing increased emphasis on Windows PowerShell as a server management tool, and provides a cmdlet called New-ADUser, which you can use to create a user account and configure any or all of the attributes associated with it. The New-ADUser cmdlet has a great many parameters, as shown in Figure 5-14, to enable access to all of the user object's attributes.

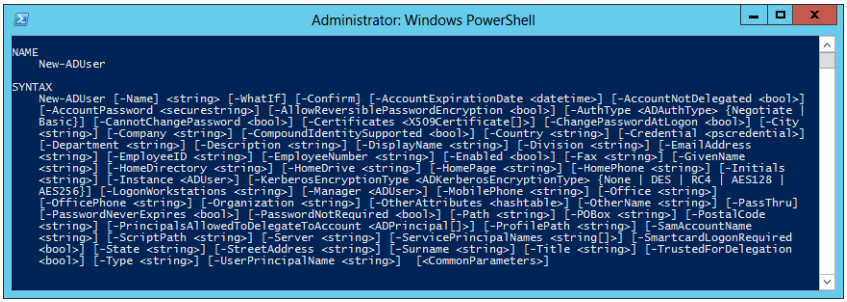

FIGURE 5-14 Syntax of the New-ADUser cmdlet.

For example to create a new user object for Elizabeth Andersen in an organizational unit (OU) called Research, you could use the New-ADUser command with the following parameters:

```
new-ADUser
-Name "Elizabeth Andersen"
-SamAccountName "eander"
-GivenName "Elizabeth"
-SurName "Andersen"
-path 'OU=Research,DC=adatum,dc=local'
-Enabled $true
-AccountPassword "Pa$$w0rd"
-ChangePasswordAtLogon $true
```

The –Name and –SamAccountName parameters are required to identify the object. The –path parameter specifies the location of the object in the AD DS hierarchy. The –Enabled parameter ensures that the account is active.

Creating user templates

In some cases, administrators have to create single users on a regular basis, but the user accounts contain so many attributes that creating them individually is time-consuming.

One way to speed up the process of creating complex user objects is to use the New-ADUser cmdlet or the Dsadd.exe program and retain your commands in a script or batch file. However, if you prefer a graphical interface, you can do roughly the same thing by creating a user template.

A user template is a standard user object containing boilerplate attribute settings. When you want to create a new user with those settings, you simply copy the template to a new user object and change the name and any other attributes that are unique to the user.

To create a user template with the Active Directory Users and Computers console, use the following procedure.

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.

2. Create a user object with the name Default Template, clearing the User Must Change Password At Next Logon check box and selecting the Account Is Disabled check box.
3. Open the user's Properties sheet and modify the attributes on the various tabs with values common to all the users you will be creating.
4. Close the Active Directory Users and Computers console.

To use the template, right-click the Default Template user object and, from the shortcut menu, select Copy. The Copy Object – User Wizard appears, as shown in Figure 5-15.

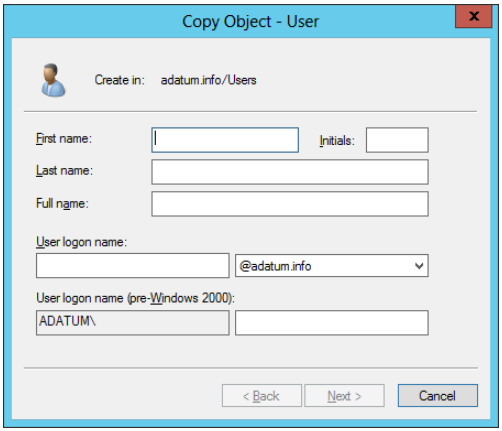

FIGURE 5-15 The Copy Object – User Wizard.

Enter the required unique information for the user and clear the Account Is Disabled check box before clicking OK. The wizard creates a new user object with all of the attributes you configured in the template.

Creating multiple users

Administrators sometimes have to create hundreds or thousands of user objects, making the single object creation procedures impractical. The previous sections described the procedures for creating single user and group object using the GUI and some of the available command-line tools in Windows Server 2012. The following sections examine some of the mechanisms for automating the creation of large numbers of Active Directory objects.

USING CSVDE.EXE

Applications such as Microsoft Excel can generate lists users, along with their accompanying information, to add to the AD DS database. In these cases, you can export information from the applications by saving it to a file in CSV format. CSV format also can be used to import information into and export it from other third-party applications.

A CSV file is a plain text file that consists of records, each on a separate line, which are divided into fields, separated by commas. The format is a means for saving database information in a universally understandable way.

The CSVDE.exe command-line utility enables administrators to import or export Active Directory objects. It uses a CSV file that is based on a header record, which identifies the attribute contained in each comma-delimited field. The header record is simply the first line of the text file that uses proper attribute names. To be imported into AD DS, the attribute names in the CSV file must match the attributes allowed by the Active Directory schema. For example, if you have a list of people and telephone numbers you want to import as users into the Active Directory database, you will need to create a header record that accurately reflects the object names and attributes you want to create. Review the following attributes that are commonly used for creating user accounts.

- **dn** Specifies the distinguished name of the object so that the object can be properly placed in Active Directory
- **samAccountName** Populates the SAM account field
- **objectClass** Specifies the type of object to be created, such as user, group, or OU
- **telephoneNumber** Populates the Telephone Number field
- **userPrincipalName** Populates the User Principal Name field for the account

As you create your CSV file, you must order the data to reflect the sequence of the attributes in the header record. If fields and data are out of order, you will either encounter an error when running the CSVDE.exe utility or you might not get accurate results in the created objects. The following example of a header record uses the previously listed attributes to create a user object.

```
dn,samAccountName,userPrincipalName,telephoneNumber, objectClass
```

A data record conforming to this header record would then appear as follows:

```
"cn=Elizabeth Andersen,ou=Research,dc=adatum,dc=com",eander,eander@adatum.com,586-555-1234,user
```

After you have added a record for each account you want to create, save the file using .csv as the extension. You then use the following command syntax to run the CSVDE.exe program and import the file:

```
csvde.exe -i -f <filename.csv>
```

The -i switch tells CSVDE.exe that this operation will import data. The -f switch is used to specify the .csv file containing the records to be imported.

USING LDIFDE.EXE

LDIFDE.exe is a utility that has the same basic functionality as CSVDE.exe and provides the ability to modify existing records in Active Directory. For this reason, LDIFDE.exe is a more flexible option. Consider an example where you have to import 200 new users into your AD DS structure. In this case, you can use CSVDE.exe or LDIFDE.exe to import the users. However, you can use LDIFDE.exe to modify or delete the objects later, whereas CSVDE.exe does not provide this option.

You can use any text editor to create the LDIFDE.exe input file, which is formatted according to the LDAP Data Interchange Format (LDIF) standard. The format for the data file containing the object records you wish to create is significantly different from that of CSVDE.exe. The following example shows the syntax for a data file to create the same user account discussed in the CSVDE.exe example.

```
dn: "cn=Elizabeth Andersen,ou=Research,dc=adatum,dc=com"
changetype: add
ObjectClass: user
SAMAccountName: eander
UserPrincipalName: eander@adatum.com
telephoneNumber: 586-555-1234
```

Using LDIFDE.exe, you can specify one of three actions that will be performed with the LDIF file:

- **Add** Creates new objects using the LDIF records
- **Modify** Modifies existing object attributes using the LDIF records
- **Delete** Deletes existing objects using the LDIF records

After creating the data file and saving it using the .ldf file extension, use the following syntax to execute the LDIFDE.exe program.

```
ldifde –i –f <filename.ldf>
```

The next example illustrates the LDIF syntax to modify the telephone number of an existing user object. Note that the hyphen in the last line is required for the file to function correctly.

```
dn: "cn=Elizabeth Andersen,ou=Research,dc=adatum,dc=com
changetype: modify
replace: telephoneNumber
telephoneNumber: 586-555-1111
-
```

USING WINDOWS POWERSHELL

It is also possible to use CSV files to create user objects with Windows PowerShell, by using the Import-CSV cmdlet to read the data from the file and piping it to the New-ADUser cmdlet. To insert the data from the file into the correct user object attributes, you use the New-ADUser cmdlet parameters to reference the field names in the CSV file's header record.

An example of a bulk user creation command would be as follows:

```
Import-CSV users Finance.csv | foreach
{New-ADUser -SamAccountName $_.SamAccountName
-Name $_.Name -Surname $_.Surname
-GivenName $_.GivenName -Path "OU=Research,DC=adatum,DC=COM" -AccountPassword Pa$$w0rd
-Enabled $true}
```

Creating computer objects

Because an AD DS network uses a centralized directory, there has to be some means of tracking the actual computers that are part of the domain. To do this, Active Directory uses computer accounts, which are realized in the form of computers objects in the Active Directory database. You might have a valid Active Directory user account and a password, but if your computer is not represented by a computer object, you cannot log on to the domain.

Computer objects are stored in the Active Directory hierarchy just as user objects are, and they possess many of the same capabilities, such as the following:

- Computer objects consist of properties that specify the computer's name, where it is located, and who is permitted to manage it.
- Computer objects inherit group policy settings from container objects such as domains, sites, and OUs.
- Computer objects can be members of groups and inherit permissions from group objects.

When a user attempts to log on to an Active Directory domain, the client computer establishes a connection to a domain controller to authenticate the user's identity. However, before the user authentication occurs, the two computers perform a preliminary authentication using their respective computer objects, to ensure that both systems are part of the domain. The NetLogon service running on the client computer connects to the same service on the domain controller, and then each one verifies that the other system has a valid computer account. When this validation is completed, the two systems establish a secure communications channel between them, which they can then use to begin the user authentication process.

The computer account validation between the client and the domain controller is a genuine authentication process using account names and passwords, just as when a user authenticates to the domain. The difference is that the passwords used by the computer accounts are generated automatically and kept hidden. Administrators can reset a computer account, but they do not have to supply passwords for them.

What all this means for administrators is that, in addition to creating user accounts in the domain, they have to make sure that the network computers are part of the domain as well. Adding a computer to an AD DS domain consists of two steps:

- **Creating a computer account** You create a computer account by creating a new computer object in Active Directory and assigning the name of an actual computer on the network.
- **Joining the computer to the domain** When you join a computer to the domain, the system contacts a domain controller, establishes a trust relationship with the domain, locates (or creates) a computer object corresponding to the computer's name, alters its security identifier (SID) to match that of the computer object, and modifies its group memberships.

How these steps are performed, and who performs them, depends on the way in which you deploy computers on your network. There are many ways to create new computer objects, and how administrators elect to do this depends on several factors, including the number of objects they need to create, where they will be when creating the objects, and what tools they prefer to use.

Generally speaking, you create computer objects when you deploy new computers in the domain. Once a computer is represented by an object and joined to the domain, any user in the domain can log on from that computer. For example, you do not have to create new computer objects or rejoin computers to the domain when employees leave the company and new hires start using their computers. However, if you reinstall the operating system on a computer, you must create a new computer object for it (or reset the existing one), because the newly installed computer will have a different SID.

The creation of a computer object must always occur before the corresponding computer can actually join the domain, although it sometimes does not appear that way. There are two basic strategies for creating Active Directory computer objects, which are as follows:

- Create the computer objects in advance using an Active Directory tool, so that the computers can locate the existing objects when they join the domain.
- Begin the joining process first and let the computer create its own computer object.

In each case, the computer object exists before the joining takes place. In the second strategy, the joining process appears to begin first, but the computer creates the object before the actual joining process begins.

When there are a number of computers to deploy, particularly in different locations, most administrators prefer to create the computer objects in advance. For large numbers of computers, it is even possible to automate the computer object creation process by using command-line tools and batch files. The following sections examine the tools you can use for computer object creation.

Creating computer objects using Active Directory Users and Computers

As with user objects, you can create computer objects with the Active Directory Users and Computers console. To create computer objects in an Active Directory domain using the Active Directory Users and Computers console or any tool, you must have the appropriate permissions for the container in which the objects will be located.

By default, the Administrators group has permission to create objects anywhere in the domain and the Account Operators group has the special permissions needed to create computer objects in and delete them from the Computers container, as well as from any new OUs you create. The Domain Admins and Enterprise Admins groups are members of the Administrators group, so members of those groups can create computer objects anywhere, as well. An administrator can also explicitly delegate control of containers to particular users or groups, enabling them to create computer objects in those containers.

The process of creating a computer object in Active Directory Users and Computers is similar to that of creating a user object. You select the container in which you want to place the object and, from the Action menu, select New > Computer. The New Object – Computer Wizard appears, as shown in Figure 5-16.

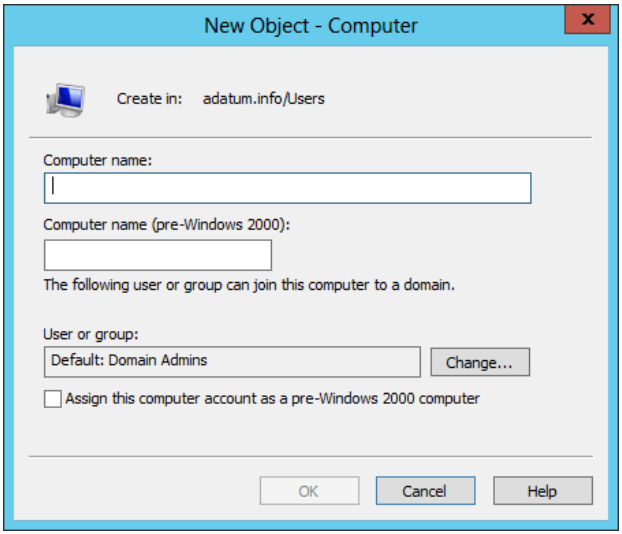

FIGURE 5-16 The New Object – Computer Wizard.

Computer objects have relatively few attributes, and in most cases, you will most likely just supply them with a name, which can be up to 64 characters long. This name must match the name of the computer joined with the object.

Creating computer objects with Active Directory Administrative Center

As with users, you can also create computer objects in the Active Directory Administrative Center. To create a computer object, you choose a container and then select New > Computer from the Tasks list to display the Create Computer dialog box.

Creating computer objects using Dsadd.exe

As with users, the graphical tools provided with Windows Server 2012 are good for creating and managing single objects, but many administrators turn to the command line when they have to create multiple objects.

The Dsadd.exe utility enables you to create computer objects from the command line, just as you created user objects earlier in this lesson. You can create a batch file of Dsadd.exe commands to generate multiple objects in one process. The basic syntax for creating a computer object with Dsadd.exe is as follows:

```
dsadd computer <ComputerDN>
```

The <ComputerDN> parameter specifies a distinguished name for the new group object you want to create. The DNs use the same format as those in CSV files, as discussed earlier.

Creating computer objects using Windows PowerShell

Windows PowerShell includes the New-ADComputer cmdlet, which you can use to create computer objects with the following basic syntax. This cmdlet creates computer objects, but it does not join them to a domain.

```
new-ADComputer -Name <computer name> –path <distinguished name>
```

Managing Active Directory objects

Once you have created user and computer objects, you can manage them and modify them in many of the same ways that you created them.

Double-clicking any object in the Active Directory Administrative Center or the Active Directory Users and Computers console opens the Properties sheet for that object. The windows appear different, but they contain the same information, and provide the same ability to alter the object attributes.

Managing multiple users

When managing domain user accounts, there are likely to be times when you have to make the same changes to multiple user objects, and modifying each one individually would be a tedious chore.

In these instances, it is possible to modify the properties of multiple user accounts simultaneously, using the Active Directory Administrative Center or the Active Directory Users and Computers console. You simply select several user objects by holding down the Ctrl key as you click each user, and then select Properties. A Properties sheet appears, containing the attributes you can manage for the selected objects simultaneously, as shown in Figure 5-17.

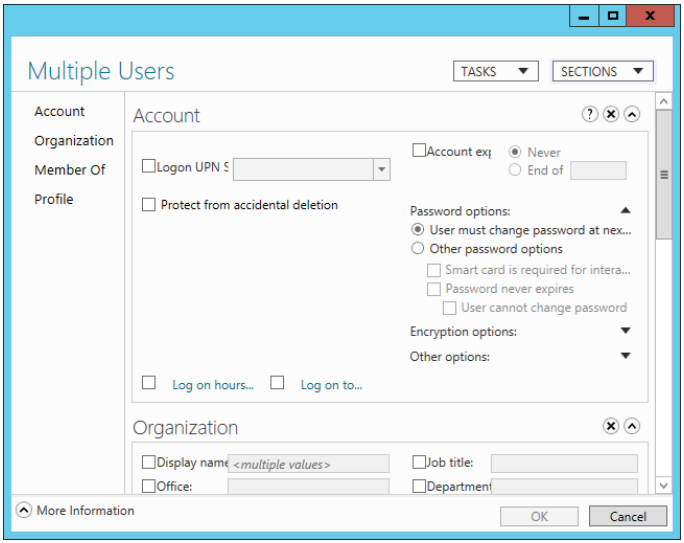

FIGURE 5-17 A Multiple Users Properties sheet in Active Directory Administrative Center.

Joining computers to a domain

The process of actually joining a computer to a domain must occur at the computer itself and be performed by a member of the computer's local Administrators group. After logging on, you join a computer running Windows Server 2012 to a domain from the Computer Name tab in the System Properties sheet. You can access the System Properties sheet from Server Manager, by clicking the Computer name or domain hyperlink on the server's Properties tile, from the Control Panel.

On a computer that is not joined to a domain, the Computer Name tab displays the name assigned to the computer during the operating system installation, and the name of the workgroup to which the system currently belongs (which is WORKGROUP, by default). To join the computer to the domain, click Change to display the Computer Name Changes dialog box shown in Figure 5-18.

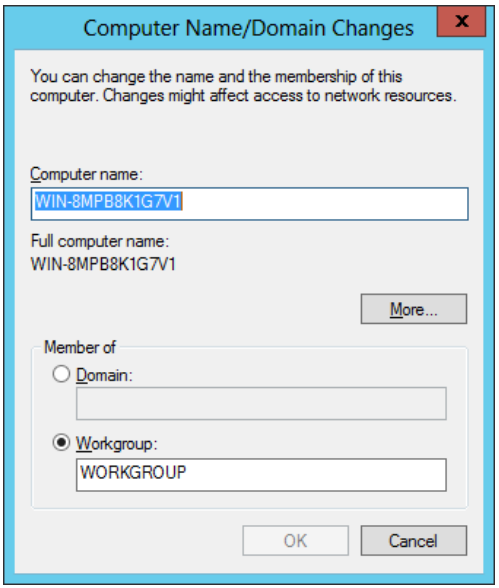

FIGURE 5-18 The Computer Name Changes dialog box.

In this dialog box, the Computer Name field enables you to change the name assigned to the computer during installation. Depending on whether you have already created a computer object, observe the following precautions:

- To join a domain in which you have already created a computer object for the system in Active Directory Domain Services, the name on this field must match the name of the object exactly.
- If you intend to create a computer object during the joining process, the name in this field must not already exist in the domain.

When you select the Domain option and enter the name of the domain the computer will join, the computer establishes contact with a domain controller for the domain and a second Computer Name Changes dialog box appears, prompting you for the name and password of a domain user account with permission to join the computer to the domain.

Once you have authenticated with the domain controller, the computer is welcomed to the domain and you are instructed to restart the computer.

JOINING A DOMAIN USING NETDOM.EXE

It is also possible to use the Netdom.exe command-line utility to join a computer to a domain. The syntax for the command is as follows:

```
netdom join <computername> /Domain:<DomainName>
[/UserD:<User> /PasswordD:<UserPassword>] [/OU:OUDN]
```

CREATING COMPUTER OBJECTS WHILE JOINING

You can join a computer to a domain whether or not you have already created a computer object for it. Once the computer authenticates to the domain controller, the domain controller scans the Active Directory database for a computer object with the same name as the computer. If it does not find a matching object, the domain controller creates one in the Computers container, using the name supplied by the computer.

For the computer object to be created automatically in this manner, one would expect that the user account you specify when connecting to the domain controller must have object creation privileges for the Computers container, such as membership in the administrators group. However, this is not always the case.

Domain users can also create computer objects themselves through an interesting, indirect process. The Default Domain Controllers Policy Group Policy object (GPO) grants a user right called Add Workstations To The Domain to the Authenticated Users special identity, as shown in Figure 5-19. This means that any user that is successfully authenticated to Active Directory is permitted to join up to ten workstations to the domain, and create ten associated computer objects, even if they do not possess explicit object creation permissions.

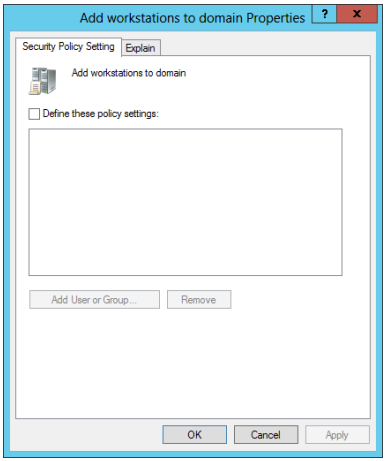

FIGURE 5-19 The Default Domain Controllers Policy user rights assignments.

NOTE **Assigning user rights**

User rights are Group Policy settings that provide users with the ability to perform certain system-related tasks. For example, logging on locally to a domain controller requires that a user has the Log On Locally right assigned to his or her account or be a member of the Account Operators, Administrators, Backup Operators, Print Operators, or Server Operators group on the domain controller. Other similar settings included in this collection are related to user rights associated with system shutdown, taking ownership privileges of files or objects, and synchronizing directory service data. For more

information on user rights assignment, see Objective 6.2, "Configure Security Policies," in Chapter 6.

JOINING A DOMAIN WHILE OFFLINE

It is typical for administrators to join computers to domains while the computers are connected to the network and have access to a domain controller. However, there are situations in which administrators might want to set up computers without access to a domain controller, such as a new branch office installation. In these cases, it is possible to perform an offline domain join, using a command-line program called Djoin.exe.

The offline domain join procedure requires you to run the Djoin.exe program twice, once on a computer with access to a domain controller, and then again on the computer to be joined. When connected to the domain controller, the program gathers computer account metadata for the system to be joined and saves it to a file. The syntax for this phase of the process is as follows:

```
djoin /provision /domain <domain name>
/machine <computer name> /savefile <filename.txt>
```

You then transport the metadata file to the computer to be joined and run Djoin.exe again, specifying the name of the file. The program saves the metadata from the file to the computer, so that the next time it has access to a domain controller, the system is automatically joined to the domain. The syntax for the second phase of the process is as follows:

```
djoin /requestODJ /loadfile <filename.txt>
/windowspath %SystemRoot% /localos
```

Managing disabled accounts

Disabling a user account prevents anyone from using it to log on to the domain until an administrator with the appropriate permissions enables it again. You can disable user accounts manually, to prevent their use while preserving all of their attributes, but it is also possible for a system to automatically disable them. For example, repeated violations of password policy settings can disable an account to prevent intruders from making further attack attempts.

To disable or enable a user or computer account in Active Directory Administrative Center or Active Directory Users and Computers, simply right-click the object and select Disable or Enable from the shortcut menu. You can also disable and enable multiple accounts by selecting multiple objects and right-clicking.

To disable or enable a user or computer account with Windows PowerShell, use the following cmdlet syntax:

```
Disable-ADAccount –Identity <account name>

Enable-ADAccount –Identity <account name>
```

Objective summary

- The user account is the primary means by which people using an Active Directory Domain Services network access resources.
- One of the most common tasks for administrators is the creation of Active Directory user objects. Windows Server 2012 includes several tools you can use to create objects.
- Windows Server 2012 has redesigned the Active Directory Administrative Center (ADAC) application, first introduced in Windows Server 2008 R2, to fully incorporate new features such as the Active Directory Recycle Bin and fine-grained password policies. You can also use the tool to create and manage AD DS user accounts.
- Microsoft Excel and Microsoft Exchange are two common applications in which you can have a number of users, along with their accompanying information, to add to the AD DS database. In these cases, you can export information from the applications by saving it to a file in CSV format.
- LDIFDE.exe is a utility that has the same basic functionality as CSVDE.exe and provides the ability to modify existing records in Active Directory.
- Because an AD DS network uses a centralized directory, there has to be some means of tracking the actual computers that are part of the domain. To do this, Active Directory uses computer accounts, which are realized in the form of computer objects in the Active Directory database.
- The process of actually joining a computer to a domain must occur at the computer itself and be performed by a member of the computer's local Administrators group.
- It is possible to perform an offline domain join, using a command-line program called Djoin.exe.

Objective review

Answer the following questions to test your knowledge of the information in this objective. You can find the answers to these questions and explanations of why each answer choice is correct or incorrect in the "Answers" section at the end of this chapter.

1. What can be used to add, delete, or modify objects in Active Directory, in addition to modifying the schema if necessary?
   A. DCPROMO
   B. LDIFDE
   C. CSVDE
   D. NSLOOKUP
2. When using CSVDE, what is the first line of the text file that uses proper attribute names?
   A. header row

B. header record

C. name row

D. name record

3. Which of the following utilities do you use to perform an offline domain join?

A. net join

B. join

C. djoin

D. dconnect

4. Which of the following is not a type of user account that can be configured in Windows Server 2012?

A. local accounts

B. domain accounts

C. network accounts

D. built-in accounts

5. Which of the following are the two built-in user accounts created automatically on a computer running Windows Server 2012?

A. Network

B. Interactive

C. Administrator

D. Guest

Thought experiment

In the following thought experiment, apply what you've learned about this objective to predict what steps you need to take. You can find answers to these questions in the "Answers" section at the end of this chapter.

You are a network administrator who is in the process of building an Active Directory network for a company called Fabrikam, Inc., and you have to create user objects for the 75 users in the Inside Sales department. You have already created the fabrikam.com domain and an OU called Inside Sales for this purpose. The Human Resources department has provided you with a list of the users' names and has instructed you to create the account names by using the first initial and the last name. Each user object must also have the value Inside Sales in the Department property and Fabrikam, Inc. in the Company property. Using the first name in the list, Oliver Cox, as an example, which of the following command-line formats would enable you to create the 75 user objects, with the required property values?

1. **dsadd "Oliver Cox" –company "Fabrikam, Inc." –dept "Inside Sales"**

2. **dsadd user CN=Oliver Cox,CN=Inside Sales,DC=fabrikam,DC=com –company Fabrikam, Inc. –dept Inside Sales**
3. **dsadd –company "Fabrikam, Inc." –dept "Inside Sales" "CN=Oliver Cox,CN=Inside Sales,DC=fabrikam,DC=com"**
4. **dsadd user "CN=Oliver Cox,CN=Inside Sales,DC=fabrikam,DC=com" –company "Fabrikam, Inc." –dept "Inside Sales"**

Objective 5.3: Create and manage Active Directory groups and organizational units (OUs)

OUs can be nested to create a design that enables administrators to take advantage of the inheritance described earlier. You should limit the number of OUs that are nested, because too many levels can slow the response time to resource requests and complicate the application of group policy settings.

When you first install Active Directory Domain Services, there is only one OU in the domain, by default: the Domain Controllers OU. All other OUs must be created by a domain administrator.

> ***NOTE*** **OUs and permissions**
>
> **OUs are not considered security principals. This means that you cannot assign access permissions to a resource based on membership to an OU. Herein lies the difference between OUs and global, domain local, and universal groups. Groups are used for assigning access permissions, whereas OUs are used for organizing resources and delegating permissions.**

There is another type of container object found in a domain, literally called a container. For example, a newly created domain has several container objects in it, including one called Users, which contains the domain's predefined users and groups, and another called Computers, which contains the computer objects for all of the systems joined to the domain.

Unlike with OUs, you cannot assign Group Policy settings to computer objects, nor can you delegate their administration. You also cannot create new container objects using the standard Active Directory administration tools, such as the Active Directory Users and Computers console. You can create container objects using scripts, but there is no compelling reason to do so. OUs are the preferred method of subdividing a domain.

Creating OUs

There is no simpler object type to create in the AD DS hierarchy than an OU. You only have to supply a name for the object and define its location in the Active Directory tree.

To create an OU object using the Active Directory Administrative Center, use the following procedure:

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.
2. From the Tools menu, select Active Directory Administrative Center to open the Active Directory Administrative Center console.
3. In the left pane, right-click the object beneath which you want to create the new OU and, from the shortcut menu, select New > Organizational Unit. The Create Organizational Unit window appears, as shown in Figure 5-20.

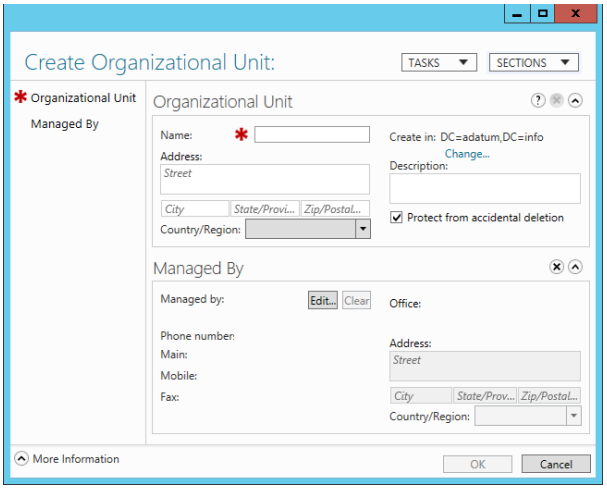

FIGURE 5-20 The Create Organizational Unit window in Active Directory Administrative Center.

4. In the Name field, type a name for the OU and add any optional information you desire.
5. Click OK. The OU object appears in the container.
6. Close the Active Directory Administrative Center console.

Creating an OU in the Active Directory Users and Computers console works in roughly the same way, although the New Object – Organizational Unit dialog box is different in appearance. Once you have created an OU, you can double-click it to open its Properties sheet, where you can modify its attributes, or right-click it and select Move, to open the Move dialog box, as shown in Figure 5-21.

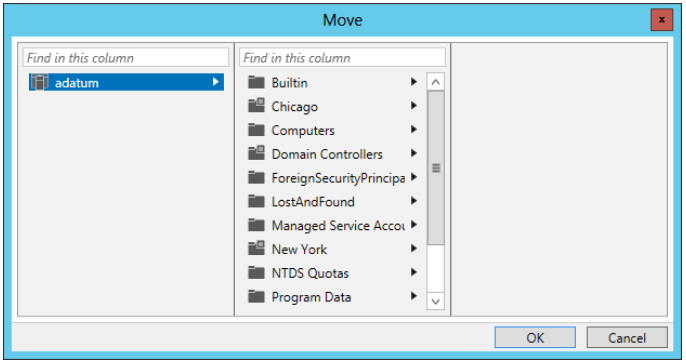

FIGURE 5-21 The Move dialog box in Active Directory Administrative Center.

Using OUs to delegate Active Directory management tasks

Creating OUs enables you to implement a decentralized administration model, in which others manage portions of the AD DS hierarchy, without affecting the rest of the structure.

Delegating authority at a site level affects all domains and users within the site. Delegating authority at the domain level affects the entire domain. However, delegating authority at the OU level affects only that OU and its subordinate objects. By granting administrative authority over an OU structure, as opposed to an entire domain or site, you gain the following advantages:

- **Minimal number of administrators with global privileges** By creating a hierarchy of administrative levels, you limit the number of people who require global access.
- **Limited scope of errors** Administrative mistakes such as a container deletion or group object deletion affect only the respective OU structure.

The Delegation of Control Wizard provides a simple interface you can use to delegate permissions for domains, OUs, or containers. AD DS has its own system of permissions, much like those of NTFS and printers. The Delegation of Control Wizard is essentially a front-end interface that creates complex combinations of permissions based on specific administrative tasks.

The wizard interface enables you to specify the users or groups to which you want to delegate management permissions and the specific tasks you wish them to be able to perform. You can delegate predefined tasks or create custom tasks that enable you to be more specific.

To delegate administrative control over an OU, use the following procedure:

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.
2. Open the Active Directory Users and Computers console, right-click the object over which you want to delegate control, and click Delegate Control. The Delegation of Control Wizard starts, displaying the Welcome page.

3. Click Next to move to the Users Or Groups page.
4. Click Add To open the Select Users, Computers, Or Groups dialog box.
5. Type the name of the user or group to which you want to delegate control of the object, and click OK. The user or group appears in the Selected Users And Groups list.
6. Click Next. The Tasks To Delegate page appears, with the following options:
   - **Delegate The Following Common Tasks** This option enables you to choose from a list of predefined tasks.
   - **Create A Custom Task To Delegate** This option enables you to be more specific about the task delegation.
7. Select Create A Custom Task To Delegate and click Next. The Active Directory Object Type page opens, displaying the following options:
   - **This Folder, Existing Objects In This Folder, And Creation Of New Objects In This Folder** This option delegates control of the container, including all of its current and future objects.
   - **Only The Following Objects In The Folder** This option enables you to select specific objects to be controlled. You can select Create Selected Objects In This Folder to allow selected object types to be created, or select Delete Selected Objects In This Folder to allow selected object types to be deleted.
8. Select This Folder, Existing Objects In This Folder, And Creation Of New Objects In This Folder and click Next. The Permissions page opens.
9. Set the delegated permissions according to your needs for the user or group to which you are delegating control. You can combine permissions from all three of the following options:
   - **General** Displays general permissions, which are equal to those displayed on the Security tab in an object's properties.
   - **Property-specific** Displays permissions that apply to specific attributes or properties of an object.
   - **Creation/deletion of specific child objects** Displays permissions that apply to creation and deletion permissions for specified object types.
10. Click Next to open the Completing The Delegation of Control Wizard page.
11. Click Finish.
12. Close the Active Directory Users and Computers console.

In this procedure, you granted permissions over a portion of Active Directory to a specified administrator or group of administrators. Although you can use the Delegation of Control Wizard to grant permissions, you cannot use it to modify or remove permissions. To perform these tasks, you must use the interface provided in the Security tab in the AD DS object's Properties sheet.

NOTE **Advanced View**

By default, the Security tab does not appear in an OU's Properties sheet in the Active Directory Users and Computers console. To display the tab, you must select Advanced Features from the console's View menu.

Working with groups

Since the early days of the Microsoft server operating system, administrators have used groups to manage network permissions. Groups enable administrators to assign permissions to multiple users simultaneously. A group can be defined as a collection of user or computer accounts that functions as a security principal, in much the same way that a user does.

In Windows Server 2012, when a user logs on to Active Directory, an access token is created that identifies the user and that user's group memberships. Domain controllers use this access token to verify a user's permissions when the user attempts to access a local or network resource. By using groups, administrators can grant multiple users the same permission level for resources on the network. If, for example, you have 25 users in the graphics department who need access to a color printer, you can either assign each user the appropriate permissions for the printer, or you can create a group containing the 25 users and assign the appropriate permissions to the group. By using a group object to access a resource, you have accomplished the following:

- When users need access to the printer, you can simply add them to the group. Once added, the user receives all permissions assigned to this group. Similarly, you can remove users from the group when you want to revoke their access to the printer.
- Administrators only have to make one change to modify the level of access to the printer for all of the users. Changing the group's permissions changes the permission level for all group members. Without the group, you would have to modify all 25 user accounts individually.

NOTE **Access tokens**

Users' access tokens are only generated when they first log on to the network from their workstation. If you add users to a group, they will need to log off and log back on again for that change to take effect.

Users can be members of more than one group. In addition, groups can contain other Active Directory objects, such as computers, and other groups in a technique called group nesting. Group nesting describes the process of configuring one or more groups as members of another group. For example, consider a company that has two groups: marketing and graphic design. Graphic design group members have access to a high-resolution color laser printer. If the marketing group personnel also need access to the printer, you can simply add the marketing group as a member of the graphic design group. This gives the marketing

group members the same permission to the color laser printer as the members of the graphic design group.

Group types

There are two group classifications in Windows Server 2012: group type and group scope. Group type defines how a group is used within Active Directory.

The two Windows Server 2012 group types are as follows:

- **Distribution groups** Nonsecurity-related groups created for the distribution of information to one or more persons
- **Security groups** Security-related groups created for purposes of granting resource access permissions to multiple users

Active Directory–aware applications can use distribution groups for nonsecurity-related functions. For example, Microsoft Exchange uses distribution groups to send messages to multiple users. Only applications that are designed to work with Active Directory can make use of distribution groups in this manner.

Groups that you use to assign permissions to resources are referred to as security groups. Administrators make multiple users that need access to the same resource members of a security group. They then grant the security group permission to access the resource. After you create a group, you can convert it from a security group to a distribution group, or vice versa, at any time.

Group scopes

In addition to security and distribution group types, several group scopes are available within Active Directory. The group scope controls which objects the group can contain, limiting the objects to the same domain or permitting objects from remote domains as well, and also controls the location in the domain or forest where the group can be used. Group scopes available in an Active Directory domain include domain local groups, global groups, and universal groups.

DOMAIN LOCAL GROUPS

Domain local groups can have any of the following as members:

- User accounts
- Computer accounts
- Global groups from any domain in the forest
- Universal groups
- Domain local groups from the same domain

You use domain local groups to assign permissions to resources in the same domain as the domain local group. Domain local groups can make permission assignment and maintenance easier to manage.

GLOBAL GROUPS

Global groups can have the following as members:

- User accounts
- Computer accounts
- Other global groups from the same domain

You can use global groups to grant or deny permissions to any resource located in any domain in the forest. You accomplish this by adding the global group as a member of a domain local group that has the desired permissions. Global group memberships are replicated only to domain controllers within the same domain. Users with common resource needs should be members of a global group, to facilitate the assignment of permissions to resources. You can change the membership of the global group as frequently as necessary to provide users with the necessary resource permissions.

UNIVERSAL GROUPS

Universal groups can contain the following members:

- User accounts
- Computer accounts
- Global groups from any domain in the forest
- Other universal groups

If a cross-forest trust exists, universal groups can contain similar accounts from a trusted forest. Universal groups, like global groups, can organize users according to their resource access needs. You can use them to provide access to resources located in any domain in the forest through the use of domain local groups.

You can also use universal groups to consolidate groups and accounts that either span multiple domains or span the entire forest. A key point in the application and utilization of universal groups is that group memberships in universal groups should not change frequently, because universal groups are stored in the global catalog. Changes to universal group membership lists are replicated to all global catalog servers throughout the forest. If these changes occur frequently, the replication process can consume a significant amount of bandwidth, especially on relatively slow and expensive WAN links.

Nesting groups

As discussed earlier, group nesting is the term used when groups are added as members of other groups. For example, when you make a global group a member of a universal group, it is said to be nested within the universal group.

Group nesting reduces the number of times you need to assign permissions to users in different domains in a multidomain forest. For example, if you have multiple child domains in your AD DS hierarchy, and the users in each domain need access to an enterprise database

application located in the parent domain, the simplest way to set up access to this application is as follows:

1. Create global groups in each domain that contain all users needing access to the enterprise database.
2. Create a universal group in the parent domain. Include each location's global group as a member.
3. Add the universal group to the required domain local group to assign the necessary permission to access and use the enterprise database.

This traditional approach to group nesting in AD DS is often referred to using the mnemonic AGUDLP: you add Accounts to Global groups, add those global groups to Universal groups, add universal groups to Domain Local groups, and, finally, assign Permissions to the domain local groups.

This same policy can apply to your administrative model as well. If you look at the Built-in container, you can see how the default domain local groups are based on administrative tasks.

Administrators can use the same method to create their own domain local groups, to which they will delegate administrative tasks and user rights for particular OUs. Then, after creating global groups (or universal groups, for forest-wide assignments), and adding them to the domain local groups, the structure is in place.

Creating groups

The procedure for creating groups in Active Directory Administrative Center or Active Directory Users and Computers is virtually identical to that for creating OUs. When you create a group, you must specify a name for the group object. The name you select can be up to 64 characters long and must be unique in the domain. You must also choose a group type and a group scope. Figure 5-22 shows the Create Group window in Active Directory Administrative Center.

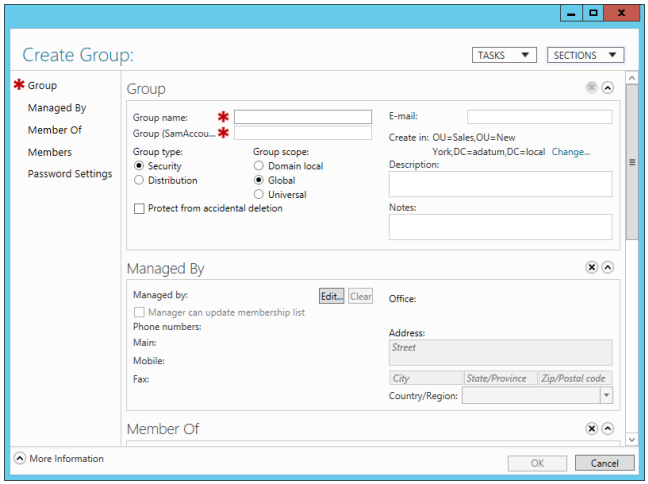

FIGURE 5-22 Creating a group in Active Directory Administrative Center.

The New Object - Group dialog box in Active Directory Users and Computers is slightly different in appearance, but contains the same basic controls.

Although the graphical AD DS utilities are a convenient tool for creating and managing groups individually, they are not the most efficient method for creating large numbers of security principals. The command-line tools included with Windows Server 2012 enable you to create and manage groups in large numbers using batch files or other types of scripts. Some of these tools are discussed in the following sections.

CREATING GROUPS FROM THE COMMAND LINE

You can use the Dsadd.exe tool to create new user objects; you can use the same program to create group objects as well. The basic syntax for creating group objects with Dsadd.exe is as follows:

```
dsadd group <GroupDN> [parameters]
```

The <GroupDN> parameter is a DN for the new group object you want to create. The DNs use the same format as those in CSV files.

By default, Dsadd.exe creates global security groups, but you can use command-line parameters to create groups with other types and scopes, as well as to specify members and memberships for the groups and other group object properties. The most commonly used command-line parameters are as follows:

- **-secgrp yes|no** Specifies whether the program should create a security group (yes) or a distribution group (no). The default value is yes.
- **-scope l|g|u** Specifies whether the program should create a domain local (l), global (g), or universal (u) group. The default value is g.
- **-samid <SAMName>** Specifies the SAM name for the group object.

- **-desc <description>** Specifies a description for the group object.
- **-memberof <GroupDN>** Specifies the DNs of one or more groups of which the new group should be made a member.
- **-member <GroupDN>** Specifies the DNs of one or more objects that should be made members of the new group.

For example, to create a new group called Sales in the Users container and make the Administrator user a member, you would use the following command:

```
dsadd group "CN=Sales,CN=Users,DC=adatum,DC=com" -member
"CN=Administrator,CN=Users,DC=adatum,DC=com"
```

To create a new group object using Windows PowerShell, you use the New-ADGroup cmdlet, with the following syntax:

```
New-ADGroup
-Name <group name>
-SamAccountName <SAM name>
-GroupCategory Distribution|Category
-GroupScope DomainLocal|Global|Universal
-Path <distinguished name>
```

For example, to create a global security group called Sales in the Chicago OU, you would use the following command:

```
New-ADGroup -Name Sales -SamAccountName Sales
-GroupCategory Security -GroupScope Global
-Path "OU=Chicago,DC=Adatum,DC=Com"
```

Managing group memberships

Unlike the Active Directory Administrative Center, which enables you to specify a group's members as you create the group, in Active Directory Users and Computers, you must create the group object first, and then add members to it.

To add members to a group, select it in the console and, from the Action menu, select Properties to open the group's Properties sheet, and then select the Members tab.

Using the Members tab, you can add objects to the group's membership list, and on the Member Of tab, you can add the group to the membership list of another group. For both of these tasks, you use the standard Select Users, Contacts, Computers, Service Accounts, Or Groups dialog box to choose objects.

Once you enter or find the objects you want to add, click OK to close the Properties sheet and add the objects to the group's membership list.

MANAGE GROUP MEMBERSHIP USING GROUP POLICY

It is also possible to control group memberships by using Group Policy. When you create Restricted Groups policies, you can specify the membership for a group and enforce it, so that

no one can add or remove members.

To create Restricted Groups policies, use the following procedure:

1. Log on to the server running Windows Server 2012 using an account with Administrative privileges. The Server Manager window opens.
2. Open the Group Policy Management console, create a new GPO and link it to your domain.
3. Open the GPO in the Group Policy Management Editor and browse to the Computer Configuration\Policies\Windows Settings\Security Settings\Restricted Groups folder, as shown in Figure 5-23.

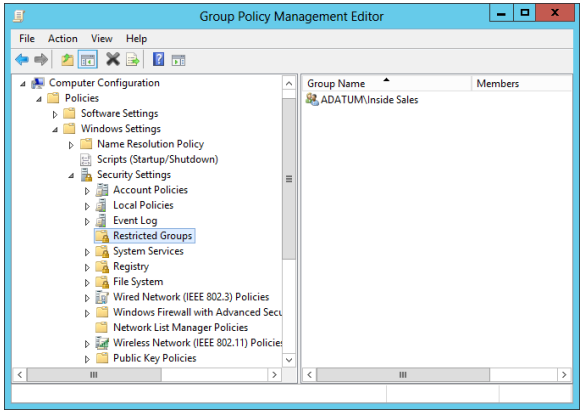

FIGURE 5-23 The Restricted Groups folder in the Group Policy object.

4. Right-click the Restricted Groups folder and from the shortcut menu, select Add Group To open the Add Group dialog box.
5. Type or browse to add a group object and click OK. The group appears in the Restricted Groups folder and a Properties sheet for the policy appears, as shown in Figure 5-24.

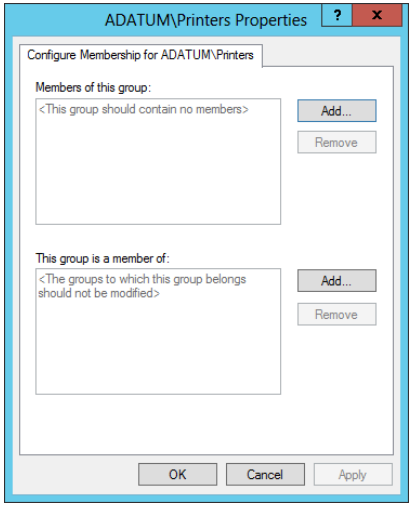

FIGURE 5-24 The Properties sheet for a Restricted Groups policy.

6. Click one or both of the Add buttons to add objects that should be members of the group, or other groups of which the group should be a member.
7. Click OK.
8. Close the Group Policy Management Editor and Group Policy Management consoles.

The members you specify for a group in a Restricted Groups policy are the only members permitted to remain in that group. The policy does not prevent administrators from modifying the group membership using other tools, but the next time the system refreshes its group policy settings, the group membership list will be overwritten by the policy.

MANAGING GROUP OBJECTS WITH DSMOD.EXE

Dsmod.exe enables you to modify the properties of existing group objects from the Windows Server 2012 command prompt. Using this program, you can perform tasks such as adding members to a group, removing them from a group, and changing a group's type and scope. The basic syntax for Dsmod.exe is as follows:

```
dsmod group <GroupDN> [parameters]
```

The most commonly used command-line parameters for Dsmod.exe are as follows:

- **-secgrp yes|no** Sets the group type to security group (yes) or distribution group (no).
- **-scope l|g|u** Sets the group scope to domain local (l), global (g), or universal (u).
- **-addmbr <members>** Adds members to the group. Replace members with the DNs of one or more objects.
- **-rmmbr <members>** Removes members from the group. Replace members with the DNs of one or more objects.

- **-chmbr <members>** Replaces the complete list of group members. Replace members with the DNs of one or more objects.

For example, to add the Administrator user to the Guests group, you would use the following command:

```
dsmod group "CN=Guests,CN=Builtin,DC=adatum,DC=com" –addmbr
"CN=Administrator,CN=Users,DC=adatum,DC=com"
```

Converting groups

As group functions change, you might need to change a group object from one type to another. To change the type of a group, open the group's Properties sheet in the Active Directory Administrative Center or the Active Directory Users and Computers console. On the General tab, you can modify the Group Type option and click OK.

The process for changing the group's scope is exactly the same, except that you select one of the Group Scope options on the General tab. The AD DS utilities only enable you to perform permissible scope changes. Table 5-1 lists the scope changes that are permitted.

TABLE 5-1 Active Directory Group Scope Conversion Restrictions

	TO DOMAIN LOCAL	TO GLOBAL	TO UNIVERSAL
FROM DOMAIN LOCAL	Not applicable	Not permitted	Permitted only when the domain local group does not have other domain local groups as members
FROM GLOBAL	Not permitted	Not applicable	Permitted only when the global group is not a member of another global group
FROM UNIVERSAL	No restrictions	Permitted only when the universal group does not have other universal groups as members	Not applicable

Deleting a group

As with user objects, each group object that you create in AD DS has a unique, nonreusable SID. Windows Server 2012 uses the SID to identify the group and the permissions assigned to it.

When you delete a group, Windows Server 2012 does not use the same SID for that group again, even if you create a new group with the same name as the one you deleted. Therefore, you cannot restore the access permissions you assigned to resources by re-creating a deleted group object. You must add the newly recreated group as a security principal in the resource's access control list (ACL) all over again.

When you delete a group, you delete only the group object and the permissions and rights specifying that group as the security principal. Deleting a group does not delete the objects that are members of the group.

Objective summary

- Once you have created a design for your Active Directory domains and the trees and forests superior to them, it is time to zoom in on each domain and consider the hierarchy you want to create inside it.
- Adding OUs to your Active Directory hierarchy is not as big an issue as adding domains; you don't need additional hardware, and you can easily move or delete an OU at will.
- When you want to grant a collection of users permission to access a network resource, such as a file system share or a printer, you cannot assign permissions to an OU; you must use a security group instead. Although they are container objects, groups are not part of the Active Directory hierarchy in the same way that domains and OUs are.
- There is no simpler object type to create in the AD DS hierarchy than an OU. You only have to supply a name for the object and define its location in the Active Directory tree.
- Creating OUs enables you to implement a decentralized administration model, in which others manage portions of the AD DS hierarchy, without affecting the rest of the structure.
- Groups enable administrators to assign permissions to multiple users simultaneously. A group can be defined as a collection of user or computer accounts that functions as a security principal, in much the same way that a user does.
- In Active Directory, there are two types of groups: security and distribution; there are also three group scopes: domain local, global, and universal.
- Group nesting is the term used when groups are added as members of other groups.
- It is possible to control group memberships by using Group Policy. When you create Restricted Groups policies, you can specify the membership for a group and enforce it, so that no one can add or remove members

Objective review

Answer the following questions to test your knowledge of the information in this objective. You can find the answers to these questions and explanations of why each answer choice is correct or incorrect in the "Answers" section at the end of this chapter.

1. Which of the following groups do you use to consolidate groups and accounts that either span multiple domains or the entire forest?

   A. Global

B. Domain local

C. Built-in

D. Universal

2. Which of the following is not a correct reason for creating an OU?

   A. To create a permanent container that cannot be moved or renamed

   B. To duplicate the divisions in your organization

   C. To delegate administration tasks

   D. To assign different Group Policy settings to a specific group of users or computers

3. Which of the following group scope modifications are not permitted? (Choose all answers that are correct.)

   A. Global to universal

   B. Global to domain local

   C. Universal to global

   D. Domain local to universal

4. In a domain running at the Windows Server 2012 domain functional level, which of the following security principals can members of a global group? (Choose all answers that are correct.)

   A. Users

   B. Computers

   C. Universal groups

   D. Global groups

5. You are attempting to delete a global security group in the Active Directory Users and Computers console, and the console will not let you complete the task. Which of the following could possibly be causes for the failure? (Choose all answers that are correct.)

   A. There are still members in the group.

   B. One of the group's members has the group set as its primary group.

   C. You do not have the proper permissions for the container in which the group is located.

   D. You cannot delete global groups from the Active Directory Users and Computers console.

Answers

Objective 5.1: Review

1. **Correct Answer:** A
   A. **Correct:** In AD DS, you can subdivide a domain into OUs and populate it with objects, but you cannot create domains within OUs.
   B. **Incorrect:** A site can contain multiple domains.
   C. **Incorrect:** A tree can contain multiple domains.
   D. **Incorrect:** A forest can contain multiple domains.
2. **Correct Answers:** B and D
   A. **Incorrect:** There is no object class called resource.
   B. **Correct:** There are two basic classes of objects: container objects and leaf objects. A leaf object cannot have subordinate objects.
   C. **Incorrect:** A domain is a specific object type, not a general classification.
   D. **Correct:** There are two basic classes of objects: container objects and leaf objects. A container object is one that can have other objects subordinate to it.
3. **Correct Answers:** A, B, C
   A. **Correct:** Some attributes are created automatically, whereas administrators must supply information for other attributes manually.
   B. **Correct:** A container object has, as one of its attributes, a list of all the other objects it contains.
   C. **Correct:** Leaf objects have attributes that contain specific information about the specific resource the object represents.
   D. **Incorrect:** Some attributes are created automatically, such as the globally unique identifier (GUID) that the domain controller assigns to each object when it creates it.
4. **Correct Answer:** D
   A. **Incorrect:** Each domain in an Active Directory installation is a separate administrative entity. The more domains you create, the greater the number of ongoing administration tasks you have to perform.
   B. **Incorrect:** Every domain requires its own domain controllers, so each additional domain you create increases the overall hardware and maintenance costs of the deployment.
   C. **Incorrect:** Applications might have problems working in a multidomain forest.
   D. **Correct:** There are no special Microsoft licenses needed for domains.
5. **Correct Answer:** B

A. **Incorrect:** DNS is used for searches within a domain

B. **Correct:** To locate an object in another domain, Active Directory clients perform a search of the global catalog first. This search provides the client with the information it needs to search for the object in the specific domain that contains it.

C. **Incorrect:** DHCP does not provide search capabilities.

D. **Incorrect:** Site link objects do not provide search capabilities.

Objective 5.1: Thought experiment

Robert should install Active Directory on a domain controller in the New York headquarters, creating a forest root domain called hq.inside.litware.com. Because the London office is well connected, but lacks its own IT staff, he can install a read-only domain controller for the hq.inside.litware.com domain there, so that the London users can authenticate using a local domain controller. For the Tokyo office, which is less well connected and has its own IT staff, the design should call for two domain controllers hosting a separate domain in the same forest, called tokyo.inside.litware.com. This will provide the Tokyo users with local domain controller access and minimize the amount of replication traffic passing over the demand-dial link between the New York and Tokyo offices.

Objective 5.2: Review

1. **Correct Answer:** B

   A. **Incorrect:** Dcpromo, now deprecated in Windows Server 2012, is a tool used to promote and demote Active Directory domain controllers.

   B. **Correct:** Like CSVDE.exe, the LDAP Data Interchange Format Directory Exchange (LDIFDE.exe) utility can be used to import or export Active Directory information. It can be used to add, delete, or modify objects in Active Directory, in addition to modifying the schema, if necessary.

   C. **Incorrect:** CSVDE.exe can create Active Directory objects from information in CSV files, but it cannot modify existing objects.

   D. **Incorrect:** NSLOOKUP is a DNS name resolution utility; it cannot create AD DS objects.

2. **Correct Answer:** B

   A. **Incorrect:** The first line of the CSV file is the header record, not the header row.

   B. **Correct:** The CSVDE command-line utility enables an administrator to import or export AD DS objects. It uses a .csv file that is based on a header record, which describes each part of the data. A header record is simply the first line of the text file that uses proper attribute names.

   C. **Incorrect:** The first line of the CSV file is the header record, not the name row.

   D. **Incorrect:** The first line of the CSV file is the header record, not the name record.

3. **Correct Answer:** C
   A. **Incorrect:** You cannot perform an offline domain join using the net join command.
   B. **Incorrect:** You cannot perform an offline domain join using the join command.
   C. **Correct:** You can perform an offline domain join on a computer running Windows Server 2012 using the Djoin.exe utility.
   D. **Incorrect:** You cannot perform an offline domain join using the dconnect command.
4. **Correct Answer:** C
   A. **Incorrect:** Local accounts can be created and configured in Windows Server 2012.
   B. **Incorrect:** Domain accounts can be created and configured in Windows Server 2012.
   C. **Correct:** Three types of user accounts can be created and configured in Windows Server 2012: local accounts, domain accounts, and built-in user accounts.
   D. **Incorrect:** Built-in accounts can be created and configured in Windows Server 2012.
5. **Correct Answers:** C and D
   A. **Incorrect:** There is no Network account in Windows Server 2012.
   B. **Incorrect:** There is no Interactive account in Windows Server 2012.
   C. **Correct:** By default, the two built-in user accounts created on a computer running Windows Server 2012 are the Administrator account and the Guest account.
   D. **Correct:** By default, the two built-in user accounts created on a computer running Windows Server 2012 are the Administrator account and the Guest account.

Objective 5.2: Thought experiment

Correct Answer: D. Answer A is incorrect because the user command is missing and because the user's name is not expressed in distinguished name (DN) format. Answer B is incorrect because the command-line variables containing spaces are not surrounded by quotation marks. Answer C is incorrect because the user command is missing and because the –company and –dept parameters appear before the DN.

Objective 5.3: Review

1. **Correct Answer:** D
   A. **Incorrect:** Global groups cannot contain users from other domains.
   B. **Incorrect:** Domain local groups cannot have permissions for resources in other domains.
   C. **Incorrect:** Built-in groups have no inherent cross-domain qualities.
   D. **Correct:** Universal groups, like global groups, are used to organize users according

to their resource access needs. You can use them to organize users to facilitate access to any resource located in any domain in the forest through the use of domain local groups. Universal groups are used to consolidate groups and accounts that either span multiple domains or the entire forest.

2. **Correct Answer:** A
   A. **Correct:** The reasons for creating an OU include duplicating organizational divisions, assigning Group Policy settings, and delegating administration. You can easily move or rename an OU at will.
   B. **Incorrect:** Duplicating organizational divisions is a viable reason for creating an OU.
   C. **Incorrect:** Delegating administration tasks is a viable reason for creating an OU.
   D. **Incorrect:** Assigning Group Policy settings is a viable reason for creating an OU.
3. **Correct Answers:** B and C
   A. **Incorrect:** Global to universal group conversions are permitted.
   B. **Correct:** Global to domain local group conversions are not permitted.
   C. **Correct:** Universal to global group conversions are not permitted.
   D. **Incorrect:** Domain local to universal group conversions are permitted.
4. **Correct Answers:** A, B, and D
   A. **Correct:** Users can be security principals in a global group.
   B. **Correct:** Computers can be security principals in a global group.
   C. **Incorrect:** Universal groups can be security principals in a global group.
   D. **Correct:** Global group can be security principals in a global group.
5. **Correct Answers:** B and C
   A. **Incorrect:** It is possible to delete a group that has members.
   B. **Correct:** If any member sets the group as its primary group, then the system does not permit the group to be deleted.
   C. **Correct:** You must have the appropriate Active Directory permissions for the container in which the group is located to delete it.
   D. **Incorrect:** It is possible to delete groups using the Active Directory Users and Groups console.