Chapter 5

Microsoft Windows Server 2008: Data Protection

Solutions in this chapter:

- **BitLocker**
- **Active Directory Rights Management Services**
- **Authorization**

- ☑ **Summary**
- ☑ **Solutions Fast Track**
- ☑ **Frequently Asked Questions**

Introduction

Computer and network security is of paramount importance for companies in the global marketplace, and a large percentage of these companies have Microsoft infrastructures in place, including domain controllers (DCs), Exchange servers, and Vista and XP workstations. A Windows server provides a number of useful functions in a company's network infrastructure. In this chapter we explain how BitLocker, Digital Rights Management Services, and authentication can help you secure your data.

BitLocker

Everyone has heard the new reports about laptops being stolen, temporarily misplaced, or lost. The data stored on the hard drive can be retrieved by means other than through the operating system. Things such as bootable CDs or USB keys can be used to bypass the operating system and get directly to the information stored on the physical media without the need to know any passwords. Once the operating system has been bypassed, all the files on the drive can be viewed, edited, or copied. The best safeguard to defend against this security issue is encryption.

BitLocker is Microsoft's answer to providing better security by encrypting the data stored on the drive's operating system volume, and is available only in the Enterprise and Ultimate versions of Vista. This new security feature goes a long way toward helping users and organizations protect their data.

You can set up BitLocker in the following configurations:

- **TPM only** In this configuration, only the hardware microchip is used to protect the data stored on the drive. The Trusted Platform Module (TPM) stores the encryption key and verifies that there have been no changes to the hard drive.
- **TPM and USB flash drive** In this configuration, the TPM will still verify the validity of the hard drive, but in addition, part of the encryption key is stored on the USB flash drive. The USB flash drive is required each time the computer starts.
- **TPM and PIN** This configuration is also a two-layer security approach. After successful verification of the drive, you will be required to enter the correct PIN for the start process to continue.

NOTE

It is important to create a recovery password in case there are any hardware failures that may prevent the system from booting. Things such as motherboard failures and USB flash drive failures, where applicable, will affect the system. If a hardware failure occurs, the only way to recover the data is through the recover mode, and a recovery password is required. There are no other ways to restore the data without the recovery password.

The default configuration for BitLocker is to be used in conjunction with a TPM. The TPM is a hardware microchip embedded into the motherboard that is used to store the encryption keys. This protects the hard drive even if it has been removed from the computer and installed into another computer. You can also use BitLocker on systems that don't have the TPM hardware manufactured on the mother board. You can do this by changing the BitLocker's default configurations with either a Group Policy or a script. When you use BitLocker without a TPM, you must store the key on a USB flash drive and insert the USB flash drive into the computer for the system to boot.

Tools & Traps…

BitLocker Vulnerabilities

BitLocker is a new security feature in Vista. As with all security technology, some people are working on creating vulnerabilities or ways around this security, so you must always be aware that new threats are coming out all the time. Therefore, BitLocker is just another technical challenge to many hackers in the world.

To use a BitLocker-enabled system, the key must be stored in RAM while the system is up and running. Universities have found that when a system is shut down, it's possible to retrieve the key from RAM for up to several minutes, giving a hacker complete control over the entire system and all files

Continued

stored on the drive. The main way to avoid this, of course, is to never leave a system unattended in an unsecured area in the first place. The next step is to completely shut down the system so that the RAM can be allowed to fully discharge.

When Vista is used in a domain environment, it is important for the domain administrators to be able to retrieve the information stored on a system in case of any emergency or other type of event. In a case where a user isn't able to work or is asked to leave the company, the information on the hard drive still needs to be accessed and recoverable. Active Directory domains in Server 2003 and 2008 provide administrators with the safeguard to set up Group Policies and have the BitLocker key backed up and stored in Active Directory on the servers.

The hardware and software requirements for BitLocker are:

- A computer that is capable of running Windows Server 2008
- A Trusted Platform Module version 1.2, enabled in BIOS
- A Trusted Computing Group (TCG)-compliant BIOS.
- Two NTFS disk partitions, one for the system volume and one for the operating system volume

Trusted Platform Modules

Developed by the Trusted Platform Group—an initiative by vendors such as AMD, Hewlett-Packard, IBM, Infineon, Intel, Microsoft, and others—a TPM is a semiconductor built into your computer motherboard. It is capable of generating cryptographic keys, limiting the use of those keys, and generating pseudo-random numbers.

Each TPM has a unique RSA key (the *endorsement key*) burnt into it that cannot be altered. The key is used for data encryption (a process known as *binding*). A TPM also provides facilities for *Secure I/O, Memory curtaining, Remote Attestation*, and *Sealed Storage*. You can secure your TPM module by assigning a TPM owner password.

With secure input and output (which is also known as *trusted path*), it is possible to establish a protected path between the computer user and the software that is running. The protected path prevents the user from capturing or intercepting data sent from the user to the software process, for example playing a media file. The trusted path is implemented in both hardware (TPM) and software and uses checksums for the verification process.

Memory curtaining provides extended memory protection. With memory curtaining, even the operating system does not have full access to the protected memory area.

Remote attestation creates a hashed summary of the hardware and software configuration of a system. This allows changes to the computer to be detected.

Sealed storage protects private information in a manner that the information can be read only on a system with the same configuration. In the preceding example, sealed storage prevents the user from opening the file on a "foreign" media player or computer system. In conjunction, it even prevents the user from making a copy (*memory curtaining*) or capturing the data stream that is sent to the sound system (*secure I/O*).

A Practical Example

You download a music file from an online store. Digital rights management protects the file. All security methods are enforced: the file plays only in media players provided by the publisher (*remote attestation*). The file can be played only on your system (*sealed storage*), and it can neither be copied (*memory curtaining*) nor digitally recorded by the user during playback (*secure I/O*).

The major features of BitLocker are full-volume encryption, checking the integrity of the startup process, recovery mechanisms, remote administration, and a process for securely decommissioning systems.

Full Volume Encryption

Windows BitLocker provides data encryption for volumes on your local hard drive. Unlike Encrypting File System (EFS), BitLocker encrypts all data on a volume—operating system, applications and their data, as well as page and hibernation files. In Windows Server 2008, you can use BitLocker to encrypt the whole *drive*, as compared to Windows Vista where you can encrypt *volumes.* BitLocker operation is transparent to the user and should have a minimal performance impact on well-designed systems. The TPM *endorsement key* is one of the major components in this scenario.

Startup Process Integrity Verification

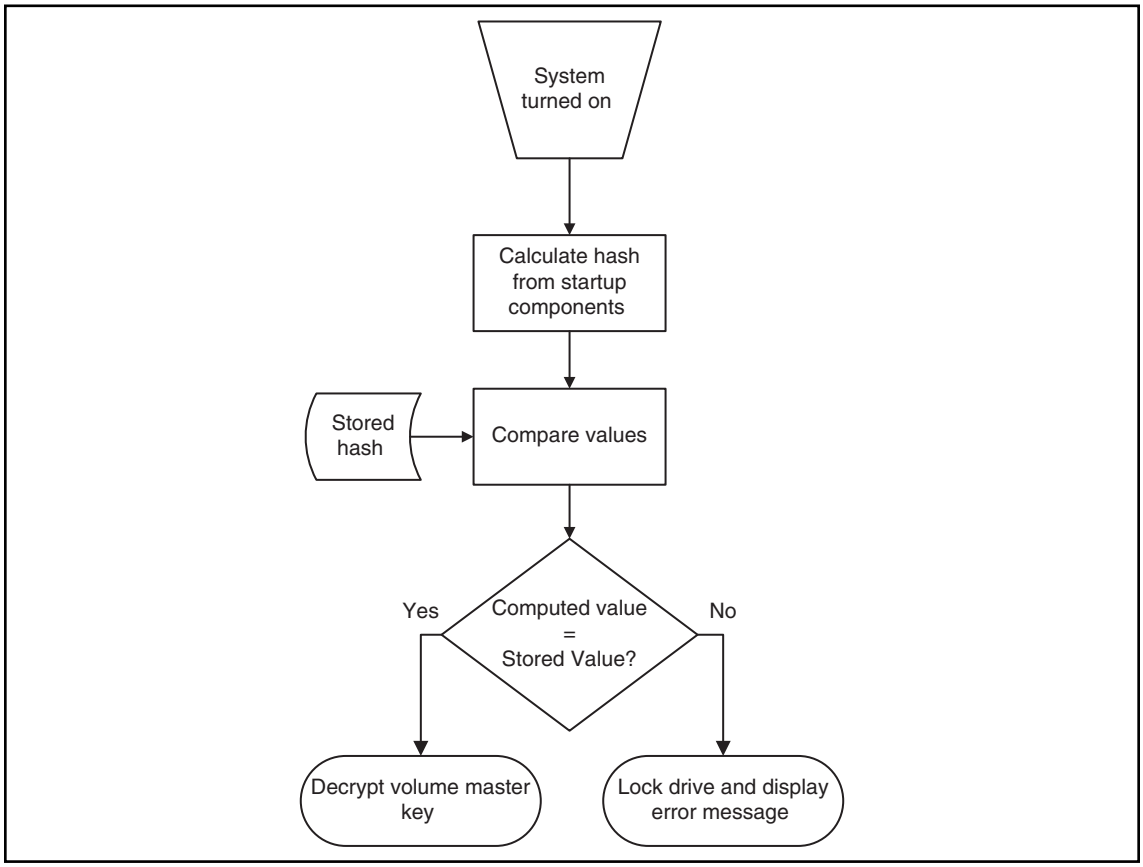
Because Windows Startup components must be unencrypted for the computer to start, an attacker could gain access to these components, change the code, and then gain access to the computer, thereby gaining access to sensitive data such as BitLocker keys or user passwords as a consequence.

To prevent such attacks, BitLocker Integrity checking ensures that startup components (BIOS, Master Boot Record (MBR), boot sector, and boot manager code) have not been changed since the last boot.

Each startup component checks its code each time the computer starts, and calculates a hash value. This hash value is stored in the TPM and cannot be replaced until the next system restart. A combination of these values is also stored.

These values are also used to protect data. For this to work, the TPM creates a key that is bound to these values. The key is encrypted by the TPM (with the endorsement key) and can be decrypted only by the same TPM. During computer startup, the TPM compares the values that have been created by startup components with the values that existed when the key was created (see Figure 5.1). It decrypts the key only if these values match.

Figure 5.1 Startup Component Integrity Verification Flowchart



Recovery Mechanisms

BitLocker includes a comprehensive set of recovery options to make sure data not only is protected, but also available. When BitLocker is enabled, the user is asked for a recovery password. This password must be either printed out, saved to file on a local or network drive, or saved to a USB drive.

In an enterprise environment, however, you would not want to rely on each user to store and protect BitLocker keys. Therefore, you can configure BitLocker to store recovery information in Active Directory. We will cover key recovery using Active Directory later in this chapter.

Remote Administration

Especially in environments with branch offices, it is desirable to have a remote management interface for BitLocker. A WMI script provided by Microsoft allows for BitLocker remote administration and management. You will find the script in the **\Windows\System32** folder after you install BitLocker.

To manage a BitLocker protected system via script:

1. Log on as an administrator.
2. Click **Start**, click **All Programs**, click **Accessories**, and then click **Command Prompt**.
3. At the command prompt type **cd /d C:\Windows\System32**.
4. For example, to view the current status of BitLocker volumes, type **cscript manage-bde.wsf -status**.

Secure Decommissioning

If you decommission or reassign (maybe donate) equipment it might be necessary to delete all confidential data so that it cannot be reused by unauthorized people. Many processes and tools exist to remove confidential data from disk drives. Most of them are very time consuming, costly, or even destroy the hardware.

BitLocker volume encryption makes sure that data on a disk is never stored in a format that can be useful to an attacker, a thief, or even the new owner of the hardware. By destroying all copies of the encryption key it is possible to render the disk permanently inaccessible. The disk itself can then be reused.

There are two scenarios when deleting the encryption key:

- Deleting all key copies from volume metadata, while keeping an archive of it in a secure location such as a USB flash drive or Active Directory. This approach allows you to temporarily decommission hardware. It also enables you to safely transfer or ship a system without the risk of data exposure.
- Deleting all key copies from volume metadata without keeping any archive. Thus, no decryption key exists and the disk can no longer be decrypted.

Notes from the Underground…

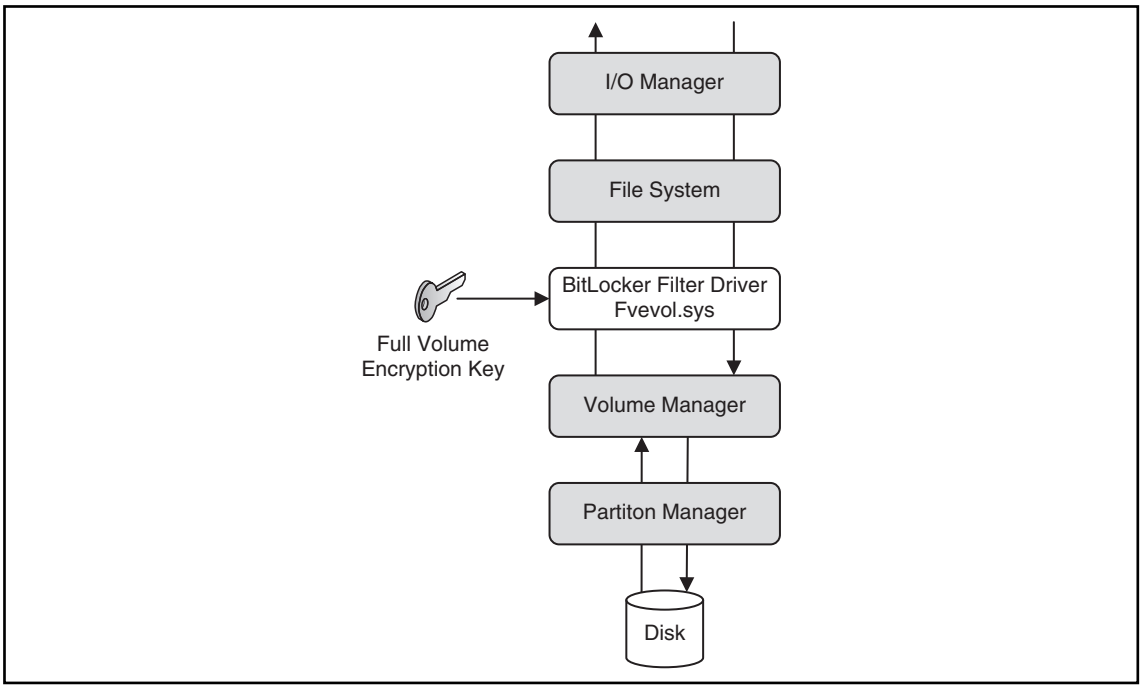
New Group Policy Settings to Support BitLocker

To support centralized administration of BitLocker, Group Policy (GPO) has been extended in Windows Server 2008 Active Directory. The new set of GPO settings allows for configuration of BitLocker as well as TPM. These can be found under Computer Configuration/Administrative Templates/Windows Components/BitLocker Drive Encryption and Computer Configuration/Administrative Templates/System/Trusted Platform Module. To configure these settings, make sure you have at least one Windows Vista or Windows Server 2008 Computer in your Active Directory to create a policy with the new settings available.

BitLocker Architecture

Once Integrity verification is successful, a filter driver encrypts and decrypts disk sectors transparently as data is written or read from the protected volume. The filter driver is a component of Windows Server 2008 or Vista and is inserted into the file system stack during BitLocker installation (see Figure 5.2), thus requiring a system restart. After the initial encryption of the volume is completed, BitLocker operation is completely transparent to the user.

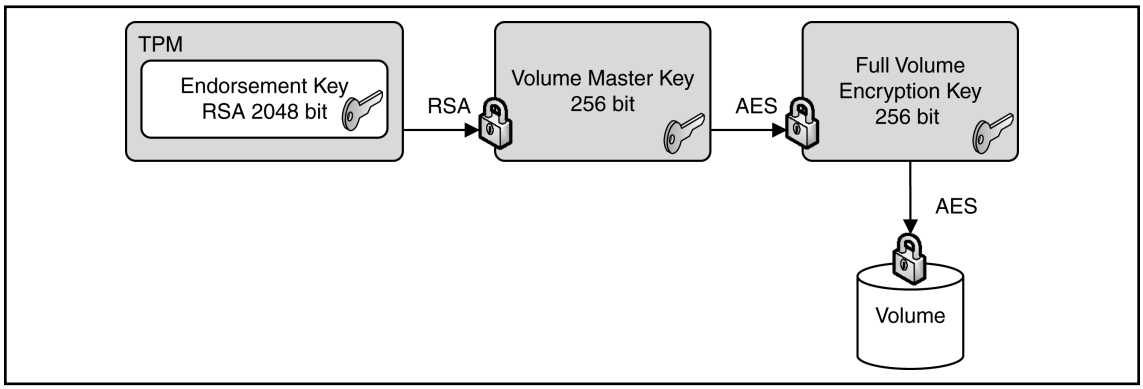
Figure 5.2 Filter Driver Inserted into the File System Stack



Keys Used for Volume Encryption

Volume encryption does not simply create a single key, which it will use to encrypt the volume. In fact, a *full volume encryption key* is used to encrypt the entire volume. This key is a 256-bit Advanced Encryption Standard (AES) key. BitLocker encrypts the full volume key with a *volume master key*. The volume master key is also 256-bit AES. Finally, the volume master key is encrypted with the TPM *endorsement key*. As mentioned before, the endorsement key is a RSA key (see Figure 5.3).

Figure 5.3 Keys Used for Volume Encryption



Notes from the Underground…

New Group Policy Settings to Support BitLocker

Why does BitLocker use a volume master key? Wouldn't it be easier to encrypt the full volume encryption key directly with the TPM endorsement key? At first glance, this would make sense. However, without the volume master key you would have to decrypt and reencrypt the entire volume in case an upstream key is lost or compromised.

Hardware Upgrades on BitLocker Protected Systems

Thanks to the use of *volume master key*, upgrades of hardware such as CPU, motherboard, and such are not very time consuming. To do so you have to disable BitLocker. Disabling BitLocker will *not* decrypt protected volumes. Instead, the volume master

key will be encrypted with a symmetric key, which is stored unencrypted on the hard drive. Moving the disk to another BitLocker-enabled system and activating the volume is possible without any additional steps. Because the encryption key for the volume master key is stored unencrypted on the disk, administrators can boot the system and the reenable BitLocker.

By reenabling BitLocker the unencrypted key is removed from the disk, the volume master key is keyed and encrypted again, and BitLocker is turned back on.

BitLocker Authentication Modes

After Installation BitLocker can be configured to seamlessly integrate into the boot process (TPM only)—therefore being transparent to the user—or can require additional information in the form of a PIN or a startup key to initiate the boot process (TPM with PIN or startup key). The later scenarios add an additional layer of security through the use *multifactor authentication* options. TPM with PIN requires something the user *knows* (e.g., the PIN), TPM with startup key requires something the user *has* (e.g., a USB device).

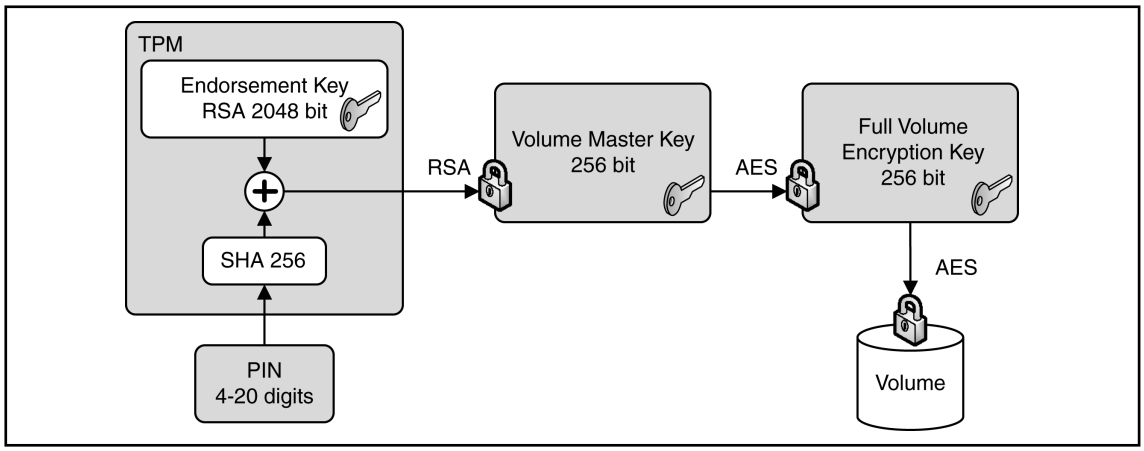
TPM Only

In this scenario, you enable BitLocker with a TPM only. No additional authentication options are used. BitLocker operation is completely transparent to the user and requires no interaction during the boot process.

TPM with PIN Authentication

Using TPM with PIN authentication, the administrator sets up a PIN during BitLocker initialization. The PIN is hashed using SHA-256 and the first 160 bits of the hash are used as authorization data for the TPM. The TPM uses the PIN data to seal the volume master key. Both the TPM and the PIN now protect the volume master key. During system startup or resume from hibernation, the user has to input the PIN to unseal the volume master key and initiate the boot process (see Figure 5.4).

Figure 5.4 Accessing a BitLocker-Enabled Disk That Is Secured with TPM + PIN



TPM with Startup Key Authentication

In this scenario the administrator creates a startup key during BitLocker initialization and stores it on any USB device that can be enumerated by the computer BIOS. During system startup or resume from hibernation, the user must insert the device. The device can be removed after the system has successfully booted.

Startup Key-Only

In this scenario, the administrator enables BitLocker on a computer without a TPM module. The startup key for the computer is generated during initialization and is stored on a USB flash drive. The computer user has to insert the USB flash drive each time the computer starts or resumes from hibernation.

A system configured to use a startup key-only configuration will not provide the same level of security as a system using one of the TPM modes. It will not check the integrity of system startup components. Using this scenario, make sure you create a Backup copy of the startup key! You do this by using the Control Panel BitLocker applet. The system saves the startup key with a .bek extension.

When to Use BitLocker on a Windows 2008 Server

In shared or unsecured environments such as branch offices, BitLocker can provide an additional level of security to a server. By securing the startup process and encrypting the operating system volume and all data volumes, BitLocker protects data from unauthorized access.

The BitLocker feature is not installed by default on Windows Server 2008. You would install it using Server Manager. Setup and maintenance are performed either by GUI tools or from the command line using a script, which also allows for remote management. On Windows Server 2008, BitLocker also integrates with Extensible Firmware Interface (EFI) computers to support IA64 hardware platforms. EFI is a newer, more flexible alternative to classical BIOS implementations. You should not install and enable BitLocker on a Windows Server 2008 Cluster machine, as it is a nonsupported scenario.

Encryption of data volumes on Windows Server 2008 is also supported. Data volumes are encrypted the same way as operating system volumes. Windows Server 2008 will automatically mount and decrypt these volumes on startup when configured to do so.

Support for Multifactor Authentication on Windows Server 2008

Multifactor authentication extends the security of BitLocker protected drives, although there are some constraints that you should think about when you plan to implement it.

PIN Authentication

Although it might not be desirable to use BitLocker with multifactor authentication on a Server, PIN authentication is a supported scenario on Windows Server 2008. If you manage a server remotely and have to reboot, who would enter the PIN?

Of course, there are third-party solutions to overcome this limitation. Most of the modern server boxes offer a built-in remote management solution that is independent of the operating system. For example, Hewlett-Packard offers a so-called Integrated Lights Out (ILO) board to remotely connect to a server and transfer the screen to your desk.

If no remote management solutions were available, another possibility would be to instruct a trustworthy person at the branch office on how and when to enter the pin.

Startup Key Authentication

Of course, startup key support also is built into Windows Server 2008 BitLocker. All the facts mentioned for PIN support apply also to the startup key scenario, plus an additional one: startup keys protect the server only if the key is not left in the server after startup completes. Hence, there must be someone to insert and remove the USB device every time you reboot the server.

Enabling BitLocker

Due to its tight integration into the operating system, enabling BitLocker is straightforward. Before you begin installing and configuring, make sure that the machine you want to secure meets all software and hardware requirements. To enable BitLocker you must be a member of the local administrators group on your computer.

Partitioning Disks for BitLocker Usage

For BitLocker to work your system must have at least two partitions configured. The first, unencrypted partition is the system partition, which contains boot information. The second partition is the boot volume, which is encrypted and contains the operating system. Both partitions must be created before you install the operating system.

If you forgot to partition your system accordingly, there's no way of reconfiguring your partitions (see Figure 5.5). Therefore, you must repartition your hard disk and reinstall the operating system from scratch.

Figure 5.5 BitLocker Refuses to Configure the System Due to an Invalid Partition Scheme

Creating Partitions for a Bitlocker Installation

In this section we'll show you how to create partitions for a Bitlocker installation.

1. Start the computer from the Windows Server 2008 Product DVD.
2. In the Install Windows screen, choose your **Installation language**, **Time** and **currency format and Keyboard layout**, and then click **Next**.
3. In the **Install Windows** screen, click **Repair your Computer**.
4. In the **System Recovery Options** dialog box, make sure no operating system is selected. Then click **Next**.
5. In the **System Recovery Options** dialog box, click **Command Prompt**.
6. At the command prompt type **Diskpart** and then type **Enter**.
7. Type **select disk 0**.
8. Type **clean** to erase all existing partitions.
9. Type **create partition primary size=1500**. This will create a primary partition with a size of 1.5 GB.
10. Type **assign letter=B** to give this partition drive letter B.
11. Type **activate** to set the partition as the active partition.
12. Type **create partition primary** to create a partition with the remaining space. Windows Server 2008 will be installed on this partition.
13. Type **assign letter=c**.
14. Type **list volume** to see a display of all the volumes on this disk.
15. Type **exit**.
16. Type **format c: /y /f /fs:ntfs** to format the C volume.
17. Type **format b: /y /f /fs:ntfs** to format the B volume.
18. Type **exit**.
19. Close the **System Recovery Options** window by clicking the close window icon in the upper right (do not click Shut Down or Restart).
20. Click **Install now** to install Windows Server 2008. Use the larger partition for installation.

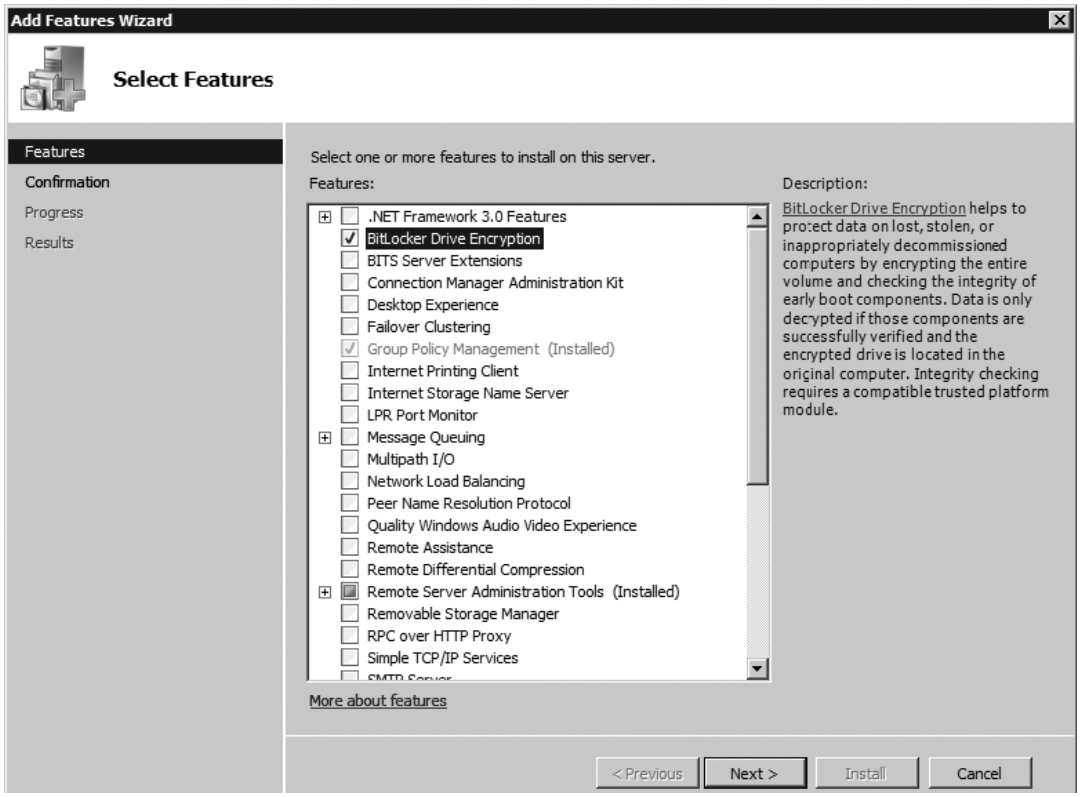
Installing BitLocker on Windows Server 2008

As we already mentioned, BitLocker is a *Feature* of Windows Server 2008 and is not installed by default. To install BitLocker you use Server Manager as you would with all other roles and features. Be aware that a restart is required after installation. You can also install BitLocker from the command line by typing **ServerManagerCmd -install BitLocker –restart**.

Here are the steps to follow to install Bitlocker on Windows Server 2008.

1. Log on as an administrator.
2. Click **Start | Administrative Tools | Server Manager**.
3. Scroll down to **Feature Summary**; click **Add Features**.
4. On the **Select Features** page, choose **BitLocker Drive Encryption** (see Figure 5.6), and then click **Next**.

Figure 5.6 Selecting the BitLocker Feature in Server Manager



5. On the **Confirm Installation Selections** page, click **Install**.
6. When installation is complete, click **Close**.
7. In the **Do you want to restart** Window click **Yes**.

NOTE

Before you start with BitLocker configuration, make sure that you open Server Manager (in case you selected the **Do not show me this console at next logon** checkbox) and let the Post-Install wizard finish the installation.

Turning on and Configuring BitLocker

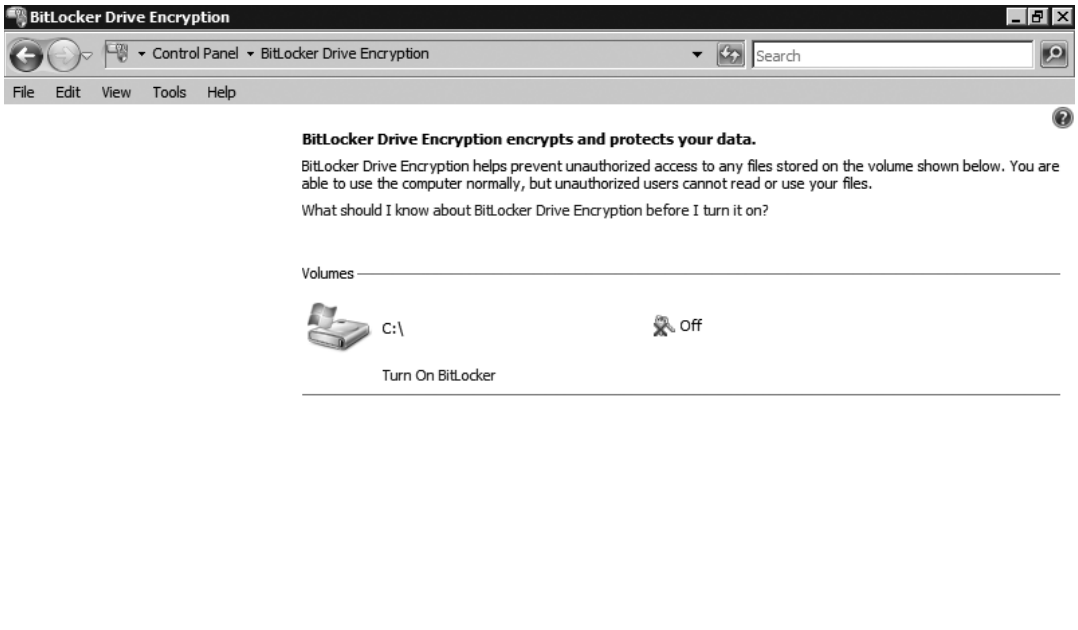
After installing the BitLocker Feature on your Server and rebooting the system, you need to turn on BitLocker via a Control Panel applet. Make sure you are logged on as an administrator on the system and you have decided where to store the recovery password. In case your computer does not have a TPM module or the TPM module is not supported, you will receive a warning (see Figure 5.7).

Figure 5.7 Warning That a TPM Is Missing or Incompatible

Here are the steps to follow for turning on BitLocker.

1. Log on as an administrator.
2. Click **Start**, click **Control Panel**, and then click **BitLocker Drive Encryption**.
3. On the **BitLocker Drive Encryption** page, click **Turn On BitLocker** on the operating system volume (see Figure 5.8).

Figure 5.8 The Server Is Ready to Turn on BitLocker



4. On the **BitLocker Drive Encryption Platform Check** dialog box click **Continue with BitLocker Drive Encryption**.
5. If your TPM is not initialized already, you will see the **Initialize TPM Security Hardware** screen.
6. On the **Save the recovery password** page, click **Save the password on a USB drive** (see Figure 5.9).

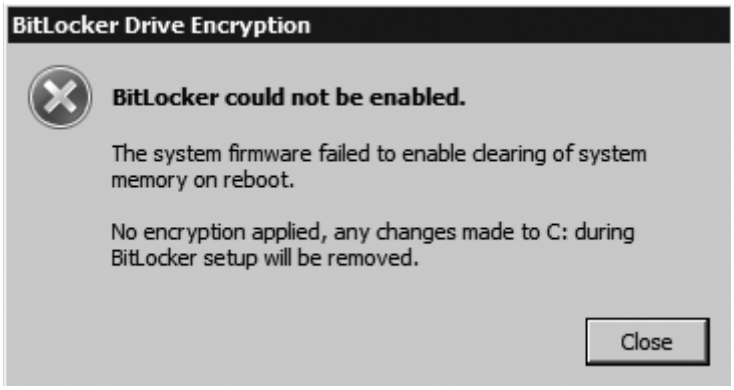
Figure 5.9 Saving the BitLocker Password

7. On the **Save a Recovery Password to a USB Drive** box, select your USB drive and click **Save**.
8. On the **Encrypt the selected disk volume** page, confirm that the **Run BitLocker System Check** checkbox is selected, and then click **Continue**.
9. Confirm that you want to reboot.

During the reboot phase, BitLocker verifies the system and makes sure it is ready for encryption. After rebooting the system, you should log back on to the system and

verify that the **Encryption in Progress** status bar is displayed in the BitLocker Control Panel applet. In case your system cannot be enabled for BitLocker, an error message pops up during logon (see Figure 5.10).

Figure 5.10 Error Enabling BitLocker



TEST DAY TIP

If you do not have a TPM module in your computer or are using virtual machines, you will not be able to configure BitLocker as described in Exercise 6.3. Alternatively, you can continue with Exercise 6.5, which first enables BitLocker operation without a TPM and then continues with the configuration.

Turning on Bitlocker for Data Volumes

Now we'll show you how to turn on BitLocker for data volumes.

1. Log on as an administrator.
2. Click **Start**, click **All Programs**, click **Accessories**, and then click **Command Prompt**.
3. At the command prompt type **manage-bde –on <volume>: -rp –rk F:**. This will encrypt the named volume, generate a recovery password, and store a recovery key on drive F:\ (which is the USB drive, in this example). Don't forget to record the recovery password!
4. At the command prompt type **manage-bde –autounlock –enable <volume>:** to enable automatic unlocking of the volume. The key to

automatically unlock the volume on each restart is stored on the operating system volume, which must be fully encrypted before this command is issued.

NOTE

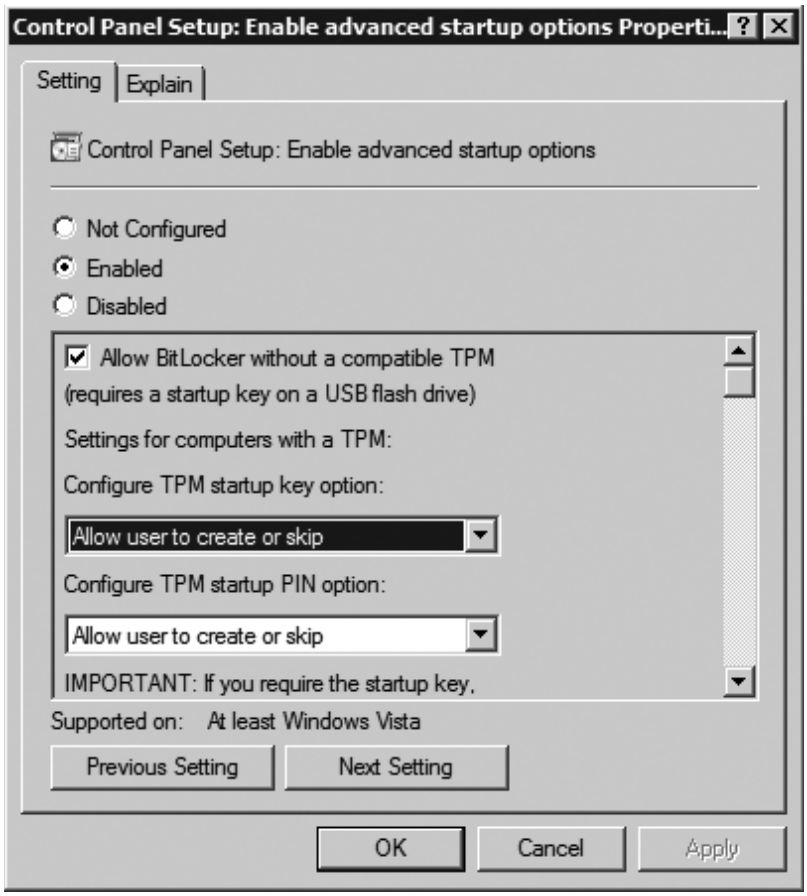
Windows Server 2008 mounts a protected data volume as normal. The keys for protecting a data volume are independent of the keys used to protect the operating system volume. The key-chain protecting the data volume is also stored on the encrypted boot volume, therefore allowing the boot volume to automatically mount any data volume after system restart.

Configuring BitLocker for TPM-Less Operation

The following steps configure your computer's Group Policy settings to turn on BitLocker on systems without a TPM.

1. Logon as an administrator.
2. Click **Start**, click **Run**, type **gpedit.msc** in the open box, and then click **OK**.
3. In the **Local Group Policy Editor** console tree, click **Local Computer Policy**, click **Administrative Templates**, click **Windows Components**, and then click **BitLocker Drive Encryption**.
4. Double-click the setting **Control Panel Setup: Enable Advanced Startup Options**.
5. Select the **Enabled** option, select the **Allow BitLocker without a compatible TPM** check box, and then click **OK** (see Figure 5.11).

Figure 5.11 Enabling TPM-less Operation in the Local Group Policy



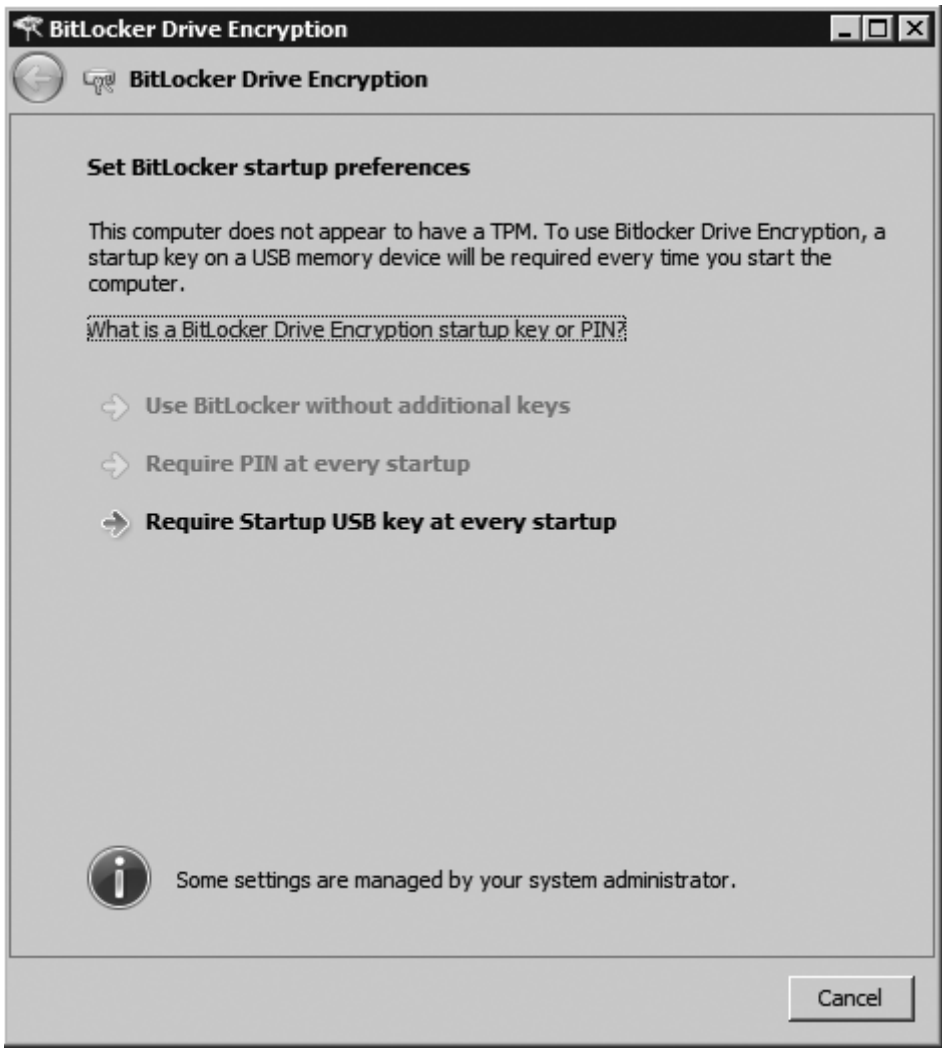
Turning on BitLocker on Systems without a TPM

Turning on BitLocker on systems without a TPM is similar to the normal activation process. Make sure you have a USB flash drive available to store the startup key.

1. Log on as an administrator.
2. Click **Start**, click **Control Panel**, and then click **BitLocker Drive Encryption**.
3. On the **BitLocker Drive Encryption** page, click **Turn On BitLocker** on the operating system volume.
4. On the **BitLocker Drive Encryption Platform Check** dialog box click **Continue with BitLocker Drive Encryption**.

5. On the **Set BitLocker startup preferences** page select **Require Startup USB key at every startup** (see Figure 5.12).

Figure 5.12 USB Startup Key Selection Screen



6. On the **Save your Startup Key** page select your USB drive from the list and click **Next**.
7. On the **Save the recovery password** page, click **Save the password on a USB drive**.
8. On the **Save a Recovery Password to a USB Drive** Box, select your USB drive and click **Save**.

9. On the **Encrypt the selected disk volume** page, confirm that the **Run BitLocker System Check** checkbox is selected, and then click **Continue**.
10. Confirm that you want to reboot.

Administration of BitLocker

In a managed Enterprise environment, it can be problematic to allow each user to enable BitLocker by themselves. Not only do you have to add the user to the local administrators group, you also give out the management of recovery passwords and/or PINs and startup keys. In the real world, users forget their passwords and PINs. So why should this be different with BitLocker recovery information? Here's an example: A user with a laptop decides to use BitLocker to make sure the data is secure even when the laptop is stolen. After enabling BitLocker, the user puts the recovery password printout into the laptop bag... A security nightmare!

One method to act upon such deficiencies is to educate users and increase their awareness so that they get more sensitive for security-related matters. Another approach might be technical. Windows Server 2008 extends well-known techniques and tools to give the administrator control over the BitLocker lifecycle. Group Policies settings were added to control the behavior of BitLocker on client and server systems. Furthermore, the Windows Management Instrumentation (WMI) Interface for BitLocker allows for local and remote management of BitLocker. We will talk about the possibilities of WMI later in this chapter.

Using Group Policy with BitLocker

Group Policy (GPO) in Windows Server 2008 has been extended to provide BitLocker-specific configuration settings. With GPO, the administrator can control BitLocker installation and configuration as well as centralized storage of recovery passwords. Table 5.1 lists Windows Server 2008's Group Policy settings for BitLocker.

Table 5.1 Overview of Windows Server 2008 BitLocker Group Policy Settings

Policy	Policy Path	Scope	Description
Configure encryption method	Windows Components\ BitLocker Drive Encryption	Machine	This policy setting allows you to configure the algorithm and key size used by BitLocker Drive Encryption.
Configure TPM platform validation profile	Windows Components\ BitLocker Drive Encryption	Machine	This policy setting allows you to configure how the computer's Trusted Platform Module (TPM) security hardware secures the BitLocker encryption key. This policy setting does not apply if the computer does not have a compatible TPM or if BitLocker has already been turned on with TPM protection.
Control Panel Setup: Configure recovery folder	Windows Components\ BitLocker Drive Encryption	Machine	This policy setting allows you to specify the default path that is displayed when the BitLocker Drive Encryption setup wizard prompts the user to enter the location of a folder in which to save the recovery password.
Control Panel Setup: Configure recovery options	Windows Components\ BitLocker Drive Encryption	Machine	This policy setting allows you to configure whether the BitLocker Drive Encryption setup wizard will ask the user to save BitLocker recovery options.
Control Panel Setup: Enable advanced startup options	Windows Components\ BitLocker Drive Encryption	Machine	This policy setting allows you to configure whether the BitLocker Drive Encryption setup wizard will ask the user to set up an additional authentication that is requested each time the computer starts. You can further configure setting options for computers with and without a TPM.

Continued

Table 5.1 Continued. Overview of Windows Server 2008 BitLocker Group Policy Settings

Policy	Policy Path	Scope	Description
Prevent memory overwrite on restart	Windows Components\ BitLocker Drive Encryption	Machine	This policy setting controls computer restart performance at the risk of exposing BitLocker secrets. BitLocker secrets include key material used to encrypt data.
Turn on BitLocker backup to Active Directory Domain Services	Windows Components\ BitLocker Drive Encryption	Machine	This policy setting allows you to manage the Active Directory Domain Services (AD DS) backup of BitLocker Drive Encryption recovery information.

Storing BitLocker and TPM Recovery Information in Active Directory

In conjunction with Group Policy and a downloadable toolkit, Active Directory can be configured to store backup information for Windows BitLocker and the Trusted Platform Module. Recovery information includes the recovery password, the TPM owner password, and the information required to identify to which computers and volumes the recovery information applies. Optionally, you can also save a package containing the actual keys used to encrypt the data as well as the recovery password required to access those keys.

As a best practice, configure Active Directory integration first and then allow BitLocker usage on clients and servers. If you enable BitLocker on clients first, recovery passwords for those computers are not stored in Active Directory, leading to an inconsistent experience in case you have to recover.

Storage of BitLocker Recovery Information in Active Directory

BitLocker recovery information is stored in Active Directory as a child object to the computer object. That is, the computer object acts as the parent container for a recovery object. Each BitLocker object includes the recovery password as well as other recovery information. Multiple recovery objects can exist under each computer account because there can be more than one recovery password for each protected volume.

BitLocker recovery information is stored in objects from type *msFVE-RecoveryInformation*. These objects are named after the following scheme:

```
<Object Creation Date and Time><Recovery GUID>
```

For example:

```
2008-01-30T08:17:05-09:00{063DC7a8-879D-DE34-FF6F-2417448D55CB}
```

Each msFVE-RecoveryInformation object contains the attributes listed in Table 5.2.

Table 5.2 Attributes Associated with the msFVW-RecoveryInformation Objects

Attribute Name	Description
ms-FVE-RecoveryPassword	Contains the 48-digit recovery password
ms-FVE-RecoveryGuid	Contains the GUID associated with a BitLocker recovery password
ms-FVE-VolumeGuid	Contains the GUID associated with a BitLocker-supported disk volume
ms-FVE-KeyPackage	Contains a volume's BitLocker encryption key

Storage of TPM Information in Active Directory

TPM owner passwords are stored as an attribute of the computer object in Active Directory. During TPM initialization or when the TPM password is changed, the hash of the password is stored in Active Directory in the *ms-TPM-OwnerInformation*.

Prerequisites

Since BitLocker Active Directory backup stores information in Active Directory objects, you need to extend the schema to support the storage of BitLocker-specific data. Schema extensions and scripts for enabling the Active Directory backup functionality are included in a downloadable toolkit from Microsoft. To access the download follow this link: http://go.microsoft.com/fwlink/?LinkId=78953. After extraction, the following sample scripts should help with the implementation:

- Add-TPMSelfWriteACE.vbs
- BitLockerTPMSchemaExtension.ldf
- List-ACEs.vbs
- Get-TPMOwnerInfo.vbs
- Get-BitLockerRecoveryInfo.vbs

NOTE

BitLocker recovery information is stored in Active Directory attributes flagged as confidential. The confidential flag is a feature introduced in Windows Server 2003 Service Pack1 and provides advanced access control for sensitive data. With this feature, only domain administrators and authorized users have read access to those attributes. Therefore Active Directory backup for BitLocker recovery information should be implemented only if your domain controllers are running Windows Server 2003 Service Pack 1, Windows Server 2003 R2, or Windows Server 2008, ensuring backed up BitLocker information is properly protected.

Extending the Schema

The first step in configuring Active Directory BitLocker backup is extending the Active Directory schema to allow storage of BitLocker specific objects (see Figure 5.13). Before you start, extract the toolkit files to a folder named **C:\BitLocker-AD**.

To extend the Active Directory schema:

1. Logon with an account that is a member of the schema admins group.
2. Click **Start**, click **All Programs**, click **Accessories**, and then click **Command Prompt**.
3. At the command prompt, type **cd /d C:\BitLocker-AD**.
4. At the command prompt, type **ldifde -i -v -f BitLockerTPMSchema Extension.ldf -c "DC=X" "distinguished name of your domain" -k -j.** Do not forget the period at the end of the command!

Figure 5.13 Schema Extension Output

```
C:\BitLocker-AD>ldifde -i -v -k -f BitLockerTPMSchemaExtension.ldf -c
"DC=X" "DC=nsoftincad,dc=internal" -j .
Connecting to "Alpha.Nsoftincad.Internal"
Logging in as current user using SSPI
Importing directory from file "BitLockerTPMSchemaExtension.ldf"
Loading entries
1: CN=ms-TPM-OwnerInformation,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry already exists, entry skipped
```

```
2: CN=ms-FVE-RecoveryGuid,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry already exists, entry skipped
3: CN=ms-FVE-RecoveryPassword,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry already exists, entry skipped
4: (null)
Entry modified successfully.
5: CN=ms-FVE-RecoveryInformation,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry already exists, entry skipped
6: CN=computer,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry modified successfully.
7: (null)
Entry modified successfully.
8: CN=ms-FVE-VolumeGuid,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry already exists, entry skipped
9: CN=ms-FVE-KeyPackage,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry already exists, entry skipped
10: (null)
Entry modified successfully.
11: CN=ms-FVE-RecoveryInformation,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry modified successfully.
12: CN=ms-FVE-RecoveryInformation,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Attribute or value exists, entry skipped.
13: CN=ms-TPM-OwnerInformation,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry modified successfully.
14: CN=ms-TPM-OwnerInformation,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry modified successfully.
15: CN=ms-FVE-RecoveryGuid,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry modified successfully.
16: CN=ms-FVE-RecoveryGuid,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry modified successfully.
17: CN=ms-FVE-RecoveryGuid,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry modified successfully.
18: CN=ms-FVE-RecoveryGuid,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry modified successfully.
19: CN=ms-FVE-RecoveryPassword,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry modified successfully.
```

```
20: CN=ms-FVE-RecoveryPassword,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry modified successfully.
21: CN=ms-FVE-RecoveryPassword,CN=Schema,CN=Configuration,DC=nsoftincad,dc=internal
Entry modified successfully.
22: (null)
Entry modified successfully.
15 entries modified successfully.
The command has completed successfully
```

Setting Required Permissions for Backing Up TPM Passwords

The second step is to set permission in Active Directory. By default Windows Vista clients can back up BitLocker recovery information in Active Directory. However, to back up the TPM owner password an Access Control Entry (ACE) must be added to the computer object. To add the ACE use the **Add-TPMSelfWriteACE.vbs** script from the toolkit. To add the ACE entry:

1. Log on with a domain administrator account.
2. Click **Start**, click **All Programs**, click **Accessories**, and then click **Command Prompt**.
3. At the command prompt type **cscript Add-TPMSelfWriteACE.vbs**.

The script will add a single ACE to the top-level domain object in your domain. The ACE is inherited by all computer child objects in Active Directory.

Enabling Group Policy Settings for BitLocker and TPM Active Directory Backup

Here are the steps to follow to configure Group Policies for clients and servers to use BitLocker Active Directory Backup.

1. Log on with a domain administrator to any Domain Controller.
2. Click **Start**, click **All Programs**, click **Administrative Tools**, and then click **Group Policy Management**.
3. In the Group Policy Management Console, expand the forest tree down to the domain level.
4. Right-click the **Default Domain Policy** and select **Edit**.

5. In the Group Policy Management Editor, open **Computer Configuration**, open **Administrative Templates**, open **Windows Components**, and then open **BitLocker Drive Encryption**.
6. In the right pane, double-click **Turn on BitLocker backup to Active Directory**.
7. Select the **Enabled** option, select **Require BitLocker backup to AD DS**, and click **OK**.

To further enable storage of TPM recovery information:

8. Open **Computer Configuration**, open **Administrative Templates**, open **System**, and then open **Trusted Platform Module Services**.
9. In the right pane, double-click **Turn on TPM backup to Active Directory**.
10. Select the **Enabled** option, select **Require TPM backup to AD DS**, and click OK.

WARNING

In this example, we use the *Default Domain Policy* to configure Active Directory backup for BitLocker and TPM recovery information. However, in a real-world scenario you would create a new GPO that contains only BitLocker specific settings!

Recovering Data

BitLocker will lock the computer when an encryption key is not available. Likely causes for this can be:

- Inserting the BitLocker-protected drive into a new computer
- Replacing the computer motherboard
- Performing maintenance operation on the TPM (such as clearing or disabling)
- Updating the BIOS
- Upgrading critical early boot components that cause system integrity validation to fail

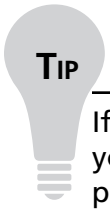
- Forgetting the PIN when PIN authentication has been enabled
- Losing the USB flash drive containing the startup key when startup key authentication has been enabled

When TPM fails to check the integrity of startup components, it will lock the computer at a very early stage before the operating system starts. When locked, the system enters recovery mode. You can use a USB flash drive with the recovery password stored on it or use the keyboard to enter the recovery password manually. In recovery mode, the keyboard assignment is somewhat different: you use functions keys to enter digits. F1 through F9 represents digits 1 trough 9, F10 represents 0.

Testing Bitlocker Data Recovery

To test BitLocker for data recovery, follow these steps:

1. Log on as an administrator.
2. Click **Start**, click **Run**, type **tpm.msc** in the open box, and click **OK**. The **TPM Management Console** is displayed.
3. Under **Actions**, click **Turn TPM Off**.
4. Provide the TPM owner password, if required.
5. When the **Status** panel in the **TPM Management on Local Computer** task panel reads "Your TPM is off and ownership of the TPM has been taken," close that task panel.
6. Click the **Safely Remove Hardware** icon in the notification area to remove the USB flash drive from the system.
7. **Restart** your computer. When you restart the computer, you will be prompted for the recovery password, because the startup configuration has changed since you encrypted the volume.
8. The **BitLocker Drive Encryption Recovery Console** should appear.
9. **Insert** your USB flash drive and press **ESC**. The computer will restart automatically.
10. The system should boot normally.



TIP

If you do not have a USB flash drive with the recovery password on it, you would press **ENTER** instead of ESC. After pressing **ENTER**, the system prompts you for the recovery password. Input the recovery password and press **ENTER** again.

Disabling BitLocker

If you want to turn off BitLocker, you need to decide if you want to disable BitLocker or decrypt the volume. Disabling BitLocker allows for TPM maintenance while the data is kept encrypted. Decrypting the volume means that the entire volume will be decrypted. Disabling BitLocker is supported only on operating system volumes and not on data volumes.

To turn off BitLocker Drive Encryption:

1. Click **Start**, click **Control Panel**, click **Security**, and then click **BitLocker Drive Encryption**.
2. On the **BitLocker Drive Encryption** page, find the volume on which you want BitLocker Drive Encryption turned off, and click **Turn Off BitLocker Drive Encryption**.
3. From the **What level of decryption do you want** dialog box, click either **Disable BitLocker Drive Encryption** or **Decrypt the volume** as needed.

Active Directory Rights Management Services

Active Directory Rights Management Services (AD RMS) is a format- and application-agnostic service designed to safeguard information by deterring inadvertent sharing of information with unauthorized people. AD RMS protects information when it is connected and when it is not connected to the corporate network. A usage policy is bound to the protected item so that no matter where it travels the rights are enforced to ensure that only the authorized recipient is able to access the contents. The policy can restrict users from actions such as viewing, copying, forwarding, and printing.

Previously shipped as an add-on for Windows Server, AD RMS is now included out-of-the-box as a role in Windows Server 2008. This release delivers a number of enhancements focused on easing administration and opening up cross-organization collaboration.

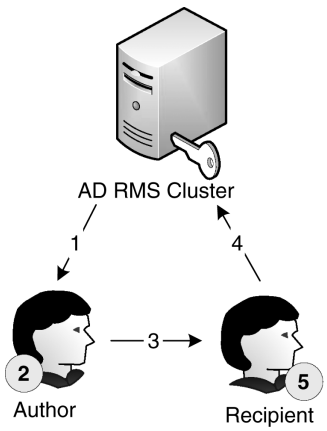
Active Directory Rights Management Services (AD RMS) includes features not available in Microsoft Windows RMS. Windows RMS was available for Windows Server 2003 and was used to restrict access to rights-protected content to files made by RMS-enabled applications. The added features were incorporated to ease administrative overhead of AD RMS and to extend use outside the organization. New features include:

- AD RMS is now a server role
- Microsoft Management Console (MMC) snap-in
- Integration with AD FS
- Self-enrollment of AD RMS servers
- The ability to delegate responsibility with new AD RMS administrative roles

As shown in Figure 5.14, AD RMS works through a service cluster providing license issuing and validation services to a group of users. As a user accesses AD RMS for the first time through an RMS-enabled application, his machine is enrolled with the RMS cluster and is issued a Client Licensor Certificate (CLC). This certificate is a key component in the protection process. It enables the user to publish content with a specific usage policy assigned. The usage policy is derived from several elements:

- **Actions** Explicitly allowed, denied, or undefined actions that include a default set (view, edit, save, export, print, forward, reply, reply all, extract, allow macros, view rights) and the ability to define new application-specific actions
- **Expiration Policy** Disable the content after a specific date, a duration following the content license being applied, or a duration following the initial opening of the document (akin to a "self-destruct" option)
- **Revocation Policy** Requiring the content to check a revocation list each time the content is accessed to ensure that the user's right to access the content has not been explicitly revoked
- **Extended Policy** Miscellaneous settings including granting the user the ability to view using a browser-based RMS viewer, forcing the user to obtain a new license every time she accesses the content, and adding additional custom attributes

Figure 5.14 AD RMS



1. Author receives a Client Licensor Certificate
2. Author create a file and defines a usage policy, RMS create a Publishing License and encrypts the file
3. Author distributes the file to the Recipient using any method
4. Recipient opens the file and the RMS client contacts the RMS server for a Use License
5. Application opens the file using the rights defined in the Publishing License

Users can create their own custom policy based on a combination of the above as exposed by the application, or they can use policies defined at the organization level. The organization-defined policies, known as *policy templates*, provide a basis for implementing uniform policies across a large number of users (e.g., "Confidential Company Information Read Only", "For Research and Development Teams Only"). The policy syntax is based on Extensible Rights Markup Language (XrML). It allows third-party developers to RMS-enable their application and extend the AD RMS service to meet their information protection needs. With the policy created the information can be distributed by any means necessary to the authorized recipients.

When the recipients receive the information their RMS-enabled application applies for a Use License. The first time the machine accesses the RMS service a Machine Certificate (MC) is issued to the computer. The RMS client then validates the viewer's identity by creating a Rights Account Certificate (RAC) for the user. With the MC and RAC the RMS client evaluates the usage policy. If everything checks out okay, the user is issued a Use License to access the content and the application enables that access. Microsoft ships two main sets of RMS-enabled applications today in the Office suite and Internet Explorer. For Office, you require the Professional edition or better to author protected content, whereas lower editions are only able to view the content. Internet Explorer acts as a viewer for RMS-protected content through an ActiveX plug-in.

You can deploy AD RMS in either a stand-alone server or a clustered configuration. This gives you the flexibility to get started with a basic configuration and scale up to handle a larger volume of usage or implement redundancy as needed.

Before you install AD RMS on Windows Server 2008, you will need to have the .NET Framework 3.0 installed. Because of this dependency, you cannot install AD RMS on a Server Core installation of Windows Server 2008.

Managing Trust Policies

When you set up AD RMS it will trust your organization's Active Directory domain by default. Depending on your business requirements you can expand or contract the boundaries of your RMS trust to specific user domains within your organization or other organizations. You can expand outside of your organization by trusting other AD RMS clusters, AD FS, as well as Windows Live ID accounts. The approach you take will depend on the types of trusts you require.

Issuing Use Licenses to users who are members of another AD RMS cluster requires that you explicitly trust the other cluster. You do this by adding the external cluster's information into the Trusted User Domain (TUD) section of the Trust Policy. Once you've added it, you can allow or deny specific users or groups of users within the external AD RMS cluster.

Follow along as we extend the Trust Policy to another AD RMS cluster.

1. Open the **Control Panel**, and under **System and Maintenance | Administration Tools** double-click the **Active Directory Rights Management Services** shortcut.
2. In the **Active Directory Rights Management Services** management console, expand the cluster node and **Trust Policie**s node in the left-hand pane, and then click the **Trusted User Domains** node.
3. In the right-hand action pane, click the **Import Trusted User Domain** link.
4. In the **Import Trusted User Domain** dialog, provide the location of the **Trusted User Domain File** given to you by the administrator of the other AD RMS cluster, provide a **Display Name**, and click **Finish**.

To minimize the administrative burden for a small or diverse group of accounts you can use the Windows Live ID service as a source of RACs for users. Before you can do this, you will need to configure your AD RMS cluster to trust the Windows Live ID service. In preparation for this, be sure to enable anonymous access and expose the AD RMS licensing Web service (located at /_wmcs/licensing on your

Web server) for external users to obtain use licenses. Now, we will extend the Trust Policy to Windows Live ID.

1. Open the **Control Panel**, and under System and Maintenance | Administration Tools double-click the **Active Directory Rights Management Services** shortcut.
2. In the **Active Directory Rights Management Services** management console, expand the cluster node and **Trust Policie**s node in the left-hand pane, and then click the **Trusted User Domains** node.
3. In the right-hand action pane, click the **Trust Windows Live ID** link.

AD RMS can also issue Publishing Licenses to users in other AD RMS clusters. This is useful if a trusted external cluster belongs to separate business units within your organization where a fully federated trust cannot be established. This process adds the external cluster's Server Licensor Certificate (SLC) to the Trusted Publishing Domain (TPD) list.

Now let's go through the steps for extending the trust policy to allow external users to receive publishing licenses.

1. Open the **Control Panel**, and under **System and Maintenance | Administration Tools** double-click the **Active Directory Rights Management Services** shortcut.
2. In the **Active Directory Rights Management Services** management console, expand the cluster node and **Trust Policie**s node in the left-hand pane, and then click the **Trusted Publishing Domains** node.
3. In the right-hand action pane, click the **Import Trusted Publishing Domain** link.
4. In the **Import Trusted Publishing Domain** dialog, provide the location of the **Trusted Publishing Domain File** given to you by the administrator of the other AD RMS cluster, provide the **Password** for the file and a **Display Name**, and click **Finish**.

Federated trusts are an alternative to adding trusted organizations to both TUD and TPD lists. It is useful when you have a trusted partner organization with which you are working and sharing information. To protect both parties these trusts are not transitive, meaning that the TUD and TPD lists of one organization do not automatically

apply to the other organization. The trust is established at the highest level only. Now, we'll go through the steps to establish a federated trust.

1. Open the **Control Panel**, and under **System and Maintenance | Administration Tools** double-click the **Active Directory Rights Management Services** shortcut.
2. In the **Active Directory Rights Management Services** management console, expand the cluster node and **Trust Policies** node in the left-hand pane, and then click the **Federated Identity Support** node.
3. In the right-hand action pane, click the **Enable Federated Identity Support** link.
4. In the right-hand actions pane. click **Properties**.
5. In the **Federated Identity Support** dialog, on the **Policies** tab, provide the **Federated Identity Certificate Service URL** for the external AD RMS cluster that will be trusted and click **OK**.

Exclusion Policies

In addition to including organizations, you can exclude certain users based on e-mail domains, specific addresses, applications, RMS client version, and Windows operating system version.

When using the Windows Live ID trust you can exclude specific users from obtaining a use certificate by adding them to the exclusion list. This could be useful if there are known users who present a security risk to corporate information. Now, we'll walk through the steps of excluding Windows Live IDs.

1. Open the **Control Panel**, and under **System and Maintenance | Administration Tools** double-click the **Active Directory Rights Management Services** shortcut.
2. In the **Active Directory Rights Management Services** management console, expand the cluster node and **Trust Policie**s node in the left-hand pane, and then click the **Trusted User Domains** node.
3. In the middle pane, right-click the **Windows Live ID** row and select **Properties**.
4. In the **Windows Live ID Properties** dialog, click the **Excluded Windows Live IDs** tab, enter the e-mail addresses or domains which you want to exclude, and click **OK**.

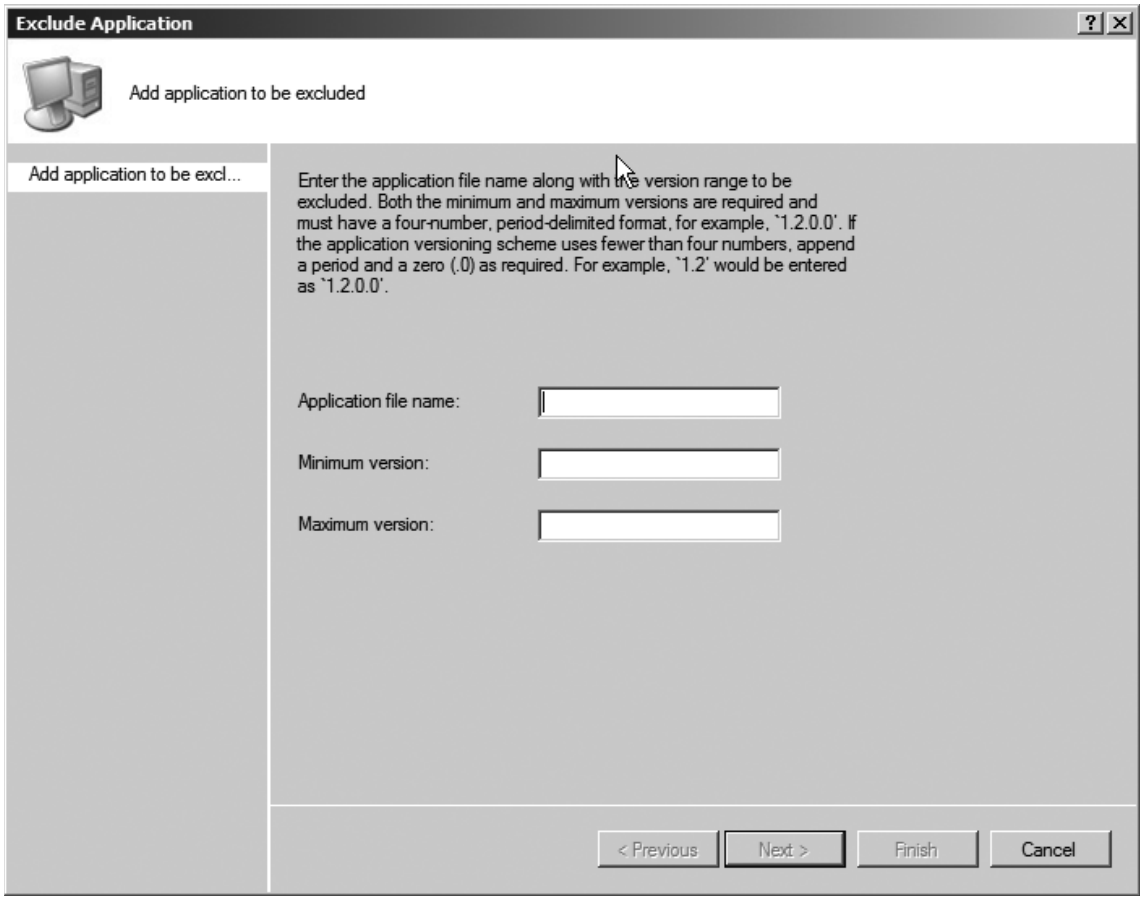
You can also exclude specific user accounts from your AD RMS cluster or other trusted clusters using the e-mail address or public key from the user's RAC. By doing this, you will prevent the user from obtaining a new Use License from your AD RMS cluster. Note that this exclusion does not apply to other AD RMS clusters that trust your users.

1. Open the **Control Panel**, and under **System and Maintenance | Administration Tools** double-click the **Active Directory Rights Management Services** shortcut.
2. In the **Active Directory Rights Management Services** management console, expand the cluster node and **Exclusion Policie**s node in the left-hand pane, and then click the **Users** node.
3. In the right action pane, click the **Enable User Exclusion** link.
4. In the right action pane, click the **Exclude User** link.
5. In the **Exclude User** dialog, provide the e-mail address of the user or the public key string and click **Finish**.

If you have an RMS client application which you no longer trust, either because an updated version is available or known defects in the application make it a risky application, you can prevent users from using that application for protected content. This policy will prevent AD RMS from issuing a new Use License to clients who are using the specified version of the software. As with the user exclusion, this will apply only to your AD RMS cluster. Now, we'll walk through the steps of excluding applications.

1. Open the **Control Panel**, and under **System and Maintenance | Administration Tools** double-click the **Active Directory Rights Management Services** shortcut.
2. In the **Active Directory Rights Management Services** management console, expand the cluster node and **Exclusion Policie**s node in the left-hand pane, and then click the **Applications** node.
3. In the right action pane, click the **Enable Application Exclusion** link.
4. In the right action pane, click the **Exclude Application** link.
5. In the **Exclude Application** dialog (see Figure 5.15), provide the filename and version range to be excluded and click **Finish**.

Figure 5.15 The Exclude Application Dialog



You can perform two other types of exclusions with AD RMS: client and operating system version exclusions. The lockbox is the RMS client component that stores a user's private key. With continued security research there is a possibility that vulnerabilities are found in a particular release of the RMS client. To mitigate the risks associated with older versions you can specify the minimum RMS client lockbox component version for which Use Licenses will be issued by the AD RMS cluster. Microsoft posts the latest released version of this component on its Web site located at http://go.microsoft.com/fwlink/?LinkID=12995.

Finally, you can restrict older versions of the RMS client running on either Microsoft Windows 98 Second Edition or Windows Millennium Edition. These operating systems do not support a number of critical security features available in later releases. Restricting them from accessing RMS content will ensure that your content is protected using the

best measures available. As with all other exclusions, you will need to enable the Windows Version exclusion on each individual AD RMS cluster.

Configuring Policy Templates

Rights policy templates provide a set of predefined rules for users to leverage in their RMS-enabled applications when making decisions regarding how to protect information. These templates form a basis for implementing uniform policies across a large number of users (e.g., "Confidential Company Information Read Only", "For Research and Development Teams Only"). They provide administrators the ability to define the information sharing parameters, and later revoke the content as a whole when the template is deleted. Let's now create a policy template.

1. Open the **Control Panel**, and under **System and Maintenance | Administration Tools** double-click the **Active Directory Rights Management Services** shortcut.
2. In the **Active Directory Rights Management Services** management console, expand the cluster node in the left-hand pane, and then click the **Rights Policy Template** node.
3. In the right action pane, click the **Create Distributed Rights Policy Template** link.
4. In the **Create Distributed Rights Policy Template** dialog, on the **Add Template Identification Information** step, click the **Add** button, provide the **Name** and **Description** of the template, and click **Next**.
5. On the **Add User Rights** step, click the **Add** button and type in the e-mail address of the user or group, or select **Anyone** to apply this policy to everyone. Then select the **Rights** from the list and click **Next**.
6. On the **Specify Expiration Policy** step, set the appropriate **Content** and **Use License** expiration and click **Next**.
7. On the **Specify Extended Policy** step, review the options and click **Next**.
8. On the **Specify Revocation Policy** step, review the options and click **Finish**.

When you are finished with a policy template it is recommended that you archive the template instead of deleting it. This will allow AD RMS to continue to issue Use Licenses for content protected with the particular template. When you do

finally delete a policy template it is recommended that you back up the configuration database before doing so to enable you to recover rights-protected content if necessary. We'll now archive a policy template.

1. Open the **Control Panel**, and under **System and Maintenance | Administration Tools** double-click the **Active Directory Rights Management Services** shortcut.
2. In the **Active Directory Rights Management Services** management console, expand the cluster node in the left-hand pane, and then click the **Rights Policy Template** node.
3. In the middle pane, select the policy to be archived.
4. In the right action pane, click the **Archive this Rights Policy Template** link.
5. On the warning dialog, click **Yes**.

Managing Your AD RMS Cluster

Now let's discuss ways to manage your AD RMS cluster.

Super User

The Super User group is an administrative group whose members can decrypt any protected content, and subsequently remove the content protection from the file. By default, this group is disabled and contains no members. To enable it you will need to assign an Active Directory Universal Group to represent the AD RMS super group. Here are the steps to follow:

1. Open the **Control Panel**, and under **System and Maintenance | Administration Tools** double-click the **Active Directory Rights Management Services** shortcut.
2. In the **Active Directory Rights Management Services** management console, expand the cluster node and **Security Policie**s node in the left-hand pane, and then click the **Super Users** node.
3. In the right-hand action pane, click the **Enable Super Users** link.
4. In the middle pane, click the **Change Super User Group** link.

5. In the **Super Users** dialog, click the **Browse** button and locate a Universal Group in Active Directory to represent the **Super Users Group**; then click **OK**.

Removing AD RMS

With usage over time, AD RMS has become a critical component of your business infrastructure. A number of items have been protected and removing AD RMS from the environment may cause those items to become inaccessible. To prevent the loss of information you should properly decommission the AD RMS environment. This changes the behavior of the AD RMS cluster to provide a decryption key for all rights-protected content which had been published using its licenses. This will give your organization and its users a chance to save their content without the content protection features and the system administrators a chance to remove all AD RMS clients from the environment. Upon decommissioning an AD RMS cluster, you will no longer be able to administer the environment. Ensure that you have adequately backed up the system before performing this step. Now, we'll walk through the steps of decommissioning a server.

WARNING

Removing AD RMS without first decommissioning it will leave all protected content inaccessible for any scenario that requires a new Use License.

1. Open **Windows Explorer** and locate the **Decommissioning.asmx** file (typically found in %SYSTEMDRIVE%\inetpub\wwwroot_wmcs\decommission); then grant the Everyone group Read & Execute permissions.
2. Open the **Control Panel**, and under **System and Maintenance** | **Administration Tools** double-click the **Active Directory Rights Management Services** shortcut.
3. In the **Active Directory Rights Management Services** management console, expand the cluster node and **Security Policie**s node in the left-hand pane, and then click the **Decommissioning** node.

4. In the right-hand action pane, click the **Enable Decommissioning** link.
5. In the middle pane, click the **Decommission** button.
6. When prompted with a warning about decommissioning, click **Yes**.
7. Repeat these steps for the rest of the servers in the AD RMS cluster.

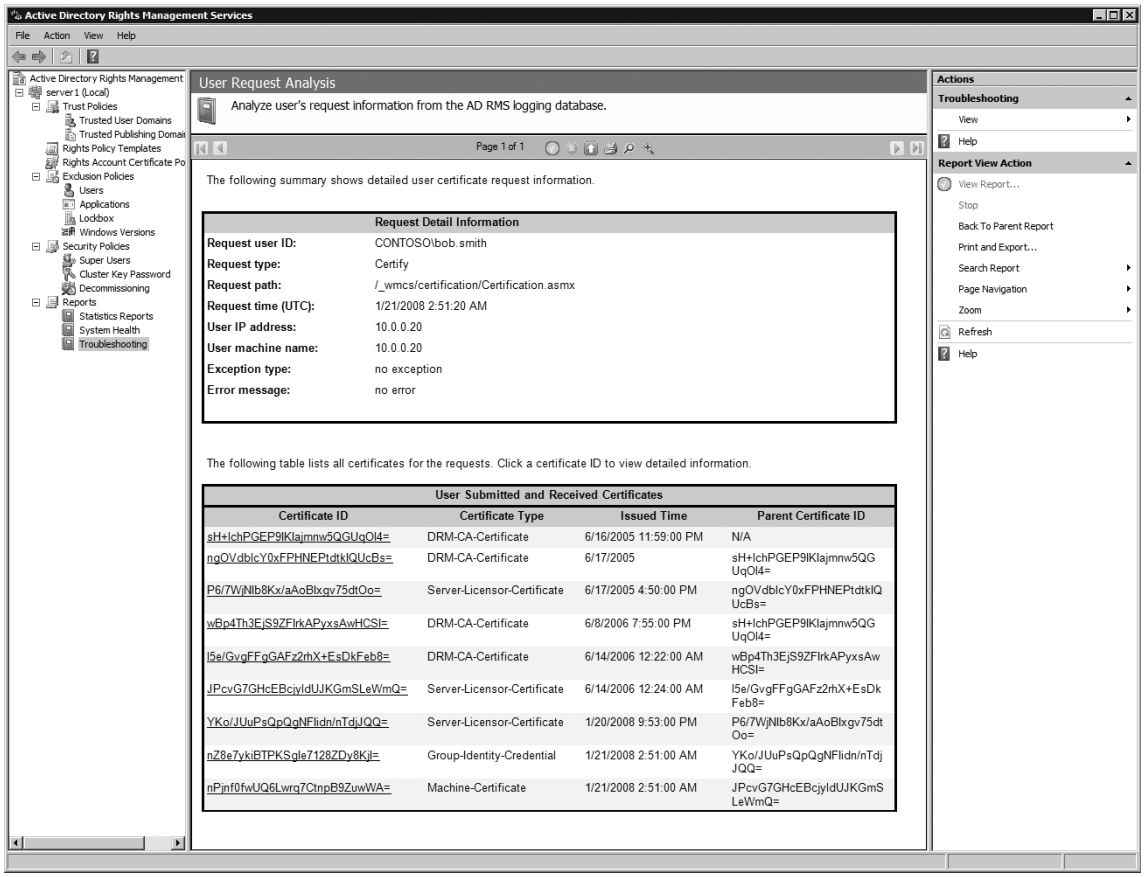
Reporting

AD RMS provides some basic usage reporting that will give you a view into who is using the rights protection services in your organization. There are three main reports:

- **Statistics Report** This report provides the number of RACs issued by the AD RMS cluster. It is mainly used for licensing purposes.
- **System Health** This provides two views—request type and performance—of the activity on a system:
  - **Request Type Summary** Outlines the number of success, failure, and total requests by request type, including a drill-down to the specific user
  - **Request Performance Report** Provides an average duration and total number of requests by type
- **Troubleshooting Report** This displays the number of success, failure, and total requests by request type for a specific user. The report is useful for determining which server responded and the details behind the request and response.

These reports are available in the AD RMS management console under Reports (see Figure 5.16).

Figure 5.16 The User Request Analysis Report



In the last three releases it would be hard to dismiss the incredible growth and maturing of the Windows Server Web application services offerings. From what was an add-on option pack item to a key component that businesses have come to rely on, you can bet that this release is nothing short of impressive. While carrying on the mandate to ship a secure, scalable solution for Web applications and services, the product group has managed to deliver an impressive foundation for Web-based solutions.

Table 5.3 is an overview of the security features available across Windows Server 2008, both Full and Server Core installations.

Table 5.3 Security Features Available for Windows Server 2008

Feature	Available on Full Install	Available on Server Core Install
Basic Authentication	Yes	Yes
Windows Authentication	Yes	Yes
Digest Authentication	Yes	Yes
Client Certificate Mapping Authentication	Yes	Yes
IIS Client Certificate Mapping Authentication	Yes	Yes
Uniform Resource Location (URL) Authorization	Yes	Yes
Request Filtering	Yes	Yes

Protecting your Web application may require one or more tactics to ensure that the application is accessed only by authorized users:

- **Transport Security** Focused on privacy of data being transmitted between the user and the server
- **Authentication** Provides a method for determining the user's identity
- **Authorization** Evaluates a set of rules to determine if the user is allowed to make the request

This section will take you further into each tactic and the details behind them. There have been few key changes that support more secure communication, authentication, and authorization:

- **IIS_IUSRS Group** Replaces the IIS_WPG group from previous releases to service as a security group to which permissions are assigned that will be required by all the application pool identities.
- **Built-in IUSR Account** Replaces the IUSR_MachineName from previous releases with a built-in account that uses a constant security identifier (SID) across servers that helps to maintain consistent access control lists (ACL). Use of the built-in account eliminates the need to have a password assigned to this account as well. For IIS installations on domain controllers

it will prevent the IUSR account from becoming a user-accessible domain account.

- **Inheritance and Merging of IP Restriction Rules** Allows more flexible ways to apply authorization rules based on a single computer, group of computers, a domain, all IP addresses, and/or any unlisted entries.
- **Request Filtering** The URLScan tool, which previously shipped as an add-on tool, is now incorporated in the HTTP protocol handler.
- **Native URL Authorization** A more efficient, globally accessible way to secure specific files and paths without having to rely on third-party tools or ASP.NET.

Transport Security

Protecting the privacy of the data being transmitted is the primary focus of transport security. There are a number of options within the Windows Server 2008 infrastructure to protect the privacy. You may want to wrap all data being transmitted, for example, through a virtual private network or IPSec tunnel. With this as the extreme at one end, IIS provides a more moderate and widely used method for protecting data using Secure Socket Layers (SSL) and Transport Layer Security (TLS). TLS is the more commonly deployed standard today and provides the ability to fall back to SSL 3.0 if the client does not support TLS. SSL/TLS uses digital certificates to encrypt the communication. At a high level the process works as follows:

1. The client makes a request to the Web server for a secure connection.
2. The server sends back its public encryption key.
3. The client checks the key to ensure:
   - The name of the host being requested matches the key.
   - The key is within the valid date range.
   - The key's issuer is trusted by the client.
4. If the client determines that it can trust the server's public key it will send its public key to the server.
5. The server will generate a password and encrypt it using both the client's public key and the server's private key, and send it back to the client.
6. The client will decrypt the password as evidence that the server is the one who sent the password, thereby establishing that only the server and the

client will be the only other party capable of reading the encrypted information.

7. The client will send the request to the server encrypted with the password that the server sent to it.

This process has been well established for quite some time and works with all major browsers. IIS fully supports using SSL/TLS certificates to encrypt communication between the server and users. Under the covers, IIS 7 now handles SSL/TLS requests in the kernel by default (it was available in IIS 6, but not enabled by default). This provides a big boost to the performance of secure requests.

Notes from the Underground…

Host Headers and SSL

As mentioned earlier in the chapter, host headers enable you to share an IP address among multiple sites. A call to www.contoso.com will result in Host: www.contoso.com:80 being passed in the header of the request. This allows the HTTP protocol handler to hand the request off to the appropriate Web site. For connections that use secure socket layer (SSL) the ability to use host headers was first introduced in Windows Server 2003 Service Pack 1.

Before you get too excited there are some restrictions that you will need to take into account. The first is that the SSL certificate must contain all the common names of the sites. For example, if you are binding www.contoso.com and store.contoso.com to the same IP address, your SSL certificate will need to contain both host names in the common name field. The most secure approach is to use multiple common names using the subjectAltName property, but it is also the most difficult to obtain as it is not commonly available through certificate authorities (CA). Most certificate authorities promote the use of wildcard certificates instead. A wildcard certificate enables you to use the certificate for all subdomains (e.g., *.contoso.com would work for www.contoso.com, store.contoso.com, foo.contoso.com, bar.contoso.com, foo.bar.contoso.com). Consult your preferred certificate authority on the cost of a wildcard or subjectAltName certificate as they are not usually supported by the typical offering.

With your new certificate in hand you need to bind the certificate to a Web site. Under the covers IIS does not bind it to the Web site, but the IP address

Continued

being used. The reason for this is simple; the HTTP header value that contains the host name is encrypted at the time that the HTTP protocol handler needs to make the decision of which certificate to use. This means that you can have only one SSL certificate per IP address and that explains why you need a wildcard certificate or one with the subjectAltName properties included. To see a list of certificates and their corresponding IP address bindings use the following NetSh command:

```
NetSh.exe HTTP Show SSLCert
```

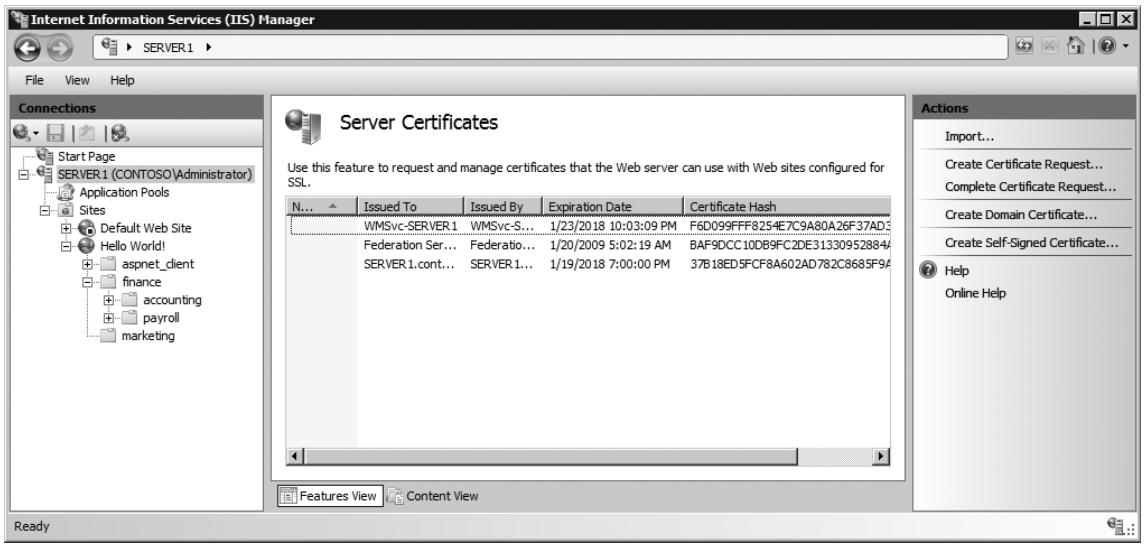
Adding an SSL binding with host header support currently is not supported through the graphical user interface. You will need to use the AppCmd tool, programmatically, or edit the ApplicationHost.config to add the binding. Here is the AppCmd syntax for adding the binding:

```
AppCmd.exe Set Site /Site.Name:"Contoso Store" /+Bindings.[Protocol="HTTPS",
BindingInformation="*:443:store.contoso.com"]
```

With that in place you can now access both of your sites using SSL.

IIS 7 also introduces a new management interface for security certificates. This new interface gives you a single point to review all the certificates installed on your server along with exposing the ability to generate a self-signed certificate from within the interface. Previously self-signed certificates were available only through the command-line SelfSSL tool that shipped with the IIS 6.0 Resource Kit tools (see Figure 5.17).

Figure 5.17 Server Certificates Module Configuration

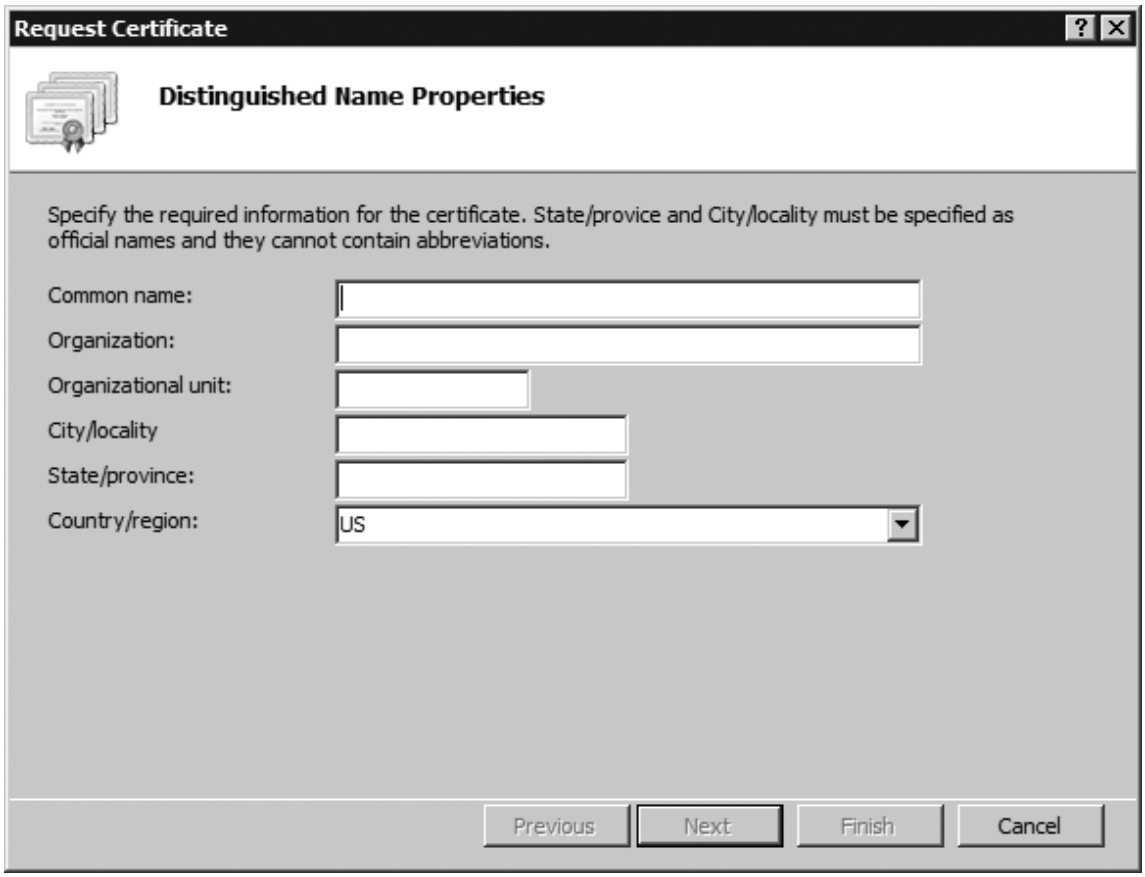


The first step to enabling a secure site is to import or create a new certificate into the server. When creating a certificate you can create one from an online connected certificate authority (CA) like the Certificate Services role that ships with Windows Server 2008, a third-party CA (e.g., Comodo, Thwarte, Verisign), or generate a self-signed certificate. Whichever path you choose the one thing to remember is that the client will need to trust the certificate's issuer in order to trust the certificate. When using a self-signed certificate no one will trust it unless they take steps to specifically add it to their trusted certificates list.

Adding a New Security Certificate

1. Open **Control Panel** and under **System and Maintenance | Administration Tools**, double-click the **Internet Information Services (IIS) Manager** shortcut.
2. In the **Internet Information Services (IIS) Manager** management console click the server node, in the middle pane click **Server Certificates**.
3. In the right-hand **Actions** pane click **Create Certificate Request**.
4. In the **Request Certificate** dialog on the **Distinguished Name Properties** page (see Figure 5.18) provide the host name that will be used to access your site (e.g., www.contoso.com) along with your company information and click **Next**.

Figure 5.18 Distinguished Name Properties Page

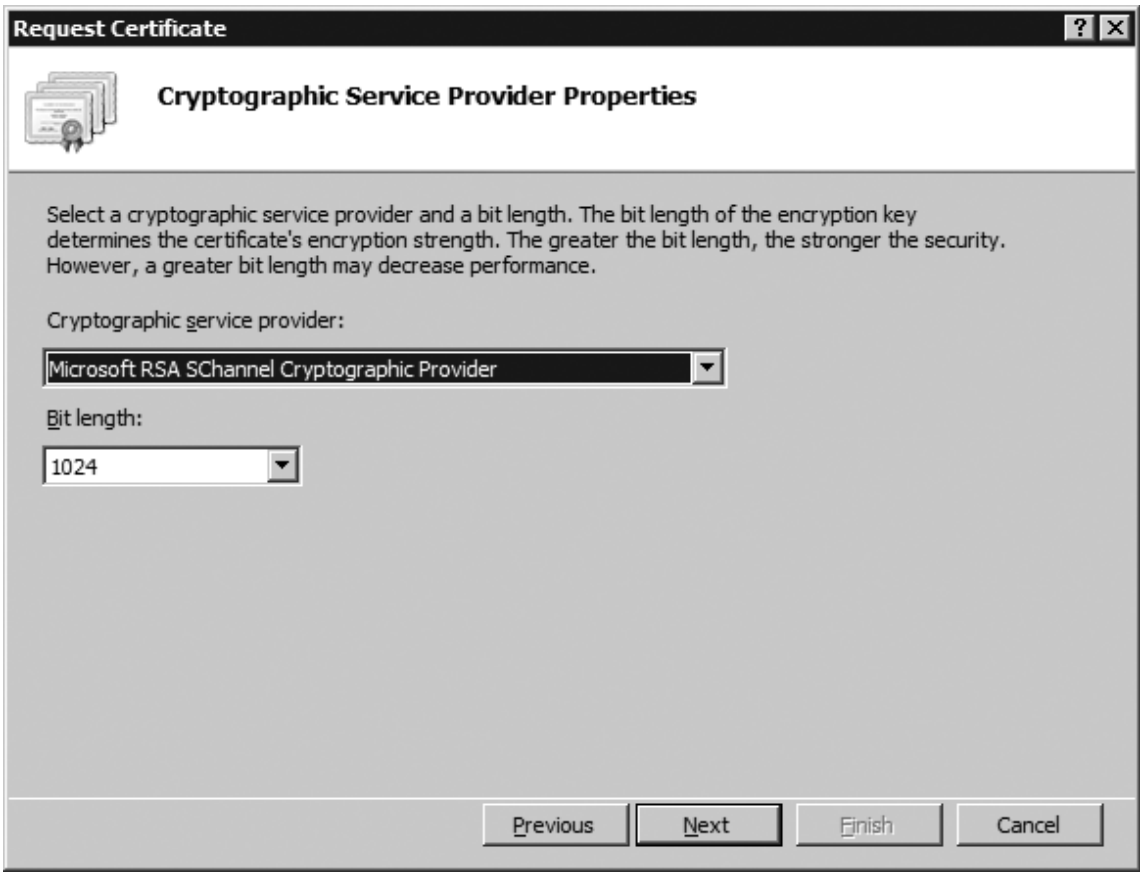


5. On the **Cryptographic Service Provider Properties** page choose a **Cryptographic Server Provider**, a minimum of 1,024 **Bit Length** for the key, and click **Next** (see Figure 5.19).

   - **RSA SChannel Cryptographic Provider** Uses an MD5 hash with an SHA hash, signed with an RSA private key. It supports SSL2, PCT1, SSL3, and TLS1 protocols.
   - **DH SChannel Cryptographic Provider** Uses the Diffie-Hellman algorithm and supports SSL3 and TLS1 protocols. Use this algorithm when you must exchange a secret key over an insecure network without prior communication with the client.
   - **Bit Length** The default length supported by most browsers and certificate authorities is 1,024 bits. With processors becoming more powerful,

expect to see a move toward 2,048 bit length certificates past the year 2010. Be sure to check with your chosen certificate authority to ensure they will support bit lengths larger than 1,024 before increasing this value.

Figure 5.19 Cryptographic Service Provider Page



6. On the **File Name** page provide a path and name of a file where to sort the certificate request and click **Next**.
7. Contact your preferred certificate authority to obtain the response file for your request.
   - If you are looking to test out the SSL functionality there are a number of providers that will give you a free trial SSL certificate that lasts for anywhere from 15 to 60 days. This is handy because they have all the trust features of regular certificates with no cost.

8. When you obtain the response file, open **IIS Manager** and return to the **Server Certificates** section.
9. In the right-hand actions pane click **Complete Certificate Request**.
10. In the **Complete Certificate Request** dialog on the **Specify Certificate Authority Response** page, locate the **Certificate Authority's Response** file, provide a **Friendly Name** for the certificate, and click **Next** to complete the process.

Configuring & Implementing…

The Real Differences between SSL Certificates

When you are out shopping for an SSL certificate it can get quite confusing as to what the differences are between the various offerings. For the most part you are buying trust in that the certificate you will be issued is trusted by the client. Under the covers the technical differences boil down to these:

- **Standard Certificate** A basic security certificate that will suit most users and will work for 40-bit encryption up to 256-bit encryption in most modern browsers
- **Server Gated Certificate** Before the United States dropped its cryptography export laws in January of 2000 these certificates added a step in the security handshake to see whether the client could support stronger cryptographic algorithms (ciphers). This allowed older browsers an opportunity to step-up their level of encryption if they did not use 128-bit or higher encryption by default.
- **Extended Validation Certificate** From a technical perspective these certificates are no different than a standard certificate with the exception that they have some additional metadata attached to the certificate. This metadata is used by browsers that are capable of reading it to determine if they should identify for the user (e.g., turn the address bar green) that the site has gone through extra validation steps. The validation steps and data included are available in the extended validation certificate guidelines at www.cabforum.org.

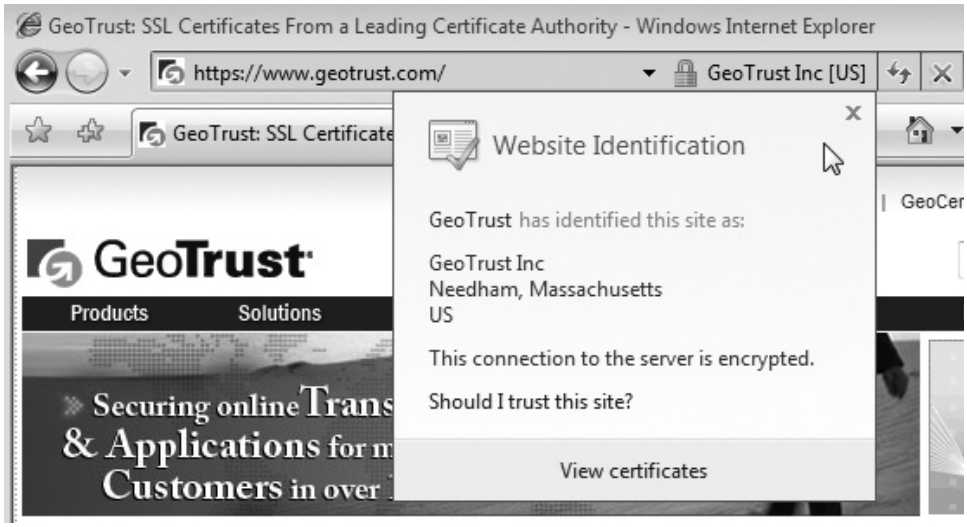
Continued

With the data in hand modern browsers will signal to the user through actions like turning the address bar green as shown in Figure 5.20. This feature of popular browsers like Internet Explorer 7 is meant to help users identify the site authenticity.

- **Wildcard Certificate** One of the three preceding certificates, but using an asterisk (*) somewhere in the domain name to signify a wildcard value. This is generally considered a premium service and commercial providers reflect this fact in their pricing model.

When choosing certificates remember that the level of encryption used in most cases is decided on as a mutual agreement between the client and the server. Both parties can choose to use a minimum level of encryption. With IIS this value is represented by a single check box to force clients to use a minimum of 128-bit encryption or have IIS refuse the connection request. Other advertised features have no impact on the security provided by the SSL-enabled session.

Figure 5.20 Internet Explorer Address Bar of a Site Using Extended Validation Certificate



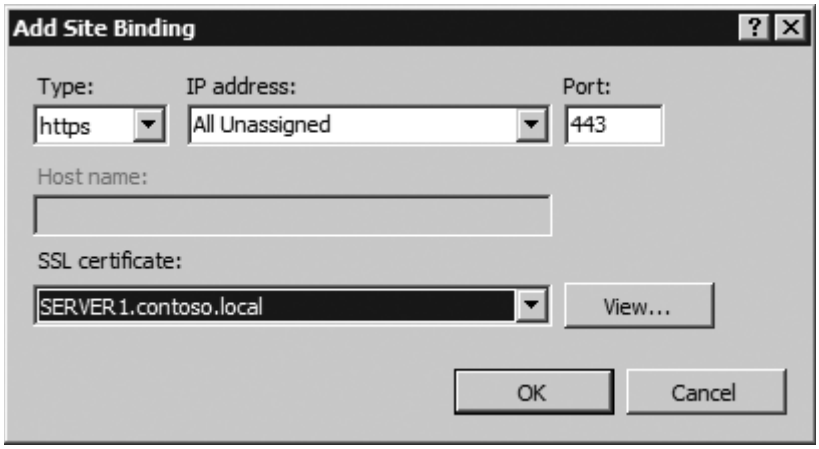
With the certificate in place you can now bind the certificate to your Web site. Under the covers the security certificate is bound to an IP address since the request header information is encrypted when the server needs to determine which certificate

to use. Once the certificate is bound you can choose to force the use of SSL on all or part of the site.

To enable secure communication on your Web site, follow these steps:

1. Open **Control Panel** and under **System and Maintenance | Administration Tools**, double-click the **Internet Information Services (IIS) Manager** shortcut.
2. In the **Internet Information Services (IIS) Manager** management console expand the server node, right-click your site, and select **Edit Bindings**.
3. In the **Site Bindings** dialog click **Add**.
4. In the **Add Site Binding** dialog set the Type to **HTTPS**. From the **SSL Certificate** list choose your certificate and click **OK** (see Figure 5.21).

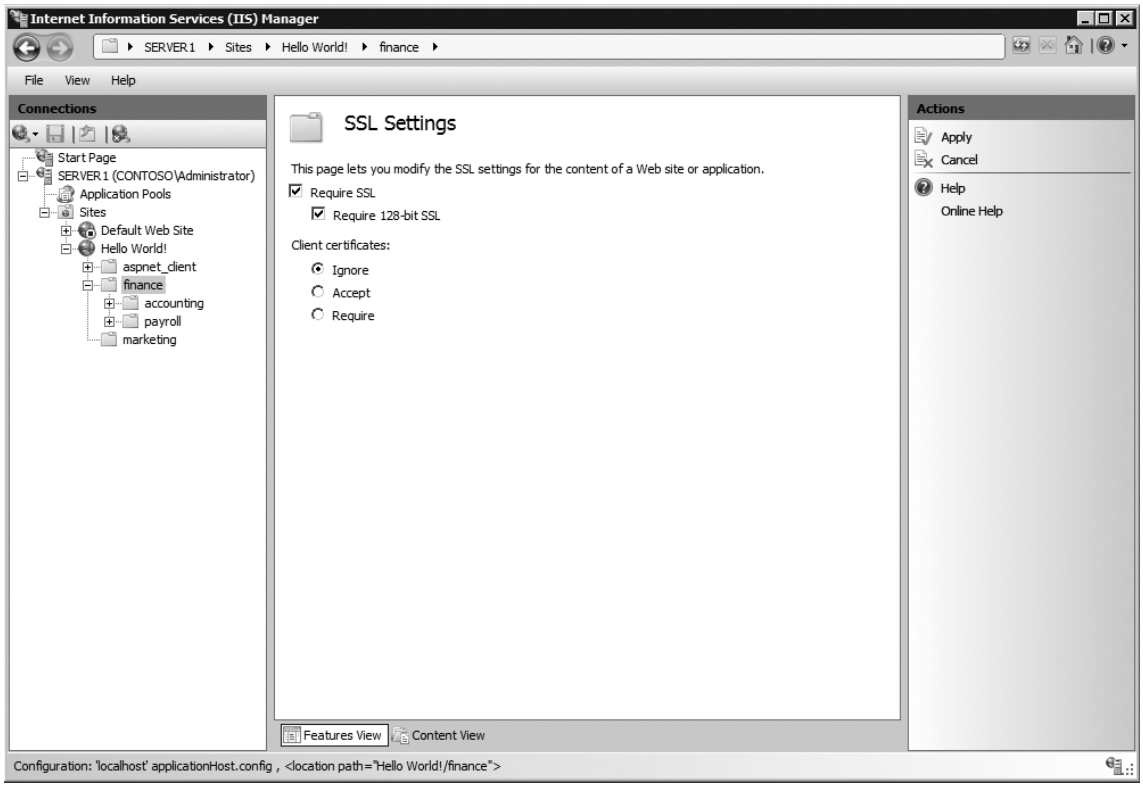
Figure 5.21 Add Site Binding Dialog



5. In the **Site Bindings** dialog click **Close**.
6. Expand your site node, locate and click a folder (or select the site to enforce SSL on the site as a whole) that you wish to secure.
7. In the middle pane under **Features View**, double-click **SSL Settings**.
8. In the **SSL Settings** module check **Require SSL**, **Require 128-bit SSL**, and in the right-hand Actions pane click **Apply** (see Figure 5.22).
   - Most modern Web browsers support 128-bit SSL. This option was put in place because up until 2000 the United State government restricted the

export of certain cryptographic algorithms, which left a good portion of the world stuck with 40- or 56-bit sessions, which provided a lesser degree of security.

Figure 5.22 SSL Settings Module Configuration



Authentication

Authentication is the process of asserting the identity of the user making a request to the Web server. With this identity we can track who is doing what and evaluate rules to determine if they are authorized to perform specific actions. IIS ships with several types of authentication modules that can be used to determine a user's identity:

- **Anonymous** Enabled by default to allow any user to access public content with a username and password.
- **Basic** Requires the user to provide a username and password. This authentication protocol is a standard across all platforms. It does not perform any

sort of encryption with the information provided by the user. As such you should use it with SSL to ensure that the credentials are sent over a secure connection.

- **Digest** Similar to basic authentication but instead of sending the password in clear text it sends an MD5 hash across the wire, which is verified by the server. One of the disadvantages to this method is that it requires that the password be stored using reversible encryption. It is also vulnerable to man-in-the-middle attacks.

WARNING

The RFC-standard Digest authentication requires HTTP 1.1, and the password must be stored in reversible encryption within the security data store (Active Directory, local SAM). Advanced Digest gets around the reversible encryption by storing the hash in Active Directory, but it works only on Internet Explorer 5.0 or later.

- **Windows** Used mainly in intranet scenarios, it allows browsers to use the current user's Windows domain credentials to authenticate the connection. Under the covers it uses NTLM or Kerberos to handle the authentication.
- **Client Certificates** Users provide a digital certificate that is mapped to a user account.

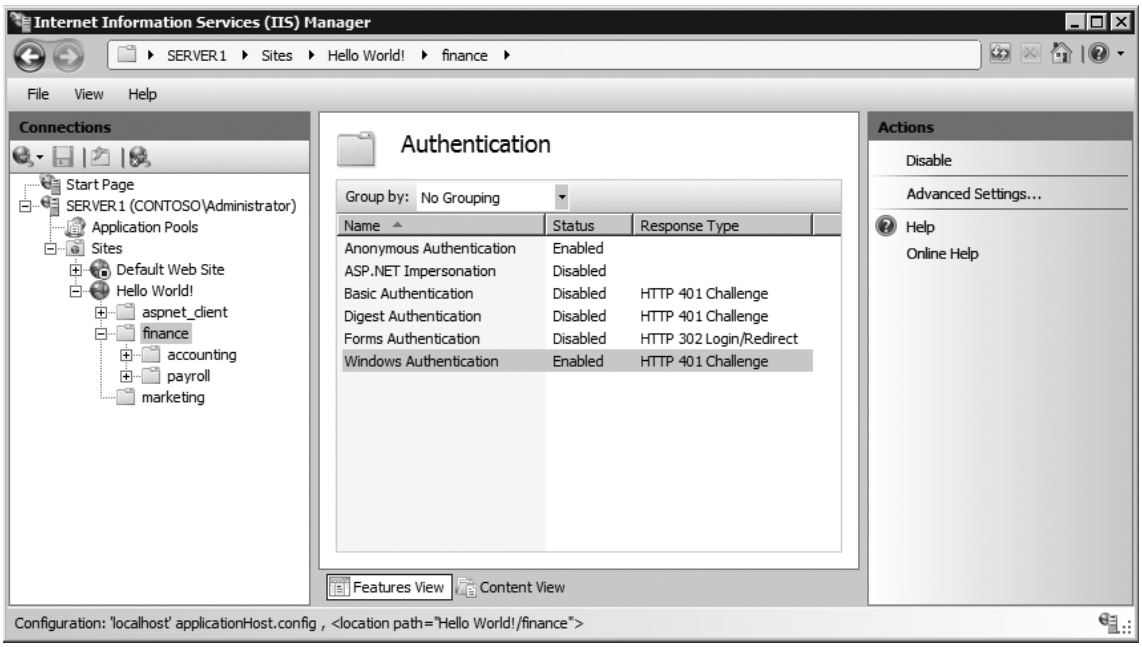
With the exception of client certificates, enabling these authentication modules usually requires nothing more than toggling of their state to enabled. The options for most of the modules are limited to either identity impersonation options or default realms for authentication.

Follow these steps to enable basic authentication on a folder:

1. Open **Control Panel** and under **System and Maintenance | Administration Tools**, double-click the **Internet Information Services (IIS) Manager** shortcut.
2. In the **Internet Information Services (IIS) Manager** management console expand the server, site node, and locate a folder to secure (or choose the site as a whole) and click your selection.

3. In the middle pane under **Features View** double-click **Authentication**.
4. Right-click the **Basic Authentication** module and select **Enable** (see Figure 5.23).

Figure 5.23 Authentication Module Configuration

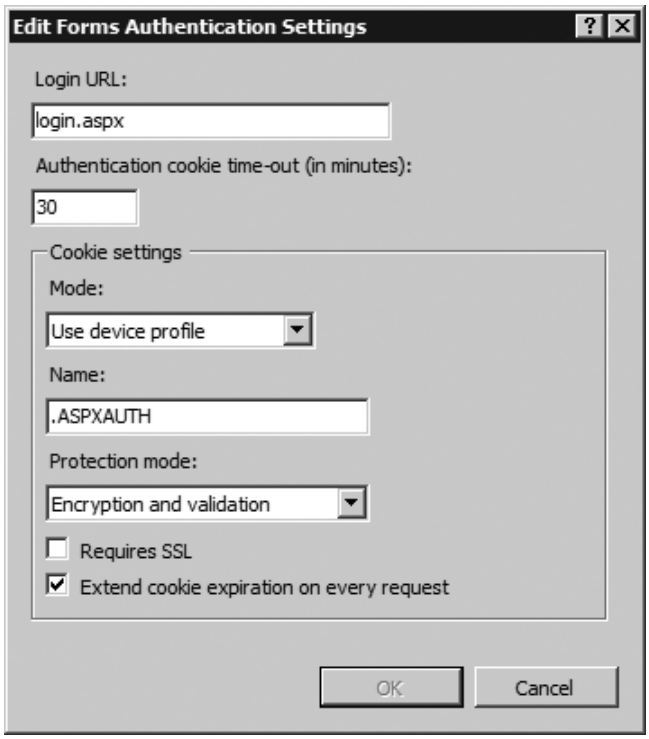


If you are using an ASP.NET runtime environment you have two other authentication modules that are specific to ASP.NET-based Web applications:

- **Forms** Enables you to provide a rich Web-based authentication and user registration experience.
- **ASP.NET Impersonation** Enables you to use a specific account, or the account specified by another IIS authentication module, to execute the application as opposed to the application pool identity.

These authentication modules have been available in ASP.NET since the 1.1 release of the .NET Framework. The IIS Manager exposes a number of the configuration options that traditionally have been managed through the ASP.NET tab in the previous release of IIS or directly in the web.config (see Figure 5.24).

Figure 5.24 Edit Forms Authentication Settings Dialog



Considerations When Using Client Certificates

In the previous section you may have noticed some options around whether or not to ignore, accept, or require client certificates. These options are contained within the SSL Settings because the client certificate submission process is a part of the SSL module. This also means that you will need SSL enabled on sites and folders where you want to use client certificate mapping. When a client certificate is received it can be mapped back to a user account in one of three ways:

- **Active Directory Client Certificate Mapping** Looks to the local Active Directory domain to locate a match for the client certificate that was applied. Note that using this option requires that it be used across all sites on the server.
- **One-to-One Mapping** Allows you to specify through the configuration the identity to be used for the user with whom the certificate matches.
- **Many-to-One Mapping** Like one-to-one mapping it allows you to control through the configuration the user identity used when the certificate is matched. This method allows you to map multiple users to a single identity.

WARNING

Active Directory Client Certificate Mapping disables the ability to use one-to-one and many-to-one certificate mapping because it is able to resolve back to both users and groups within the directory, effectively doing the same thing as both one-to-one and many-to-one certificate mapping.

At the time of this writing there was no graphical interface to the one-to-one and many-to-one certificate mapping controls. Listing 5.1 shows an example of the configuration values for both of these mapping methods.

Listing 5.1 One-to-One and Many-to-One Certificate Mapping Configuration

```
<configuration>
…
<system.webServer>
  …
  <security>
    …
    <authentication>
      <iisClientCertificateMappingAuthentication enabled="true">
        <manyToOneMappings>
          <add name="FinanceUsers" description="Finance Users"
               enabled="true" permissionMode="Allow"
               userName="CONTOSO\FinanceDelegate" password="DF923uD@#2">
            <rules>
              <add certificateField="Subject"
                matchCriteria="john@contoso.com" />
              <add certificateField="Subject"
                matchCriteria="jane@contoso.com" />
              <add certificateField="Subject"
                matchCriteria="sam@contoso.com" />
              <add certificateField="Subject"
                matchCriteria="sally*@contoso.com" />
            </rules>
          </add>
        </manyToOneMappings>
        <oneToOneMappings>
          <add enabled="true" certificate="-----BEGIN CERTIFICATE-----
```

```
MIIBqDCCARECAQAwaTELMAkGA1UEBhMCVVMxDjAMBgNVBAgTBVRleGFzMRMwEQYD
VQQHEwpMYXNDb2xpbmFzMRIwEAYDVQQKEwlNaWNyb3NvZnQxDjAMBgNVBAsTBUl0
ZWFtMREwDwYDVQQDFAhOVFZPT0RPTzCBnjANBgkqhkiG9w0BAQEFAAOBjAAwgYgC
gYBxmmAWKbLJHg5TuVyjgzWW0JsY5Shaqd7BDWtqhzy4HfRTW22f31rlm8NeSXHn
EhLiwsGgNzWHJ8no1QIYzAgpDR79oqxvgrY4WS3PXT7OLwIDAQABoAAwDQYJKoZI
hvcNAQEEBQADgYEAVcyI4jtnnV6kMiByiq4Xg99yL0U7bIpEwAf3MIZHS7wuNqfY
acfhbRj6VFHT8ObprKGPmqXJvwrBmPrEuCs4Ik6PidAAeEfoaa3naIbM73tTvKN+
WD30lAfGBr8SZixLep4pMIN/wO0eu6f30cBuoPtDnDulNT8AuQHjkJIc8Qc=
-----END CERTIFICATE-----"
                userName="CONTOSO\FinanceDelegate" password="DF923uD@#2"
            </oneToOneMappings>
          </iisClientCertificateMappingAuthentication>
        </authentication>
        …
      </security>
      …
  </system.webServer>
  …
</configuration>
```

Unlike the other two methods, enabling Active Directory Client Certificate is exposed through the graphical interface. The option is exposed at the server node level, and when it is set, it disables the ability to use one-to-one and many-to-one mappings on the server. To learn how to associate a certificate with an Active Directory user account refer to the Windows Server 2008 documentation around public key infrastructure.

Follow these steps to enable Active Directory Client Certificate mapping:

1. Open **Control Panel** and under **System and Maintenance | Administration Tools**, double-click the **Internet Information Services (IIS) Manager** shortcut,
2. In the **Internet Information Services (IIS) Manager** management console click the server node,
3. In the middle pane under **Features View** double-click **Authentication**.
4. Right-click the Active Directory Client Certificate Mapping module and select **Enable**.

Authorization

With the user's identity established the next step is to determine if the user can perform the action that is being requested. Authorization encompasses a set of rules that are evaluated based on a number of conditions, which could include the user's identity, to provide a decision as to whether or not to allow the user's request to be acted upon. IIS provides three core modules focused on authorization and supporting services—URL authorization, IP authorization, and request filtering.

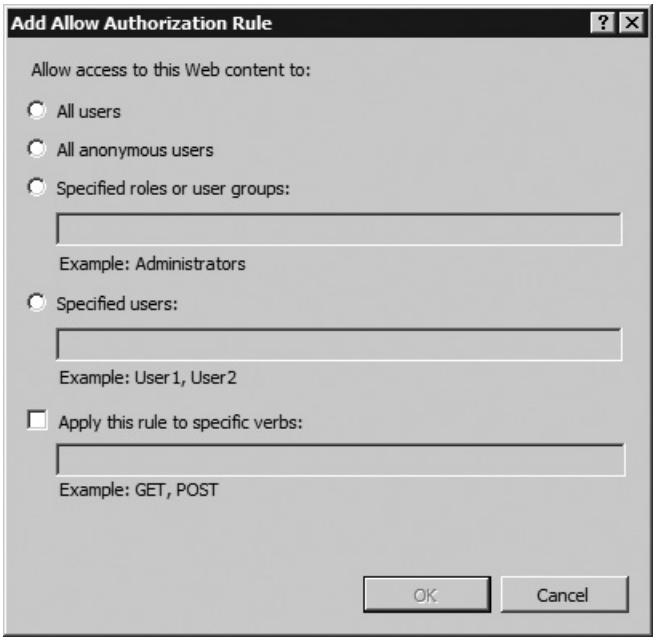
URL Authorization

Originally brought into the IIS environment by ASP.NET, the URL Authorization module has been rewritten as a native IIS module to allow everyone to take advantage of an easy way of restricting access to specific folders and files. This module allows Web content managers the ability to control access in a manner similar to the use of NTFS permissions. Unlike NTFS permissions, you do not need file system access to the server to apply permissions since everything is managed through the web.config file stored at the root of the site or within a given folder. As well, this allows you to easily carry the permissions with the site as it moves environments. In the following steps, we'll show you how to restrict access to a folder:

1. Open **Control Panel** and under **System and Maintenance** | **Administration Tools,** double-click the **Internet Information Services (IIS) Manager** shortcut.
2. In the **Internet Information Services (IIS) Manager** management console expand the server, site node, and locate a folder to secure (or choose the site as a whole) and click your selection.
3. In the middle pane under **Features View** double-click **Authentication**.
4. On the **Authentication** page ensure that the **Anonymous Authentication** module is **Disabled**, select one of the other authentication modules, and click **Enable** in the right-hand **Actions** pane.
5. Click the Back arrow in the top left-hand corner.

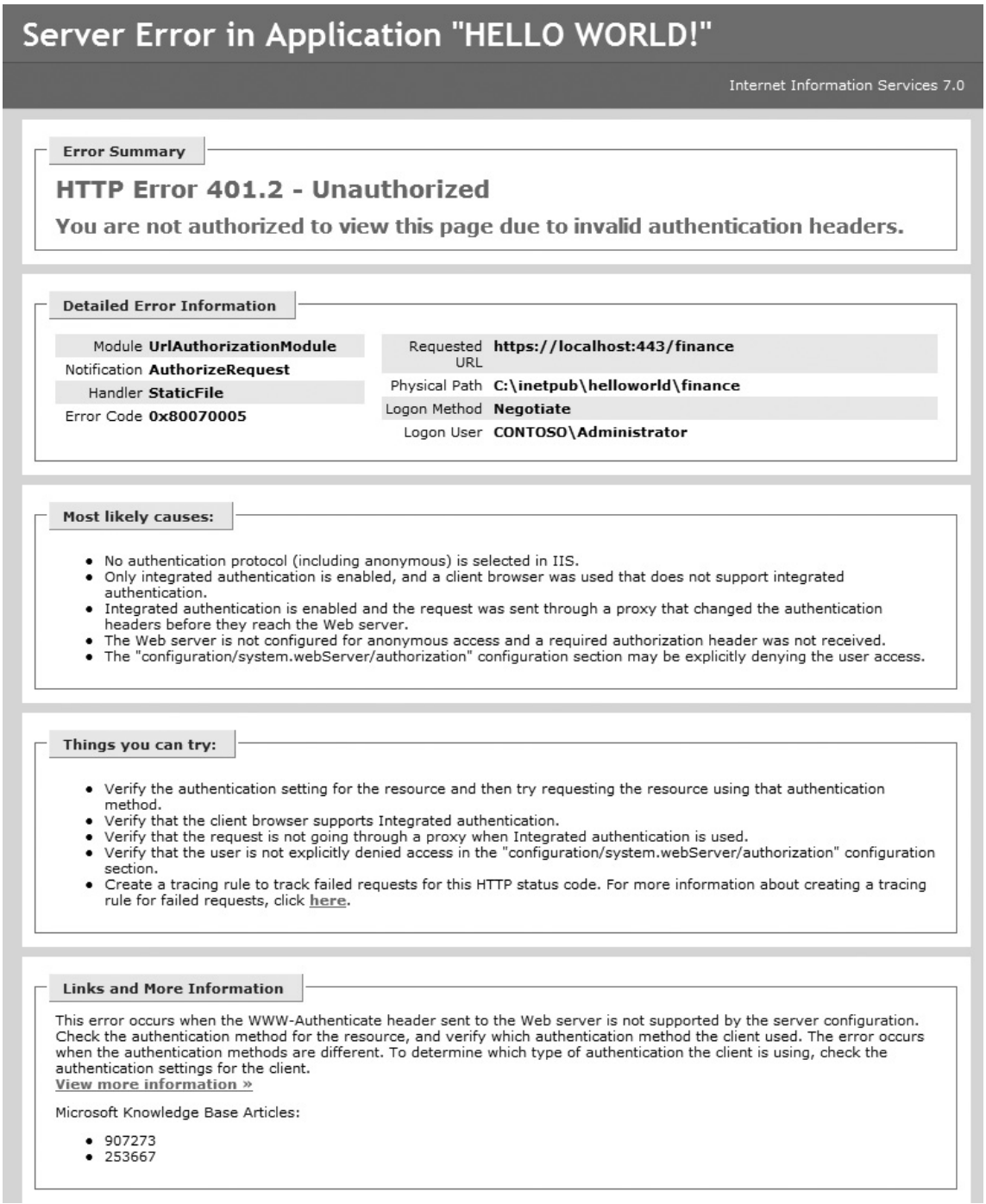
6. On the folder page in the middle pane under **Features View**, double-click **Authorization Rules**.
7. On the **Authorization Rules** page click the **Add Allow Rule** in the right-hand action page.
8. Select the **Specified Users** radio button, provide a username, and click **OK** (see Figure 5.25).

Figure 5.25 Add Allow Authorization Rule Dialog



When users attempt to access a page to which they have been denied, they will receive a 401.2 unauthorized error. With the addition of detailed error requests the server-side error message gives you a number of useful elements to help you troubleshoot access denied issues being caused by URL authorization. As shown in Figure 5.26, you can see that we are dealing with the URL Authorization Module, that the file is a static file, along with the logon method and user account being used to access the URL.

Figure 5.26 Server-Side Version of Unauthorized Page Access Error Message



IP Authorization

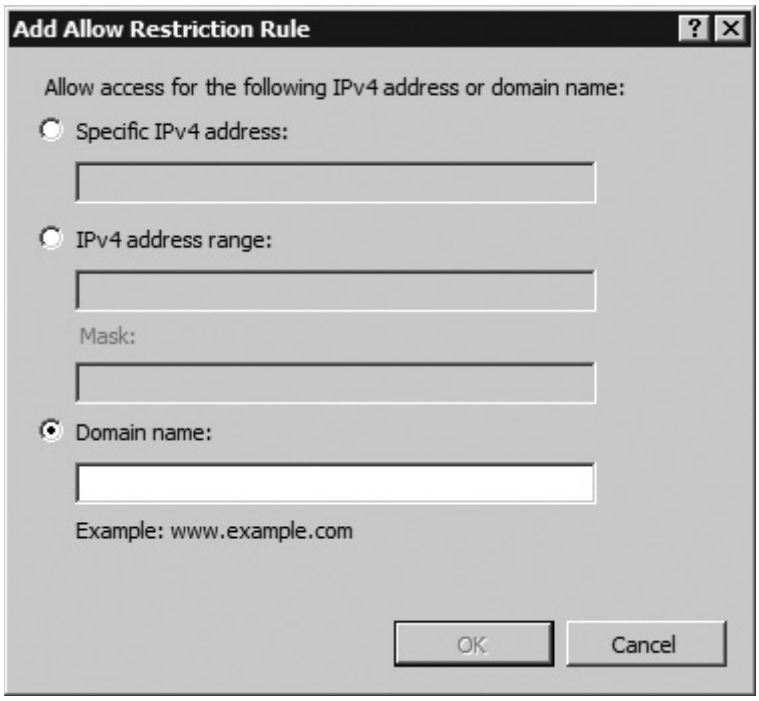
The ability to restrict access to specific IP addresses has existed for quite some time across both servers and networking devices such as firewalls. In the past this function, like file permissions, was available only through IIS Manager and was tough to replicate across to other servers as it was stored in the metabase. This setting, along with all other configuration options, has been moved to the new XML-based configuration files. This allows you to centralize, copy, and manipulate the settings using new programming interfaces and command-line tools as well as the traditional graphical user interface.

Here are the steps to follow for restricting access to users based on their IP addresses.

1. Open **Control Panel** and under **System and Maintenance | Administration Tools**, double-click the **Internet Information Services (IIS) Manager** shortcut.
2. In the **Internet Information Services (IIS) Manager** management console expand the server, site node, and locate a folder to secure (or choose the site as a whole) and click your selection.
3. In the middle pane under **Features View**, double-click **IPv4 Address and Domain Restrictions**,
4. In the right-hand actions pane click **Add Deny Entry**.
5. In the **Add Deny Restriction Rule** dialog, select the **Specific IPv4 Address** radio button, provide an IP address (e.g. 127.0.0.1 if you want to test http://localhost) and click **OK**.

When users attempt to access a page to which they have been denied, they will receive a 403.6 forbidden error. Another option is to restrict users based on their domain names (see Figure 5.27). You will need to enable this through the Edit Feature Settings link in the module page on IIS. Be aware that the added overhead of DNS resolution for each IP address could negatively affect the performance of your application.

Figure 5.27 Add Allow Restriction Rule Dialog with Domain Restrictions Enabled



Configuring & Implementing…

Authorization Manager

You can use Authorization Manager in applications that require role-based authorization, such as ASP.NET Web applications, ASP.NET Web services, and client/server systems based on .NET Remoting. Windows Server 2008's enhanced version of Authorization Manager includes support for custom object pickers and business rule groups. Authorization Manager is also now capable of storing authorization stores in AD, SQL, or XML.

Request Filtering

Previously available through an add-on known as URLScan, the request filtering features provide an additional layer of security by inspecting incoming requests for seven different characteristics that might indicate a malformed or malicious attack:

- **Double-Encoded Requests** Attackers may encode a request twice to get around a first layer of filtering. This filter will detect it, reject the request, and log a 404.11 error.
- **High Bit Characters** You may choose to not want to accept non-ASCII characters (e.g., Unicode characters) because your application has not been tested or does not support it. This filter will detect the non-ASCII characters, reject the request, and log a 404.12 error.
- **File Extensions** Your Web application may contain certain files that you do not want anyone to download in any case (e.g., a DLL file in an ASP.NET application). You can add a list of allowed and denied extensions, which will cause IIS to reject the request and log a 404.7 error.
- **Request Limits** This filter will look at how long the content is in the request, the length of the URL, and more specifically the length of the query string. If any of those measurements exceed the maximum values provided this filter will reject the request and log a 404.13, 404.14, or 404.15, respectively.
- **Verbs** There are different types of requests that are identified using verbs (e.g., PUT, GET, and POST). If your application uses only specific types you can tell IIS to reject the request for other types and log a 404.6 error.
- **URL Sequences** There are certain character sequences that you may wish to never have in your request (e.g., a double period "…" often signifies someone trying to relatively traverse your folder structure). This filter will reject requests that match the sequences and log a 404.5 error.
- **Hidden Segments** This filter will enable you to reject requests for content from certain segments. Listing 4.7 contains an example where the bin folder has been specified causing IIS to reject requests that contains the bin folder in the URL. Note that the filter is able to distinguish between a request for http://contoso.com/bin/somefile.dll and http://contoso.com/binary/somefile.zip. The latter request would be allowed through because the filter looks at the URL segment as a whole. It will reject the first request and log a 404.8 error.

Unfortunately IIS Manager does not expose these configuration values. If you want to enable request filtering and tune it to your environment you will need to do it directly in the configuration file or through one of the programmatic APIs. Listing 5.2 shows a sample excerpt of the configuration settings.

Listing 5.2 Request Filtering Configuration Example

```
<configuration>
  …
  <system.webServer>
    …
    <security>
      …
      <requestFiltering allowDoubleEscaping="false"
                        allowHighBitCharacters="true"
                        maxAllowedContentLength="1024768"
                        maxQueryString="64"
                        maxUrl="260">
        <denyUrlSequences>
          <add sequence="…"/>
        </denyUrlSequences>
        <fileExtensions allowUnlisted="true" >
          <add fileExtension=".dll" allowed="false"/>
          <add fileExtension=".xml" allowed="false"/>
        </fileExtensions>
        <hiddenSegments>
          <add segment="BIN"/>
        </hiddenSegments>
        <verbs allowUnlisted="false">
          <add verb="GET" allowed="true" />
          <add verb="PUT" allowed="false" />
        </verbs>
      </requestFiltering>
      …
    </security>
    …
  </system.webServer>
    …
</configuration>
```

Even though you may have to work with the application developer to gain necessary input, this module in particular is extremely useful in reducing the attack surface of your Web application. It is recommended that you take the time to take full advantage of the filters offered by this module.

.NET Trust Levels

With a number of new IIS features based around the .NET Framework it is important to understand how .NET Trust Levels impact your Web applications and IIS itself. A trust level conveys a policy of permissions that an application is allowed to perform. Each trust level has a different set of permissions applied. By default the policies build upon one another from Minimal, which can do very few things to Full, which can perform a number of things:

- **Full Trust** The application is able to execute anything with the security bounds granted to the process identity.
- **High Trust** Restricts applications from calling unmanaged code (e.g., Windows APIs, COM objects, etc.), writing to the event log, message queues, or databases.
- **Medium Trust** Restricts the application from navigating any part of the file system except its own application directory, accessing the registry, or making network and Web service calls.
- **Low Trust** Restricts the application from writing to the file system.
- **Minimal** Restricts the code to doing basic algorithmic work.

If the out-of-the-box trust levels do not suffice application developers can define a custom trust policy based on a series of intrinsic and custom permissions. For a complete list of permissions see the .NET Framework Developer's Guide at http://msdn2.microsoft.com/en-us/library/5ba4k1c5.aspx.

WARNING

The trust levels and permissions system, known as Code Access Security, is confusing to many developers as well as administrators. Think of the policies as another type of access control list (ACL) with individual access control entries (ACE) that allow or deny you from performing an action on a resource. The resources can vary from external services, such as databases and Web services, to internal Windows subsystems, such as the registry, event log, and file system.

The trust level that you chose for an application should be sufficient for it to function, but like all good security practices, not excessive beyond the needs of the application. In most environments application developers will communicate the level of trust their application needs. As IT professionals understanding what that means helps us understand the boundaries in which the application can function in the server environment. The trust levels can be set at the site and folder level. It is most practical, however, to set it at the level a Web application is defined or the root of the site.

Summary

Protecting data is extremely important for companies in the global marketplace today, and many of these companies' networks are based on Microsoft infrastructure. A Windows server provides a number of useful functions in a company's network infrastructure. BitLocker is Microsoft's answer to providing better security by encrypting the data stored on the drive's operating system volume, and is available only in the Enterprise and Ultimate versions of Vista. This new security feature goes a long way toward helping users and organizations protect their data.

AD RMS protects information when it is connected and when it is not connected to the corporate network. A usage policy is bound to the protected item so that no matter where it travels the rights are enforced to ensure that only the authorized recipient is able to access the contents. The policy can restrict users from actions such as viewing, copying, forwarding, and printing.

Authorization encompasses a set of rules that are evaluated based on a number of conditions, which could include the user's identity, to provide a decision as to whether or not to allow the user's request to be acted upon. IIS provides three core modules focused on authorization and supporting services—URL authorization, IP authorization, and request filtering.

Solutions Fast Track

BitLocker

- ☑ BitLocker is Microsoft's answer to providing better security by encrypting the data stored on the drive's operating system volume
- ☑ You can set up BitLocker in three configurations: TPM only, TPM and USB flash drive, and TPM and PIN.
- ☑ To use a BitLocker-enabled system, the key must be stored in RAM while the system is up and running. Universities have found that when a system is shut down, it's possible to retrieve the key from RAM for up to several minutes, giving a hacker complete control over the entire system and all files stored on the drive. The main way to avoid this, of course, is to never leave a system unattended in an unsecured area in the first place. The next step is to completely shut down the system so that the RAM can be allowed to fully discharge.

Active Directory Rights Management Services

- ☑ Previously shipped as an add-on for Windows Server, AD RMS is now included out-of-the-box as a role in Windows Server 2008.
- ☑ AD RMS is a format- and application-agnostic service designed to safeguard information by deterring inadvertent sharing of information with unauthorized people.
- ☑ When you set up AD RMS it will trust your organization's Active Directory domain by default. Depending on your business requirements you can expand or contract the boundaries of your RMS trust to specific user domains within your organization or other organizations.

Authorization

- ☑ Authorization encompasses a set of rules that are evaluated based on a number of conditions, which could include the user's identity, to provide a decision as to whether or not to allow the user's request to be acted upon.
- ☑ IIS provides three core modules focused on authorization and supporting services—URL authorization, IP authorization, and request filtering.
- ☑ Previously available through an add-on known as URLScan, the request filtering module provides an additional layer of security by inspecting incoming requests for seven different characteristics that might indicate a malformed or malicious attack.

Frequently Asked Questions

Q: I want to use BitLocker but my motherboard doesn't have a TPM. Is it still possible to enable BitLocker?

A: Yes. The default BitLocker configurations will have to be modified to use a USB flash drive.

Q: I have a media file that uses Digital Rights Management. How can I remove the DRM protection from the file?

A: It's impossible to remove the DRM protection from any file that has been created with it.

Q: Does Rights Management work with mobile devices?

A: Yes, there is a mobile module for Rights Management Services. However, only Windows Mobile devices are supported with Rights Management. Check with your wireless vendor or mobile manufacturer for support and availability on particular models.