Realtime
publishers
"Leading the Conversation"

The Definitive Guide™ *To*

Building a Windows Server 2008 Infrastructure

sponsored by

triCerat

Greg Shields

Copyright Statement

Chapter 9: Securing Servers & the Domain

Throughout this guide, I've attempted to show you the features and functionality now available with Windows Server 2008 that are designed to help build and manage your Windows infrastructure. Some of these capabilities are new to this OS version, while others have minor upgrades or remain relatively unchanged.

Yet all these great technologies found in Windows Server 2008 amount to exactly nothing if you can't properly secure them against external attack. This idea is central to much of what's different about Windows Server 2008. Beneath the covers and the systems administrator's radar are a host of changes to the core OS itself that improve its security, enable better resistance against external attack, and ultimately improve its reliability. But those kernel-level improvements are only one part of the story. Layering atop the core enhancements are a set of features that make the management of security easier and more reliable.

In this chapter, we'll talk about those features that enhance your data center's security posture. By making the upgrade to Windows Server 2008, the workloads you run in your organization stand to gain a higher level of uptime.

Windows Server 2008 Incorporates New and Improved Security Features

Although this chapter cannot touch on all the new security-related capabilities found in Windows Server 2008, we'll focus our attention on a few that can make the most impact in your infrastructure. Some of these features have essentially no user interface, making them low-level improvements for across-the-board use, while others must be specifically enabled and managed for use in your environment.

> ☞ Be aware that due to the sharing of code base between Windows Server 2008 and Windows Vista, much of what you see here also relates to Windows Vista. Thus, the improvements enjoyed by Windows Server 2008 are also realized at the desktop upon the upgrade to Windows Vista.

Componentization

We've discussed throughout this guide how the concept of componentization is a major shift in the development of the Windows OS. With previous versions of Microsoft Windows, Microsoft elected to heap virtually all the files associated with OS functionality onto every installed instance. This is the case even when some of that functionality was unused. This decision made it easy to add new functionality to an existing system, but at the same time, unnecessarily left potentially exploitable code on a system.

The componentization activities that went into the development of Windows Server 2008 changed all that. By breaking apart the Windows OS into its disparate components and logically defining the linkages between each, functionality that is not used on a system instance is simply not present on the system. This componentization, much of which is below the level of visibility of the administrator, accomplishes three very important things:

- It means that Microsoft now has a much-improved view of system components as well as how each relates to and relies on others for functionality. By breaking apart and mapping system functions in relation to each other, Microsoft can better support and enhance the OS now and in the future.
- With a known functional map of components and their interrelations, unnecessary and extraneous files and folders need not be present on a system instance. This means that exploit code will not be able to use these system files in the case that they are not properly patched.
- Knowing how each function relies on others means that any desired component can be assuredly installed with its necessary prerequisites. This helps eliminate the situation in which administrators attempt to install new functionality without knowing what prerequisite components are necessary.

Security Configuration Wizard

Although the componentization activity itself goes far into locking down system functionality to just the resources a server needs, sometimes additional hardening is necessary. For these cases, the Security Configuration Wizard (SCW) makes a return in Windows Server 2008 with special added functionality that supports Roles, Role Services, and Features. The SCW further tightens server configurations for environments that require a very high level of service lockdown.

> 📖 Though we won't discuss the use of the SCW in this chapter, you can find out more at http://technet.microsoft.com/en-us/library/cc771492.aspx.

Windows Service Hardening

Windows services and their "always on" nature have traditionally been a source of security concern with the Windows OS. Windows Server 2008 reduces the threat from these sources by first reducing the number of services that are necessary to run by default. A few other capabilities are also introduced:

- Services are given the new ability to run with a per-service security identifier (SID). This isolates the running of the service to a particular SID while allowing explicit access control lists (ACLs) to be assigned to resources required by the service.
- Many services that used to run under the LocalSystem context have been moved to lesser-privileged accounts such as LocalService or NetworkService.
- Services are now linked to firewall policies, which limit the network exposure of the service to its intended functionality.
- Write-restricted access tokens can now be assigned to service processes. This restricts the services' ability to update data on a system outside its intended functionality.

> ✎ For the most part, these hardening tactics will be used by software developers in further securing their applications. Systems administrators will likely not make much use of these new capabilities except when instructed by application vendors or security checklists.

Fine-Grained Password Policies

In all previous versions of the Windows OS, password policies were only applied at the domain level. This implementation meant that classes of accounts that required different policies—such as password length, expiry, complexity requirements, and so on—could only be implemented through the creation of a completely new domain. With Windows Server 2008 and Fine-Grained Password Policies (FGPP), it is possible to create separate policies for individual groups within the same domain.

> ✎ FGPPs aren't necessarily an added security measure for your Windows infrastructure, but their capability for creating more than one password policy per domain makes them handy in a few special cases. For example, you might want to create a separate password policy for service accounts that does not require their passwords to be changed. This gives you the ability to change them on a regular basis rather than waiting for a password change event that could impact the functionality of the service. For detailed instructions on creating an FGPP, check out http://technet.microsoft.com/en-us/library/cc770842.aspx.

User Account Control

User Account Control (UAC), originally released with Windows Vista, is a mechanism for enforcing the principle of least privilege. UAC splits administrator authentication "tokens" into two halves, FGPP—with and without administrative privileges, and manages the use of the correct half as needed by the administrative user.

Windows Firewall with Advanced Security

The Windows firewall gains new management flexibility with Windows Server 2008, making this version of the firewall the easiest to use to date. Integrating the Windows firewall with Group Policy and enhancing its management GUI allows administrators to easily create policy across the domain for cohesive firewall management.

BitLocker Drive Encryption

Lastly, with Windows Server 2008 comes the same full-drive encryption capabilities seen with Windows Vista. BitLocker Drive Encryption provides a mechanism for encrypting an entire Windows drive, effectively eliminating the risk of data exposure as a result of a stolen system.

Although Microsoft's implementation of new security controls under the covers is a boon to your environment security, there is little that you can do to manage these features. So for the rest of this chapter, we'll focus the discussion on these last three topics, providing the specific steps necessary to implement them in your Windows Server 2008 infrastructure.

Successfully Managing UAC

Depending on whom you ask, UAC can be considered Microsoft's greatest addition to this latest round of OS releases or its greatest blunder. UAC has definitely gotten its share of press—both good and bad—in relation to how often it makes itself known to the individual user. Although this chapter won't get into the political discussion of UAC's efficacy, it will discuss the nature of UAC and some ways in which you can manage it in your organization.

UAC is probably best described by looking at the tool it is intended to replace. The principle of least privilege suggests that users should operate computer systems and launch processes and applications with as few privileges as possible. For any action that needs to be run on a system, that action should be run with the lowest level of privileges necessary to accomplish its task.

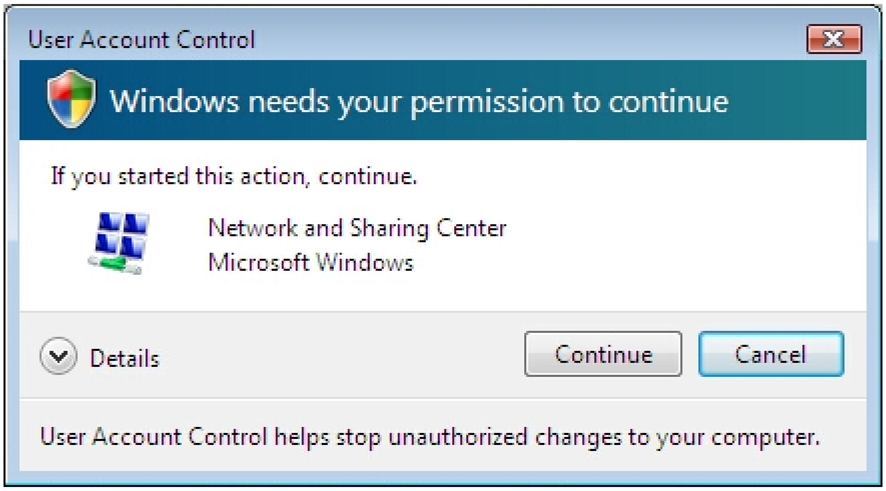
For regular users without administrative privileges, this has been accomplished by granting Domain User privileges along with the level of group membership necessary so that they can access the data necessary for their jobs. Administrators are a different breed entirely. With administrators, privileges tend toward the "all or nothing" variety, with membership in a computer's Administrators group granting what is effectively full control over all objects on a system.

In the days prior to Windows Vista and Windows Server 2008, Microsoft recommended that administrators maintain two separate accounts. Administrators would login using a non-administrative account with minimal privileges to accomplish the majority of their tasks. Only when it was necessary to accomplish tasks that required elevated privileges would they login using their privileged account. This was typically done with tools such as *runas*, which could launch individual privileges under the context of their administrative-privileged user.

> More information about the runas command can be found at http://support.microsoft.com/kb/294676.

The problem with tools like runas is in the administrative burden of managing two separate accounts and multiple logins. Administrators burdened with the overhead of double logins often found themselves not following proper procedures out of frustration. Standard users given administrative privileges on their local desktops were rarely capable of dealing with the complexities of double logins.

Thus, with the release of Windows Vista and Windows Server 2008, Microsoft implemented UAC as a mechanism for automating this process. Using UAC, when a user logs into a computer with administrative privileges, the local system creates two separate sessions. In the first session, the user does not have access to their administrator privileges. In the second, the administrator privileges are available for use. The Windows OS detects when administrator privileges are required and automatically switches to the session with elevated privileges when necessary for the activity required. UAC notifies the user and requests permission before any session switching occurs to request the user's "consent" for the elevation. The dialog box that does this looks similar to Figure 9.1.



User Account Control

Windows needs your permission to continue

If you started this action, continue.

Network and Sharing Center
Microsoft Windows

User Account Control helps stop unauthorized changes to your computer.

Figure 9.1: The UAC elevation request prompt.

The elevation prompt shown in Figure 9.1 appears each and every time an administrative user attempts to complete a task that requires the elevation. With numerous activities on the system requiring administrative privileges to correctly function—especially for IT professionals who perform IT functions constantly during their daily workflow—this elevation prompt has the tendency to appear a lot.

Moreover, this elevation behavior does not occur for just members of the Administrators group. A user that is a member of any of the following groups will also experience the double-sessioning effect:

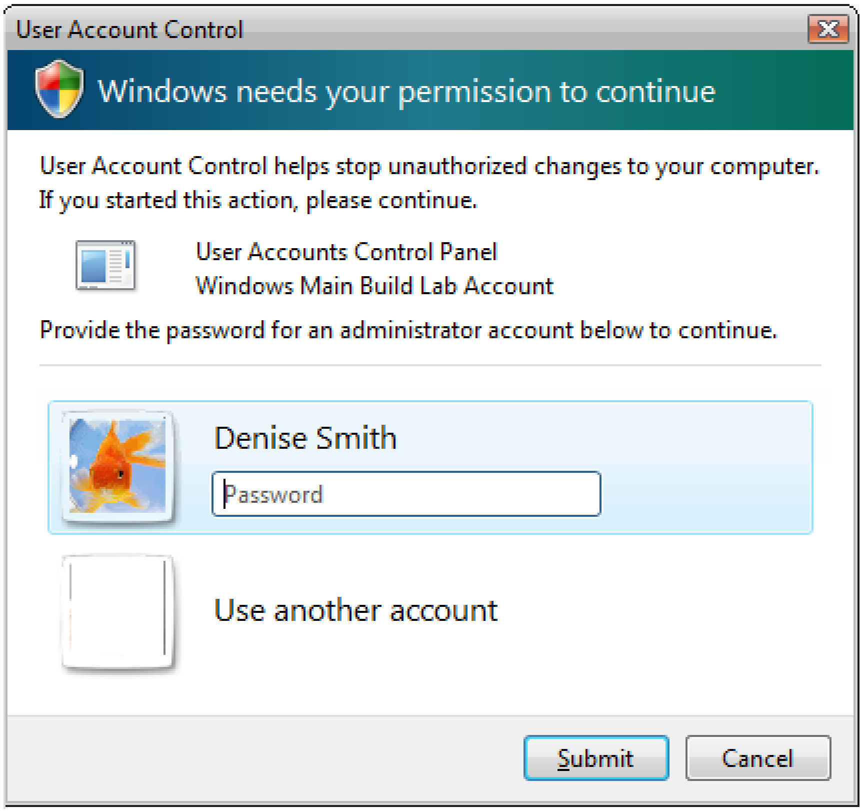
- Administrators
- Enterprise Administrators
- Policy Administrators
- Backup Operators
- Cryptographic Operators
- Power Users (deprecated)
- Network Configuration Operators
- RAS Servers
- Read-Only Domain Controllers in W2K8
- Domain Administrators
- Schema Administrators
- Certificate Administrators
- Account Operators
- System Operators
- Print Operators
- Domain Controllers
- Enterprise Read-Only Domain Controllers
- Pre-Windows 2000 Compatible Access

In addition to these groups, a user who has been assigned any of the following nine privileges will experience the same behavior:

- Create Token object (SeCreateTokenPrivilege)
- Act as part of the OS (SeTcbPrivilege)
- Take Ownership (SeTakeOwnershipPrivilege)
- Backup files and directories (SeBackupPrivilege)
- Restore files and directories (SeRestorePrivilege)
- Debug programs (SeDebugPrivilege)
- Impersonate client after authentication (SeImpersonate)
- Modify object label (SeRelabelPrivilege)
- Load and unload device drivers (SeLoadDriverPrivilege)

> By default, UAC is disabled for the original Administrator account. Initial logins to a freshly installed Windows Server 2008 instance login using this account.

The prompting doesn't stop there. Users who are not a member of any of the above groups and have not been assigned any of these nine privileges will also be prompted if they attempt to accomplish an action that requires administrative privileges. Unlike previous versions of the OS, instead of seeing an "Access Denied" error, users will instead be prompted with what is called an over the shoulder (OTS) elevation prompt. This prompt, shown in Figure 9.2, provides a mechanism for an administrator to enter credentials allowing the user to elevate in order to accomplish the task.



User Account Control

Windows needs your permission to continue

User Account Control helps stop unauthorized changes to your computer. If you started this action, please continue.

User Accounts Control Panel
Windows Main Build Lab Account

Provide the password for an administrator account below to continue.

Denise Smith

Password

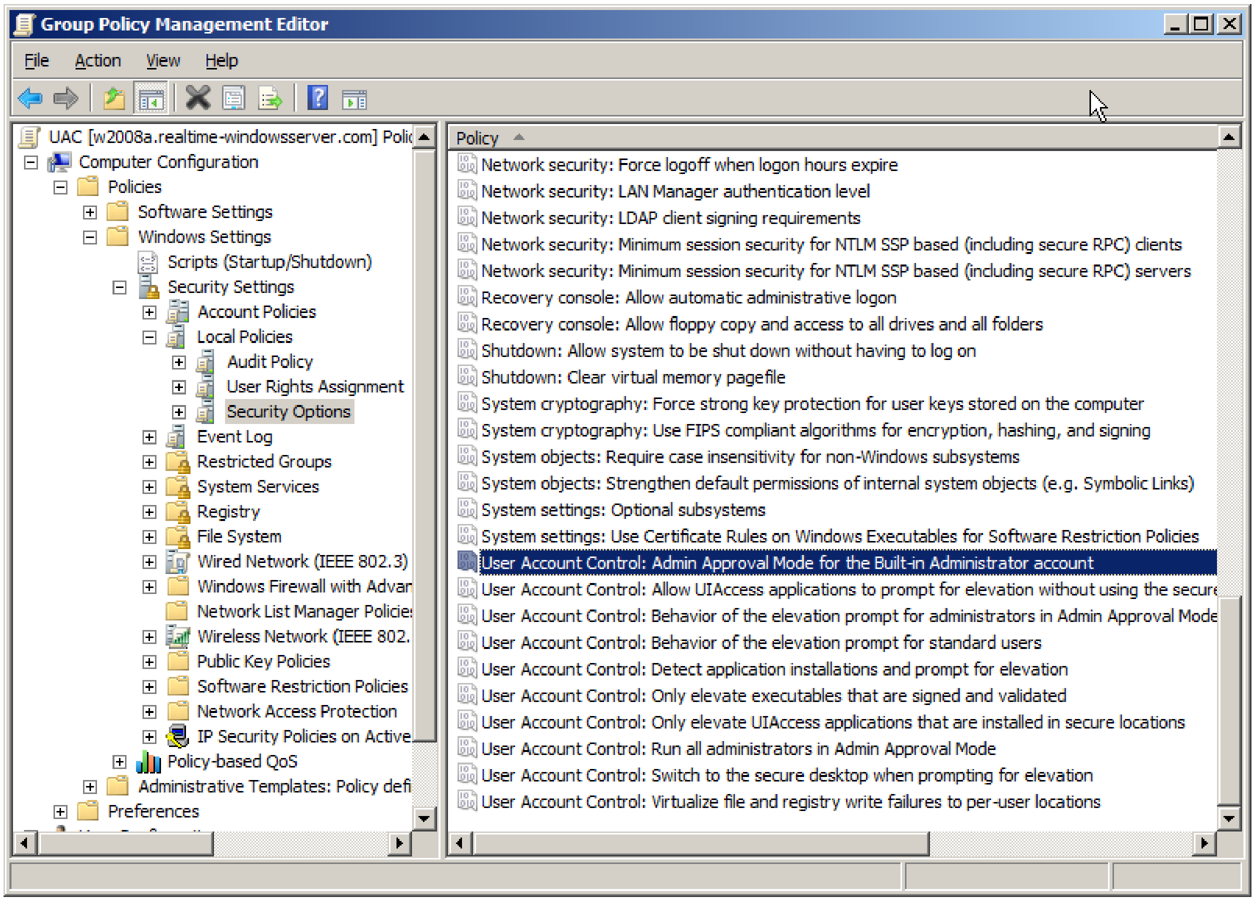
Use another account

Figure 9.2: An OTS elevation prompt seen by a non-administrative user who attempts to complete an action that requires administrative privileges.

Although ostensibly this prompt appears useful for helping out a user when you can't be present to assist with their problems, be careful with their use. When you hand over an administrative password to a user, that password can then be used for subsequent elevations by the user. If you want to use OTS elevations as a troubleshooting tool of last resort, remember to change any passwords after the user completes the necessary action.

Group Policy and UAC

Within a Windows domain, UAC is probably best controlled through the use of Group Policy. Ten settings are available for controlling UAC's behavior on configured desktops and servers. All are found at the location Computer Configuration | Policies | Windows Settings | Security Settings | Security Options, which is displayed in Figure 9.3.



UAC [w2008a.realtime-windowsserver.com] Polic
Computer Configuration
Policies
Software Settings
Windows Settings
Scripts (Startup/Shutdown)
Security Settings
Account Policies
Local Policies
Audit Policy
User Rights Assignment
Security Options
Event Log
Restricted Groups
System Services
Registry
File System
Wired Network (IEEE 802.3)
Windows Firewall with Advar
Network List Manager Policie
Wireless Network (IEEE 802.
Public Key Policies
Software Restriction Policies
Network Access Protection
IP Security Policies on Active
Policy-based QoS
Administrative Templates: Policy defi
Preferences

Policy

Network security: Force logoff when logon hours expire
Network security: LAN Manager authentication level
Network security: LDAP client signing requirements
Network security: Minimum session security for NTLM SSP based (including secure RPC) clients
Network security: Minimum session security for NTLM SSP based (including secure RPC) servers
Recovery console: Allow automatic administrative logon
Recovery console: Allow floppy copy and access to all drives and all folders
Shutdown: Allow system to be shut down without having to log on
Shutdown: Clear virtual memory pagefile
System cryptography: Force strong key protection for user keys stored on the computer
System cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing
System objects: Require case insensitivity for non-Windows subsystems
System objects: Strengthen default permissions of internal system objects (e.g. Symbolic Links)
System settings: Optional subsystems
System settings: Use Certificate Rules on Windows Executables for Software Restriction Policies
User Account Control: Admin Approval Mode for the Built-in Administrator account
User Account Control: Allow UIAccess applications to prompt for elevation without using the secure
User Account Control: Behavior of the elevation prompt for administrators in Admin Approval Mode
User Account Control: Behavior of the elevation prompt for standard users
User Account Control: Detect application installations and prompt for elevation
User Account Control: Only elevate executables that are signed and validated
User Account Control: Only elevate UIAccess applications that are installed in secure locations
User Account Control: Run all administrators in Admin Approval Mode
User Account Control: Switch to the secure desktop when prompting for elevation
User Account Control: Virtualize file and registry write failures to per-user locations

Figure 9.3: UAC settings for computers in an Active Directory domain are best configured through Group Policy.

In the following bulleted list is a discussion on each setting, its use, and the impacts its configuration can have for your infrastructure. The section following will include a discussion of a few common ways in which UAC's behavior can be adjusted to reduce the impact of its prompts on the environment:

- *Admin Approval Mode for the Built-in Administrator Account.* By default, the built-in Administrator account does not have UAC applied to it. Enabling this setting forces UAC to apply to this account.
- *Allow UIAccess applications to prompt for elevation without using the secure desktop.* When an application attempts to bring forward the prompt for elevation, by default it also switches to a special desktop called the "secure desktop." This special desktop grays out the standard desktop and displays only the elevation prompt itself. It is strictly limited in what it can process. Its limitations generally allow the user to click the elevation prompt and little else. The secure desktop is in place to prevent malware from attempting to spoof the user into clicking something inappropriate and undesired. However, there may be cases in which the secure desktop actually breaks some applications. This Group Policy setting allows the elevation prompt to appear without switching to this special desktop.

- *Behavior of the elevation prompt for administrators in Admin Approval Mode.* This setting configures how UAC behaves during a request for elevation. Three settings are possible. The first, *Prompt for consent* is the default behavior and configures UAC to behave as has been discussed to this point. *Prompt for credentials* takes the elevation prompt another step and requires a user to re-enter their credentials to complete the elevation. The idea with this selection is that elevations will be more closely scrutinized when the added step of re-entering credentials is required for elevation. The final selection, *Elevate without prompting*, instructs UAC to elevate the user automatically and without prompting.
- *Behavior of the elevation prompt for standard users.* Standard users can be forced to *Prompt for credentials*, which is the default behavior explained earlier. Alternatively, *Automatically deny elevation requests* can be selected. This has the effect of reverting UAC behavior for standard users back to what was experienced in Windows XP. By configuring the setting in this way, users without administrative privileges will again see the equivalent of an "Access Denied" error when they attempt an administrative activity.
- *Detect application installations and prompt for elevation.* This setting configures the behavior seen when attempting to install applications. UAC includes logic that looks for certain types of executables that appear to be software installations. When this setting is enabled and UAC sees an attempt to launch what it believes to be an installation, it automatically prompts for elevation. This alleviates the administrator from having to right-click an installation file and select Run as Administrator to elevate the installation. This setting is often disabled in situations in which desktop management applications such as Group Policy Software Installation or System Center Configuration Manager handle application installations.

> ☞ UAC looks at the file name to identify whether the executable is a software installation. If the filename includes the characters "install," "setup," or "uninst," then UAC considers the executable a valid installation and will automatically elevate on a double-click. Other files that require elevation for their processing would need to be right-clicked and launched through the Run as Administrator context menu.

- *Only elevate executables that are signed and validated.* This setting forces all executables to be digitally signed if they are to be elevated. Any executables that are not digitally signed will not be elevated, which has the effect of preventing their full functionality (or, indeed, their functionality at all). This can be an effective protection against malware but requires each and every executable in the environment to be correctly digitally signed, which can add a major administrative burden.
- *Only elevate UIAccess applications that are installed in secure locations.* Windows considers only the following locations as secure locations for the installation of software: \Program Files (and subfolders), \Program Files (x86) (and subfolders), and \Windows\system32\. By enabling this setting, if an application is not installed into one of these locations and requests elevation, its request will be denied. Effectively, this setting prevents the elevated launching of applications that are installed to insecure locations. As most legitimate applications install to secure locations, prevention of malware is a major reason for this setting.

- *Run all administrators in Admin Approval Mode.* This setting can be considered the master toggle switch for UAC. Setting this to Disabled effectively disables UAC and its related functionality.
- *Switch to the secure desktop when prompting for elevation.* Similar to the previous setting related to the secure desktop, this setting can be considered the master toggle switch for whether the secure desktop is used for all elevations or none.
- *Virtualize file and registry write failures to per-user locations.* As part of the security model for UAC, applications installed to the secure locations noted earlier are not allowed to store user-specific information in those locations. This protects the core installation of these applications from down-the-road user customizations. File and registry virtualization is a process used by UAC to spoof any writes to these locations towards alternate locations in the user profile and within HKEY_CURRENT_USER. This setting enables or disables file and registry virtualization for these legacy applications.

Common UAC Implementations

In comparison with the relative simplicity of runas and double logins, UAC's behavior can be somewhat complicated to understand. When and why it requires administrative elevation can be a challenge for the uninitiated. But the requirement for UAC to prompt for consent was designed to illustrate which processes require additional privileges. Even more important, when unwanted software such as malware attempts to elevate to accomplish its nefarious mission, this provides a way for an alert administrator to stop its activity.

The default behavior of UAC is achieved through no manipulation of its policy settings at all. Simply living with UAC's default behavior is one common UAC implementation. By using Windows Server 2008 and Windows Vista with the default configuration, UAC operates exactly as its designers intended. In short, it alerts when processes require administrative privileges.

Although the purported benefits of UAC are obvious, some in IT disagree with the level of responsibility it places on the individual person. Answering its oft-seen elevation prompts can grow irritating for the IT administrator who constantly requires administrative privileges to do his or her job. Even worse, users who have been given administrative privileges but are untrained on what they are seeing may simply always choose *Continue* rather than truly analyzing the process that is attempting to elevate. Thus, while UAC's intentions are excellent, its implementation may not be effective for some IT environments.

In some environments, it might be desirable to turn off UAC until an alternative solution is found. To completely disable UAC, configure the following policy settings:

- *Detect application installations and prompt for elevation.* Set to *Disabled*.
- *Behavior of the elevation prompt for standard users.* Set to *Automatically deny elevation requests*.
- *Run all administrators in Admin Approval Mode.* Set to *Disabled*.

There is a downside to completely eliminating UAC. One consequence of disabling UAC is that it eliminates a worthwhile OS protection against external attack. Also, shutting down UAC eliminates other protections gained as a function of Internet Explorer Protected Mode (IEPM), a special security mode that is also disabled with the disabling of UAC.

Because of these downsides, completely disabling UAC is not considered a best practice. An alternative solution that retains many of UAC's protections while eliminating its prompts is to configure UAC to operate in "quiet mode."

When UAC is configured to operate in quiet mode, it continues processing elevations as necessary. Users with administrative privileges still use two login sessions and switch between them as necessary to gain administrative privileges. However, this switching occurs in the background without prompting the user. Also, with quiet mode, the protections of IEPM remain in place. To configure UAC for quiet mode configure the following Group Policy setting: *Behavior of the elevation prompt for administrators in Admin Approval Mode.* Set to *Elevate without prompting*.

> The downside to quiet mode is that users are unaware of when elevations occur because elevations occur automatically without an expressed consent. Yet for some environments this may be a better solution than charging users with the responsibility for consent and living with its prompts.

Introducing the Windows Firewall with Advanced Security

The Windows firewall has historically been a feature of equal disdain and appreciation with Windows administrators. With Windows Server 2003 and Windows XP, the firewall was available but notoriously difficult to use. Administering the firewall through Group Policy was difficult due to Group Policy's text-based mechanisms for configuring the firewall and its necessary exclusions. Consequently, the built-in Windows firewall went relatively unused in many IT environments.

With the release of Windows Server 2008, the firewall gains some much-needed maturity, both in the functions it can support as well as your ability to manage it. Built-in to the Windows Firewall with Advanced Security (WFAS) are new profiles, a new ability for outbound filtering, better mechanisms for creating IPSec-based connection security between clients, and much improved management control through Group Policy. If you're familiar with its old and painful management through Administrative Templates in previous editions, you'll be happy to know that its configuration through Group Policy now closely mirrors its local GUI toolset.

Three Profiles

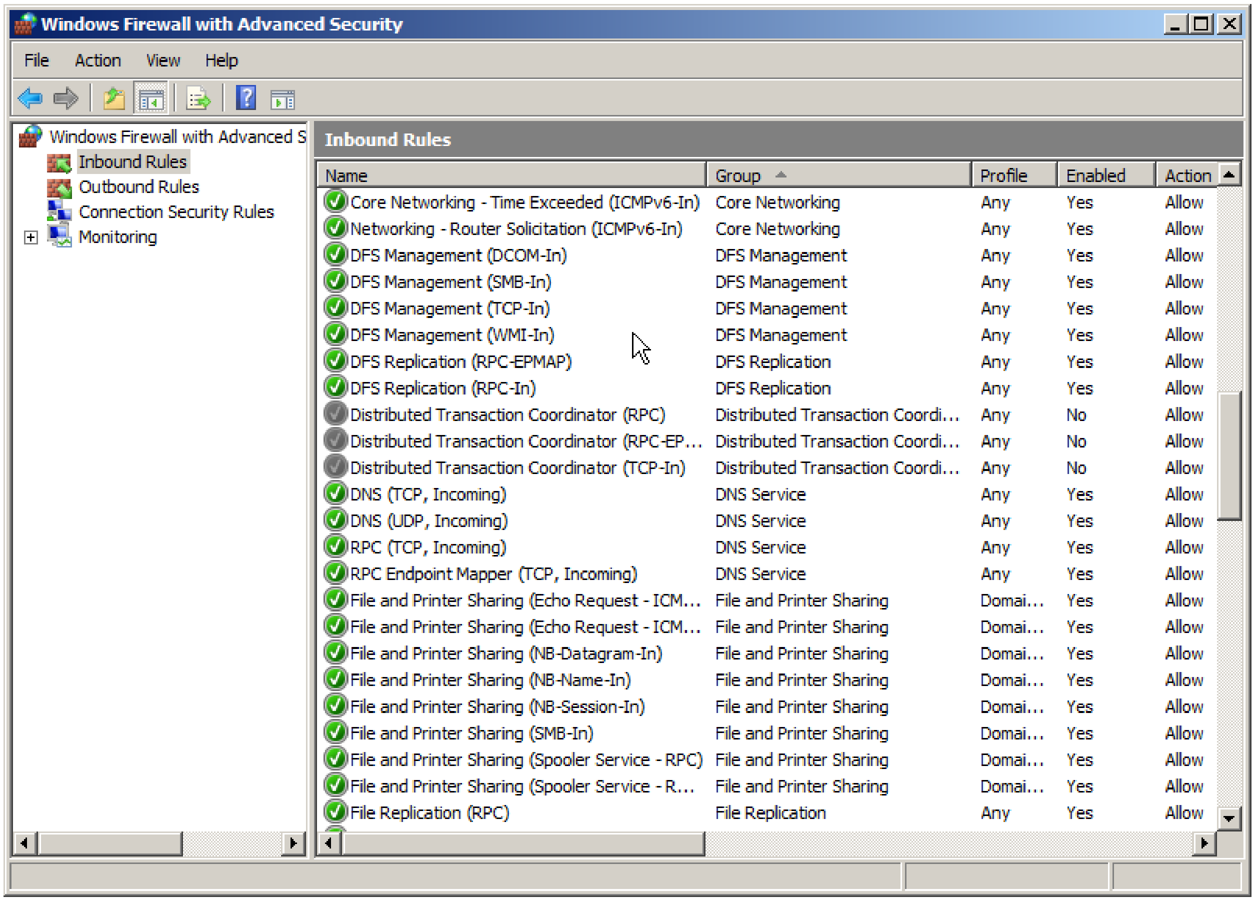
With Windows Server 2008 as well as with Windows Vista, WFAS includes three separate profiles for determining the type of currently attached network. Although these profiles are arguably more used by Windows Vista clients roaming between networks, they remain available for Windows Server 2008 instances as well:

- *The Public Profile.* This profile is roughly equivalent to what was called the Standard Profile in Windows XP. This profile is intended for use in untrusted network situations such as coffee shops and airports. The Public Profile is chosen by the logged on user as the computer connects to a new network and corresponds to the *Public location* network, seen as an icon of a park bench.
- *The Private Profile.* This profile, new to Windows Server 2008, is designed for use in partially trusted situations such as partner companies, home offices, or other areas where some expectation of security can be assured. Similar to the previous profile, the Private Profile is chosen by the logged on user as the computer connects to a new network. However, both the *Work* network and the *Home* network, seen as icons of an office building and a home, respectively, take the user to the Private Profile.
- *The Domain Profile.* This profile is automatically chosen when the server connects to a network where it can locate a domain controller for its attached domain. There is no user interface for choosing this network as it is always automatically chosen when the server can contact a domain controller.

Because they are generally always attached to the domain, there is a high likelihood that all your servers will always utilize the Domain Profile. These two additional profiles are available in the situation in which servers may move between networks.

Inbound & Outbound Rules

At the time of a Windows Server 2008 installation, around 90 inbound rules and 40 outbound rules are created by default. As seen in Figure 9.4, these rules relate to core networking, file and printer sharing, and numerous other core requirements for a new system. Because the server shown in Figure 9.4 is a domain controller, also present are additional rules in place to support Active Directory Domain Services and DNS functionality.

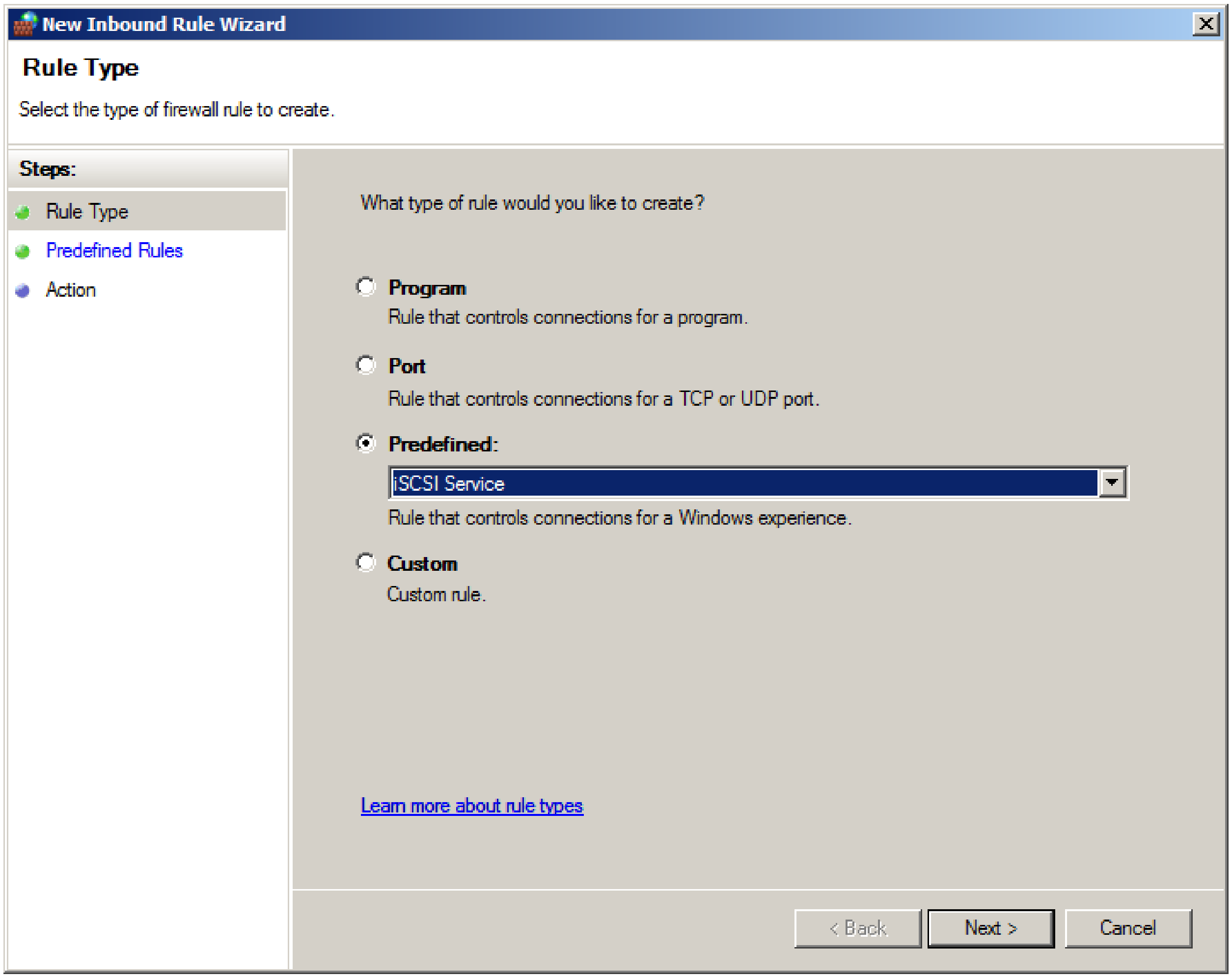

Windows Firewall with Advanced Security
File
Action
View
Help
Windows Firewall with Advanced S
Inbound Rules
Outbound Rules
Connection Security Rules
Monitoring
Inbound Rules

Name	Group	Profile	Enabled	Action
Core Networking - Time Exceeded (ICMPv6-In)	Core Networking	Any	Yes	Allow
Networking - Router Solicitation (ICMPv6-In)	Core Networking	Any	Yes	Allow
DFS Management (DCOM-In)	DFS Management	Any	Yes	Allow
DFS Management (SMB-In)	DFS Management	Any	Yes	Allow
DFS Management (TCP-In)	DFS Management	Any	Yes	Allow
DFS Management (WMI-In)	DFS Management	Any	Yes	Allow
DFS Replication (RPC-EPMAP)	DFS Replication	Any	Yes	Allow
DFS Replication (RPC-In)	DFS Replication	Any	Yes	Allow
Distributed Transaction Coordinator (RPC)	Distributed Transaction Coordi...	Any	No	Allow
Distributed Transaction Coordinator (RPC-EP...	Distributed Transaction Coordi...	Any	No	Allow
Distributed Transaction Coordinator (TCP-In)	Distributed Transaction Coordi...	Any	No	Allow
DNS (TCP, Incoming)	DNS Service	Any	Yes	Allow
DNS (UDP, Incoming)	DNS Service	Any	Yes	Allow
RPC (TCP, Incoming)	DNS Service	Any	Yes	Allow
RPC Endpoint Mapper (TCP, Incoming)	DNS Service	Any	Yes	Allow
File and Printer Sharing (Echo Request - ICM...	File and Printer Sharing	Domai...	Yes	Allow
File and Printer Sharing (Echo Request - ICM...	File and Printer Sharing	Domai...	Yes	Allow
File and Printer Sharing (NB-Datagram-In)	File and Printer Sharing	Domai...	Yes	Allow
File and Printer Sharing (NB-Name-In)	File and Printer Sharing	Domai...	Yes	Allow
File and Printer Sharing (NB-Session-In)	File and Printer Sharing	Domai...	Yes	Allow
File and Printer Sharing (SMB-In)	File and Printer Sharing	Domai...	Yes	Allow
File and Printer Sharing (Spooler Service - RPC)	File and Printer Sharing	Domai...	Yes	Allow
File and Printer Sharing (Spooler Service - R...	File and Printer Sharing	Domai...	Yes	Allow
File Replication (RPC)	File Replication	Any	Yes	Allow

Figure 9.4: Inbound firewall rules displayed on an example server operating as a domain controller.

This shows another benefit to Microsoft's componentization activity. Any time a new Role, Role Service, or Feature is installed, part of its installation is to enable the correct inbound and outbound firewall ports that ensure proper functionality. This process alleviates much of the need to manually identify and open the necessary ports sometimes required with previous versions.

Should you find the need to create additional exclusions, creating a new rule is done through the *New Inbound Rule Wizard* or the *New Outbound Rule Wizard* depending on the direction of the rule you need to create. As you can see in the first screen of the wizard, shown in Figure 9.5, it is possible to create rules based on a stated port, all ports used by an installed application, or even predefined "Windows experiences" that relate to some functionality being used on the server. In the example seen in Figure 9.5, the Windows experience being shown is the *iSCSI Service*. Configuring a rule for any Windows experience automatically opens the necessary exclusions in the firewall to support the needs of that experience.



New Inbound Rule Wizard

Rule Type

Select the type of firewall rule to create.

Steps:

- Rule Type
- Predefined Rules
- Action

What type of rule would you like to create?

○ **Program**
Rule that controls connections for a program.

○ **Port**
Rule that controls connections for a TCP or UDP port.

● **Predefined:**
iSCSI Service
Rule that controls connections for a Windows experience.

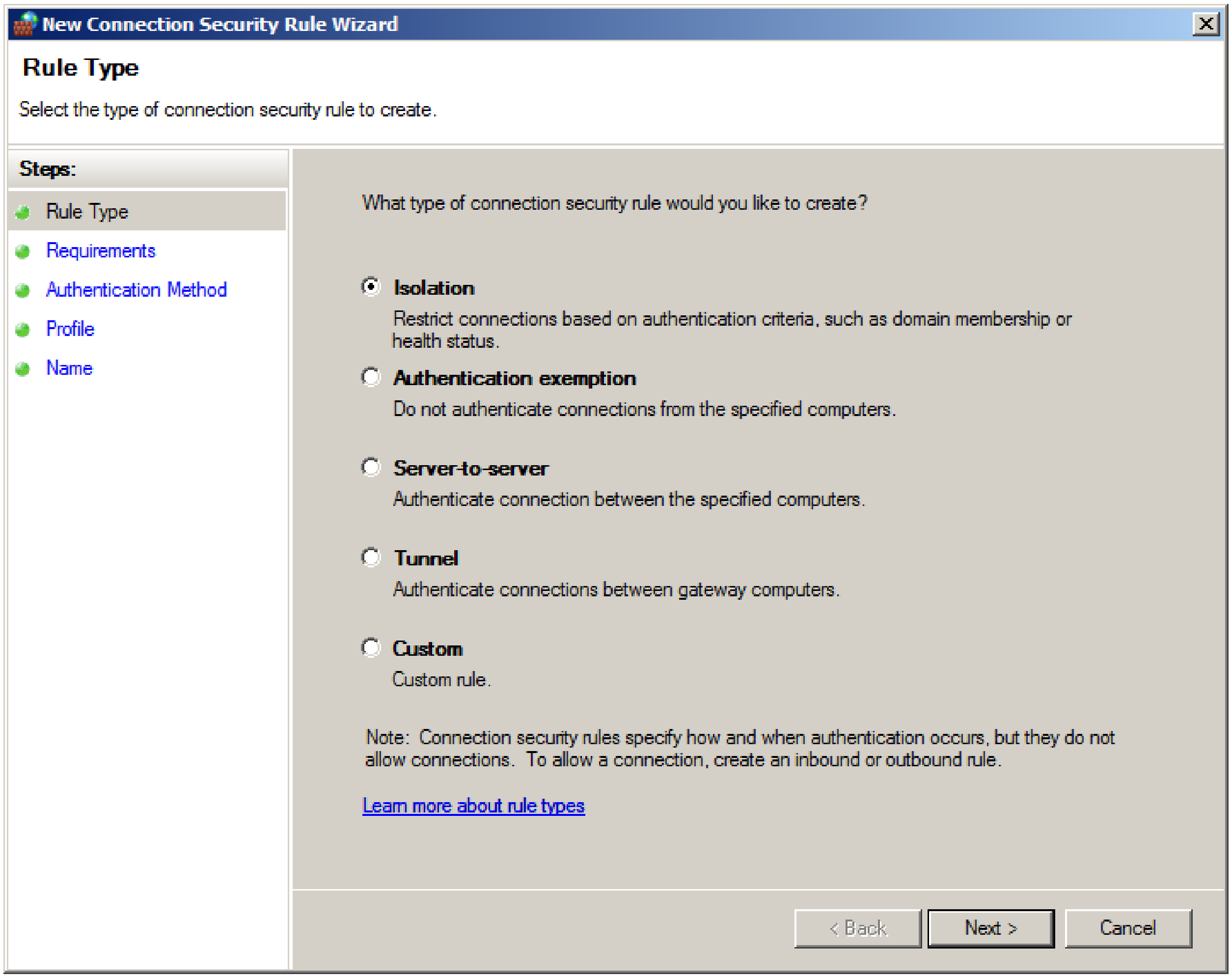
○ **Custom**
Custom rule.

Learn more about rule types

Figure 9.5: Configuring a new rule uses a wizard-like interface that enables rules based on actual ports, ports used by an installed application, or preconfigured Windows experiences.

Connection Security Rules

The process for creating exclusions was challenging in previous OS versions yet not excessively complicated. However, creating connection security between two or more clients was. Connection security rules are mechanisms for two or more computers to authenticate themselves to each other prior to communicating. These rules leverage IPSec authentication to validate each other at a server level before either is willing to send or receive traffic from the other host. Connection security rules in Windows Server 2008 as well as the wizard used to create them are a much-improved way of setting up server-level authentication and even network traffic encryption. Rules can be created for setting up isolation groups, server-to-server authentication, and even tunnel authentication between gateway computers. Figure 9.6 shows an example of the first screen in the *New Connection Security Rule Wizard* that shows these options.



Rule Type

Select the type of connection security rule to create.

What type of connection security rule would you like to create?

● **Isolation**
Restrict connections based on authentication criteria, such as domain membership or health status.

○ **Authentication exemption**
Do not authenticate connections from the specified computers.

○ **Server-to-server**
Authenticate connection between the specified computers.

○ **Tunnel**
Authenticate connections between gateway computers.

○ **Custom**
Custom rule.

Note: Connection security rules specify how and when authentication occurs, but they do not allow connections. To allow a connection, create an inbound or outbound rule.

Learn more about rule types

Figure 9.6: Connection security rules of various types are created through the New Connection Security Rule Wizard.

Centralized Management & Group Policy

Making these handy new graphical wizards even more effective is their mirroring into the Group Policy Management Editor (GPME). Unlike previous OS versions which used the text-based Administrative Tools interface, Windows Server 2008 leverages a more user-friendly graphical interface for enabling the firewall, creating new rules, and configuring connection security. Effectively, what you see in a local firewall configuration is very similar to what you see within a configured Group Policy.

Two Common Uses of the Windows Firewall with Advanced Security

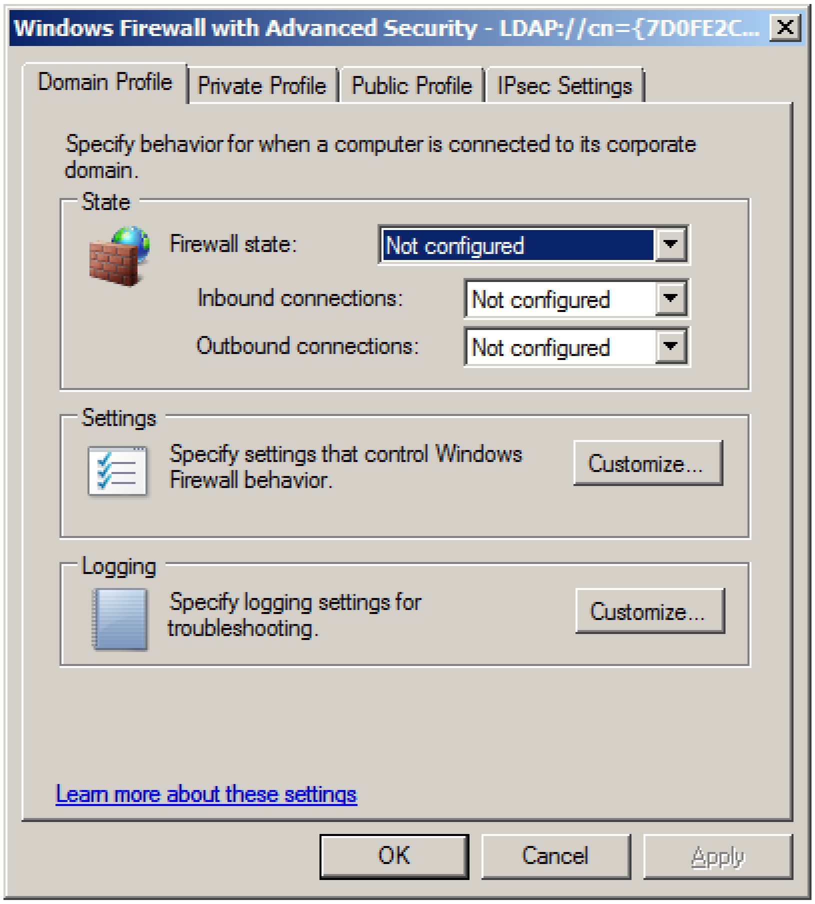
Although understanding the benefits of WFAS is helpful, learning a few specific ways in which it is often used by IT organizations will help in securing your Windows infrastructure. Though there are as many ways of configuring the firewall as there are firewall settings, we'll focus here on two common uses. These uses are selected because they are easy to set up and can offer some of the biggest "bang for the buck" in environments of any size.

> The following two examples assume an environment that is fully comprised of Windows Vista and Windows Server 2008 computers. For environments that contain down-level OS versions, the steps used in these examples will be somewhat different.

Example 1: Securing Laptops While off the Domain

Back when Microsoft released Windows XP Service Pack 2 (SP2), its automatic enabling of the firewall for all connections immediately broke networking in many environments. Consequently, many decided to disable the firewall completely. This first example leaves WFAS disabled for all computers while they remain attached to their home domain. Any computer that detaches from its home domain and roams to an alternative network location—such as the aforementioned coffee shops or airport lounges—will automatically enable the firewall for all inbound connections. This has the effect of automatically enforcing a high level of protection for laptop computers that leave the LAN and roam elsewhere.

To do so, first create a new Group Policy Object (GPO) and edit that GPO in the GPME. Navigate to Computer Configuration | Policies | Windows Settings | Security Settings | Windows Firewall with Advanced Security | Windows Firewall with Advanced Security. Once there, click the Windows Firewall Properties link to see a screen similar to Figure 9.7.



Specify behavior for when a computer is connected to its corporate domain.

Firewall state: Not configured

Inbound connections: Not configured

Outbound connections: Not configured

Specify settings that control Windows Firewall behavior.

Specify logging settings for troubleshooting.

Figure 9.7: The master properties page for configuring WFAS through Group Policy.

Within this wizard, we first need to configure the Domain Profile to disable the firewall. This means that computers attached to the domain will not filter traffic through the firewall's rules. For the Public and Private Profiles, we want to block any inbound connections while allowing all outbound connections. While doing this, we also want to ensure that local users cannot change the firewall settings, enforcing the policy for all computers. To do so, use the following steps:

- On the *Domain Profile* tab, set the *Firewall state* to *Off*. In the *Settings* box, click *Customize*. In the *Rule merging* box of the resulting screen, set both *Apply local firewall rules* and *Apply local connection security* rules to *No*. These two settings prevent local users from changing firewall settings at their local computer.
- On both the *Private Profile* and *Public Profile* tabs, set the *Firewall state* to *On (recommended)*. Set *Inbound connections* to *Block all connections*. Set *Outbound connections* to *Allow (default)*. In the *Settings* box, click *Customize*. In the *Rule merging* box of the resulting screen, set both *Apply local firewall rules* and *Apply local connection security rules* to *No*. In the *Firewall settings* box of the same screen, set *Display a notification* to *No*. This prevents the user from seeing notifications about blocked inbound programs.

Once these settings have been configured, close the GPO and apply it to the domain or an OU that contains the computers you want to manage with this policy. Once computers have processed the policy, any time they leave the LAN, their firewalls will automatically enable and reject all inbound traffic. Upon returning to the LAN, the firewall will automatically disable and process all traffic.

☞ In this example, we configure both the Private and Public Profiles because the Vista user has the option of choosing a profile any time they connect to a new network. If you want to provide an option for users who connect to partially trusted networks that includes some firewall exclusions, consider configuring them into the Private Profile. Users will then be able to choose either the Home or Work network as they connect to lessen the policy-assigned restrictions.

Example 2: Simple Domain Isolation

One issue with most Windows networks is that authentication is traditionally done after two computers make the decision to communicate with each other. With authentication occurring after communication begins, this leaves open the possibility for compromise should a rogue computer plug into the network. If your environment allows computers that are not members of the domain to attach to the network, you are putting yourself at risk should an attached computer contain some form of replicating malware. More importantly, any computer that connects to the network can also begin searching around that network for information it may contain. If any file share anywhere on your network has been configured to grant *Read* access to the *Everyone* group, that information could get disclosed to the rogue computer.

Either of these two possibilities are nightmares for open networks. But with the combination of Windows Vista and Windows Server 2008 comes an easy way to prevent non-domain computers from ever communicating with computers on your domain. Called domain isolation, this way of configuring WFAS requires computers to authenticate to each other before they ever begin communication. Computers that aren’t a member of the domain don’t have the necessary IPSec authentication, and therefore are denied access to communicate with computers on the domain.

💣 Because domain isolation is designed to require authentication, it can impact your network operations. If you have non-domain computers like UNIX/Linux machines or Macintosh desktops that don’t support connection security, implementing domain isolation can prevent these computers from communicating with others on the domain. Before implementing the information shown here, test it in a separate environment first.

To set up domain isolation, first create a new GPO and edit it in the GPME. Configuring domain isolation is done from the same *Windows Firewall with Advanced Security* location as discussed in the earlier example. Configure the GPO using the following steps:

- Within the *Windows Firewall with Advanced Security* wizard, navigate to the *Domain Profile*. Set the *Firewall* state to *On (recommended)*. Set *Inbound connections* to *Block*. Set *Outbound connections* to *Allow (default)*. In the *Settings* box click *Customize*. In the *Rule merging* box of the resulting screen, set both *Apply local firewall rules* and *Apply local connection security rules* to *No*. On the *IPSec Settings* tab in the *IPSec exemptions* box, set *Exempt ICMP from IPSec* to *Yes*. Click *OK* to close the wizard. This process enables the firewall but instructs it to block any connection not specifically excluded. In the next step, we'll configure an exclusion that allows all traffic. This "double negative" is necessary to ensure that all traffic is passed through the firewall's filters.
- In this step we will create the rule that allows traffic. Right-click *Inbound Rules* and choose to create a *New Rule*. In the resulting screen, choose to create a *Custom* rule. For each of the following screens in the wizard, do the following:
    - *Program.* Choose *All Programs* and click *Next*.
    - *Protocol and Ports.* Leave the default configuration and click *Next*.
    - *Scope.* Leave the default configuration and click *Next*.
    - *Action.* Select *Allow the connection* and click *Next*.
    - *Users and Computers.* Leave the default configuration and click *Next*.
    - *Profile.* Select to apply the rule to only the *Domain* profile.
    - *Name.* Provide a *Name* and *Description* for the rule.
- Now that all traffic is passing through the firewall's filters, the next step is to require authentication. Once the inbound rule has been created, right-click *Connection Security Rules* and choose to create a *New Rule*. In the resulting screen, choose to create an *Isolation* rule. For each of the following screens in the wizard, do the following:
    - *Requirements.* Choose *Require authentication for inbound and request authentication for outbound connections*.
    - *Authentication Method.* Leave the default configuration and click *Next*.
    - *Profile.* Select to apply the rule to only the *Domain* profile.
    - *Name.* Provide a *Name* and *Description* for the rule.

- Due to issues with the propagation of Group Policy, it is a good idea to exclude client-to-domain controller communication from the authentication requirement. In this case, we configure this traffic to request authentication from domain controllers rather than require it. By doing so, clients will always be guaranteed the ability to communicate with domain controllers. To do this, right-click *Connection Security Rules*, and choose to create a *New Rule*. In the resulting screen, choose to create an *Isolation* rule. For each of the following screens in the wizard, do the following:
    - *Requirements*. Choose *Request authentication for inbound and outbound connections*.
    - *Authentication Method*. Leave the default configuration and click *Next*.
    - *Profile*. Select to apply the rule to only the *Domain* profile.
    - *Name*. Provide a *Name* and *Description* for the rule.
- To complete the previous step and target this exemption for domain controllers only, under *Connection Security Rules*, double-click the rule just created. On the Computers tab of the resulting window in the *Endpoint 2* box, add the IP addresses for all domain controllers.

Once complete, close the GPME and apply the GPO to the domain. Once computers begin to receive and apply this Group Policy, they will begin requiring authentication from any computer that attempts to communicate with the single exception of the domain controllers configured in the last steps. Any rogue computer that connects to the LAN will not be able to communicate with any domain-attached computer.

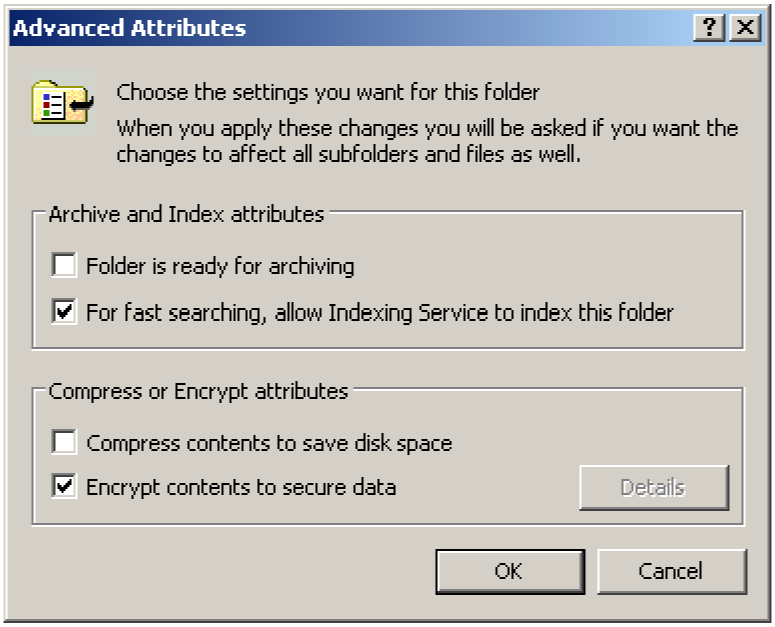
Be aware that if your environment includes non-domain computers, computers of other OSs, network appliances, or any other network-attached equipment that is not a Windows Vista or Windows Server 2008 computer, you will need to create authentication exemptions for these devices. Additionally, you might want to exempt domain services such as DNS, WINS, or DHCP from authentication.

Installing and Managing BitLocker

Prior to the release of Windows Vista and Windows Server 2008, Microsoft's primary solution for encrypting files on a Windows machine was the Encrypting File System (EFS). Using EFS, users and administrators could choose to encrypt individual files and folders on a Windows system, protecting them from inappropriate data disclosure.

But there has always been a problem with EFS's file- and folder-based architecture. Encrypting specific files and folders of interest to a user indeed protects those locations from exposure. However, the processing of those files and folders by applications does not always keep every copy of the object in a protected location. This is perhaps best explained through an example.

Using EFS, let's assume that a user desires to encrypt the contents of the *C:\MyData* folder. Using Windows XP, they view the properties of that folder and by clicking *Advanced*, they see the wizard shown in Figure 9.8. There, they are able to select the *Encrypt contents to secure data* check box. This process encrypts the contents of the *C:\MyData* folder.



Choose the settings you want for this folder

When you apply these changes you will be asked if you want the changes to affect all subfolders and files as well.

Archive and Index attributes

□ Folder is ready for archiving

☑ For fast searching, allow Indexing Service to index this folder

Compress or Encrypt attributes

□ Compress contents to save disk space

☑ Encrypt contents to secure data

Figure 9.8: The Windows XP wizard page used to encrypt individual files and folders with EFS.

At some point in the future, the user needs to work with a document in that folder. The user double-clicks the document to spawn its linked application. To process the document, this application requires a temporary copy of the document to be created, which it stores in the location *C:\Temp*. This temporary copy is now stored in a folder that is outside the protections of the encrypted folder, storing the document in clear text. Making this situation even more problematic, even if the application deletes the clear-text temporary copy when it has finished processing, the deleted version can likely be later undeleted from the temporary location.

Due to these limitations with file- and folder-based encryption, Microsoft recognized that the only way to ensure documents remain encrypted no matter how or where they are used is in encrypting the entire disk drive. The BitLocker Drive Encryption feature found in both new OSs accomplishes that goal.

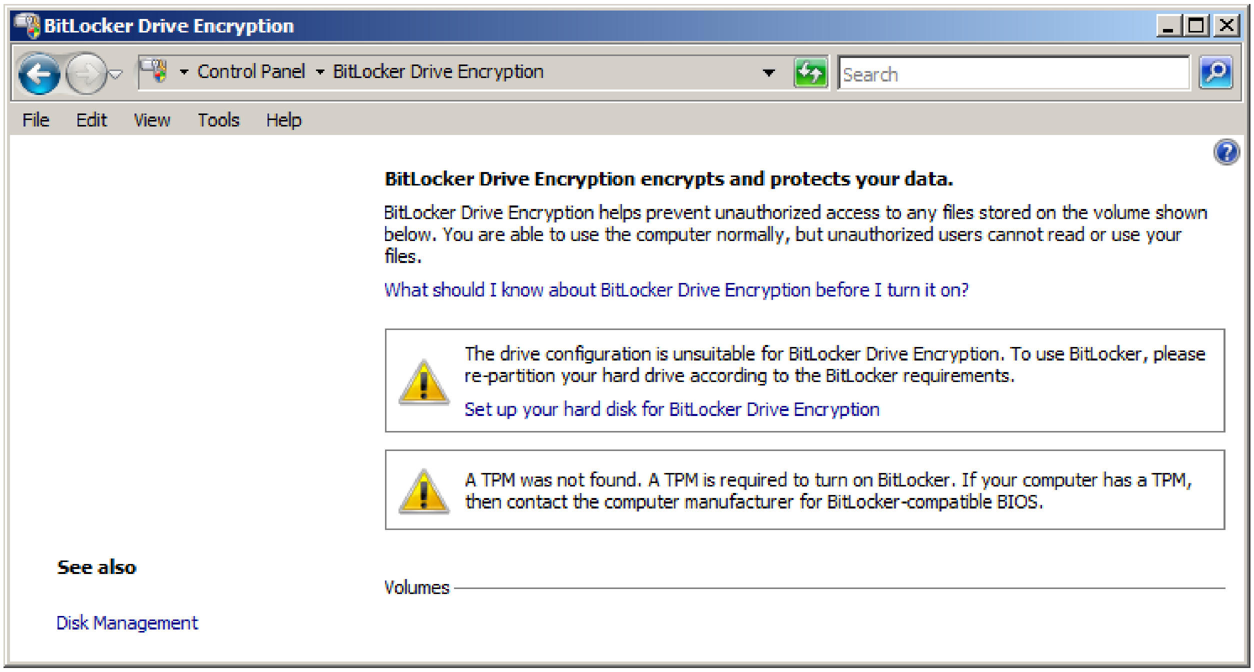
BitLocker is a volume encryption service that arrives in Windows Server 2008 as an installable feature. When BitLocker is installed to a computer, it works with on-board hardware to identify at every boot whether the hardware of the computer has been modified in any way. If the system drive appears to have been modified—such as moved to another computer, for example—BitLocker will not decrypt the encrypted volume until a recovery password is entered. That recovery password, which can take the form of a 48-digit password or a 256-bit key stored in a file, instructs BitLocker to decrypt the volume and allow the system to boot. Although BitLocker may not be necessary for all systems in an IT environment—its encrypting and decrypting activities add a small performance hit to a server—it can be an excellent addition to servers in quasi-secured locations such as branch offices or servers that contain highly sensitive material.

✎ If an attacker were to remove a drive from a server protected by BitLocker, they would find little more than random characters on the drive. As BitLocker uses AES encryption in various strengths, it is virtually impossible using today's technology to decrypt the drive without the required restore key. For extremely sensitive uses, it is also possible to require a startup password to be entered every time the server is booted.

Prerequisites & Installation

There are a number of special prerequisites required for a system to support BitLocker. First, a system must be configured with a Trusted Computing Group (TCG)-compliant BIOS. It must also include a special chip built-in to its hardware called a Trusted Platform Module (TPM) with a minimum version of 1.2. BitLocker leverages these special hardware components for its encrypting and decrypting activities as well as its hardware verification. Contact your server manufacturer to determine whether your hardware includes the correct components necessary for BitLocker functionality.

Another way to check for the presence of a TPM is by installing the *BitLocker Drive Encryption* feature to a Windows Server 2008 instance. Once installed, navigate to Control Panel | BitLocker Drive Encryption where you will see a screen similar to Figure 9.9. You'll notice there that two prerequisites are not present, one of which explains that the necessary TPM module is not present on the hardware.


BitLocker Drive Encryption
Control Panel ▸ BitLocker Drive Encryption
Search
File Edit View Tools Help

BitLocker Drive Encryption encrypts and protects your data.

BitLocker Drive Encryption helps prevent unauthorized access to any files stored on the volume shown below. You are able to use the computer normally, but unauthorized users cannot read or use your files.

What should I know about BitLocker Drive Encryption before I turn it on?

The drive configuration is unsuitable for BitLocker Drive Encryption. To use BitLocker, please re-partition your hard drive according to the BitLocker requirements.

Set up your hard disk for BitLocker Drive Encryption

A TPM was not found. A TPM is required to turn on BitLocker. If your computer has a TPM, then contact the computer manufacturer for BitLocker-compatible BIOS.

See also

Disk Management

Volumes

Figure 9.9: The BitLocker Drive Encryption Control Panel will alert when necessary prerequisites are not present on the system.

Also necessary and shown in Figure 9.9 is a separate and unencrypted partition of at least 1.5 gigabytes that will act as the system volume. This volume will contain the bootstrap files necessary to boot the computer. Once the system has sufficiently booted, BitLocker can then decrypt any encrypted drives and enable the OS to continue the boot process.

> ✎ If you've already built your server and have partitions in place, you will need to reconfigure them to create this necessary system volume. Microsoft provides a *BitLocker Drive Preparation Tool* that assists with the process of repartitioning the system. Once installed, the tool will create a new volume of the proper 1.5 gigabyte size and mark it as the active partition for booting the system. The tool along with information on its use can be found at http://support.microsoft.com/kb/933246.

Once your system drives have been properly partitioned and a TPM correctly identified on the system, turning on basic drive encryption with BitLocker is done through the same Control Panel discussed earlier. Simply click the link titled *Turn on BitLocker*. If the TPM in the system has not been initialized at this point, the *Initialize TPM Security Hardware* wizard will appear. On the *Save the recovery password* page of the wizard, choose to save the BitLocker recovery password to a USB drive, a folder, or printed out. This password must be stored in a secure location as it is the only mechanism for recovering the volume.

On the *Encrypt the volume* page, select the *Run BitLocker system check* box to have BitLocker run a set of tests that ensure that the service can read the recovery and encryption keys prior to encrypting the drive. At this point, the system will reboot and test key retrieval. If the key retrieval test functions correctly, the system will begin encrypting the drive.

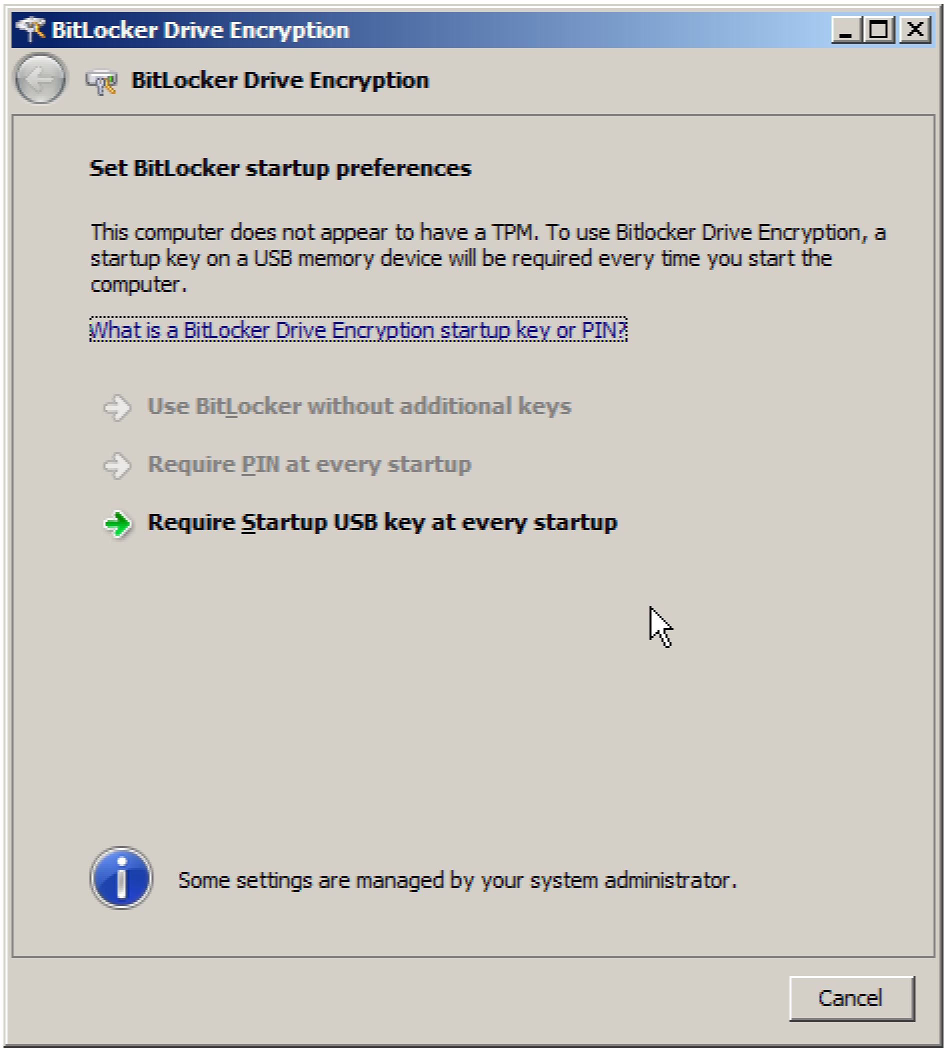
> ✎ BitLocker can also be configured at the command line using the script *manage-bde.wsf*. More information about using this script for encrypting data drives as well as detailed instructions on enabling, disabling, and recovering BitLocker can be found at http://technet.microsoft.com/en-us/library/cc732725.aspx.

Installing BitLocker Without a TPM

Although strongly suggested, the TPM chip is not a true prerequisite. It is possible to run BitLocker using a separate startup key stored on a USB flash drive. The startup key must be inserted into the computer during the boot process to authenticate the startup of the system. Installing without a TPM requires enabling BitLocker's advanced functions through either Group Policy or a Local Policy.

To enable the advanced functions, navigate to Computer Configuration | Administrative Templates | Windows Components | BitLocker Drive Encryption. There, enable the setting *Control Panel Setup: Enable advanced startup options*. When enabling the policy, select the *Allow BitLocker without a compatible TPM* check box.

Once this policy has been applied to the computer, the remaining steps are equivalent to those discussed earlier with one exception. Without a TPM, a USB flash drive must be plugged into the system at every startup. Figure 9.10 shows the extra screen in the installation wizard that is displayed when attempting to install BitLocker without a TPM.


BitLocker Drive Encryption
BitLocker Drive Encryption

Set BitLocker startup preferences

This computer does not appear to have a TPM. To use Bitlocker Drive Encryption, a startup key on a USB memory device will be required every time you start the computer.

What is a BitLocker Drive Encryption startup key or PIN?

Use BitLocker without additional keys
Require PIN at every startup
Require Startup USB key at every startup

Some settings are managed by your system administrator.

Figure 9.10: The extra screen in the installation wizard that is displayed when attempting to install BitLocker without a TPM.

BitLocker and Group Policy

Although BitLocker's configuration can be controlled using Local Policy at the individual machine, using Group Policy to cohesively control its installation across all instances is considered a best practice. Using Group Policy ensures that recovery keys are always stored in Active Directory (AD), the right level of encryption is set across all systems, and the correct password settings are used. Seven Group Policies are available:

- *Turn on BitLocker backup to Active Directory Domain Services.* This setting enables the automatic backup of recovery information to AD. By backing up this information to AD, the likelihood of the loss of a recovery key is reduced.
- *Control Panel Setup: Configure recovery folder.* This setting identifies the local folder where recovery information is stored on each computer protected with BitLocker. Enabling the local storage of recovery information in addition to a secondary AD backup further ensures that recovery information is readily available in the case that a system protected with BitLocker needs to be recovered.
- *Control Panel Setup: Configure recovery options.* This setting identifies which options can be used to recover a system. Options include the 256-bit recovery key as well as the 48-digit recovery password.
- *Control Panel Setup: Enable advanced startup options.* Discussed previously, this setting is used to instruct BitLocker to operate on systems that do not have a TPM installed. It can additionally be used to control whether startup keys and PINs are required, disallowed, or optional.
- *Configure encryption method.* With this setting can be controlled which of the four types of AES encryption and encryption strengths will be used by BitLocker.
- *Prevent memory overwrite on restart.* BitLocker by default instructs the computer to overwrite memory at every restart to ensure that BitLocker secrets do not remain resident. This memory overwrite process also increases the time needed to complete a restart. This setting selects whether memory overwrites are enforced or prevented at each restart.
- *Configure TPM platform validation profile.* For systems with a TPM, this setting allows the selection of which components will be validated by the TPM before allowing the system to gain access to the encrypted volume. These validations ensure that the drive has not been relocated to new hardware. When the TPM has identified that a change has occurred, the system will enter into recovery mode.

As stated earlier, BitLocker is likely not a solution for every server in your environment. But for those servers that have the highest risk for compromise along with the greatest impact if they do, it can be a lifesaver.

Windows Server 2008 Is Microsoft's Most Secure OS to Date

With features both exposed to the administrator as well as those under its covers, Windows Server 2008 provides a stable platform upon which to host workloads required by today's businesses. As discussed in this chapter, controllable elements such as UAC, WFAS, and BitLocker, as well as core changes to its operations, all aggregate to increase Windows Server 2008's overall security.

Along with those security improvements comes a related increase in overall system uptime. This chapter has attempted to show where Windows Server 2008's built-in capabilities can enhance your data center's security profile while at the same time ensuring you systems remain up and operational.

In the next and final chapter, we'll delve into one more specific feature of Windows Server 2008 that continues this discussion on uptime and reliability: Windows Server Failover Clustering. With the right hardware in place, Microsoft's improvements to failover clustering can significantly improve the resiliency of services for business with high-reliability needs.