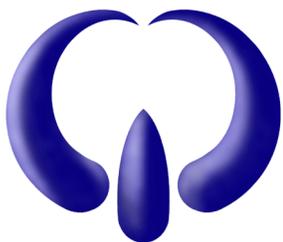




realtimepublishers.com<sup>tm</sup>

# *The Definitive Guide<sup>tm</sup> To*

# Windows Server 2003 Terminal Services



**triCerat**  
Software®

*Gresyon Mitchem*

Chapter 6: Managing Security and Virus Protection .....	128
Viruses, Worms, and Trojan Horses...Oh My! .....	128
Internet Explorer Enhanced Security Configuration.....	129
Changes Made by Internet Explorer Enhanced Security Configuration.....	131
Managing Approved ActiveX Controls .....	133
Implementing Windows Automatic Updates.....	134
Using SUS.....	137
Deploying Service Packs and Hotfixes.....	142
Using Group Policy to Deploy Service Packs .....	142
Deploying Hotfixes.....	143
Using a ZAP File .....	144
Using a Shutdown Script .....	145
Virus Protection Software Best Practices .....	147
Putting It All Together .....	147
Example One: Anytown Little Theatre.....	147
Example Two: BigBusiness, Inc.....	148
Summary .....	150

## Copyright Statement

© 2003 Realtimedpublishers.com, Inc. All rights reserved. This site contains materials that have been created, developed, or commissioned by, and published with the permission of, Realtimedpublishers.com, Inc. (the "Materials") and this site and any such Materials are protected by international copyright and trademark laws.

THE MATERIALS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. The Materials are subject to change without notice and do not represent a commitment on the part of Realtimedpublishers.com, Inc or its web site sponsors. In no event shall Realtimedpublishers.com, Inc. or its web site sponsors be held liable for technical or editorial errors or omissions contained in the Materials, including without limitation, for any direct, indirect, incidental, special, exemplary or consequential damages whatsoever resulting from the use of any information contained in the Materials.

The Materials (including but not limited to the text, images, audio, and/or video) may not be copied, reproduced, republished, uploaded, posted, transmitted, or distributed in any way, in whole or in part, except that one copy may be downloaded for your personal, non-commercial use on a single computer. In connection with such use, you may not modify or obscure any copyright or other proprietary notice.

The Materials may contain trademarks, services marks and logos that are the property of third parties. You are not permitted to use these trademarks, services marks or logos without prior written consent of such third parties.

Realtimedpublishers.com and the Realtimedpublishers logo are registered in the US Patent & Trademark Office. All other product or service names are the property of their respective owners.

If you have any questions about these terms, or if you would like information about licensing materials from Realtimedpublishers.com, please contact us via e-mail at [info@realtimedpublishers.com](mailto:info@realtimedpublishers.com).

## Chapter 6: Managing Security and Virus Protection

Blaster, Love Bug, Nimda, Melissa—computer viruses and worms are a fact of life these days, so Windows infrastructure design must take them into account. Virus scanning software, firewalls, and patch management should be a part of even the smallest environments. If users will be accessing email or browsing the Internet from terminal servers, you will need to be vigilant in keeping your servers safe and virus free. Luckily, the restrictive permissions on a terminal server make it difficult for users to introduce viruses to the system; nonetheless, you should be prepared.

In this chapter, we will explore available options for keeping servers secure and up to date, including Microsoft Automatic Updates and Software Update Services, as well as best practices for implementing these options in your environment. In addition, I will highlight considerations for installing virus scanning software on terminal servers.

### Viruses, Worms, and Trojan Horses...Oh My!

There are many types of malicious code (malware) from which you must protect your systems. Although malware is often referred to as viruses, there are distinct differences between viruses, worms, and Trojan horses.

A *virus* is a small program that is written to alter—without the knowledge of the user—the way that a computer operates. To be called a virus, the program must be executable code—either in the form of a standalone program or as a macro contained in another file—and must be able to copy itself so that it can continue to execute once the initial program or macro is terminated. Examples of viruses include:

- Boot sector viruses, such as Michelangelo and Disk Killer, which install themselves in the boot sector of your hard disk so that they begin execution before the OS
- Macro viruses, such as Melissa and Nice Day, which are contained inside macros in Office documents. These viruses execute when the document is opened, and attempt to infect other Office documents or the normal template so that all new documents created on the computer are infected.

A *worm* is a program that takes advantage of poor security design or security flaws in the OS. Worms are able to spread from one computer to another without a host file, although some spread by copying an infected file from one computer to another. Blaster is an example of a worm that takes advantage of a security flaw in the Windows remote procedure call (RPC), allowing it to remotely execute code on other computers on the network to replicate itself.

A Trojan horse is a program that is contained inside of another, seemingly desirable, application. When you execute the host program (for example, a screen saver, an e-greeting card, or a shareware application), the Trojan horse is installed on your system. Trojan horses can act as spyware, sending private information to a third party, and are used in distributed Denial of Service (DoS) attacks. The main difference between a virus and a Trojan horse is that a Trojan horse does not self-replicate—it must be manually executed to infect your system.

 A DoS attack makes a large number of requests against a specific server or URL, keeping the server busy so that it cannot service legitimate requests. A distributed DoS attack uses a virus, worm, or Trojan horse to enslave multiple computers across the Internet and directs them all to attack the server simultaneously.

You must protect your terminal servers from all three types of malware. You can do so through a combination of file system permissions, Group Policy settings, virus protection software, and security patches distributed by Microsoft.

If you accept the default file system permissions and keep your users restricted to the Users group (not elevating them to Power Users), your system is protected from most viruses and Trojan horses that attempt to install themselves into system files or the system root or program files directories. In addition, you can use Group Policy to enforce additional restrictions—limiting which processes and applications a user can execute, preventing users from saving files to the C drive of the server, and so on—to further protect your systems.

 For more information about using Group Policy to secure your terminal server, see Chapter 4.

Even with the enhanced security that WS2K3 implements on the file system and the additional restrictions you can apply through Group Policy, it is a good practice to install and maintain virus protection software and a patch-management system—virus authors are always coming up with new ways to harm your system.

## Internet Explorer Enhanced Security Configuration

Internet Explorer Enhanced Security Configuration is a new option included with WS2K3. When installed, it changes the default security settings in Internet Explorer to limit the exposure to potentially damaging code found in Web content and application scripts.

Internet Explorer Enhanced Security Configuration is installed by default for all user groups on WS2K3 unless you are adding the Terminal Services role—the configuration of Internet Explorer Enhanced Security Configuration on a terminal server depends on the method used to install the OS. Table 6.1 shows the configuration of Internet Explorer Enhanced Security Configuration under different installation methods.

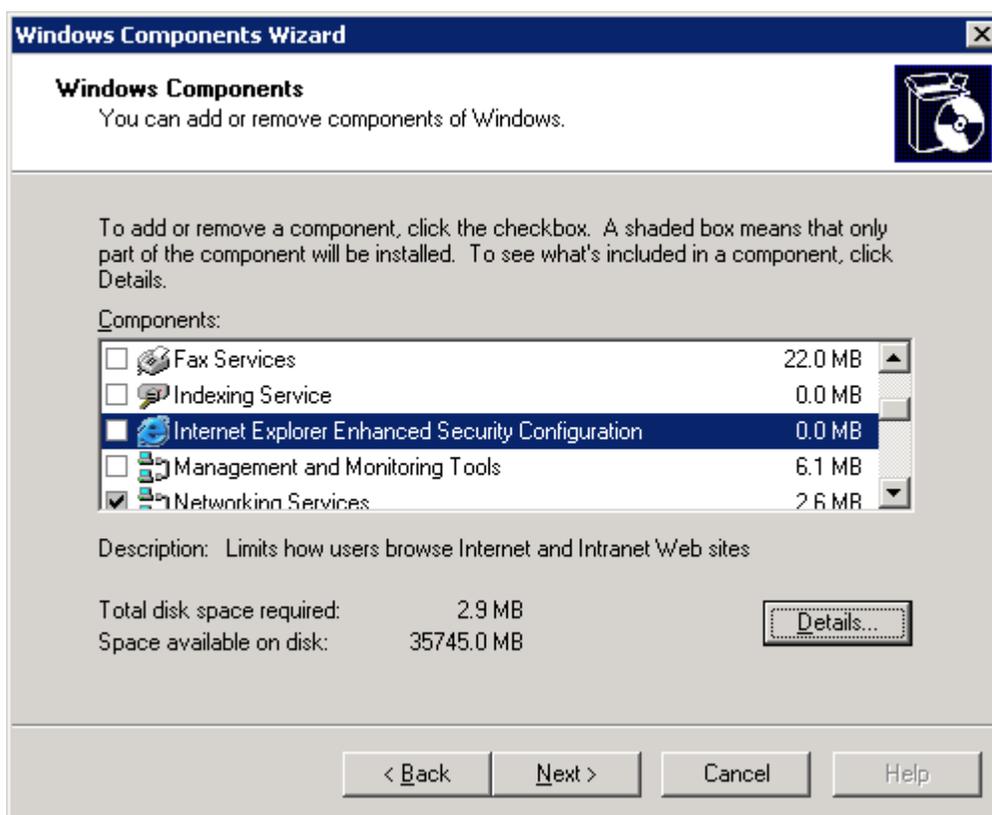
Type of Installation	Enhanced Security Configuration is Applied			
	Administrators	Power Users	Limited Users	Restricted Users
Upgrading the OS	Yes	Yes	No	No
Unattended installation of the OS	Yes	Yes	No	No
Manual installation of Terminal Services	Yes	Yes	Prompt	Prompt

**Table 6.1: Configuration of Internet Explorer Enhanced Security Configuration on a terminal server.**

When installing the Terminal Services role manually, the Configure Your Server Wizard will prompt you to disable Internet Explorer Enhanced Security Configuration for the Limited Users and Restricted Users groups. Doing so improves the browsing experience for these users and relies on the native security of the file system and OS to prevent these users from running or installing malicious code.

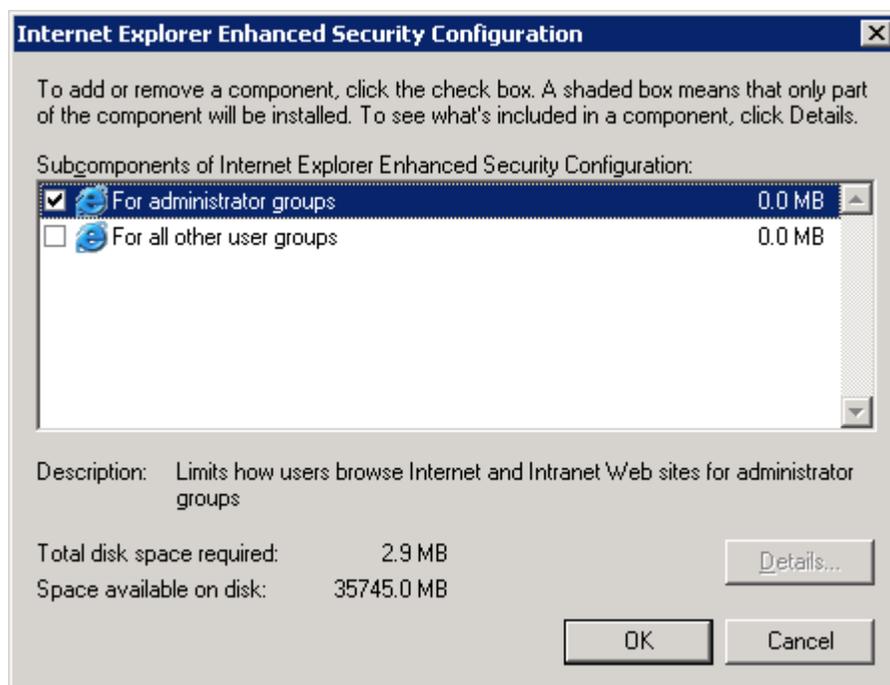
Regardless of whether Internet Explorer Enhanced Security Configuration is enabled, the Limited Users and Restricted Users groups are prevented from installing ActiveX controls on a terminal server. An administrator must manually install all approved controls on the server. Methods for doing so are described later in this section.

To manually enable or disable Internet Explorer Enhanced Security Configuration, use the Add/Remove Programs Control Panel applet, as Figure 6.1 shows.



**Figure 6.1: Enabling or disabling the Internet Explorer Enhanced Security Configuration through the Add/Remove Windows Components tool.**

If you want to enable Internet Explorer Enhanced Security Configuration for all users, simply select the associated check box. To specify to which groups Internet Explorer Enhanced Security Configuration is applied, click Details. In the resulting window, which Figure 6.2 shows, you can apply the enhanced security to administrators and all other groups individually. On a terminal server, a best practice is to enable it for administrators and leave it disabled for other user groups.



**Figure 6.2:** Applying Internet Explorer Enhanced Security Configuration to specific user groups.

### Changes Made by Internet Explorer Enhanced Security Configuration

Internet Explorer Enhanced Security Configuration changes the default zone settings for IE and adjusts several IE advanced settings to further increase protection from malicious code. Table 6.2 shows the changes made to the security levels of each zone.

Zone	Default Level	Internet Explorer Enhanced Security Configuration Level
Internet Zone	Medium	High
Local Intranet Zone	Medium Low	Medium Low
Trusted Sites Zone	Low	Medium
Restricted Sites Zone	High	High

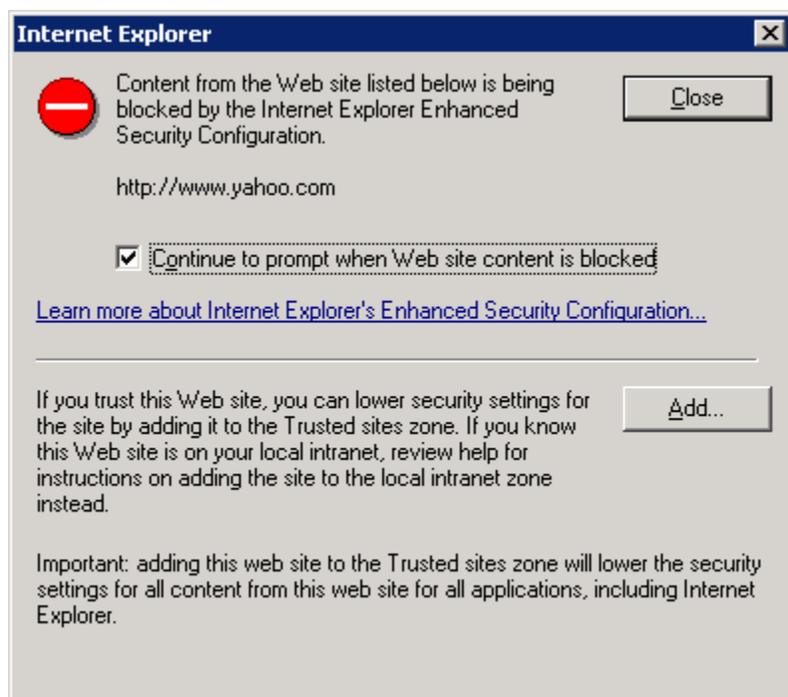
**Table 6.2:** Zone security level changes made by Internet Explorer Enhanced Security Configuration.

In addition to the security level changes, Internet Explorer Enhanced Security Configuration changes the default zone for all intranet Web sites. By default, all intranet sites (determined by your DNS suffix) are in the Local Intranet Zone. With Internet Explorer Enhanced Security Configuration enabled, intranet sites are placed in the Internet Zone and are treated with High security until you add them to the Local Intranet Zone manually. The only sites in the Local Intranet Zone under Internet Explorer Enhanced Security Configuration are local machine sites (<http://localhost>, <https://localhost>, [hcp://system](http://localhost/hcp://system)). Local machine sites must be in the Local Intranet Zone for many of the Help and Administrative Tools to work properly. Internet Explorer Enhanced Security Configuration also changes advanced options, as Table 6.3 shows.

Feature	Entry	New Setting	Result
Browsing	Display enhanced security configuration dialog box	On	Displays a dialog box to notify you when an Internet site tries to use scripting or ActiveX controls.
Browsing	Enable Browser Extensions	Off	Disables features you installed for use with IE that might have been created by companies other than Microsoft.
Browsing	Enable Install On Demand (Internet Explorer)	Off	Disables installing IE components on demand if needed by a Web page.
Browsing	Enable Install On Demand (Other)	Off	Disables installing Web components on demand if needed by a Web page.
Microsoft VM	JIT compiler for virtual machine enabled (requires restart)	Off	Disables the Microsoft VM compiler.
Multimedia	Don't display online content in the media bar	On	Disables playback of media content in the IE media bar.
Multimedia	Play sounds in Web pages	Off	Disables music and other sounds.
Multimedia	Play animations in Web pages	Off	Disables animations.
Multimedia	Play videos in Web pages	Off	Disables video clips.
Security	Check for server certificate revocation (requires restart)	On	Automatically checks a Web site's certificate to see whether it has been revoked before accepting it as valid.
Security	Check for signatures on downloaded programs	On	Automatically verifies and displays the identity of programs you download.
Security	Do not save encrypted pages to disk	On	Disables saving secured information in your Temporary Internet Files folder.
Security	Empty Temporary Internet Files folder when browser is closed	On	Automatically clears the Temporary Internet Files folder when you close the browser.

**Table 6.3: Advanced changes made by Internet Explorer Enhanced Security Configuration.**

The browsing experience will be very restrictive while Internet Explorer Enhanced Security Configuration is enabled. If you use Web-based tools or Web pages with active scripting for systems administration, you will need to add these sites to either the Local Intranet Zone or Trusted Sites Zone. To make zone changes easier, Microsoft adds an *Add this site to...* selection to the IE File menu when Internet Explorer Enhanced Security Configuration is enabled. Also, if you encounter a Web page that uses active scripting, a warning dialog box, which Figure 6.3 shows, gives you quick access to the Add to Trusted Sites interface.



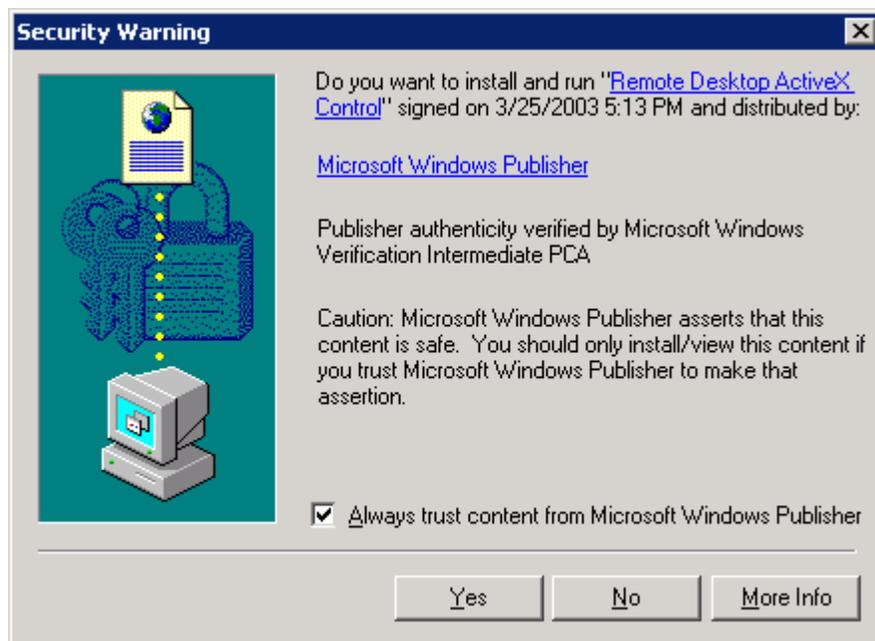
**Figure 6.3: Internet Explorer Enhanced Security Configuration's enhanced security configuration dialog box.**

### **Managing Approved ActiveX Controls**

Regardless of whether Internet Explorer Enhanced Security Configuration is enabled, the Limited Users and Restricted Users groups are prevented from installing any ActiveX controls or other active Web code on a terminal server—members of these groups simply do not have write access to the locations on the file system where the controls are stored.

There are, of course, many cases in which your users will need to access Web pages that use active code in the course of their normal work. As an administrator, you will need to manage these approved controls. You can do so using several methods. To manually install a control for users:

1. Log on to the terminal server using an administrative account, and use IE to browse to the Web site containing the control.
2. If Internet Explorer Enhanced Security Configuration is enabled, you will receive the security configuration dialog box warning. Either click *Add...* to add the site to your trusted sites list or select *Add this site to...* from the File menu and add the site to the Local Intranet Zone.
3. Another dialog box will appear, prompting you to accept and install the control; select the *Always trust...* check box, which Figure 6.4 shows, and click OK.
4. The control is now installed for all users on the terminal server.



**Figure 6.4: Security warning when installing an active Web control.**

For administering a large number of terminal servers, you will not want to manually install controls on each server. To automate the installation process, you can either include the controls in your base image (if you are using a server cloning process such as Sysprep) or you can capture the files contained in the control and repackage them into MSI files to use Group Policy to install them on your servers. Which method you choose will depend on your environment and software distribution methods as well as the frequency at which new or updated controls are required in your environment.

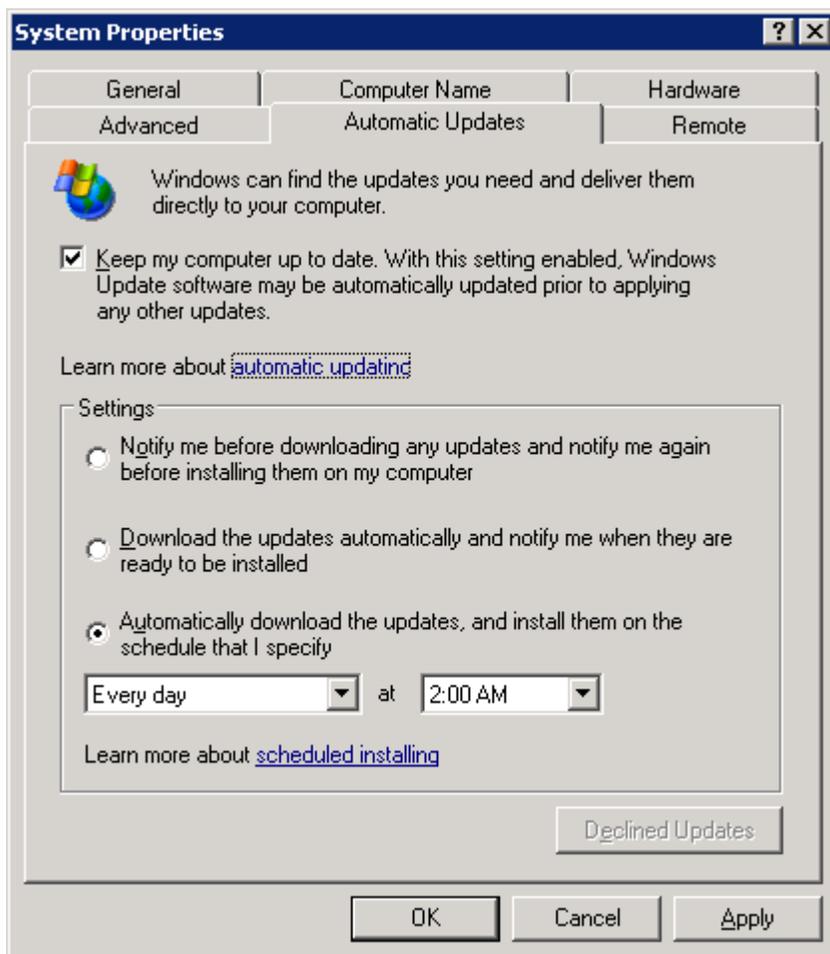
 Internet Explorer Enhanced Security Configuration applies only to IE. If you use a third-party Internet browser, you will need to implement a separate security model.

## Implementing Windows Automatic Updates

The Internet Explorer Enhanced Security Configuration can help protect you from malicious code found on Web pages—worms and viruses can be designed to take advantage of security flaws that are discovered in Windows over time. Microsoft is proactive in providing patches and updates to plug discovered security holes, but you are responsible for installing these updates on your systems. In the days of NT 4.0, this meant subscribing to the Microsoft Security Bulletin and frequently checking the Microsoft Security Web page for new updates and information, then downloading any critical patches and distributing them to your servers and workstations.

Since the release of W2K3 and Windows XP, Microsoft has included the Windows Automatic Updates client. WS2K3 also includes this service. With Windows Automatic Updates, you can configure your servers and workstations to download and install any updates that are deemed critical by Microsoft. The Automatic Updates Client is very versatile and can be configured to fit most situations.

You can configure Automatic Updates manually or via Group Policy. For manual configuration, access the System Control Panel applet, and select the Automatic Updates tab (see Figure 6.5).

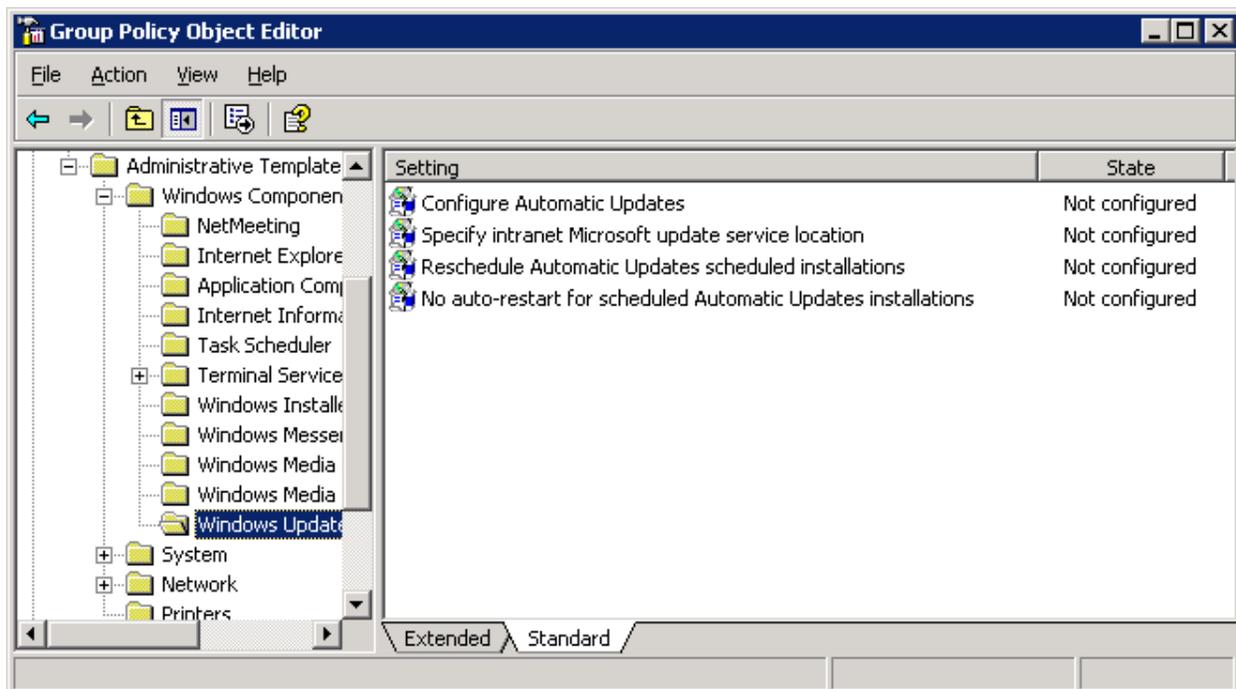


**Figure 6.5: Configuring Automatic Updates.**

Through this interface, you can enable or disable Automatic Updates, and configure the mode in which the client operates. The available modes are:

- **Notify before downloading**—The client regularly checks for new critical updates. When one is available from Microsoft, the client places a notification icon in the system tray whenever an administrator is logged on to the server. The administrator clicks on this icon to begin the download and install the update.
- **Automatically download but notify before installing**—The client regularly checks for new critical updates. When one is available, the client automatically downloads the update, then places a notification icon in the system tray whenever an administrator is logged on to the server. The administrator clicks on the icon to begin the installation.
- **Automatically download and install at a specific time**—The client regularly checks for new critical updates. When one is available, the client cues up the update for installation at the next specified day and time.

You can also configure Automatic Updates via Group Policy. In the Group Policy Object Editor, drill down to Computer Configuration, Administrative Templates, Windows Components, Windows Update. Figure 6.6 shows the available settings.



**Figure 6.6: Configuring Automatic Updates through Group Policy.**

The *Configure Automatic Updates* setting specifies the mode and schedule. It offers the same options as the System Control Panel applet. You can specify the mode and, if automatic installation is selected, the day(s) and time that the installation should take place.

The next setting, *Specify intranet Microsoft update service location*, redirects the Automatic Updates client to an internal Software Update Services (SUS) server.

 We will explore SUS in more detail in the next section.

*Reschedule Automatic Updates scheduled installations* specifies how long after boot the service should wait before installing updates if the computer was shut down at the last scheduled installation time. This setting is very useful on workstations, but isn't often necessary on servers.

Finally, the *No auto-restart for scheduled Automatic Updates installations* setting suppresses the reboot (if the update being installed requires one). This setting is helpful if you are running automated reboot scripts on your terminal servers and do not want to reboot twice on nights that critical updates are installed.

When configuring Automatic Updates, there are some factors to consider. If your servers are used only during business hours, it is very easy to set up an automatic installation schedule. However, if your servers are in use  $24 \times 7$ , stagger the installation schedule so that not all servers are rebooting at the same time. You can use a scheduled task to disable logons in advance of the scheduled installation time so that the server has a chance to drain before updates are installed and the server is rebooted.

In a very large environment, you can create multiple GPOs filtered by security group, each specifying a different installation schedule. This way, you can group your servers by which day they install updates.

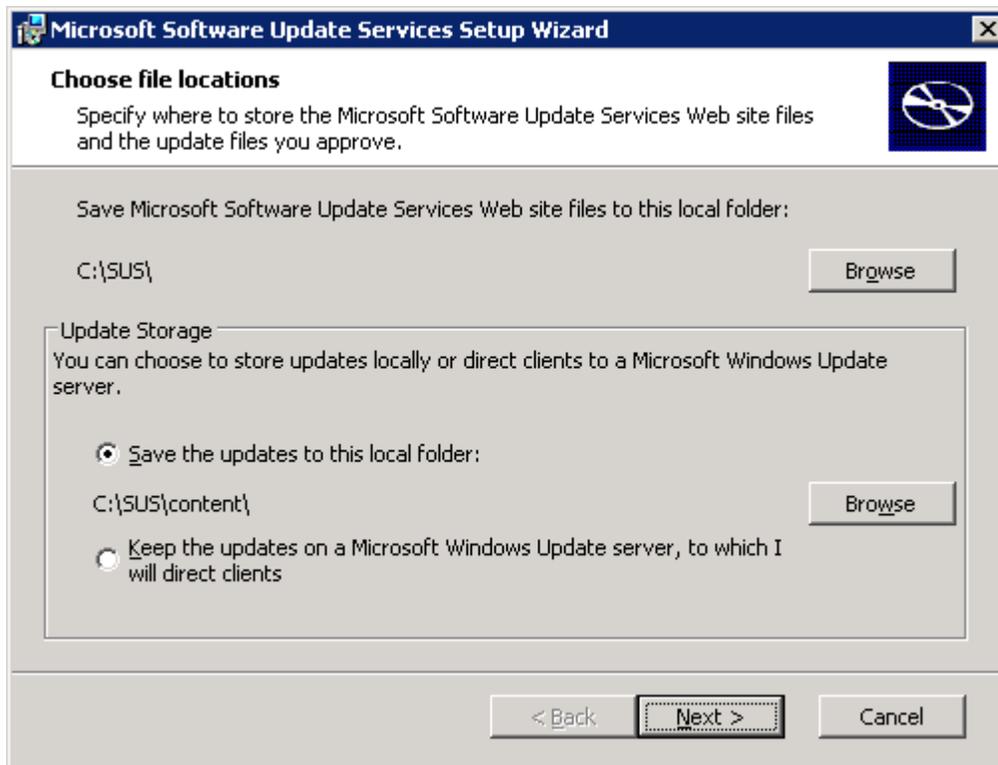
☞ When using multiple GPOs to schedule Automatic Updates, place the first GPOs in the policy processing order to apply to the Authenticated Users group, then higher GPOs to be filtered by security group. This way, if a server is not in a specific group, it will receive the default schedule and receive updates.

## Using SUS

In large or mission-critical environments, you might want the ability to select which critical updates are necessary in your environment or have the ability to test updates in a lab before installing them on your production servers. To meet these requirements, Microsoft provides SUS.

SUS is installed on a Win2K IIS or WS2K3 Application Server, and takes the place of Microsoft's own Automatic Updates site. You then use a GPO to redirect the Automatic Updates client to your internal SUS server. The client then downloads updates from the SUS server for installation rather than polling Microsoft for updates.

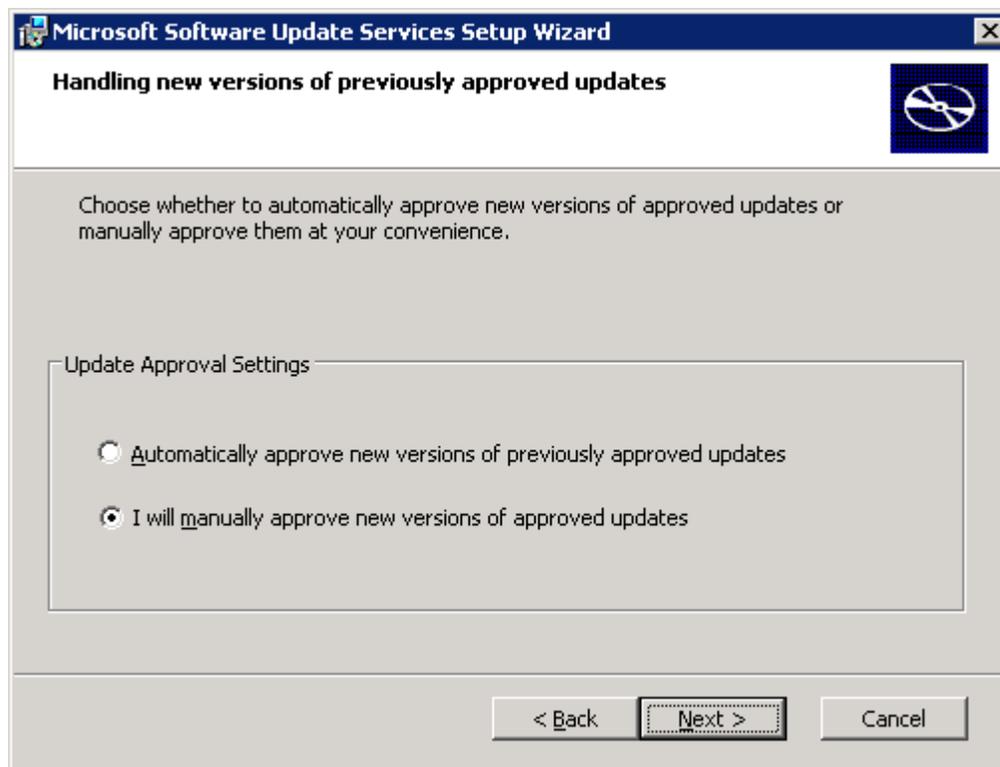
To install SUS, download the installer from the Microsoft Web site (the most recent location is at <http://www.microsoft.com/windows2000/windowsupdate/sus/default.asp>). The default installation stores updated source files in C:\sus\content and is set to require you to manually approve new versions of previously approved updates before clients can download and install them. You can change these options either after the installation is complete or by selecting a custom installation from the Windows Installer wizard. Figures 6.7 and 6.8 show the installation options dialog boxes.



**Figure 6.7:** Specifying where to store update source files during the installation of SUS.

If your application server is tight on disk space and your terminal servers have an open connection to the Internet, you can choose to leave the update source files on Microsoft's update servers. You will still be able to selectively approve and test updates, but you will not have a local copy of the source files.

Occasionally, Microsoft releases an updated version of a previously released patch. SUS allows you to treat the updated version as an entirely new patch (which you will want to do if you are testing updates in a lab before deploying them) or automatically approve the new version if you have already deployed the original version (see Figure 6.8).



**Figure 6.8:** Configuring manual or automatic approval of new versions of previously approved updates.

After SUS is installed, you can configure and manage the service through a Web interface located at <http://localhost/susadmin> or remotely from <http://<servername>/susadmin>. Through the interface, you can configure a proxy server (if one is required for the SUS server to access Microsoft's Web site), specify the name that the clients will use to access the SUS server, select whether to synchronize updates from Microsoft or from another SUS server (useful if you have more than one SUS server for fault tolerance), and change the two options you set during installation.

If you are using a local store for update source files, you can select which localized versions of the updates you want to download. Be sure to select only the languages of the clients you support, as the additional source files can consume a large amount of disk space and increase the time it takes to synchronize the server. Figure 6.9 shows the section of the options page used to specify localized updates.

**Set options**

Set your Software Update Services options, and then click **Apply**.

**Select how you want to handle new versions of previously approved updates:**

Automatically approve new versions of previously approved updates

Do not automatically approve new versions of approved updates. I will manually approve these updates later.

---

**Select where you want to store updates:**

Maintain the updates on a Microsoft Windows Update server

Save the updates to a local folder

Synchronize installation packages only for these locales:

<input checked="" type="checkbox"/> Arabic	<input checked="" type="checkbox"/> Italian
<input checked="" type="checkbox"/> Arabic enabled	<input checked="" type="checkbox"/> Japanese
<input checked="" type="checkbox"/> Chinese Simplified	<input checked="" type="checkbox"/> Japanese NEC
<input checked="" type="checkbox"/> Chinese Traditional	<input checked="" type="checkbox"/> Korean

**Figure 6.9: Specifying SUS options.**

The SUS administrative interface is also where you initiate synchronization. During synchronization, the server downloads the latest catalog of updates from Microsoft as well as the source files for the updates (if you have selected a local store). After performing a synchronization manually to confirm that your proxy configuration is valid, you should set up a synchronization schedule so that your SUS server is always up to date.

Once your SUS server is synchronized, you use the administrative interface to approve new updates. These approved updates will be downloaded and installed at the next scheduled installation window on all clients configured to use the SUS server. Figure 6.10 shows the approval interface.

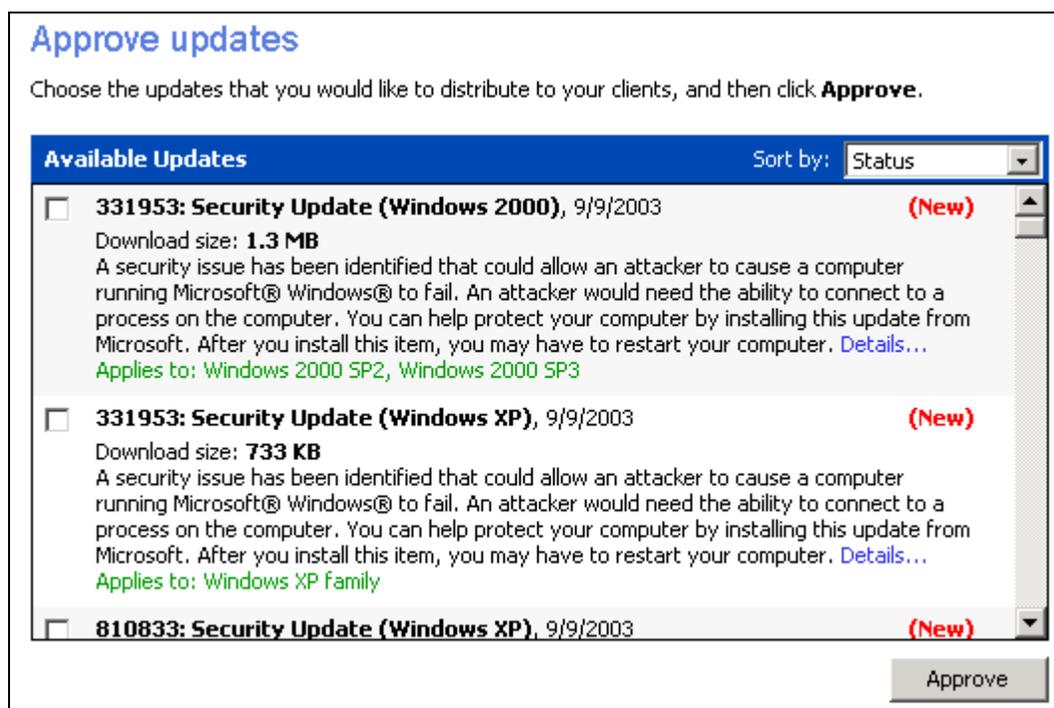


Figure 6.10: Approving updates on an SUS server.

SUS provides a description of each update, a link to the installation details, and a list of the OSs to which the update applies. In the details screen, there is a hyperlink to the Microsoft security bulletin for the vulnerability, which provides additional information you can use to determine whether the update is required in your environment.

☞ If you plan to test updates before approving them for distribution to your production systems, you can either download the updates from the Windows Update Web site manually or set up two SUS servers—one for your lab and one for production clients. Approve a new update on the lab server first, perform your testing, then approve it on the production server.

To configure your terminal servers to pull approved updates from an SUS server, configure a GPO in the same way you would for Microsoft Automatic Updates but use the *Specify Intranet Microsoft update service location* setting to configure the URL of your SUS server. The Automatic Updates client will now look to SUS for updates.

Implementing SUS puts you in control of which updates are installed on your terminal servers. Although this method gives you the opportunity to test updates before installation, it also requires that you manually approve updates. You must be vigilant in staying up to date on Microsoft security bulletins or frequently check your SUS server for new updates awaiting approval. You do not want to be caught with a security exposure that could have been prevented by simply approving an update.

 As of this writing, the Automatic Updates client has one major flaw. Once an update or set of updates is downloaded and cued for installation, the client will not poll for new updates again until the first set is installed. Thus, you should either set up a daily installation schedule or only approve updates on your SUS server on a weekly basis. This way, when the client polls SUS, it will download and install all pending updates.

## Deploying Service Packs and Hotfixes

In addition to using security patches, Microsoft keeps Windows at its best through service packs and hotfixes. These should be thoroughly tested before deploying them to your production servers—test service packs because of how large the scope of the changes are and test hotfixes because of Microsoft's limited support of them.

### *Using Group Policy to Deploy Service Packs*

Once you determine that a service pack is ready for prime time, you need a way to deploy it to your terminal servers. Beginning with Win2K, Microsoft began including an MSI file that you can use to assign service packs via Group Policy. Doing so saves you the time of manually installing the service pack on each server.

 The UPDATE.MSI file should only be used for deploying a service pack via Group Policy. Never install a service pack manually with the MSI file—use UPDATE.EXE instead.

You assign a service pack to a computer in the same way you would any other piece of machine-based software. Begin by extracting the service pack files to a network share that can be accessed by all your terminal server computer accounts. To perform the extraction, launch the service pack executable with a -X switch (for example, WS2K3SP1.EXE -X). You will then be prompted for the location to which the files should be extracted.

Next, use the Group Policy Management Console to edit the GPO that will assign the service pack to your computers. Drill down to Computer Configuration, Software Settings, Software installation. Right-click this node, and select New, Package. You will then be prompted to select an MSI package to deploy. Enter the UNC path to the UPDATE.MSI file in the folder to which you extracted the service pack, and click Open.

You will be offered the option to assign the package with default settings or open the advanced dialog box. You can select Assigned, then click OK—there are no advanced settings that need to be specified when deploying a service pack.

The next time a terminal server that is configured to receive the GPO reboots, the service pack will be installed. If you are performing regular maintenance scripts on your terminal servers, you can relax knowing that within a week all servers will have the service pack installed.

Microsoft always makes service packs inclusive of updates found in previous service packs. This way, you only need to install one service pack when building a new server. When you are ready to deploy a new service pack, you will need to remove the GPO assignment of the previous one so that both service packs are not installed on new systems.

 You do not need to uninstall a previous service pack before installing the new one, just remove the assignment.

To remove the assignment of the earlier service pack, use Group Policy Management Console to edit the GPO used to assign the service pack. Right-click the package, and select All Tasks, Remove. You are then prompted with the dialog box that Figure 6.11 shows, giving you the option to either uninstall the software from all computers that are managed by the GPO or simply prevent new installations; select the latter option.



**Figure 6.11: Removing software assigned via GPO.**

Once you have removed the assignment for the old service pack, you can create the assignment of the new service pack. At the next reboot, all servers will receive the latest pack, either over the previous one or as the first pack on a new server.

### Deploying Hotfixes

Windows Automatic Updates and SUS will only take care of critical and security updates. If you determine that one or more non-critical updates are needed in your environment, you need a way to deploy them to your servers.

For new servers, the best method is to integrate the hotfixes into your server build process. If you are using a server-cloning process or RIS images, this task can be accomplished by either installing them manually before running Sysprep or Riprep. If you are building servers with an unattended install process, you can add them via a CMDLINES.TXT file.

 For additional information about integrating hotfixes into an unattended installation of Windows, see the [Hotfix Installation and Deployment Guide](http://www.microsoft.com/Windows2000/downloads/servicepacks/sp3/hfdeploy.htm) found at <http://www.microsoft.com/Windows2000/downloads/servicepacks/sp3/hfdeploy.htm>.

Microsoft does not, however, provide a clear method to deploy hotfixes to multiple existing servers. The deployment guide only covers manual installation from a network source. This process is fine in smaller environments, but for large terminal server farms, you will want to automate the installation. The following suggested methods for doing so will not work in all situations and is not a comprehensive list. Select the best option for your situation and environment.

## Using a ZAP File

If you need to install a single hotfix at a time, a ZAP file is a good option. ZAP files are used to install non-MSI-based software via Group Policy. ZAP files are simply text files containing the information needed to install the application. Listing 6.1 provides a sample ZAP file.

```
[Application]
; Only FriendlyName and SetupCommand are required,
; everything else is optional.

; FriendlyName is the name of the program that
; will appear in the software installation snap-in
; and the Add/Remove Programs tool.
; REQUIRED
FriendlyName = "Hotfix Q911001"

; SetupCommand is the command line used to
; Run the program's Setup. If it is a relative
; path, it is assumed to be relative to the
; location of the .zap file.
; Long file name paths need to be quoted. For example:
; SetupCommand = "long folder\setup.exe" /unattend
; or
; SetupCommand = "\\server\share\long _
; folder\setup.exe" /unattend
; REQUIRED

SetupCommand = "Q#####_WS2K3_SP1_x86_en.exe" /M

; Version of the program that will appear
; in the software installation snap-in and the
; Add/Remove Programs tool.
; OPTIONAL
DisplayVersion = 1.0

; Version of the program that will appear
; in the software installation snap-in and the
; Add/Remove Programs tool.
; OPTIONAL
Publisher = Microsoft
```

**Listing 6.1: A sample ZAP file for installing a hotfix.**

The main disadvantage of using ZAP files to install hotfixes is that the server will reboot after each hotfix installation. Thus, this method is not useful for installing multiple hotfixes.

 The command-line arguments for all Microsoft hotfix installer programs are:

- /F Forces any open applications to close when the hotfix reboots the computer
- /N Does not back up files for removing the hotfix
- /Z Does not restart the computer after the installation is completed
- /Q Uses quiet mode; no user interaction is required
- /M Uses unattended setup mode
- /L Lists installed hotfixes

## Using a Shutdown Script

Most hotfixes require you to reboot the server after installation (or at the end of a string of hotfixes), so a shutdown script is a perfect option for installing them. You don't want to install the hotfixes *every time* the server shuts down, so you will need to script in some checks before calling the installation program. The easiest way to check whether a hotfix is already installed is to query the registry. All Microsoft hotfixes register themselves in two locations in the registry:

- HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Updates\WS2K3\<SPx>\<HotfixName>—Where <SPx> is the pending service pack that will contain the hotfix and <HotfixName> is the Microsoft Knowledge Base article that describes the hotfix
- HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Hotfix\<HotfixName>—Where <HotfixName> is the Microsoft Knowledge Base article that describes the hotfix

Listing 6.2 shows a sample Visual Basic script you can use to install multiple hotfixes.

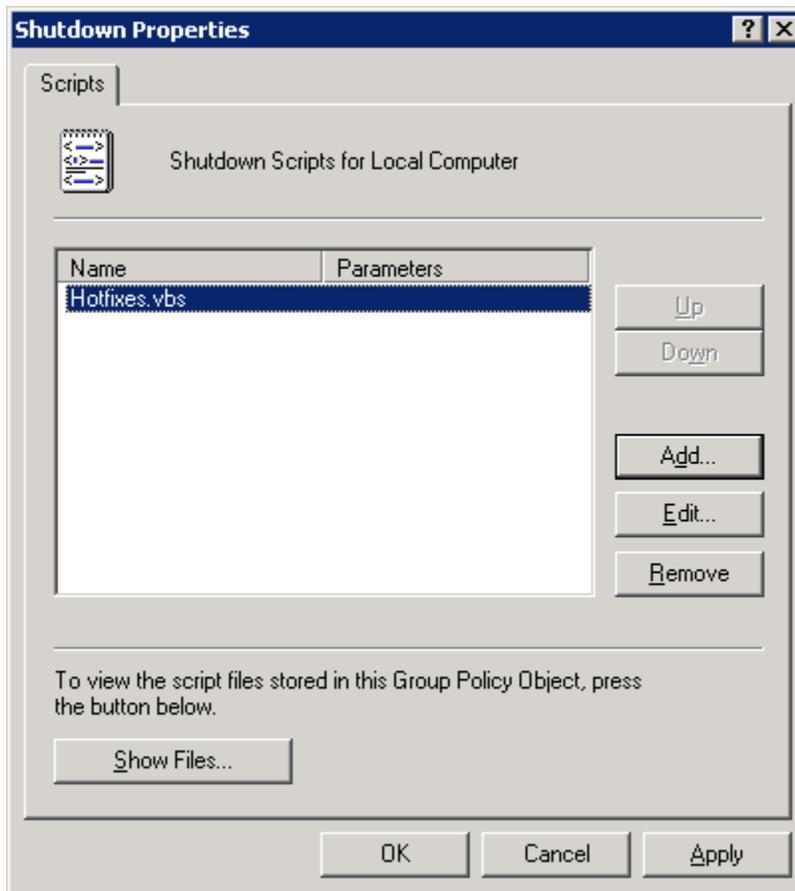
```
On Error Resume Next
Set WshShell = WScript.CreateObject("WScript.Shell")

WshShell.RegRead("HKLM\SOFTWARE\Microsoft\Updates\WS2K3\SP1\Q819639\Description")
If Err then
    Hotfix1="Q819639_WS2K3_SP1_x86_en.exe /Z /M"
    WshShell.Exec(Hotfix1)
End If

WshShell.RegRead("HKLM\SOFTWARE\Microsoft\Updates\WS2K3\SP1\KB818529\Description")
If Err then
    Hotfix2="KB818529_WS2K3_SP1_x86_en.exe /Z /M"
    WshShell.Exec(Hotfix2)
End If
```

**Listing 6.2:** A sample Visual Basic script for installing multiple hotfixes.

To set the script as a shutdown script, use the Group Policy Management Console to edit a GPO that applies to your terminal servers. Drill down to Computer Configuration, Windows Settings, Scripts (Startup/Shutdown), and double-click Shutdown in the right pane. Doing so will open the Script Properties dialog box that Figure 6.12 shows.



**Figure 6.12: Configuring a shutdown script via a GPO.**

In this dialog box, click Show Files and copy your script file and all the hotfix installation files to the folder. Then use the Add button to set the script file to run at system shutdown. Every time your servers are shut down or rebooted, the script will install any hotfixes not already present on the system.

- ❗ Although hotfixes have built-in logic to prevent them from being installed on a system with a newer service pack, disable or update your hotfix script after you deploy a new service pack to reduce the overhead—no need to spend the time attempting to install fixes that are not needed.

## Virus Protection Software Best Practices

Virus protection software is your last line of defense against malware. You can implement filters on your corporate email system, use an email client such as Microsoft Outlook 2002 that blocks all executable attachments, and implement strong firewall and proxy server rules on your Internet connection. Regardless, a virus will always find its way into the environment.

Virus protection software works by scanning both individual files and active processes in memory and comparing them with a database of known signatures. Most virus protection software protects you from all three types of malware; however, the protection is only as good as the database. You must keep the database of virus definitions updated, so be sure to select a software product that provides you with a method of updating the database.

The following list highlights best practices when implementing and configuring virus protection software:

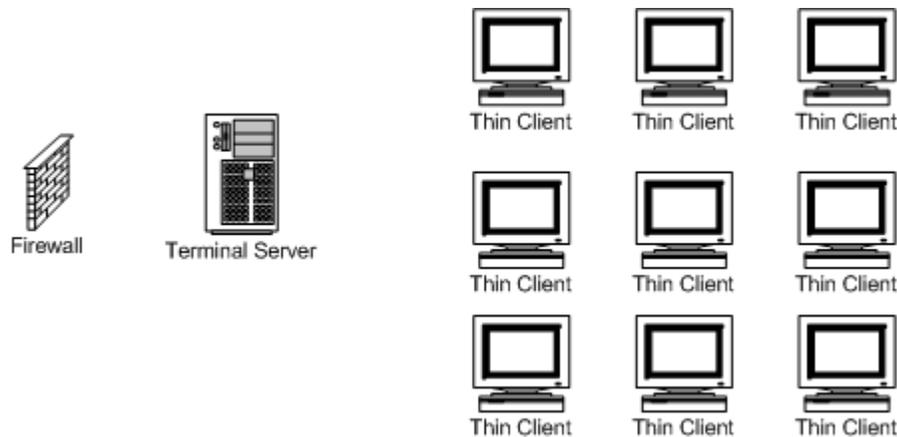
- Select a product that allows you to update virus definitions without rebooting the server.
- If your terminal servers are mission critical, select a product that enables you to pull updates from an internal source, giving you the opportunity to test and validate the updates before deploying them on production servers.
- Many virus protection products install a status icon into the system tray. After checking with the vendor, disable the icon (usually by removing a value from the HKEY\_LOCAL\_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run registry key). Doing so will reduce the load placed on your server by having the status program running in every user session.
- In addition to “real-time” scanning (scanning every file that users access and processes in memory), most products perform scheduled scans of every file on the system. Be sure to schedule these scans during off-hours.

## Putting It All Together

As you can see, there are several options available to help keep your terminal servers secure and virus-free. In this section, we’ll explore two examples of how these options can be implemented.

### **Example One: Anytown Little Theatre**

David is a terminal server administrator at a local company. During evenings and weekends, he volunteers doing systems administration and tech-support work at a local community theater, Anytown Little Theatre. To help keep costs down and the network as stable as possible, he used some grant money to set up a terminal server infrastructure for the theatre. Figure 6.13 illustrates the network.



**Figure 6.13: Anytown Little Theatre network.**

The infrastructure consists of nine thin clients used by the theatre staff, a terminal server, and a personal router/firewall connected to a DSL line. Users connect to the terminal server and receive a full desktop environment with all the needed applications installed. They store their documents in their My Documents folders on the terminal server, and access a public share for common documents. The share is also on the terminal server. User profiles, the public share, and the System State are backed up nightly at 1AM to an external hard disk connected to the server.

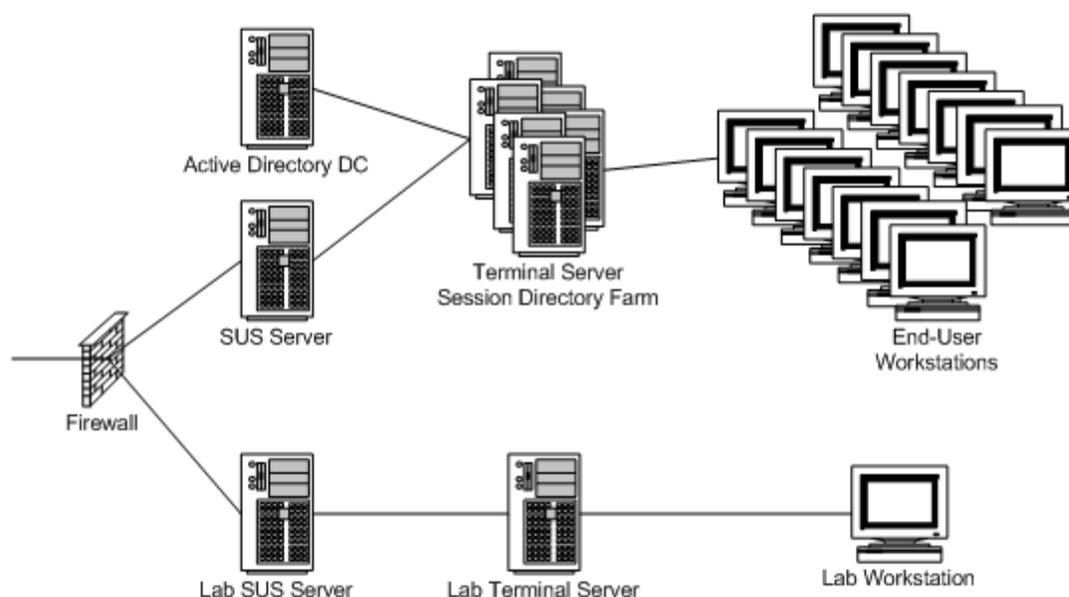
David wants to keep the network as stable and secure as possible, so he implements the following security measures:

- The firewall is configured to use Network Address Translation (NAT) and block inbound requests. This setup allows the staff to surf the Web, but prevents worms from attacking the server from the Internet.
- All theatre staff members are set up as Limited Users on the terminal server. This setup prevents them from installing software or ActiveX controls but does not apply Internet Explore Enhanced Security Configuration to them, so they can browse the Internet without receiving unnecessary warnings and error messages.
- The Automatic Updates client on the server is configured to automatically download and install critical updates from Microsoft on a nightly basis at 4AM. Automatic reboots are allowed.
- Virus protection software is installed on the server and configured to download updated definition files from the vendor's Web site automatically every night at 3AM.
- There is only one server, so David installs service packs and hotfixes manually.

### **Example Two: BigBusiness, Inc.**

BigBusiness, Inc is a mid-sized company of about 2000 employees. Because of the variety of tasks users perform, each user has a Windows XP workstation with applications installed locally. There are two mission-critical applications that receive frequent updates and access large databases of customer data.

To optimize performance and make deploying application updates easier, BigBusiness' IT staff has implemented a terminal server Session Directory Farm to serve the two applications to end users. Users access the applications through a customized Remote Desktop Web Connection Web page that connects the users to the specific applications instead of a terminal server desktop. Figure 6.14 shows BigBusiness' network.



**Figure 6.14:** The network of BigBusiness, Inc.

To keep the terminal servers as stable and secure as possible, BigBusiness' IT staff has set up the following configuration:

- The corporate firewall blocks all inbound traffic to the terminal servers.
- All users are configured as Limited Users on the terminal servers, and Internet Explorer Enhanced Security Configuration is enabled for all users. Because users do not access IE on the server, this configuration can be enabled without impacting the end-user experience.
- Virus protection software is installed on all the terminal servers and configured to pull updates from an internal FTP site. Virus definition files are tested in the lab before being uploaded to the FTP site.
- Two SUS servers have been implemented—one for production and one for the lab. New critical updates are first approved on the lab SUS server. The lab terminal server's Automatic Updates client is configured to pull updates daily at 1AM from the lab SUS server. The day after a new update is approved, the lab server and its applications are tested. If the update does not impact the server or applications, it is then approved on the production SUS server.
- The production servers are grouped by maintenance schedules. Each afternoon a script runs on a specific group of servers, disabling new logons. That night, the Automatic Updates client on that group of servers is configured via Group Policy to download and install new updates from the production SUS server, then reboot. Within a week, all servers receive any new updates.

- In the event of a serious threat, the Group Policies can be changed to have the Automatic Updates client on all servers pull updates that night. This way, a new critical update can be deployed to all servers within one day.
- When a new service pack is available, it is thoroughly tested in the lab. Once it is deemed stable, it is assigned to the production servers via GPO. At the next maintenance night for each group of servers, the service pack is installed during the automatic reboot. Within a week, all servers have received the new service pack.

## Summary

The enhanced security model of WS2K3 and the inclusion of the Automatic Updates client make it easier to keep your servers secure and updated with the latest security patches. You can use Group Policy and the Internet Explorer Enhanced Security Configuration to further protect your terminal server systems from malicious code.

“Do more with less” is the slogan that Microsoft used to launch WS2K3. With the new features and enhancements in Terminal Services, terminal server administrators can certainly take this slogan to heart. We now have native methods of grouping servers into load-balanced farms, managing their configuration and behavior through Group Policy, and distributing connection files to our users.

Through the course of this book, I have introduced you to the new features and taken you through the process of building a terminal server infrastructure, both small and large. You have seen how to enable the Terminal Services Role, set up a Session Directory, assign applications to your servers through Group Policy, and configure the Automatic Updates client to pull updates from either Microsoft or an SUS server. I hope you see how powerful Terminal Services can be, and how easy it is to maintain if set up and configured properly.

From the smallest business to the largest corporation, Terminal Services provides a powerful tool for centralizing your computing needs. It is versatile enough to provide entire desktop replacements or give users access to a single application. With a little research and planning, you are ready to begin building terminal server infrastructures that are stable, highly available, easy to scale, and easily managed.