realtimepublishers.com™

The How-To Guide™ To

Windows Server 2003 Terminal Services

triCerat®

Greyson Mitchem

Copyright Statement

[**Editor's Note**: This eBook was downloaded from Content Central. To download other eBooks on this topic, please visit http://www.realtimepublishers.com/contentcentral/.]

Chapter 3: User Session and Environment Configuration

Once you have the Terminal Server role enabled and applications installed, you are ready to allow users to start logging onto your terminal server. Before you do so, however, you should consider configuring their user profiles, sessions, and resource redirection settings. These settings are central to the overall experience your users will have when working on your terminal server. Although your users might like to have the freedom to customize every setting in their terminal server environment—such as unlimited session lengths, and the ability to access every drive, printer, and peripheral on their local workstations—this freedom would not be a very efficient use of your terminal servers resources. Your goal in configuring them should be to find a balance between end-user functionality and server performance.

Session Length Limits

There are three distinct user session timeouts you can configure on a terminal server:

- Active session—Limits the amount of time that a user can actively work on the terminal server. At the end of the timeout duration, the user will be logged off of the server. This setting is commonly used in public kiosk environments.
- Idle session—This setting limits the amount of time that a user can leave a session idle. If the user minimizes the terminal server window or locks his or her workstation, the terminal server session becomes idle but continues to take up some resources on the server to maintain the user session.
- Disconnected session—When a user closes a terminal server window without logging off or experiences a network interruption, the session can become disconnected. This setting determines how long the session will remain active on the server waiting for the user to reconnect. Like an idle session, a disconnected session continues to use some server resources.

If your servers have the capacity to handle all your users simultaneously, you might choose to leave theses timeouts set to unlimited so that your users can leave sessions running for days at a time and disconnect and reconnect at will. More often, however, you will want to configure at least the disconnected session timeout to reclaim the resources on the server. The next section will show you how to configure these settings.

Types of User Profiles

A user profile consists of the user’s HKCU registry hive as well as all per-user files and data (Internet Explorer—IE—Favorites, Outlook Signature files, and so on). Since the release of Windows NT, Windows has supported three distinct types of user profiles:

- Local profiles—These profiles are specific to one computer and are stored on the local hard disk only (on WS2K3 systems, they are stored in C:\Documents and Settings).
- Roaming profiles—These profiles are stored centrally on a file server and are copied down to every machine that a user logs onto. At logoff, any changes made during the user session are copied back up to the central file server. Roaming profiles allow users to have consistent settings across multiple computers.
- Mandatory profiles—A mandatory profile is a preconfigured profile that is stored on a central file server. At logon, a copy of the profile is made for the user with all the administrator-defined settings in tact. At logoff, any changes that the user might have made during the session are discarded. Mandatory profiles provide a consistent user experience and minimize the chances of profile corruption by creating a fresh profile for the user at every logon.

In a terminal server environment, there are some additional options available, but they are based on these three types of profiles. You can configure a separate roaming or mandatory profile to be used exclusively on terminal servers so that your users can have separate settings for servers and workstations. You can also configure a single server or group of servers to use a distinct roaming profile from the one defined in the user account.

> There are also third-party tools that offer alternatives to the native profile types, which will be introduced later in this chapter.

Client Device Redirection and Resource Mapping

In order for a user to have a rich, full-featured terminal server experience, you may want to take advantage of client device redirection and resource mapping. These features allow the user to seamlessly work across both local and terminal server-based applications by automatically connecting them to resources available on the user’s local workstation. RDP 5.2 supports the following features:

- Client drive mapping—Automatically connects to the client workstation’s local drives (hard disk, diskette, and CDROM) so that users can access files on these drives from within applications on the terminal server.
- Client printer mapping—Automatically connects to printers that users have mapped on their workstations (both locally attached and network printers) so that users can print without needing to re-map the printers on the terminal server.

- Client time zone mapping—Detects the time zone to which the client workstation is set and adjusts the time within the user session to match. This mapping is vital when your users are distributed across multiple time zones.
- Client COM port mapping—Allows users to access peripherals that are connected to the client workstation's serial ports. This mapping can be used for POS devices (bar-code readers, cash drawers, and so on) as well as serial-port–connected PDAs.
- Client audio redirection—You can configure sounds that are generated by applications on the terminal server to be played through the client workstation's speakers, played through the terminal server's speakers (if it has a sound card), or disabled altogether.

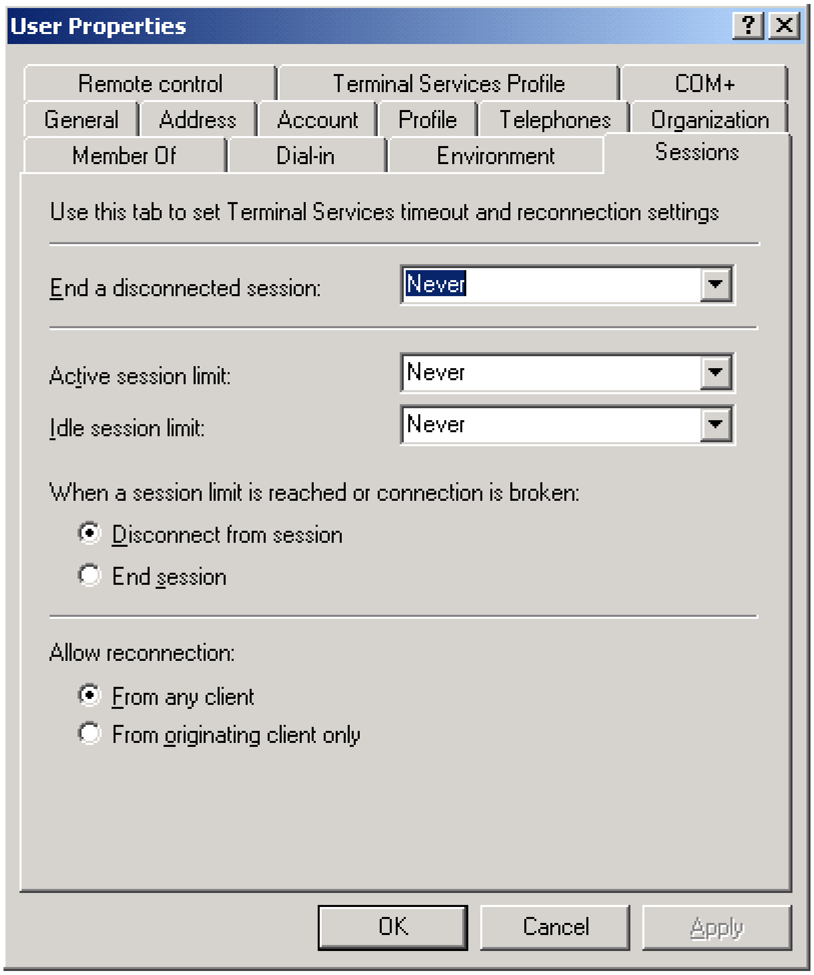
> Each of these client redirection features consume additional bandwidth. Take this consideration into account before enabling these features if your users are working over a low-bandwidth connection.

How to Configure User Session Timeouts

If you want each user to have unique session timeout settings, you can configure them on a per-user basis via Active Directory Users and Computers, ADSI, and Group Policy, and on a per-server basis via GUI and Group Policy. The following sections explore how to do so.

On a Per-User Basis via Active Directory Users and Computers

The Active Directory Users and Computers tool enables you to set session timeouts on each user account independently. To do so, launch Active Directory Users and Computers, then open the properties interface for the user account you want to configure. Figure 3.1 shows the Sessions tab of a user account.



Use this tab to set Terminal Services timeout and reconnection settings

End a disconnected session: Never

Active session limit: Never

Idle session limit: Never

When a session limit is reached or connection is broken:

● Disconnect from session

○ End session

Allow reconnection:

● From any client

○ From originating client only

Figure 3.1: Configuring session timeouts on a user account.

In addition to setting active, idle, and disconnected session timeouts with this tool, you can configure whether to disconnect from the client or end the session when the idle or active session limit is reached. You also have the option of limiting the user to only reconnecting to a disconnected session from the client device that originally established the session. In other words, you can prevent your users from moving a terminal server session from one client device to another.

> Per-server and Group Policy-based timeouts override per-user settings, so if you want the settings on the user account to apply, you must leave the timeouts on the server unconfigured.

On a Per-User Basis via Active Directory Scripting Interface

If you want to use per-user session timeout settings and you have a large number of users in your environment, configuring each account manually can be very time consuming. WS2K3 allows you to configure these settings through the Active Directory Scripting Interface (ADSI). This way, you can batch-edit a large number of user accounts with very little effort.

You access ADSI by using the Windows Script Host (WSH); thus, you can choose whether to write your scripts in Visual Basic Script (VBScript) or Java Script. The following examples are in VBScript.

Configuring user properties through ADSI is a three-step process. First, you must open a connection to the user account, then set the properties, and finally write the changes back to the user account. To open a connection to the user account, use either the WinNT provider or the Lightweight Directory Access Protocol (LDAP).

> Although you can use the scripts in this section to configure both NT 4.0 domain and Win2K AD accounts, you can only run the scripts on a WS2K3 server; they will not work if run on Win2K or even Windows XP.

The syntax for the connection is either

```
Set objUser = GetObject(“WinNT://<domain name>/<username>,user”
```

or

```
Set objUser = Get Object(“LDAP://<distinguished name of user>”)
```

As you can see, to use LDAP, you must know the distinguished name of the user object (for example, cn=joe.user,ou=users,dc=example,dc=domain,dc=com), which is difficult if your users are spread across multiple organizational units (OUs). To make it easier, Microsoft enables the ability to use the WinNT provider for AD accounts as well. The domain controller will automatically translate the WinNT call into an LDAP call for you.

Once you have the user account open, set the parameters that you want to change. The names of the Terminal Services session timeouts and the syntax for setting them are provided in Listing 3.1.

```
objUser.MaxDisconnectionTime = [minutes, 0 for never]
objUser.MaxConnectionTime = [minutes, 0 for never]
objUser.MaxIdleTime = [minutes, 0 for never]
objUser.BrokenConnectionAction = [1,0]
      1 = end session, 0 = disconnect the session
objUser.ReconnectionAction = [1.,0]
      1 = original client only, 0 = any client
```

Listing 3.1: Names and syntax for setting Terminal Services session timeouts.

Finally, you must write the changed attributes back to the user account:

```
objUser.SetInfo
```

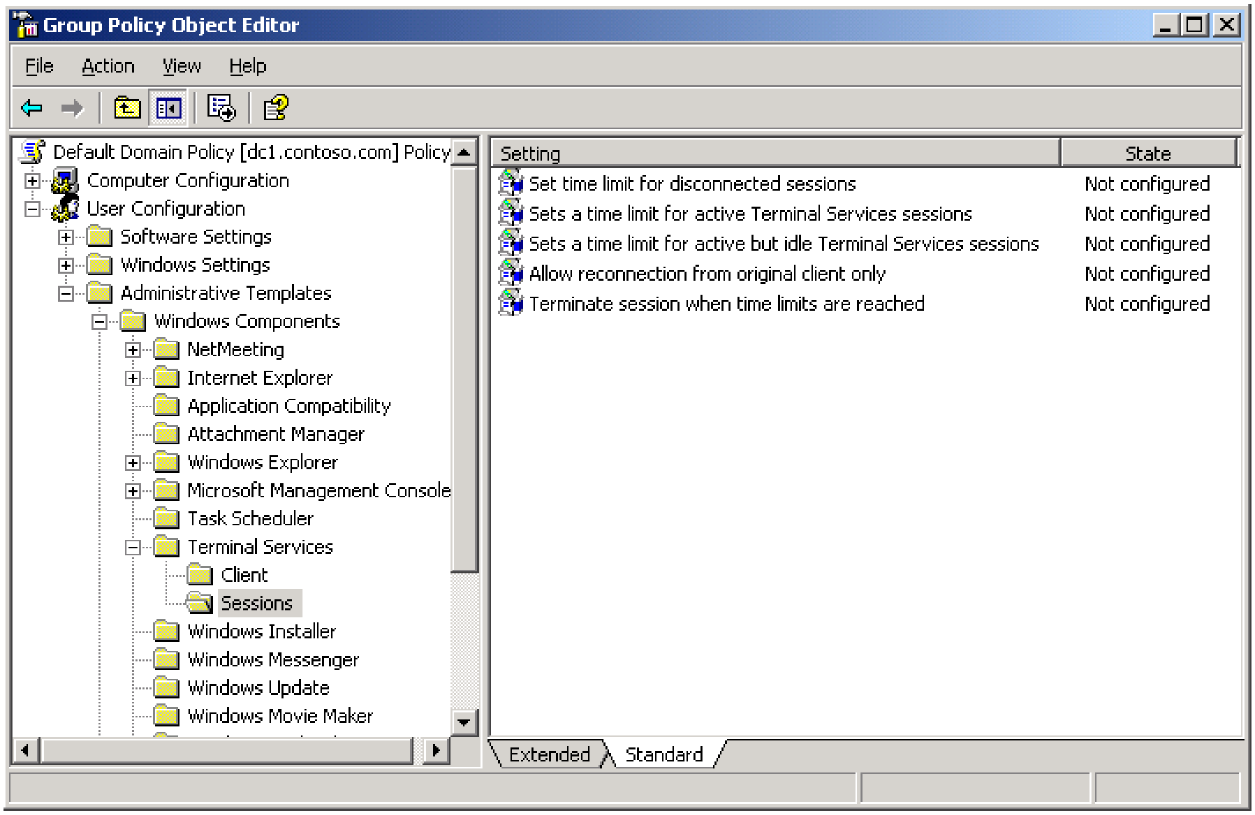
Obviously, if you want to configure a single user account, it would be faster to just use the GUI tool. However, ADSI is a useful option for configuring properties for multiple users at the same time.

📖 The Command Line Reference section at the end of this chapter provides an example script that configures all the terminal server attributes of a user account.

🖫 The Microsoft TechNet Script Center (http://www.microsoft.com/technet/scriptcenter) is a useful resource for administrative scripting. With a little scripting know-how, you can modify the example scripts to meet your unique needs.

On a Per-User Basis via Group Policy

If the desired session timeouts align with either AD group membership or OU structure, you can configure per-user session timeouts via Group Policy. To do so, use the Group Policy Management Console to create or edit a GPO that applies to the user accounts you want to configure. Then drill to User Configuration | Administrative Templates | Windows Components | Terminal Services | Sessions and configure the timeouts (see Figure 3.2).



Setting	State
Set time limit for disconnected sessions	Not configured
Sets a time limit for active Terminal Services sessions	Not configured
Sets a time limit for active but idle Terminal Services sessions	Not configured
Allow reconnection from original client only	Not configured
Terminate session when time limits are reached	Not configured

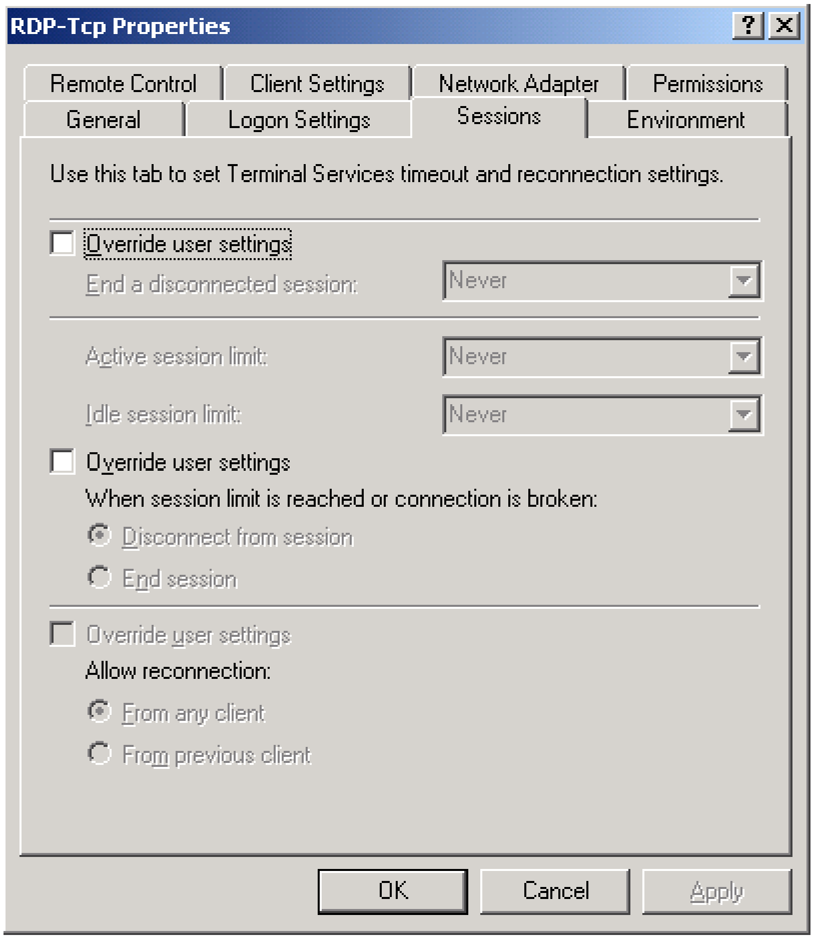
Figure 3.2: Configuring per-user session timeouts via Group Policy.

It is common to use Loopback Policy Processing mode in a terminal server environment. Loopback mode applies user configuration settings from the computer object's GPOs instead of those that apply to the user object. If you use Loopback mode, session timeouts you configure in a GPO that is scoped to the user object may not apply when logging into a terminal server.

On a Per-Server Basis via GUI

Although configuring session timeouts on a per-user basis provides the greatest amount of flexibility, it can also be very time consuming and difficult to manage. Most terminal server administrators choose to find settings that are appropriate for all users and configure timeouts on a per-server basis.

To configure per-server session timeouts, launch the Terminal Services Configuration administrative tool. Open the properties dialog box of the RDP-Tcp protocol, and go to the Sessions tab, as Figure 3.3 shows.



Use this tab to set Terminal Services timeout and reconnection settings.

□ Override user settings

End a disconnected session: Never

Active session limit: Never

Idle session limit: Never

□ Override user settings

When session limit is reached or connection is broken:

● Disconnect from session

○ End session

□ Override user settings

Allow reconnection:

● From any client

○ From previous client

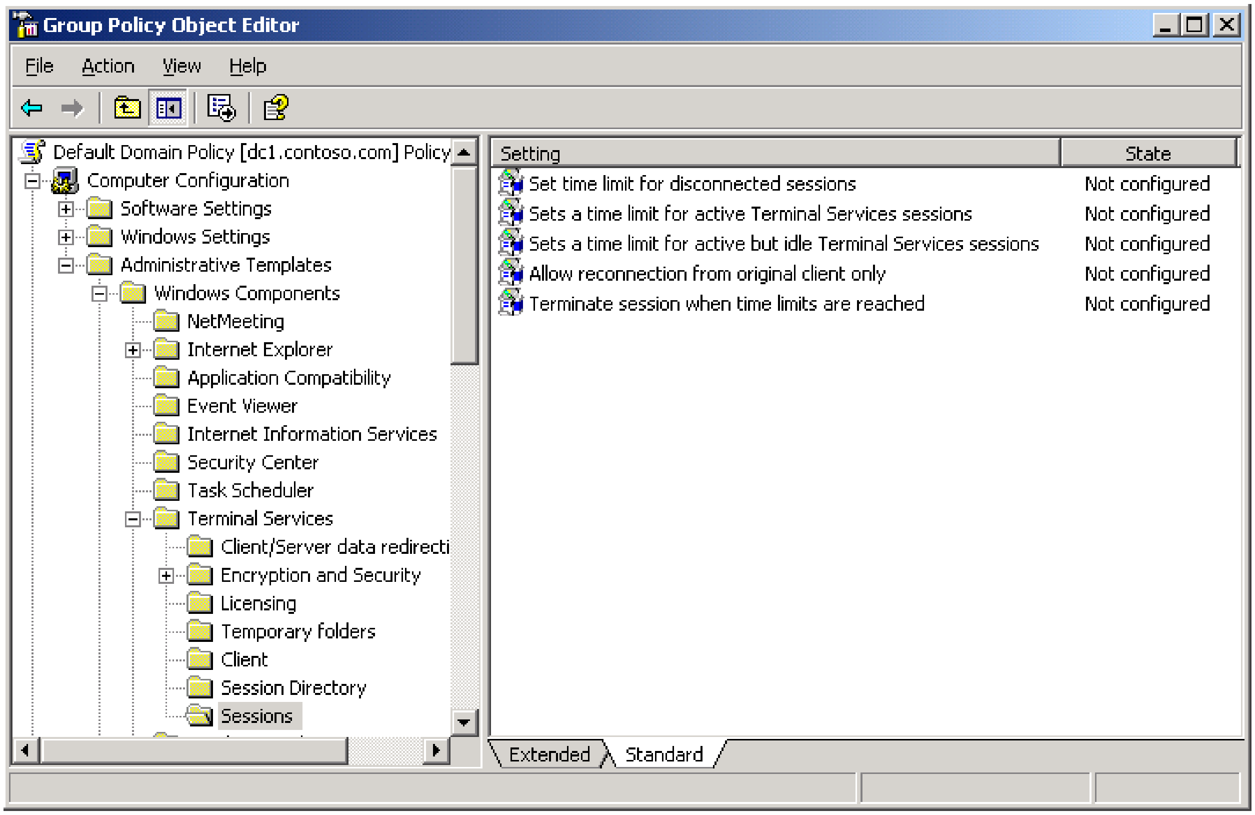
Figure 3.3: Configuring session timeouts on a per-server basis.

On this tab, you can enable the server to override the settings configured on the user account, then set server-specific timeouts. You have the ability to override the client reconnection setting as well.

On a Per-Server Basis via Group Policy

If you want to use per-server session timeout but have multiple servers to configure, it may be easier to centrally configure the session timeouts via Group Policy. This way, any new terminal servers that are added to your domain automatically acquire the correct timeout settings (provided they are placed in AD within the scope of the GPO).

To configure session timeouts via Group Policy, use the Group Policy Management Console to create or edit a GPO that applies to your terminal servers. Then drill to Computer Configuration | Administrative Templates | Windows Components | Terminal Services | Sessions and configure the timeouts (see Figure 3.4).



Setting	State
Set time limit for disconnected sessions	Not configured
Sets a time limit for active Terminal Services sessions	Not configured
Sets a time limit for active but idle Terminal Services sessions	Not configured
Allow reconnection from original client only	Not configured
Terminate session when time limits are reached	Not configured

Figure 3.4: Configuring per-server session timeouts via Group Policy.

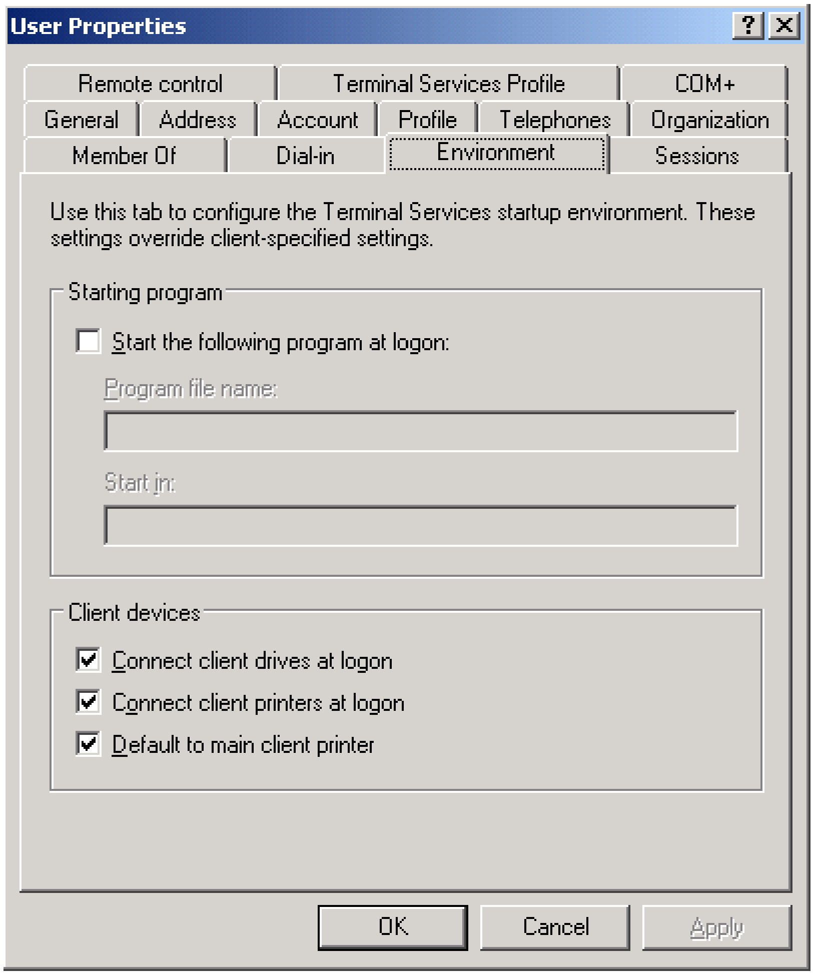
☞ It is also possible to configure per-server session timeouts via Windows Management Instrumentation (WMI). To learn more about this option, see Microsoft’s Web site at http://www.microsoft.com/technet/prodtechnol/windowsserver2003/library/ServerHelp/f826112b-c88d-42cc-a52c-c99ea467ab87.mspx.

How to Configure Client Device Redirection and Resource Mapping

Like session length limits, you can configure client device redirection and resource mapping on a per-user or per-server basis. To do so, you can use Active Directory Users and Computers and ADSI for per-user settings or on a per-server basis via a GUI or Group Policy.

On a Per-User Basis via Active Directory Users and Computers

To configure client device redirection and resource mapping via Active Directory Users and Computers, launch the tool, and open the properties window of the user object you want to configure. Figure 3.5 shows the Environment tab of a user object.



Use this tab to configure the Terminal Services startup environment. These settings override client-specified settings.

Starting program

☐ Start the following program at logon:

Program file name:

Start in:

Client devices

☑ Connect client drives at logon

☑ Connect client printers at logon

☑ Default to main client printer

Figure 3.5: Configuring client redirection via Active Directory Users and Computers.

On this tab, you can enable or disable client drive mapping and client printer mapping and set whether to default to the main client printer if it is different than the default printer set in the user's terminal server profile. This tab also allows you to set a specific program to be used instead of the Windows Explorer Shell whenever this user account logs into a terminal server.

> The starting program setting does not launch the specified program within a desktop shell on the terminal server. If you configure this setting, the user will only see the specified application and will not have access to a desktop shell.

On a Per-User Basis via ADSI

If you want to script the configuration of these user settings, use the following ADSI interfaces:

```
objUser.ConnectClientDrivesAtLogon = [1,0]
objUser.ConnectClientPrintersAtLogon = [1,0]
objUser.DefaultToMainPrinter = [1,0]
objUser.TerminalServicesInitialProgram = [“path to program”]
objUser.TerminalServicesWorkDirectory = [“path to directory”]
```

Remember, you must first open a connection to the user object using either the WinNT or LDAP provider:

```
Set objUser = GetObject(“WinNT://<domain name>/<username>,user”
```

or

```
Set obUser = Get Object(“LDAP://<distinguished name of user>”)
```

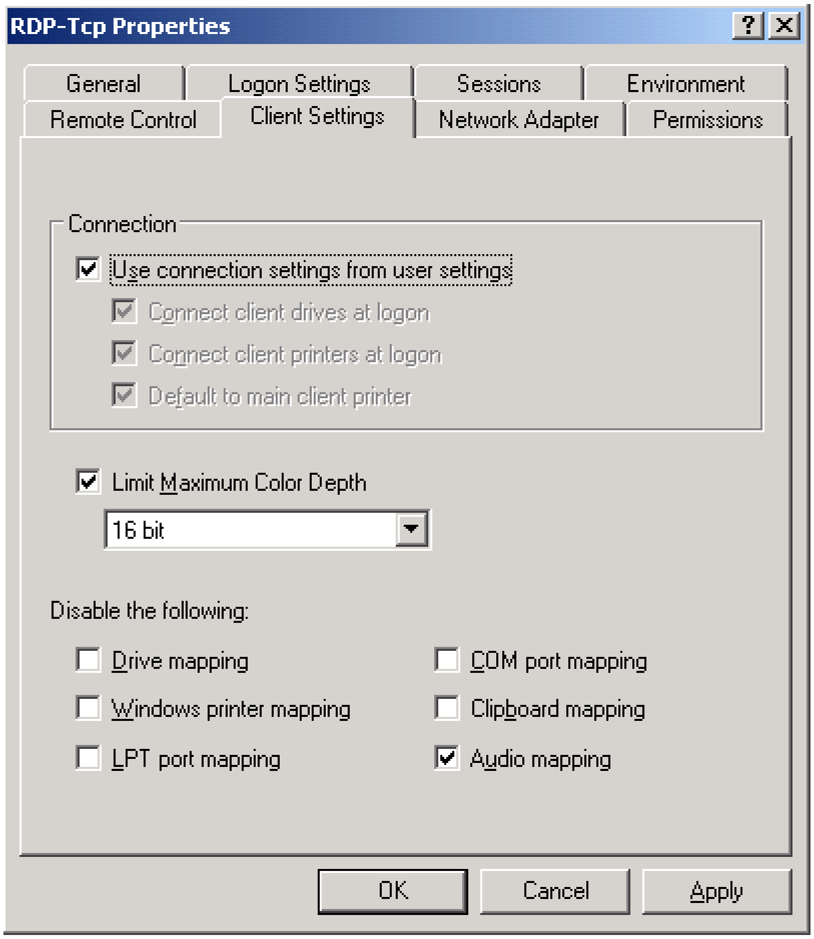
And then save your changes after configuring the attributes:

```
objUser.SetInfo
```

> Once again, if you want the per-user settings to be obeyed, you must leave both the per-user and Group Policy settings unconfigured.

On a Per-Server Basis via GUI

To configure client device redirection at the server, launch the Terminal Services Configuration administrative tool, and open the properties dialog box of the RDP-Tcp protocol. Figure 3.6 shows the Client Settings tab.



Connection

☑ Use connection settings from user settings

☑ Connect client drives at logon

☑ Connect client printers at logon

☑ Default to main client printer

☑ Limit Maximum Color Depth

16 bit

Disable the following:

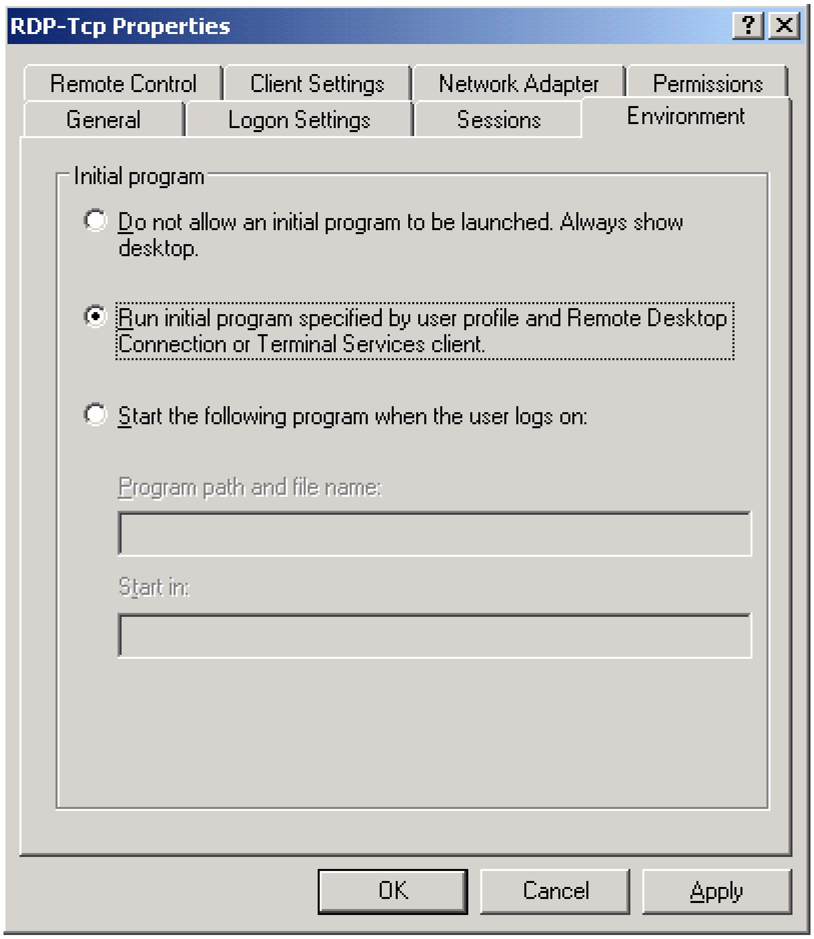
□ Drive mapping	□ COM port mapping
□ Windows printer mapping	□ Clipboard mapping
□ LPT port mapping	☑ Audio mapping

Figure 3.6: Configuring client device redirection at the server.

On this tab, you can override the same three settings that are available on the user account. In addition, you can limit the maximum color depth that is sent to the client and disable specific device mapping features.

Audio mapping is disabled by default on WS2K3 SP1.

To configure the starting program feature that you saw on the user account, go to the Environment tab, which Figure 3.7 shows.



Initial program

○ Do not allow an initial program to be launched. Always show desktop.

● Run initial program specified by user profile and Remote Desktop Connection or Terminal Services client.

○ Start the following program when the user logs on:

Program path and file name:

Start in:

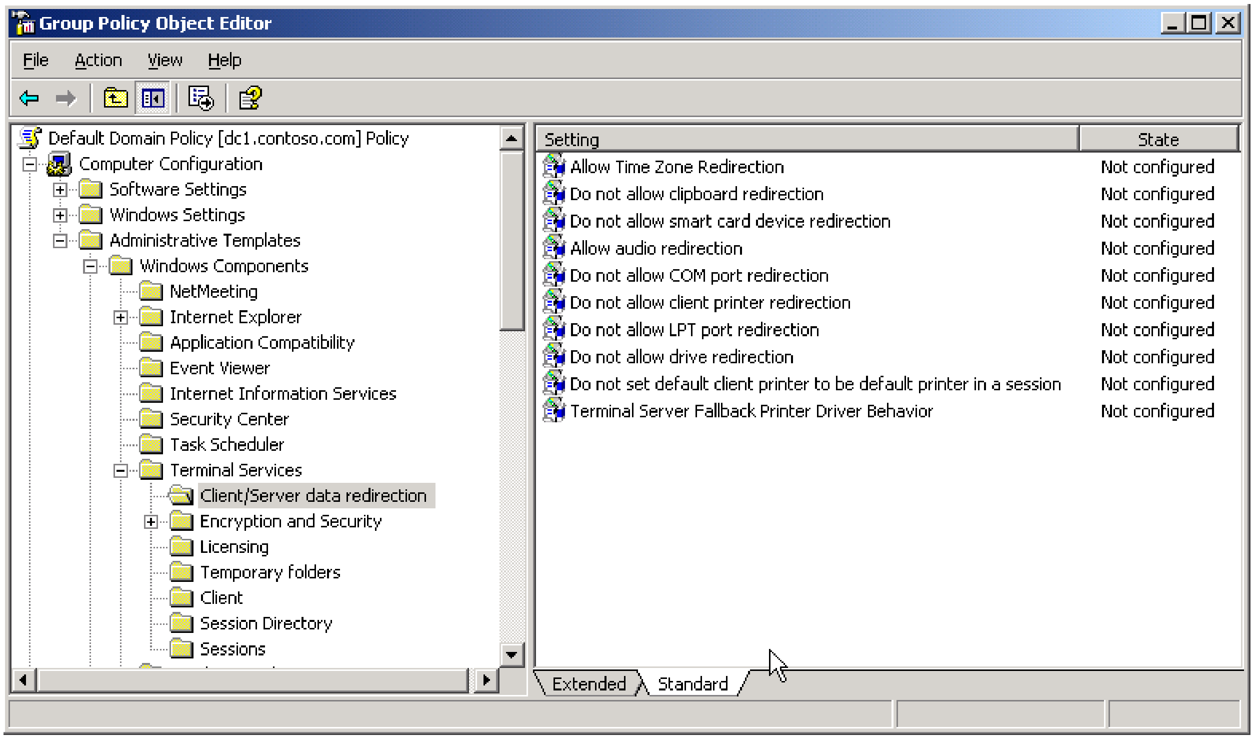
Figure 3.7: Configuring initial program settings at the server.

On this tab, you have three options: override both terminal server client and per-user settings and always display a desktop shell, obey the client and per-user settings, or always start a specific program.

Be very careful when configuring an initial program at the server level as doing so will apply to administrative accounts as well as users, and you might lose your own ability to see a desktop shell.

On a Per-Server Basis via Group Policy

You can centrally configure client device redirection via Group Policy. To do so, launch the Group Policy Management Console, and create or edit a GPO that applies to your terminal servers. Drill to Computer Configuration | Administrative Templates | Windows Components | Terminal Services | Client/Server Data Redirection, as Figure 3.8 illustrates.



Setting	State
Allow Time Zone Redirection	Not configured
Do not allow clipboard redirection	Not configured
Do not allow smart card device redirection	Not configured
Allow audio redirection	Not configured
Do not allow COM port redirection	Not configured
Do not allow client printer redirection	Not configured
Do not allow LPT port redirection	Not configured
Do not allow drive redirection	Not configured
Do not set default client printer to be default printer in a session	Not configured
Terminal Server Fallback Printer Driver Behavior	Not configured

Figure 3.8: Configuring client device redirection via Group Policy.

In addition to the settings you have in the Terminal Services Configuration tool, you can also control client time zone mapping and smart card redirection from the Group Policy Management Console.

> Client time zone mapping is disabled by default on WS2K3.

> To configure an initial program via GPO, go up a level to Computer Configuration | Administrative Templates | Windows Components | Terminal Services, and configure the application in the *Start a program on Connection* setting. Once again, be careful as doing so will also prevent an administrator from receiving a desktop shell over Terminal Services.

How to Configure User Profiles

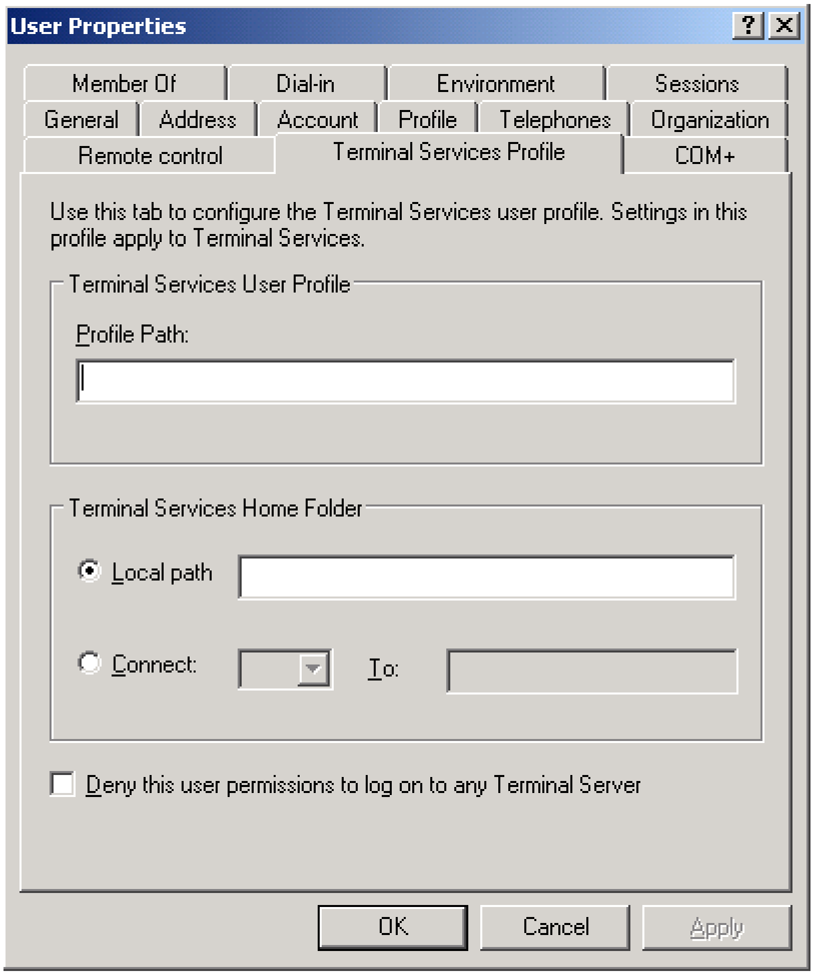
By default, all Windows systems will use local profiles. If you have only one terminal server, this default can be fine; however, most environments have multiple servers that are either load balanced or provide different applications. If you want users to have a consistent experience across the servers, you must configure either roaming or mandatory profiles.

On a Per-User Basis via Active Directory Users and Computers

To configure a profile on a per-user basis, use Active Directory Users and Computers, and open the properties dialog box for the user account. You will see that there is both a Profile and a Terminal Services Profile tab. The Profile tab is primarily used for workstation logons, and the settings on the Terminal Services Profile tab will be loaded only by terminal server sessions.

> If you use roaming or mandatory profiles for your workstations and you do not configure a separate profile on the Terminal Services Profile tab, the terminal server will load the same profile that is defined on the Profile tab.

To configure a specific roaming or mandatory profile to be used on terminal servers, go to the Terminal Services Profile tab, which Figure 3.9 shows.



Use this tab to configure the Terminal Services user profile. Settings in this profile apply to Terminal Services.

Terminal Services User Profile

Profile Path:

Terminal Services Home Folder

● Local path

○ Connect: To:

□ Deny this user permissions to log on to any Terminal Server

Figure 3.9: Configuring a Terminal Services profile on a user account.

On this tab, specify the full UNC path to the profile you want this user to load at logon. If you want to use mandatory profiles, set all users to the same path, configure the central profile, and change the NTUSER.DAT filename to NTUSER.MAN. On this tab, you can also configure a separate user home directory to be used on terminal servers, but doing so is fairly rare, as most likely your users will need the same home directory for both workstations and terminal servers.

> ☞ You can also flag a specific user account to deny access to any terminal server on this tab.

On a Per-User Basis via ADSI

Like the other Terminal Services user attributes, you can configure Terminal Services profile settings via ADSI. After opening a connection to the user object:

```
Set objUser = GetObject(“WinNT://<domain name>/<username>,user”
```

or

```
Set obUser = Get Object(“LDAP://<distinguished name of user>”)
```

Set any of the following attributes:

```
objUser.TerminalServicesProfilePath = [“path to directory”]
objUser.TerminalServicesHomeDirectory = [“path to directory”]
objUser.TerminalServicesHomeDrive = [“drive letter:”]
objUser.AllowLogon = [1,0]
```

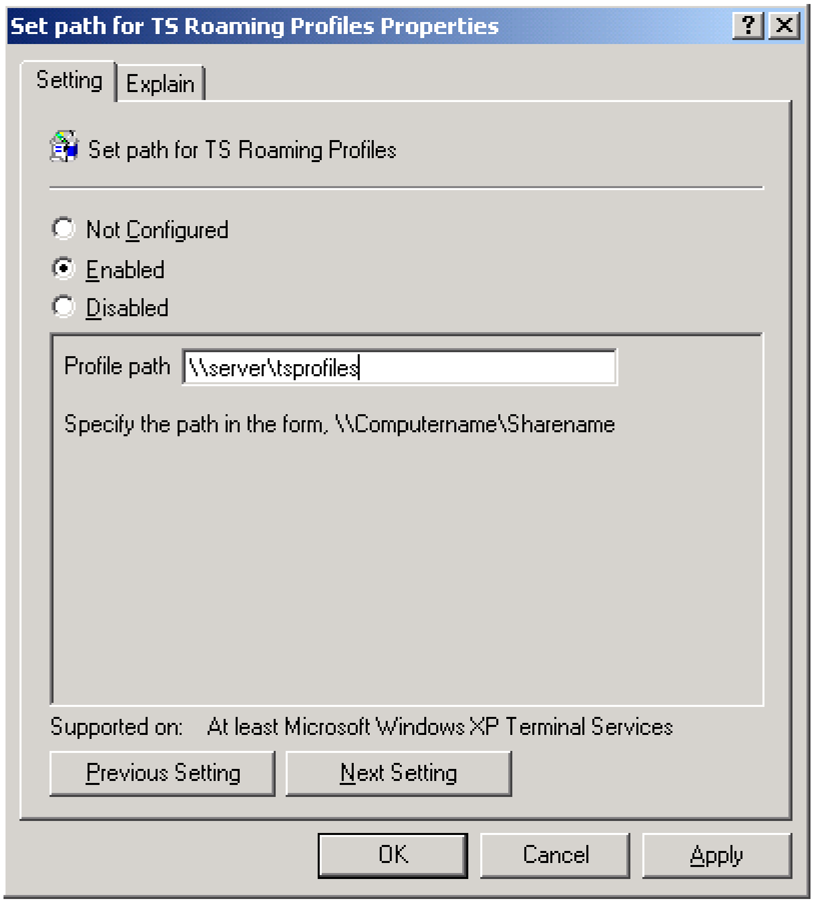
and then save your changes:

```
objUser.SetInfo
```

Via Group Policy

A feature that is new in WS2K3 is the ability to override the user account settings and configure a terminal server to use only local profiles or to use a separate roaming profile share from the one defined in the user account. To take advantage of either of these options, use the Group Policy Management Console to create or edit a GPO that applies to your terminal servers.

To set an alternative roaming profile path, drill to Computer Configuration | Administrative Templates | Windows Components | Terminal Services. Configure the *Set Path for TS Roaming Profiles* setting with the UNC path of the file share you want to use. Do not specify the %username% variable, as it is automatically added to the path for each user. Figure 3.10 shows the UI for this setting.



Set path for TS Roaming Profiles Properties

Set path for TS Roaming Profiles

○ Not Configured
● Enabled
○ Disabled

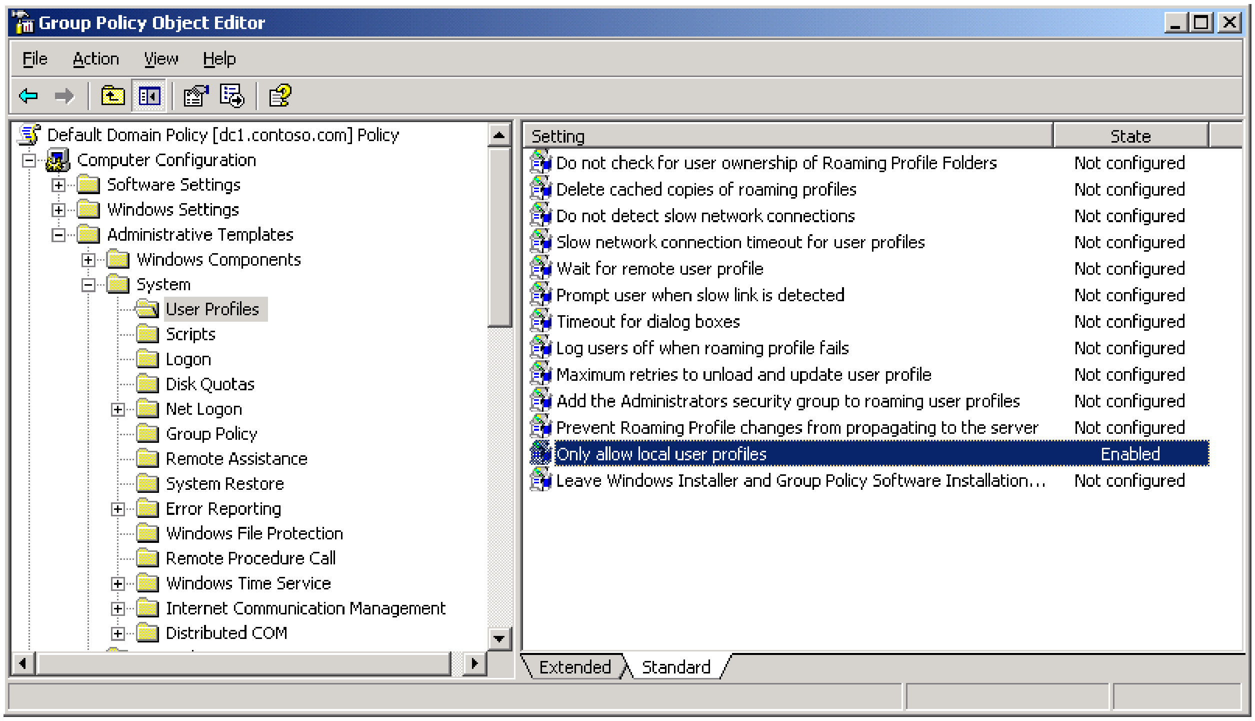
Profile path: \\server\tsprofiles

Specify the path in the form, \\Computername\Sharename

Supported on: At least Microsoft Windows XP Terminal Services

Figure 3.10: Configuring an alternative Terminal Services profile share via GPO.

To configure the terminal server to ignore the profile path in the user account and exclusively use local profiles, drill to Computer Configuration | Administrative Templates | System | User Profiles, and set *Only allow local user profiles* to enabled (see Figure 3.11).



Setting	State
Do not check for user ownership of Roaming Profile Folders	Not configured
Delete cached copies of roaming profiles	Not configured
Do not detect slow network connections	Not configured
Slow network connection timeout for user profiles	Not configured
Wait for remote user profile	Not configured
Prompt user when slow link is detected	Not configured
Timeout for dialog boxes	Not configured
Log users off when roaming profile fails	Not configured
Maximum retries to unload and update user profile	Not configured
Add the Administrators security group to roaming user profiles	Not configured
Prevent Roaming Profile changes from propagating to the server	Not configured
Only allow local user profiles	Enabled
Leave Windows Installer and Group Policy Software Installation...	Not configured

Figure 3.11: Configuring a server to only use local profiles.

Per Server (Local Machine Policy)

You can configure either of the previously mentioned settings on a per-server basis without creating a separate GPO in your domain. To do so, edit the local machine policy by launching the Group Policy Editor (GPEDIT.MSC) from a Run dialog box.

Third-Party Products for Profile Management

User profiles can be the biggest challenge for any Windows systems administrator. Users want the convenience of roaming profiles, but with frequent logons to multiple machines, they can become corrupt. Systems administrators want the central control of mandatory profiles but have to face the challenge of editing them when changes are needed. There are third-party products—such as triCerat's Simplify Suite, AppSense's AppSense Management Suite, and BrsSuite from BrainSyS—that help you bridge the gap between the two.

One of the components of Simplify Suite is Simplify Profiles, which gives you the ability to leverage the stability of mandatory profiles while still enabling users to save designated settings across sessions and servers. Simplify Profiles also makes changes to settings in the profiles easy by providing a central UI for edits. Alternatively, BrsSuite is a free option that stores user profile information in a SQL database, allowing you to make single-point changes to all profiles while enabling your users to store designated per-user settings.

User Profile Hive Cleanup Service

Regardless of whether you use local, roaming, or mandatory profiles, user sessions can get locked in memory at logoff. This can occur because of software defects, quick logon/logoffs, or hung applications. When a user session is locked, the server cannot properly unload the profile and cannot reclaim all the resources used by the session. Microsoft's User Profile Hive Cleanup service can assist in this process.

The User Profile Hive Cleanup service monitors the user logoff process, and if any user process handles are keeping the user hive from being unloaded, the service will proactively kill the process so that the profile can be unloaded. The User Profile Hive Cleanup service is fully supported by Microsoft and should be installed on all your terminal servers.

How to Install UPH Clean

To install the User Profile Hive Cleanup service, download the MSI package from Microsoft's Web site at http://www.microsoft.com/downloads/details.aspx?familyid=1b286e6d-8912-4e18-b570-42470e2f3582&displaylang=en. Either install the package manually or assign it to your terminal servers via Group Policy. The User Profile Hive Cleanup service is a simple "set it and forget it" tool, although there are some advanced logging options that you can control via the registry. See the readme.txt file for the current information about these options.

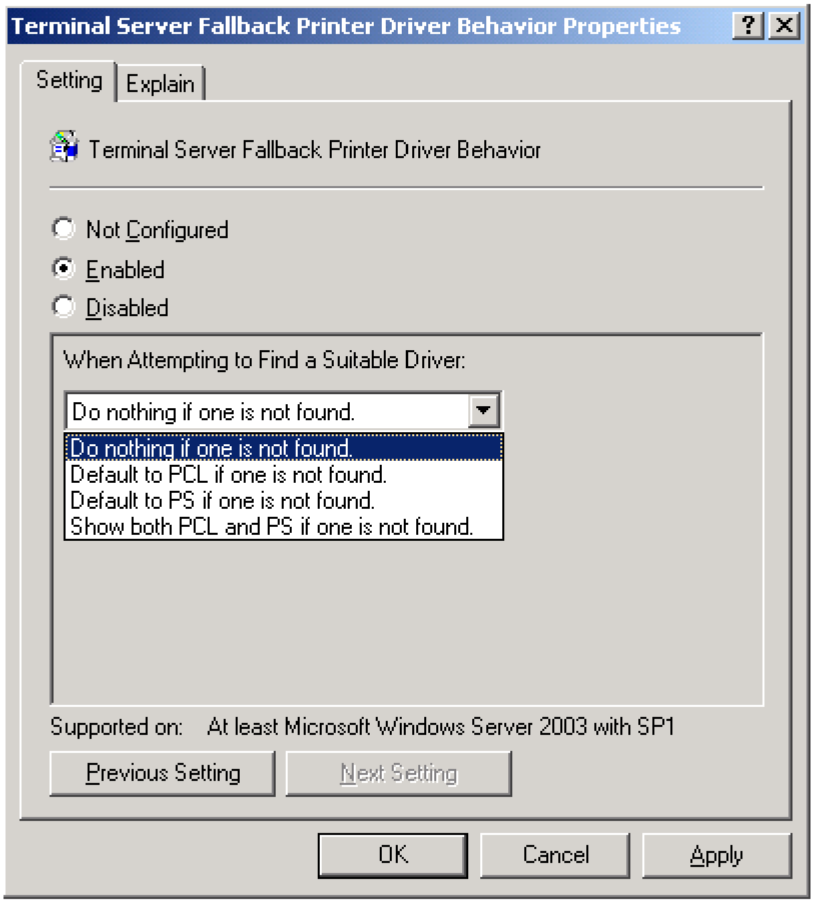
How to Manage Printing in a Terminal Server Environment

When a user connects to a terminal server, if client printer mapping is enabled, the terminal server enumerates all printers defined on the workstation and attempts to connect to them. RDP 5.2 supports USB, Serial (COM), Parallel (LPT), and network printers. Before the release of WS2K3 SP1, the terminal server had to have a native driver for the printer pre-installed by an administrator for the mapping to be successful. SP1 introduced a fallback printer driver that can be used whenever a native driver is not available.

Microsoft Fallback Printer Driver

The Microsoft fallback printer driver can be used whenever the native driver is not installed on the terminal server. The fallback driver feature is disabled by default, so if the native driver is not available, the client printer will not be mapped. The driver supports both Printer Control Language (PCL) and PostScript printer languages, but only basic functionality, so advanced features—such as duplex, multiple paper trays, high resolution, and so on—will not be available.

To enable the fallback printer driver, use the Group Policy Management Console to create or edit a GPO that applies to your terminal servers, then drill to Computer Configuration | Administrative Templates | Windows Components | Terminal Services | Client/Server Data Redirection. Open the *Terminal Server Fallback Printer Driver Behavior* setting, which Figure 3.12 shows.



Terminal Server Fallback Printer Driver Behavior Properties

Terminal Server Fallback Printer Driver Behavior

○ Not Configured
● Enabled
○ Disabled

When Attempting to Find a Suitable Driver:

Do nothing if one is not found.

- Do nothing if one is not found.
- Default to PCL if one is not found.
- Default to PS if one is not found.
- Show both PCL and PS if one is not found.

Supported on: At least Microsoft Windows Server 2003 with SP1

Figure 3.12: Configuring the fallback printer driver behavior.

After enabling this setting, you must select the appropriate behavior for the server to take:

- Do nothing if one is not found—Disables the fallback driver. This option is the same as not configuring or disabling the setting.
- Default to PCL if one is not found—If no suitable printer driver can be found, the terminal server uses the Hewlett-Packard compatible PCL fallback printer driver.
- Default to PS if one is not found—If no suitable printer driver can be found, terminal server uses the Adobe PostScript fallback printer driver.
- Show both PCL and PS if one is not found—In the event that no suitable driver can be found, show both PostScript-based and PCL-based fallback printer drivers.

> ☞ You can also enable the fallback printer driver locally on a specific server by editing the local machine policy (GPEDIT.MSC).

Third-Party Products for Printer Driver Management

The fallback printer driver is a useful advancement for Microsoft Terminal Services, however, it is still a very basic driver and does not support advanced features of many printers. If your users need the full functionality of their printers when working on the terminal server, you should look to third-party tools to enhance the native printing functionality.

One such tool is triCerat's Simplify Printing, part of the Simplify Suite. Simplify Printing enables users to access the local printer properties and features even when working in a terminal server-based application. Simplify Printing does require both a server and client install, but the advantages of not having to install drivers on your terminal servers while still providing users will the full features of their printers is worth the effort.

Another third-party option for printer management is ThinPrint, which uses Driver Free Printing to spool documents to local printers while managing the bandwidth and performance latency that can arise during printing. ThinPrint also requires both client and server components.

Group Policy Reference

The following lists highlight the Group Policy settings user environment considerations discussed in this chapter:

- To configure per-user session timeouts—User Configuration | Administrative Templates | Windows Components | Terminal Services | Sessions
- To configure per-server session timeouts—Computer Configuration | Administrative Templates | Windows Components | Terminal Services | Sessions
- To configure client device redirection—Computer Configuration | Administrative Templates | Windows Components | Terminal Services | Client/Server Data Redirection
- To configure an alternative path for Terminal Services profiles—Computer Configuration | Administrative Templates | Windows Components | Terminal Services; configure the *Set Path for TS Roaming Profiles* setting
- To configure a server to use only local profiles—Computer Configuration | Administrative Templates | System | User Profiles; set *Allow only local user profiles* to enabled
- To enable the Microsoft fallback printer driver—Computer Configuration | Administrative Templates | Windows Components | Terminal Services | Client/Server Data Redirection; set the *Terminal Server Fallback Printer Driver Behavior* setting

Command-Line Reference

Listing 3.2 provides an example script to configure all the terminal services-related attributes of a user object via ADSI.

```
Set objUser = GetObject _
      (“LDAP://cn=joe.user,ou=users,dc=example,dc=domain,dc=com”)
‘ or: Set objUser = GetObject(“WinNT://example/joe.user,user”)
objUser.ConnectClientDrivesAtLogon = 1
objUser.ConnectClientPrintersAtLogon = 1
objUser.DefaultToMainPrinter = 1
objUser.TerminalServicesInitialProgram = “C:\windows\notepad.exe”
objUser.TerminalServicesWorkDirectory = “c:\windows
objUser.TerminalServicesProfilePath = _
“\\server\tsprofiles\joe.user”
objUser.TerminalServicesHomeDirectory = _
“\\server\home\joe.user”
objUser.TerminalServicesHomeDrive = “H:”
objUser.AllowLogon = 1
objUser.MaxDisconnectionTime = 15
objUser.MaxConnectionTime = 0
objUser.MaxIdleTime = 180
objUser.BrokenConnectionAction = 0
objUser.ReconnectionAction = 0
objUser.EnableRemoteControl = 1
objUser.SetInfo
```

Listing 3.2:An example script to label all the terminal services-related attributes of a user object via ADSI.

Summary

This chapter covered session timeouts, client device redirection, and user profile management. It walked you through the steps needed to configure these features manually, via Group Policy, and by script. The next chapter will cover the configuration of load balancing, session directory, and user session management.

Content Central

Content Central is your complete source for IT learning. Whether you need the most current information for managing your Windows enterprise, implementing security measures on your network, learning about new development tools for Windows and Linux, or deploying new enterprise software solutions, Content Central offers the latest instruction on the topics that are most important to the IT professional. Browse our extensive collection of eBooks and video guides and start building your own personal IT library today!

Download Additional eBooks!

If you found this eBook to be informative, then please visit Content Central and download other eBooks on this topic. If you are not already a registered user of Content Central, please take a moment to register in order to gain free access to other great IT eBooks and video guides. Please visit: http://www.realtimepublishers.com/contentcentral/.