Interested in learning more about security?

SANS Institute
InfoSec Reading Room

This paper is from the SANS Institute Reading Room site. Reposting is not permitted without express written permission.

Secure Remote Server Administration of the Windows Server Family Using Windows Terminal Services

Secure remote server administration is part of a multi-layered secure network environment. Servers and networking environments need to be properly configured, and analysts need to use appropriate access levels while performing their tasks. The tools that an analyst uses must be secure or used in a secure fashion. With the advent of Windows 2000 Server and Windows Terminal Services, secure remote administration is possible across a networked environment. In this paper, I propose a method for installing and configuring W...

Copyright SANS Institute
Author Retains Full Rights

AD

Secure Remote Server Administration of the Windows Server Family Using Windows Terminal Services

John B. "Bill" Evrigenis, Jr.
GSEC
v1.4b

Abstract

Secure remote server administration is part of a multi-layered secure network environment. Servers and networking environments need to be properly configured, and analysts need to use appropriate access levels while performing their tasks. The tools that an analyst uses must be secure or used in a secure fashion. With the advent of Windows 2000 Server and Windows Terminal Services, secure remote administration is possible across a networked environment. In this paper, I propose a method for installing and configuring Windows Terminal Services in a secure environment, I propose methods for developing an analyst's work environment, and I propose an essential set of tools.

This paper provides an overview of Windows Terminal Services (WTS) and its operating scenarios. I discuss how to install and secure Windows Terminal Services. However, because WTS can be used as part of defense in depth strategy for secure computing, I also provide details for configuring a workstation environment to utilize WTS. In addition, I provide guidance for user account administration as it is relates to remote server support. Finally, I provide a framework for securely managing remote servers through standardized tools and workstation environment.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Overview of Windows 2000 Terminal Services

Running Terminal Services allows an analyst to perform administrative tasks as if he or she is sitting at the server console. Remote administration is becoming increasingly popular due to it being a low cost alternative for server support. There are several benefits to running Terminal Services. Terminal Services provides a graphical user interface for any Windows 2000 Servers running Terminal Services. In addition, almost all administrative tasks are available with negligible performance impact. Low bandwidth connections are available using Remote Desktop Protocol (RPD) and the RDP communications protocol can be encrypted using 128-bit key encryption.[1]

Terminal Services enables an analyst to establish a graphical window or desktop session from a Windows client computer (Windows 95, Windows NT, Windows XP, or Windows 2000 Professional), with the session actually being executed on the remote console, a Windows 2000 Server. The analyst actually sees the remote desktop or console and is able to perform administrative functions without being present at the console.[2] This remote support functionality can be utilized to support servers miles and even oceans away.

Remote Desktop Protocol (RDP) is an International Telecommunications Union (ITU) standard protocol (T.120) that Microsoft uses for communication between the Terminal Services server and client. The RDP protocol has built-in keyboard and mouse drivers as well as graphics device API calls. RDP is the basis of communication for Terminal Services. In addition, the kernel of the operating system with Terminal Services installed is configured to run multiple sessions simultaneously. The terminal server then uses RDP to package the communications and transmits the data across the network.[3] Thus a client computer, such as Windows XP, with Terminal Services Client installed, can remotely control a Windows 2000 Server with Terminal Services. Carrying this framework one step further, one Terminal Services client can be used to remotely administer multiple remote Windows 2000 Server with Terminal Services.

Windows 2000 Terminal Services (WTS) can be configured in one of two ways: it can be configured in Application server mode or in Remote Administration mode. It is my intention to provide an overview of how to implement the use of WTS in a remote server administration environment. I will provide guidelines on how to implement the server environment for secure remote administration purposes. In addition, part of the configuration that I recommend utilizes Terminal Services in Application server mode. Therefore, I will provide guidelines on establishing Terminal Services in Application server mode as well.

Application Server Mode

WTS configured in Application server mode allows more than one user to simultaneously establish network connections and run applications on the server itself. This is the traditional configuration of WTS and is the original "thin-client" solution from Microsoft, where a large

---

[1] "Using Terminal Services for Graphical Remote Administration of the Windows 2000 Server Family", URL http://www.microsoft.com/technet/prodtechnol/win2kts/maintain/monitor/tsremote.asp?frame=true

[2] Smith, Randy Franklin. "Terminal Services, Part 1", Windows and .Net Magazine Network, February 1, 2001. URL: http://www.windowsitsecurity.com/Articles/Index.cfm?ArticleID=19791

[3] Harwood, Ted. Windows NT Terminal Server and Citrix MetaFrame. New Riders Publishing, 1998, page 6.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

central server manages application processing requirements and the client workstations are small, implying a "thin-client".[4] Applications run in allocated memory on the WTS application server, each user maintains separate profiles and user configuration settings.[5]

Remote Administration Mode

Each Windows 2000 Server with WTS Remote Administration installed supports 2 simultaneous user sessions. There is no license fee required for using WTS in Remote Administration mode. Terminal Services in remote administration mode has been tightly integrated with the Windows 2000 Server kernel. As a result, CPU utilization is minor and memory requirements are about 2MB. Microsoft recommends that Terminal Services in remote administration mode is enabled on every Windows 2000 BackOffice Server or domain controller.[6]

Recommended Environment Configuration

I recommend that WTS configured in Remote Administration mode should be installed on all Windows 2000 Servers that require remote administration. Depending on the size of your organization, at least one or two servers should be established to support a WTS Application server. The WTS Application server should be used as the connecting point for managing all remote Windows 2000 Servers and Windows NT Servers. Analysts establish network connections to the Application server using Windows Terminal Services Client. Through the Windows Terminal Services Client session, each remote server can be management using the appropriate tools. To help illustrated this concept; I have diagrammed a sample logical configuration in Figure 1.

© SANS Institute 2003, Author retains full rights.

[4] Harwood, page 3.
[5] Harwood, page 7.
[6] "Using Terminal Services for Graphical Remote Administration of the Windows 2000 Server Family", URL http://www.microsoft.com/technet/prodtechnol/win2kts/maintain/monitor/tsremote.asp?frame=true

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

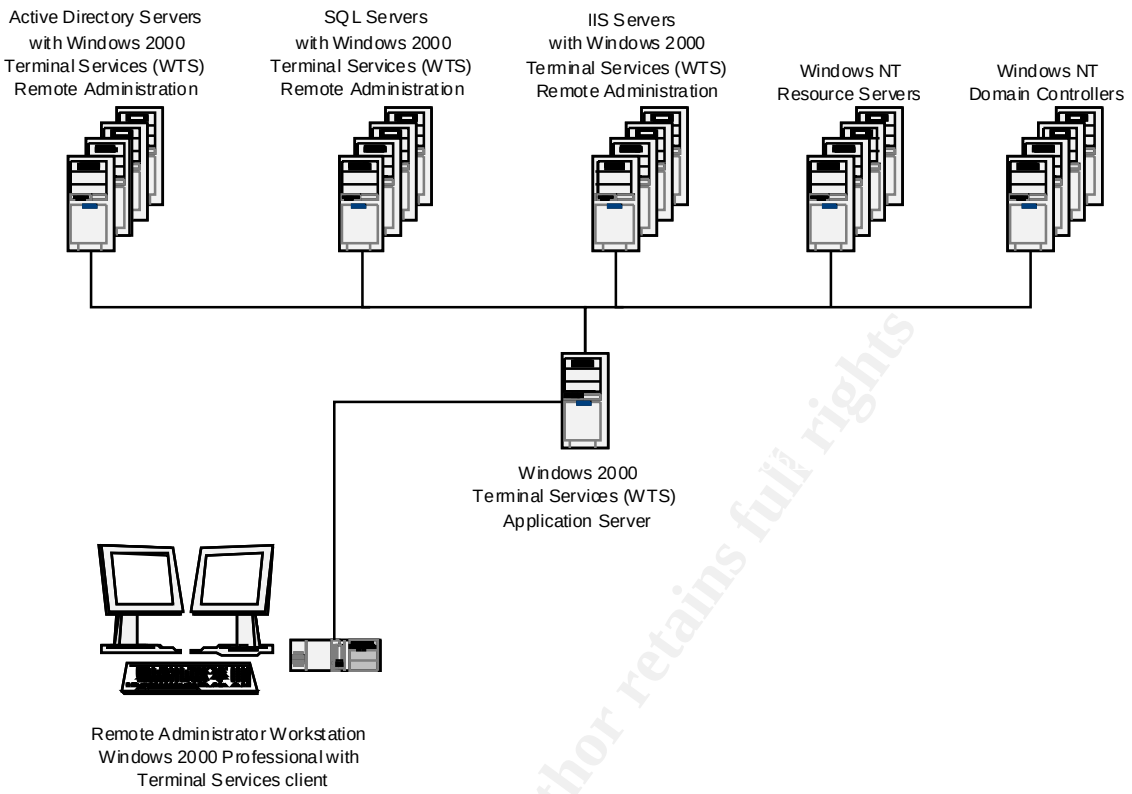

Figure 1
Sample Configuration of Terminal Services Environment

It is my intention to provide an overview of the requirements for establishing the environment shown in Figure 1. In this regards, I will establish a plan for configuring the Windows Terminal Services environment and establish a baseline configuration for the remote administrator's workstation.

Windows NT Server Configuration. Terminal Services is available for Windows NT Server, however it is sold as a separate product. In addition, Microsoft announced this year that it would cease support of Windows NT, by December 2003.[7] Fortunately, organizations can still provide remote support using a number of standard tools. I have documented a few tools available to the analyst in Appendix C. However, in light of the fact that Windows NT Server is no longer being supported, I will focus the remainder of the discussion on Windows 2000 Server.

Installing Windows Terminal Services

WTS must be installed separately using the Add/Remove Programs Control Panel applet.[8] The default configuration allows members of the local Administrators group to configure and manage the server, but depending on the security environment of your organization, this

[7] "Retiring Windows NT Server 4.0: Changes in Product Availability and Support", URL: http://microsoft.com/ntserver/ProductInfo/Availability/Retiring.asp
[8] Boswell, Bill. "Remote Operator", Microsoft Certified Professional Magazine Online, June 2002, page 5. URL: http://mcpmag.com/Features/article.asp?EditorialsID=275

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

default option may need to be modified to accommodate your organization's security requirements. The Terminal Server mode may only be selected during the installation process. If there is a need to ever change the Terminal Server mode from one mode to another, the Terminal Services will need to be reinstalled and all existing configuration settings will be lost.

Installing WTS to use Remote Administration Mode

As mentioned previously, Windows Terminal Services is added through the Control Panel Add/Remove Programs applet. The service can be then selected from the Add/Remove Windows Components screen. The process is straight forward, when prompted, choose the Remote administration mode radio button.[9] The screen shots below show two of the screens displayed the Windows Components Wizard.

Figure 2
Windows Components Wizard to enable Terminal Services

[9] Jones, Don. "Must-Have Remote Administration Tools", Windows and .Net Magazine Network, May 2002, page 2. URL: http://www.winnetmag.com/Articles/Index.cfm?ArticleID=24536

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Figure 3
Windows Components Wizard -Terminal Services Mode Selector

When the WTS installation is complete you will be prompted to reboot the server.

Installing WTS to use Remote Administration Mode

Installing WTS to use Application server mode is very similar to Remote administration mode install process. When prompted by the Windows Components Wizard, select the Application server mode button and then press the Next button (See Figure 3). You then be prompted to choose permission compatibility. Select "Permissions compatible with Windows 2000 Server", as shown in Figure 4.

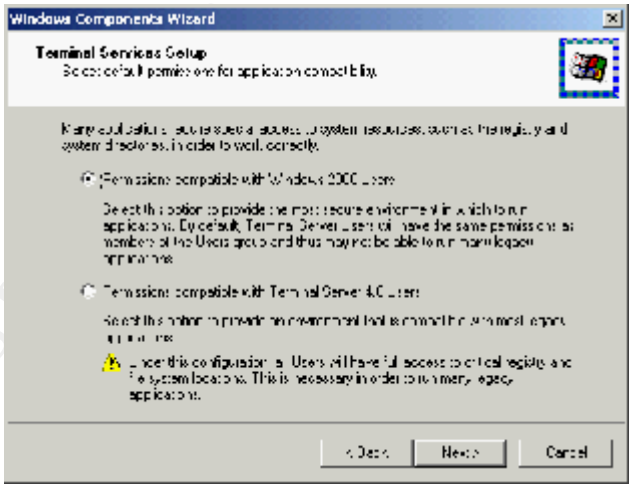

Figure 4
Windows Components Wizard - Terminal Services
Permissions Compatibility Selector

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

You should not select “Permissions compatible with Terminal Server 4.0 Users” as this may give users undesirable permissions.[10]

After selecting the “Next” button, you may be given a caution regarding the programs currently installed on the Windows 2000 Server. The caution indicates that the programs installed may not work properly in an application server environment. However, the only purpose of the WTS Application server in our example is to create a safe launching pad to other remote servers. Only a few applications, if any, will be required to be shared, for example, SQL Enterprise Manager and Query Analyzer if you support Microsoft SQL servers. Depending on your environment, you may not need to install any additional applications as a number of remote administrations tasks can be managed from the standard tools installed with the basic server configuration, such as the Microsoft Management Console (MMC).

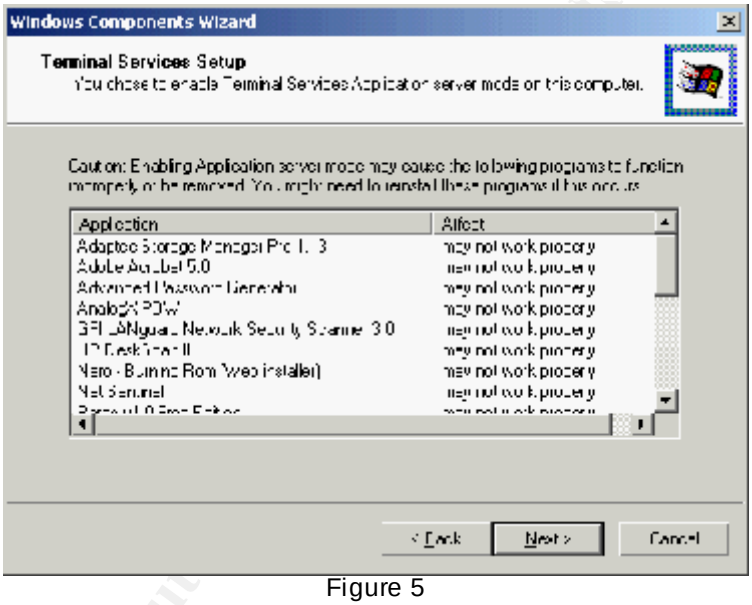

Figure 5
Windows Components Wizard - Terminal Services Set
Enable Application Serving

Pressing the “Next” button completes the installation process. Once again, when the WTS installation is complete you will be prompted to reboot the server.

Configuring Windows Terminal Services Security

While this paper does not provide specific measures for physically securing Windows servers, there are several basic security principles that should be understood and applied to the remote management environment. The most secure foundation is to harden the Windows 2000 Server operating system. The steps required to harden the server is to begin with a clean installation of Windows 2000 Server, install the latest patches, and then configure the

[10] U.S. National Security Agency. Guide to Securing Microsoft Windows 2000 Terminal Services, [by Vincent J. DiMaria, et al] Ft. Meade, MD, July 2, 2001, page 9 URL: http://www.nsa.gov/snac/win2k/guides/w2k-19.pdf

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

server's security settings. Then install Windows Terminal Services and apply the latest WTS service packs.[11]

WTS is configured using the Microsoft Management Console (MMC) snap-in: Terminal Services Configuration. When Terminal Services is installed, two objects are made available for configuration: Connections and Server Settings. From a security standpoint, the Terminal Services Configuration tool is used to make the majority of the security settings for WTS. The Server Settings is used to govern the behavior of the server when WTS sessions are established. The Connections setting is used to manage how each individual session or connection established.

Server Settings Configuration

As stated earlier, WTS Remote Administration mode should be enabled on all remote servers that require the functionality. Once the software has been installed, there are few basic settings that should be configured. These options can be starting Terminal Services Configuration program, found under Start, Programs, and Administrative Tools. The Server Settings object is used configure the basic functionality of WTS. The Server Settings options should be configured using the attributes listed in Figure 6.[12]

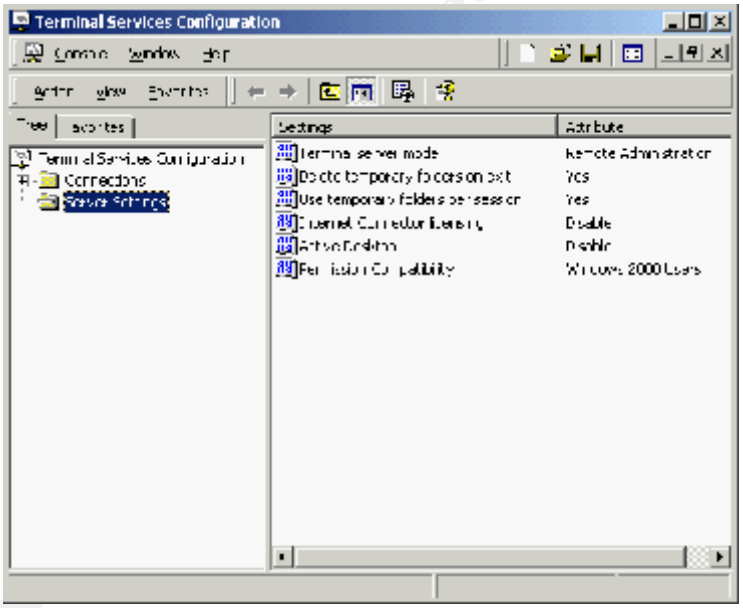

Figure 6
Terminal Services Configuration – Server Settings

I would like to note two attributes of the Server Settings object in Figure 6 above. First, the attribute for Terminal server mode can be either Remote Administration or Application. However, this particular attribute cannot be modified through the Terminal Services Configuration tool. The Terminal server mode can only be set through the Terminal Services installation process described previously. Secondly, the Active Desktop attribute should be

[11] Mackey, David, "Securing Windows 2000 Terminal Services", page 1. URL: http://www.microsoft.com/technet/prodtechnol/win2kts/maintain/optimize/secw2kts.asp?frame=true
[12] DiMaria, et al, page 29.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

set to disabled. This setting, if enabled, allows client sessions to use Active Desktop, which can unknowingly execute scripts in Web pages. Microsoft recommends that the Active Desktop setting should be disabled.[13]

Connections Configuration

The Connections object in Terminal Services Configuration provides several options for configuring connection, session, and user settings. The Connections object is used to configure the RDP-TCP connection object, double clicking this object displays the RDP-TCP Properties configuration screen. From the RDP-TCP Properties, there are several tabs and configuration settings used to specify the security settings. Figure 7 below shows a sample screen of the RDP-TCP Properties dialog box.

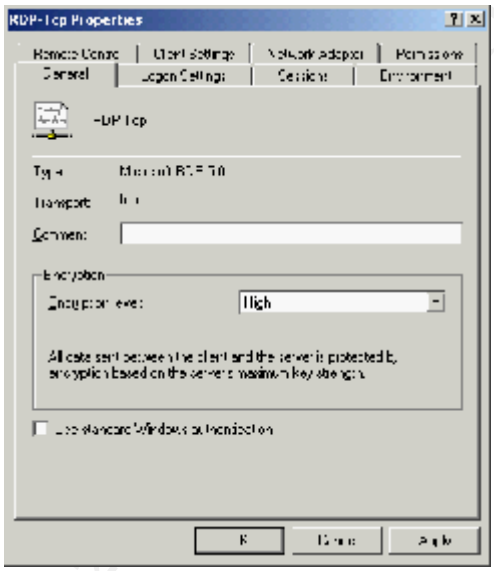

Figure 7
RDP-TCP Properties Dialog Box - General Tab

I have created a table of recommended settings the RDP-TCP Properties for both Remote Control mode and Application mode configurations. The table is located in Appendix A. However, I will provide a brief synopsis of a few of the tab settings below and then later in this document.

General Tab. As I stated earlier, Microsoft utilizes RDP as the communication protocol for WTS. Fortunately, RDP supports the RSA RC4 encryption algorithm so secure communications channels can be established. The General tab is used to select the data communication encryption level. There are three levels of encryption available to choose from: Low, Medium, and High. The Low setting utilizes 56-bit key length encryption and specifies encryption for transmissions from the client to the server and this level is useful for protecting passwords from being sniffed. The Medium setting again utilizes 56-bit key length encryption, but it specifies encryption for both send and receive directions. The High setting uses the stronger 128-bit key length encryption for data sent in both directions and is the

[13] Mackey, David, page 4.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

recommended setting. The High encryption is only available in the US and Canada, so you may have to use the Medium setting in some circumstances.[14]

Remote Control Tab. Remote control permits another user to control the session. Users who are granted the remote control permissions could take control of another user's WTS session. This is potential security risk as it allows one user to perform actions under another user's account. Remote control therefore is not recommended. To disable remote control, select "Do not allow remote control" under the Remote Control tab of the RDP-TCP connection properties dialog box.[15]

Logon Settings Tab. When establishing WTS sessions it is important to maintain standard logon procedures. Although WTS can be configured to bypass normal user authentication and use a standardized account name and password for authentication, it is recommended that each session is established using an account name and password. The Logon Settings tab of the RDP-TCP connection properties dialog box, should have the "Use client provided logon information" and "Always prompt for password" options selected. This will force each session to be established with an account name and password as well as it will prevent the usage of embedded passwords.[16]

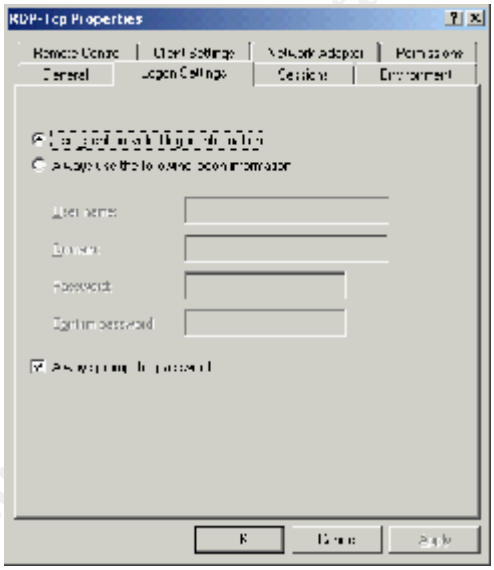

Figure 8
RDP-TCP Properties Dialog Box - Logon Settings Tab

Environment Tab. The Environment tab of the RDP-TCP connection properties dialog box is used to control which single application users will run when they log on to WTS. Its function is to support Application server mode, by forcing one application to start when a session is established. While this feature might be useful in some environments, such as in the use of a kiosk type application, it is not useful for remote administration. Remote administration generally requires analysts to access more than one application. Therefore, the

[14] Smith, Randy Franklin. "Terminal Services, Part 2", Windows and .Net Magazine Network, February 15, 2001, page 2. URL: http://www.windowsitsecurity.com/Articles/Index.cfm?ArticleID=19936
[15] DiMaria, et al, page 21.
[16] DiMaria, et al, page 14.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

option “Override settings from user profile and Client Connection Manager wizard” should not be selected.

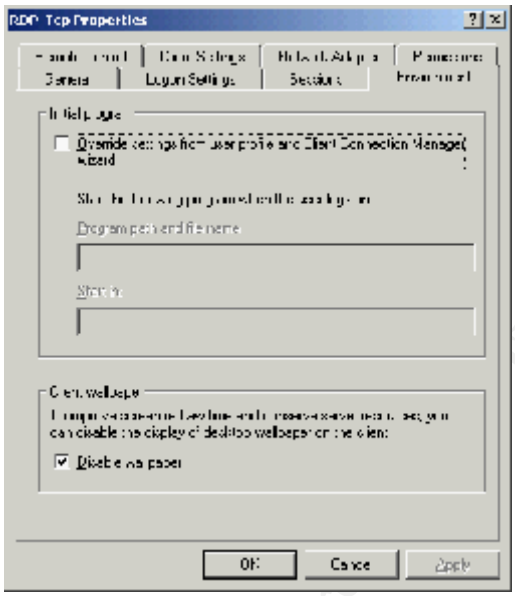

Figure 9
RDP-TCP Properties Dialog Box - Environment Tab

Configuring Terminal Services Permissions

The Windows Terminal Services environment can be configured with security policies to help reduce the threat of comprise of intentional intrusion. WTS permissions need to be configured differently for both the remote administration mode and application server mode. With a well-protected WTS server environment, administrative tasks can be performed from within terminal sessions. However, care must be taken with administrative rights for the WTS server itself[17].

The permissions settings are made under the Permissions tab for RCP-TCP Properties of the Terminal Server Configuration menu. The Add and Remove buttons are used to make certain that only the appropriate users and groups have been granted access. The permissions settings for each mode of WTS operation require different levels of permissions. I will explain how to configure the permissions settings below.

Remote Administration Mode Permissions. The permission for WTS in Remote Administration mode should be configured so general users have no access and only the System account and Administrators group are granted access. Below is a sample screen on how to configure an administrators group. The screen is selected from the Advanced button the Permissions tab.

[17] “Configure Computers for Secure Remote Administration”, CERT® Coordination Center, URL: http://www.cert.org/security-improvement/practices/p073.html

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

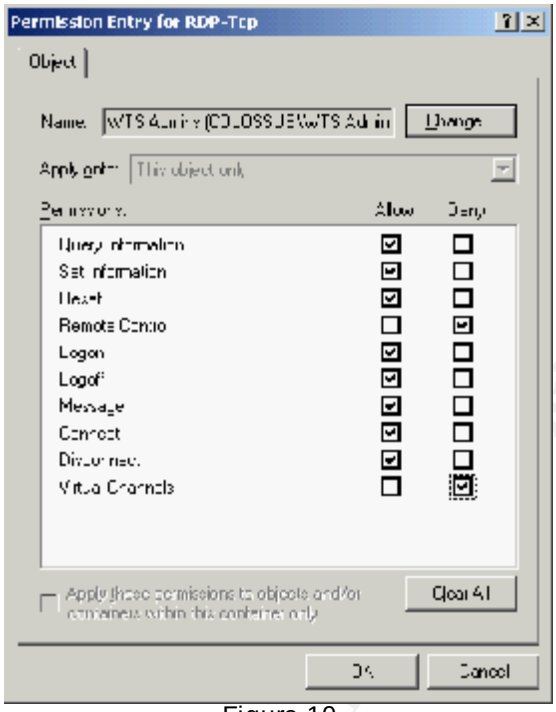

Figure 10
RDP-TCP Properties Dialog Box - Permissions Tab
Administrator Permissions Entry

In the sample screen above, the WTS Admins group is actually a group that I created on my lab network to hold all administrator accounts that should be granted the ability to perform remote administrative tasks. The administrative group has been granted all rights except Remote Control and Virtual Channels. Remote control would allow an administrator to take control of another user's session. This poses a security risk as it allows one user to take actions using another user's account.[18] Virtual Channels are used to allow users the ability to map drives and user local printers during a Terminal Services session. It is recommended that the Virtual Channels setting be set to deny for both users and administrators.[19]

Application Server Mode Permissions. Application server mode permissions are slightly different from Remote Administration mode permissions in that Application server mode requires the setting of permissions for a users group. Although administrators responsible for supporting the remote servers will require access to the application server, they should only be granted permissions as general users. Based on my proposed WTS environment, the idea is to use the WTS Application server only as a portal to support remote servers. The analyst should establish WTS server sessions with administrative accounts, but the analyst's accounts have no administrative rights on the WTS Application server, they only have normal user rights. Therefore, the analyst should not be able perform administrative task on the WTS Application server itself. Figure 11 below shows settings that standard users should have on the WTS Application server. For the true administrators of the WTS Applications server, use the settings shown in Figure 10.

[18] DiMaria, et al, page 28.
[19] Mackey, David, page 10.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

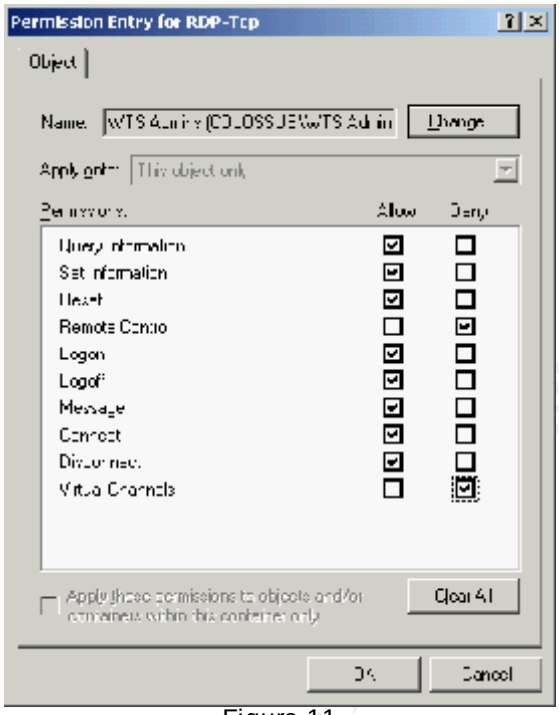

Figure 11
RDP-TCP Properties Dialog Box - Permissions Tab
User Permissions Entry

Appendix B lists all the RDP-TCP permission settings and their descriptions along with recommended settings.

Comments on Account Privileges

System administrators should perform their duties using two accounts. One account should be established with normal user rights and another distinct account should be for administrative purposes. The user account should be equivalent to that of the typical user account with access to private data files and normal file and print services. The administrative account should be assigned administrative privileges as necessary.

User Account. I recommend that the user account use the standard naming conventions and have the equivalent access rights as all typical users in the organization. It has been my experience that it is important for administrators to understand the systems from the user's perspective. Accessing system resources from this level will make the analyst aware of the problems and issues that users face on a regular basis. The user account should have no administrative rights.

Administrator Account. It is obvious that system administrators and analysts need some privileged levels access in order to perform their fundamental job functions. Fortunately, Windows 2000 Active Directory and Windows NT Directory Services offer multiple levels of granularity for managing access control rights. All administrative functions should be managed using accounts designated for administrative purposes.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Account Naming Conventions

User accounts and accounts with administrative rights should be clearly labeled and managed appropriately. I suggest that part of the establishment of account naming conventions include standards for administrative accounts. The difference between users accounts and administrative accounts should be easily identifiable. For example, if the account naming convention calls for accounts to be created as a concatenation, i.e. the user's first initial, middle initial, and first 6 characters of their last name, then the following user account my result for a user named John Q. Public.

Name: John Q. Public – resulting account name: `jqpublic`

Additionally, a convenient way to identify administrative accounts could be to add a prefix to indicate administrative rights, such as the prefix `adm_`. Therefore using the same user name as the example above, the users account with administrative rights could be as follows:

Name: John Q. Public – resulting account name: `adm_jqpublic`

Once the accounts have been established than it is only a matter of assigning the appropriate user rights. As I stated earlier, the user account should have the access privileges equivalent to the typical user in your organization. Typical access rights might include access to publicly shared folders, secured access to personal data, access to group data and normal print queue access. The administrative accounts can be managed on a per server level, by domain, or by organizational unit. The idea behind controlling the administrative account is to utilize a policy of least privilege and to grant only the rights necessary for the administrator to effectively carry out their responsibility.[20]

Using the Accounts Together

While there are several alternatives available, one suggestion that I would like to detail is how to utilize the user and administrative accounts.

Using Windows 2000 Professional workstation, for example, the analyst should log in to the domain using the user account (`jqpublic`). All desktop tools such Microsoft Office and Microsoft Outlook should be run using the Access Control Levels (ACL) of a normal user. For all administrative tasks, WTS sessions should be established. The analyst should log on to the WTS server using the administrative account (`adm_jqpublic`).

I do not recommend the use of the RunAs[21] feature for executing commands and programs. Instead, the analyst should simply log on with the account they intend to use with the program. While logging in and establishing a user session may seem like a cumbersome method, it provides an extra layer of security to the networked environment.

Figure 12 below diagrams the suggested authentication method.

---

[20] Zimmerman, Scott, C. "Secure Infrastructure Design", CERT Coordination Center, URL: http://www.cert.org/archive/pdf/Secure_Infrastructure_Design.pdf

[21] Denial of Service Vulnerability in Windows 2000 RunAs Service, URL: http://www.securiteam.com/windowsntfocus/6K00F1535I.html

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

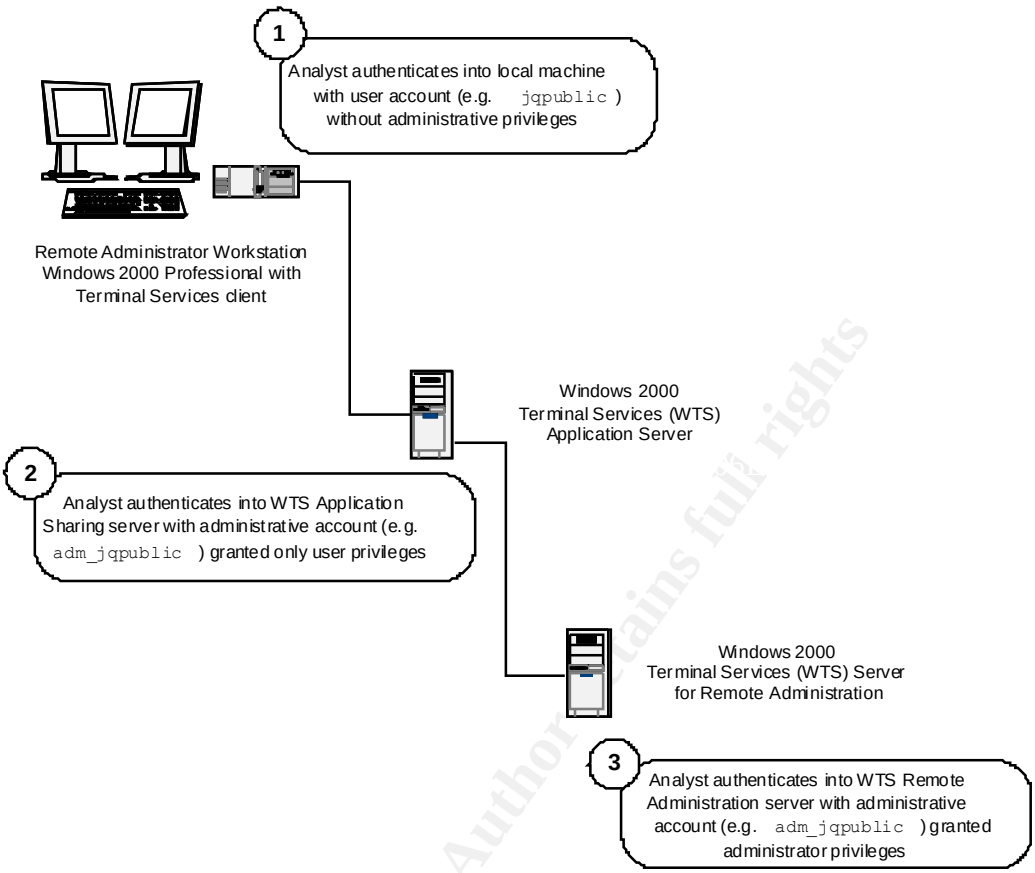

Figure 12
Sample Authentication Scheme Using Windows Terminal Services

Installing the Terminal Services Client

When Terminal Services is installed on Windows 2000 Server, the Terminal Services Client Creator tool is automatically placed into the Administrative Tools folder. The Terminal Services Client Creator tool is used to create a floppy disk installation. However, the Terminal Services Client setup files, which are used to create the floppies, can be found in %systemroot%\system32\clients\tsclient. The setup.exe can be run across the network from the "net\win32" subfolder.

Floppy Disk Installation. Terminal Services Client can be installed using two floppy disks, which are created using the Terminal Services Client Creator tool. Once the two floppy disks are created, run setup.exe on disk 1. You will be prompted for the installation destination directory; in addition, you will be prompted if you wish to install the same settings for all users or just the current user.

Using Terminal Services

The Terminal Services Client (TSC) application is launched from the Terminal Services Client folder in the Programs menu. Once the application has started you can select the Terminal Services server that you wish to connect to. You can also specify the screen size area for the terminal session. Once the appropriate server is selected, press the Connect button. The figure below shows a Terminal Services Client dialog box with the sever

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

COLOSSUS selected and screen area of 1024x768 selected. The Available Servers window indicates all known servers with WTS installed.

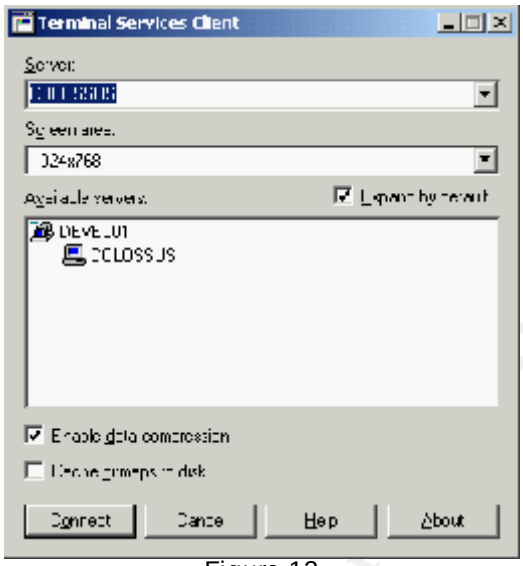

Figure 13
Terminal Services Client – Server Selection Tool

After pressing the Connect button, a Terminal Services Client window will open with a logon dialog box. The session window can be resized to fit your screen. After logging on to the remote server, you will have access to the remote computer as if you were working at the computer's console. The figures below show the log on screen to the server COLOSSUS. Note that the screen can be resized and has scroll bars to help position the window view.

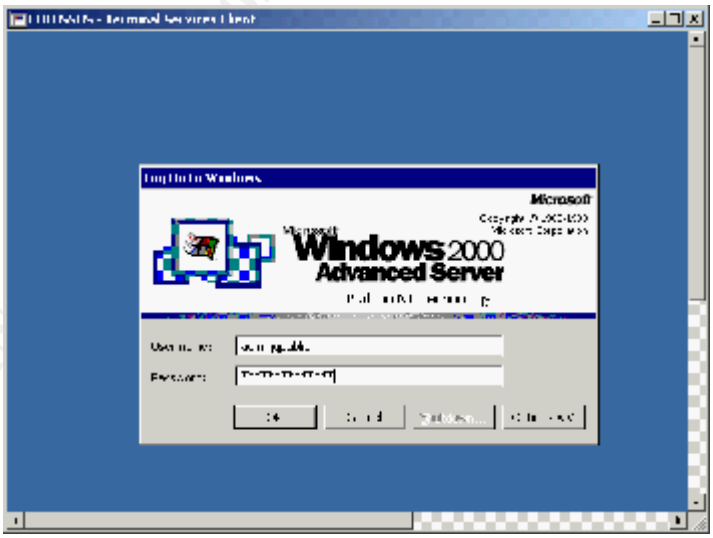

Figure 14
Terminal Services Client – Server Log On Windows

After user authentication, the Terminal Services Client will display the remote server console. The Terminal Services Client screen can be manipulated just as any other application window. The screen can be minimized, resized, or relocated on the desktop. Below is a sample screen shot of a desktop that includes a Terminal Services Client session.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

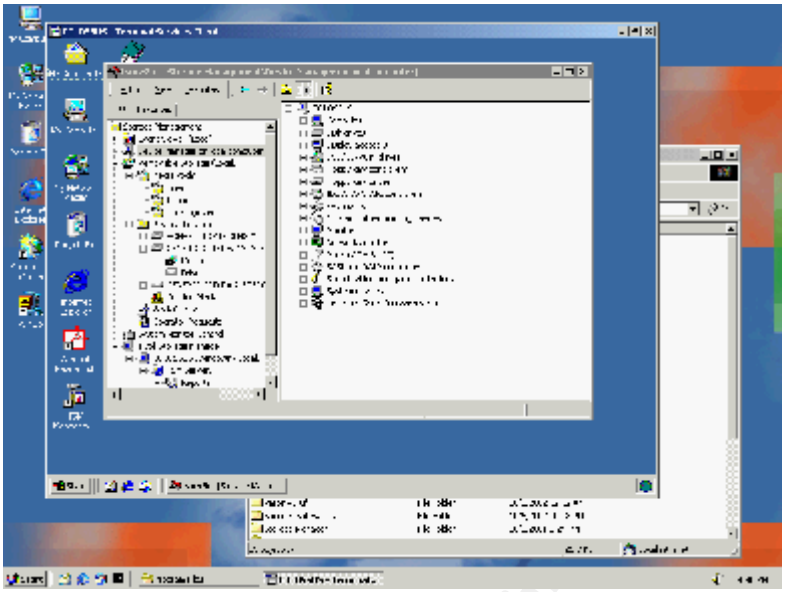

Figure 15
Sample Terminal Services Window Display

Disconnecting or Terminating the Terminal Session

There are essentially three ways to stop a Terminal Services Client session. The first method is performed within the client session. The standard menu based log off method from within the WTS session, using Start, Shutdown, Log Off user dialog box, cleanly ends a client session (See Figure 16 below). Pressing Ctrl-Alt-Del will only affect the workstation or console desktop not the Terminal Services Client session.

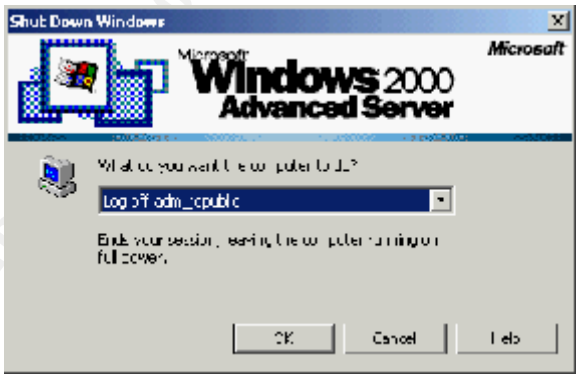

Figure 16
Sample End of Session Dialog Box

The second method for end a session is by closing the TSC window without logging off the server. In actuality, this method will disconnect the session. Reestablishing a TSC session to the same server may restart the disconnected session, depending on the Terminal Serves configuration settings. The session disconnection is a useful feature as it allows the screen settings to be maintained while the session is temporarily closed or locked. When the session is disconnected, the programs or processes the user has running will continue to run. The processes will continue to run in the state they were left prior to disconnecting the session.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

The screen shot below is the message when the TSC user will see when exiting the TSC without logging off.

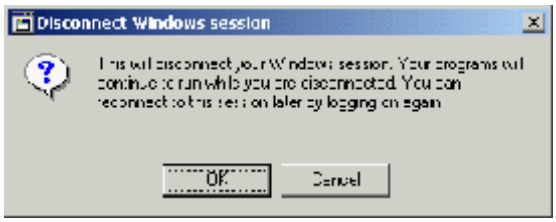

Figure 17
Disconnecting a Terminal Services Client Session

After a TSC session has been disconnected, it may be reestablished by connecting back to the server through the Terminal Service Client server selection dialog box. The user will be prompted to authenticate into the server. When authentication is complete, the session is returned to the state in which it was left.

The settings for session termination and disconnection are set in the Sessions tab of the RCP-TCP Properties of the Terminal Server Configuration menu (see Figure 18). The NSA recommends the following session time out settings: 1 day for Remote Administration mode server and 3 hours for Application server mode servers.

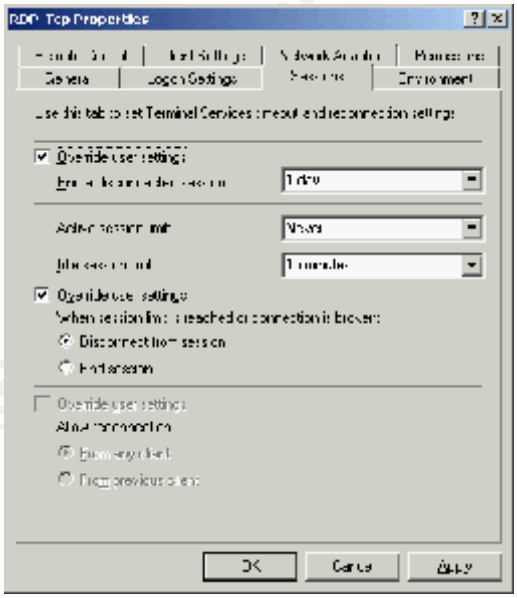

Figure 18
RDP-TCP Properties Dialog Box (Sessions Tab)

Lastly, the terminal session can be disconnected from the Terminal Services Manager tool on the Terminal Services computer. This feature will only disconnect the session and not actually terminate the session.

Care must be taken when disconnecting from WTS sessions. The Remote Administration mode server only allows two concurrent sessions. If two administrators disconnect, then no one else will be able to manager the remote server. This would force the analyst to connect to the server via an alternate method and stop and restart the Terminal Services process

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

(TERMSRV.EXE). For example, this process "re-cycle" can be managed through the use of the MMC to control services remotely from the dedicated WTS Application server. Data may be lost from an open session using this type of brute force method.[22]

One additional note regarding the use of WTS Remote Administration mode is that it is not always necessary to establish a remote session. A number of administrative tasks can be performed from the WTS Application server against a remote server; therefore, there may be no need to log on to the WTS Remote Administration server and tie up a user session.

Workstation Standards

The system administrators' or analyst's tool set should include a fairly standard set of applications. Email, word-processing, and spreadsheet applications should be available. The list of applications should include at least the following set of applications or their equivalent:

Microsoft Outlook

Microsoft Word

Microsoft Excel

WinZip

Adobe Acrobat Reader

Microsoft Internet Explorer

Power users should consider the full suite of applications available with Microsoft Office.

While it is not necessary for system analysts to use Microsoft Word or Microsoft Excel, or equivalent applications such as Word Perfect and Quattro Pro, I highly recommend that some sort of text processing tools be available to the remote administrator. In addition, advanced remote administration requires documentation. The recommended tools are to be used to create effective documents and they can aid in the analysis of problematic issue. Remote administration can certainly be accomplished without "Office" type applications, but the applications facilitate the use of de facto standards for documentation.

Internet Browser. An Internet browser is an important tool for system administration. Several tools can be utilized with a browser. Documentation is probably one of the most valuable tools that a system analyst can utilize. System documentation, scripts, escalation procedures, and escalation contacts should be available for reference. Well-organized web based documentation and reference material can enable the system analyst to quickly troubleshoot problems and properly escalate severe issues.

Workstation Operating System Recommendation

Both Windows 2000 Professional and Windows XP Professional support multiple monitors. I recommend that system analysts utilize two monitors while performing their duties, since the use of multiple monitors has many benefits for the system analyst. My recommendation

[22] Boswell, page 6.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

would be to have reference tools available on one monitor and use the other monitor for performing maintenance and troubleshooting tasks. For example, one monitor can be displaying a web browser focused on a procedure for troubleshooting an application issue, while the other monitor displays a WTS session. The analyst can perform the tasks documented in web browser using the WTS session: however, this is a simple scenario. Advanced analysts utilize several on-line tools and two monitors can facilitate easy access to those tools and the workstation desktop can be configured for their ease of use.

Ultimately, in my experience, it is preferable to have reference documentation available while I am working to resolve an issue. Documentation may not always be available; that is why the tools in the Microsoft Office suite are recommended, so that the advanced analyst can create required documentation. Well-organized teams should have scripted procedures and it is to the system analyst's benefit to be able to read documentation while performing their duties.

Documentation and Procedural Scripts

System documentation, procedures, and scripts are important tools that should be available to the analyst as standard tools. The tools should be used as a consistent framework for providing high levels of quality service. Documentation can be of various degrees of detail, but what is important is that the documented tasks can be carried out in a standard and consistent fashion. While I am not recommending that all tasks and procedures be scripted, I am trying to say that certain functions should be scripted as a matter of requirement and accountability. For example, based on the severity level of an alert, it may be necessary to escalate an issue to management at certain time intervals, such as every half hour. Having scripts or web page links, which can direct the analyst along a proper path of escalation, may help to reduce the severity of an outage.

Beyond escalation procedures, standard tasks based on event errors should be documented and available to the analyst. If an SQL environment is supported, then perhaps a set of SQL queries can be detailed and then the analyst can chose an appropriate SQL query based on the SQL event. The same can be said about errors generated by any third party hardware or software environment. Backup and restore scripts should be documented for the analyst to follow based on event errors.

Terminal Server Desktop Tools

The WTS Application server desktop has several built in tools for remote server support. The standard desktop with a few additional tools can provide the analyst with remote support capabilities. Outside of application specific support tools, I would recommend that the Microsoft Windows 2000 Professional Resource Kit[23] utilities should be added to the WTS Application server configuration or that the Windows 2000 Resource Kit utilities are at least available to the analyst through a shared resource drive mapping. Application specific support tools may include SQL Server Enterprise Manager and SQL Server Query Analyzer if one is supporting SQL environments.

The command line interface is an indispensable tool to the analyst. I recommend that a short

[23] Microsoft Windows 2000 Professional Resource Kit, Microsoft Press; March 15, 2000.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

cut should be added to the desktop to reference the `cmd.exe`. The command prompt interface can be used to execute scripts, batch files and standard programs. I recommend that analysts configure the command prompt properties to allow for 'QuickEdit Mode' and 'Insert Mode', the window size height setting should be increased to at least 50 lines, the screen buffer size height should be set to 400, and that an appropriate screen color should be selected.

Depending on the environment you are supporting you should probably have the Microsoft Management Console (MMC) available on your desktop. At the very least, the analyst should be able to access the Manage Computer option available through the "My Computer" icon. Through Computer Manager, connections to remote servers can be established using the 'Connect to another computer…' option from within Computer Management icon. The Computer Manager tool allows access to logged events, user and group management, and service management among others.

The MMC can provide the analyst access to several administrative tools, including Active Directory. However, it is not always the case where an analyst requires access to the Active Directory objects. Depending on the organization, it is quite possible to have one team manage Active Directory and other teams manage the resource servers. In this scenario it is plausible for the analyst to never need the MMC tool and all remote administration tasks can be managed with the Computer Management tool.

Unfortunately, not all tasks can be managed using graphical interface tools. Sometimes the analyst is required to use command line tools. Below is a list of essential command line tools for the analyst:

```
ping.exe

srvinfo.exe

uptime.exe

rkill.exe

net use \\servername\ipc$

net helpmsg
```

This list is certainly not intended to be exhaustive and at first glance they may appear to be high-level tools, yet these commands are very powerful tools that analysts should utilize for remote administration. While there are several hundred tools available in the Windows 2000 Professional Resource Kit that can benefit the analyst, I have tried to compile a short list of tools that I would consider to be essential command line programs for remote administration. I would also like to note that for some tools there is little documentation that can be found. For example, if one was to query Microsoft's TechNet for 'rkill', the resulting data set lists only three references. Appendix C to this text provides further details to the command line tools listed above.

Reducing email Virus Threats

Because of the threat of worm viruses and vulnerabilities in Microsoft Outlook, it is important to reduce the threat of email based virus acts by adding layers of security to your network environment. Firewalls and packet filtering gateways, virus detection and

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

eradication software, network and host based intrusion detection systems can all help to establish layers of security. Additionally, account security and user account management also plays a key role in the maintenance of secure computing environments.

To reduce the threat of viruses that spread through email or application macros, log on security can be utilized. The analyst should access tools to perform routine tasks such as reviewing email or working on projects using a standard desktop tool set. Windows XP allows for multiple monitors, one monitor can be used to establish a standard desktop, and a second monitor can be used for working with WTS server sessions. The WTS server sessions should be accessed using the administrative account. The administrative account should then not access any email system. While the analyst can perform administrative functions, the threat of email based viruses, worms, or Trojan horses can be eliminated from affecting multiple servers for which the analyst has established administrative connections.

Certainly, it almost goes without saying, that at the server level, there are several methods for hardening the security of a server. For instance, unnecessary shares should be removed, and file and directory permissions have appropriate ACLs, anti-virus software is installed and updated, and the latest service packs and security fixes are installed.

Hardening Resources

The Windows server environment provides a number of security challenges to system administrators. Hardening is one of the security challenges administrators face. Security hardened servers and workstations environments are extremely important to the analyst. It is not the intent of this report to outline all security issues nor is there intent to enumerate the measures for hardening the Windows server environment. There are several papers that are available that discuss hardening for both the Windows NT Server and the Windows 2000 Server environments. Keep in mind; hardening servers and networks is a task that should integrate a 'Defense in Depth'[24] strategy. Firewalls, proxy servers, and network intrusion detection software should also be included as part of the security strategy.

The following links provide detailed explanations for hardening server resources:

- Micorosoft Technet - Securing Windows 2000 Terminal Services[25]
- Guide to Securing Microsoft Windows 2000 Terminal Services - NSA (pdf)[26]
- Guide to Securing Microsoft Windows 2000 File and Disk Resources[27]
- Guide to Securing Microsoft Windows 2000 Active Directory[28]
- Guide to Securing Microsoft Windows NT Networks[29]

---

[24] Brooke, Paul. "Building an In-Depth Defense", Network Computing, July 9, 2001, 1-2. URL: http://www.networkcomputing.com/1214/1214ws1.html

[25] U.S. National Security Agency. Guide to Securing Microsoft Windows 2000 Terminal Services, [by Vincent J. DiMaria, et al] Ft. Meade, MD, July 2, 2001.

[26] U.S. National Security Agency. Guide to Securing Microsoft Windows 2000 Terminal Services, [by Vincent J. DiMaria, et al] Ft. Meade, MD, July 2, 2001.

[27] U.S. National Security Agency. Guide to Securing Microsoft Windows 2000 File and Disk Resources, [McGovern, Owen R., EDS, DISA, and Haney, Julie M., NSA] Ft. Meade, MD, April 19, 2001.

[28] U.S. National Security Agency. Guide to Securing Microsoft Windows 2000 Active Directory, [Sanderson, Mark J. and Rice, David C.] Ft. Meade, MD, December, 2000.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Additionally, consider IPSec for encrypting IP datagrams. IPSec provides authenticated end-to-end security between hosts. IPSec is supported by Windows 2000 servers and can be used to establish secure connections between WTS server sessions and Windows 2000 servers.

Conclusion

I have tried to demonstrate a secure way to utilize Windows Terminal Services for performing remote server administration. I have documented a secure WTS environment and shown how to install and configure the servers in the environment. The configuration recommendations are designed for secure computing using a principle of least privilege and defense in depth strategy. The process for setting up the environment is not a difficult task, but it does take some forethought and testing. I would strongly recommend that you develop a lab environment to test your own system utilizing WTS, prior to rolling out WTS in a production environment.

Secure remote server administration is more than just having secure servers and tools. Remote server administration requires using the servers and tools properly. I have described how to design a server environment for managing remote servers. I have described how to utilize separate user and administrative accounts for remote server support. I have also described standards for presenting documentation to the administrator. With the use of clear documentation, carefully assigned administrative privileges, and properly configured tools such as Windows Terminal Services, secure remote server administration can be accomplished.

I have also provided a list of what I consider essential tools for remote administration. I have intentionally tried to keep the list of essential tools short. Remote server administration can be a daunting task and analysts need tools that they know will work. The tools I have described allow the analysts to gain a quick understanding of the systems they support. Once an understanding of the problem is made, the analyst can then spend his or her time performing root cause analysis, resolving chronic issues, and escalating issues that affect mission critical business processes.

There are certainly plenty of exploits for circumventing security measures, but by implementing a defense in depth security structure, secure remote server administration is possible. But implementing the tools I have described is not a panacea for secure remote administration. The tools, processes, and methods for secure remote administration need to be carefully scrutinized for new exploits. It is imperative for the system analyst to be aware of security issues and measures for defending their environment. Beyond the tools that I have described, continuing education and investment in resources is required for secure environments.

---

[29] U.S. National Security Agency. Guide to Securing Microsoft Windows NT Networks, [Bartock, Paul F. et al.] Ft. Meade, MD, September 18, 2001.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Appendix A

RDP-TCP Settings Recommendations[30]

General Tab	**Remote Administration Mode**	**Application Server Mode**
Encryption Level	High	High
Use standard Windows authentication	Not Selected	Not Selected
Logon Settings Tab		
Use client-provided logon information	Selected	Selected
Always use the following logon information	Not Selected	Not Selected
Always prompt for password	Selected	Selected
Sessions Tab		
Override user settings for session limits	Selected	Selected
End a disconnected session	1 day	3 hours
Active session limit	Never	Never
Idle session limit	15 minutes	15 minutes
Override user settings for action When session limit is reached	Selected	Selected
Disconnect from session	Selected	Selected
End session	Not Selected	Not Selected
Environment Tab		
Override settings from user profile and Client Connection Manager wizard	Not Selected	Selected (if applicable)
Program path and file name	Blank	Full path to application to users for Intranet Application Sharing
Start in	Set as applicable	Set as applicable
Disable wallpaper	Environment Specific	Environment Specific

[30] DiMaria, et al, pages 13 - 23.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Remote Control Tab	**Remote Administration Mode**	**Application Server Mode**
Use remote control with default user settings	Not Selected	Not Selected
Do not allow remote control	Selected	Selected
Use remote control with the following settings:	Not Selected	Not Selected
Client Settings Tab		
Use connection settings from user settings	Not Selected	Not Selected
Connect client printers at logon	Not Selected	Not Selected
Default to main client printer	Selected	Selected
Disable Windows printer mapping	Selected	Selected
Disable LPT port mapping	Selected	Selected
Disable COM port mapping	Selected	Selected
Disable Clipboard mapping	Selected	Selected

© SANS Institute 2003, Author retains full rights

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Appendix B

WTS Permissions Recommendations[31]

Permission		Remote Administration Mode	Application Server Mode
Query Information	Allows a user to access information about the session, such as IP addresses and connection states	Allow	Allow
Set Information	Allows the user change the connection settings	Allow	Deny
Reset	Allows the administrator to abruptly close a user's session with without warning. Can result in lost data.	Allow	Deny
Remote Control	Required permission for the Remote Control feature	Deny	Deny
Logon	Minimum permission required to establish a WTS session	Allow	Allow
Logoff	Lets one user log off another user without warning	Allow	Deny
Message	Allows a user to send a message to another log on Terminal Services user	Allow	Deny
Connect	Allows a user a connection to a disconnected session	Allow	Deny
Disconnect	Allows a user to disconnect another user without warning	Allow	Deny
Virtual Channels	Allows a user to access a virtual channel, such as mapping a drive or using a local COM port during a Terminal Services session	Deny	Deny

[31] DiMaria, et al, pages 24 - 28

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Appendix C

Analyst Tools

The following description of tools and their use is intended to be an essential list, not an all-inclusive list. There are certainly plenty of tools to choose from and each tool may have several available options, but the purpose of this document is to point out essential tools. It is also my intent to simplify normal administrative tasks by limiting the tools set. In this way, the system analyst can spend more time resolving problems, communicating findings to appropriate team members and management, and resolving chronic issues.

Below is a listing of essential remote analyst tools. Each tool includes a brief description of its purpose and usage, as well as sample output.

`ping`

The general use of `ping` command is to show connectivity to a server's IP address or server name.

```
H:\bat>ping sqlserver7

Pinging sqlserver7 [10.19.252.46] with 32 bytes of data:

Reply from 10.19.252.46: bytes=32 time=20ms TTL=122
Reply from 10.19.252.46: bytes=32 time=20ms TTL=122
Reply from 10.19.252.46: bytes=32 time=20ms TTL=122
Reply from 10.19.252.46: bytes=32 time=20ms TTL=122
```

There are two options that are generally utilized with the `ping` command: `-a` and `-t`.

```
ping -a 10.19.252.46
```

`-a` is used for reverse name lookup, this command can be used to identify a server name from it's IP address.

```
ping -t sqlserver1
```

`-t` is used to send the ping until the command is stopped with a Control-C. This option is useful to see if downed server has come back to life or to monitor server connectivity during a server reboot.

`srvinfo.exe`

The `srvinfo.exe` command is a useful command for displaying configuration details regarding Windows NT and Windows 2000 computers. The resulting output from the command provides the analyst with several useful details. The details include the server name, domain name, primary domain controller, build type, and all patches that have subsequently installed. In addition, IP address information, drive space data, information on all services, and server uptime is also detailed.

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

```
H:\>srvinfo \\SQLSERVER1

Server Name: SQLSERVER1
Security: Users
NT Type:  NT Advanced Server -
Version: 5.0
Build: 2195, Service Pack 2
Current Type: Uniprocessor Free
Product Name: Microsoft Windows 2000
Registered Owner: SampleSQL
Registered Organization: Sample Company Name
ProductID: 49672-OEM-0050753-06708
Original Install Date: Tue Apr 02 11:17:10 2002
Domain: SQLDOMAIN01
PDC: \\SQLRD001
IP Address: 192.168.25.104
CPU[0]: x86 Family 15 Model 1 Stepping 2: 1794 MHz
Hotfixes:
   [SP2SRP1]:
   [Q314147]:
   [Q313829]:
   [Q313450]:
   [Q311967]:
   [Q300845]:
   [Q295688]:
   [Q147222]:
Drive:  [FileSys]  [ Size ]  [ Free ]  [ Used ]
  C$      NTFS        8192      4180      4012
  D$      NTFS       38171     34180      3991
Services:
   [Stopped]    Alerter
   [Running]    ISS Agent Service
   [Stopped]    Application Management
   [Running]    AVSync Manager
   [Running]    BlackICE
   [Running]    Computer Browser
   [Stopped]    Indexing Service
   [Stopped]    ClipBook
   [Running]    DHCP Client
   [Stopped]    Logical Disk Manager Administrative Service
   [Running]    Logical Disk Manager
      (Multiple Lines Deleted)
   [Running]    DNS Client
   [Running]    Event Log
   [Stopped]    QoS RSVP
   [Stopped]    Uninterruptible Power Supply
   [Stopped]    Utility Manager
   [Running]    Windows Time
   [Running]    Windows Management Instrumentation
   [Running]    WMDM PMSP Service
   [Running]    Windows Management Instrumentation Driver Extensions
Network Card [0]:
System Up Time: 0 Days, 2 Hr, 14 Min, 17 Sec
```


© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

The `srvinfo.exe` command can also be used with the switch `-ns`. When specified with the `-ns` switch `srvinfo.exe` will not display any of the details regarding services (think: `-ns` = no services).

```
H:\>srvinfo -ns \\SQLSERVER1

Server Name: SQLSERVER1
Security: Users
NT Type:  NT Advanced Server -
Version: 5.0
Build: 2195, Service Pack 2
Current Type: Uniprocessor Free
Product Name: Microsoft Windows 2000
Registered Owner: SOC
Registered Organization: Intel
ProductID: 49672-OEM-0050753-06708
Original Install Date: Tue Apr 02 11:17:10 2002
Domain: SQLDOMAIN01
PDC: \\SQLRD001
IP Address: 192.168.25.104
CPU[0]: x86 Family 15 Model 1 Stepping 2: 1794 MHz
Hotfixes:
   [SP2SRP1]:
   [Q314147]:
   [Q313829]:
   [Q313450]:
   [Q311967]:
   [Q300845]:
   [Q295688]:
   [Q147222]:
Drive:  [FileSys]  [ Size ]  [ Free ]  [ Used ]
  C$      NTFS        8192      4180      4012
  D$      NTFS       38171     34180      3991
Network Card [0]:
System Up Time: 0 Days, 2 Hr, 14 Min, 17 Sec
```

Note: the `srvinfo.exe` command can also be use to identify server for which you have administrative rights. The tool can be used against any Windows servers, but when the analyst has not been granted administrative rights, then IP address information, disk drive capacity and uptime information is not displayed.

rkill

`rkill` provides the analyst with a method to show processes and their associated process id numbers (PID), and to remotely kill a process.

To show all processes and enumerate each associated PID you can run the following command:

```
rkill /view \\sqlserver1
```

Sometimes processes fail and cannot be stopped or restarted with a GUI tool such as Computer Manager or the Windows Task Manager. If this is case, the `rkill` command

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

can be used to terminate the process. The following is an example of an attempt to kill process 543 on server \\sqlserver1.

```
rkill /kill \\sqlserver1 543
```

Notes:

Some analysts find that using the `rkill` command to view processes is a little cumbersome since you have to specify the `/view` option. The resource Kit command `pulist.exe` can be used to enumerate processes running on a remote server. `pulist` and `rkill /view` provide essentially the same output.

A Windows GUI version of `rkill` is also available; it is called `wrkill.exe`. A GUI is nice to have, but if you consider writing a script to kill a process on a remote server, the command line tools are much easier to work with.

net helpmsg message#

Several command line tools provide error level indicators in their output. Most programs return and error code number, but not all error messages have descriptive information. The `net helpmsg message#` command can be used by the analyst to display a brief description of the error code. The following example shows how `net helpmsg message#` is used to determine the error message returned by the `srvinfo` command:

```
H:\>srvinfo \\sqlserver5
Network Error #53

H:\>net helpmsg 53
The network path was not found.
```

net use \\server\ipc$

Microsoft servers and clients establish communications channels using interprocess communication protocol (IPC). The hidden share point IPC$ is available on Windows servers and workstations for IPC between programs and networked computers. Establishing a network connection to a server's IPC$ share point can be a useful method of determining connectivity to a server. If the IPC$ share point is not available, it is very strong indication that the server has experienced a major failure.

Occasionally, Windows servers may crash, but the network interface card can remain active. When this occurs, the server may respond to a `ping`, which would initially indicate the server is on-line, however further probing with the `net use \\server\ipc$` command can return validation of server availability. In the example below, the server `sqlserver7` responds to a ping request, but the `net use \\server\ipc$` command can be used to verify connectivity.

```
H:\bat>ping sqlserver7

Pinging sqlserver7 [10.19.252.46] with 32 bytes of data:
```


© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

```
Reply from 10.19.252.46: bytes=32 time=20ms TTL=122
Reply from 10.19.252.46: bytes=32 time=20ms TTL=122
Reply from 10.19.252.46: bytes=32 time=20ms TTL=122
Reply from 10.19.252.46: bytes=32 time=20ms TTL=122

H:\bat>net use \\sqlserver7\ipc$
System error 240 has occurred.

The session was cancelled.
```

The 'System error 240 has occurred' error message indicates that the communication session with the server has been lost. In all likelihood, a process has run away and communication with the server is not possible, and a visit to the server console is probably required

uptime.exe

The `uptime` tool is a powerful command line program that has recently been upgraded. Microsoft provides plenty of documentation with the tool. Its basic usage provides a quick look at a server's hours of availability. There are several options available to the analyst to review times of reboots, length of previous uptimes, and length of time since the previous reboot. Below are examples of `uptime`'s basic usage, as well as the `-a` option for displaying application failure events:

```
H:\bat>uptime sqlserver1
\\sqlserver1 has been up for: 2 day(s), 23 hour(s), 34 minute(s), 22 second(s)

H:\bat>uptime -a sqlserver1
Uptime Report for: \\sqlserver1

OS: Windows NT 4.0 (Build 1381), Service Pack 6, Multiprocessor Free.
Time Zone: Pacific Daylight Time

System Events as of 6/21/02 9:37:28:

Date:      Time:        Event:               Comment:
---------- -----------  -------------------  ---------------------------------

           3/7/02    16:18:32  Boot
    3/7/02    16:38:05  Shutdown             Prior uptime:0d 0h:19m:33s
    3/7/02    16:41:36  Boot                 Prior downtime:0d 0h:3m:31s
    3/7/02    16:46:37  Abnormal Shutdown    Prior uptime:0d 0h:5m:1s
    3/7/02    16:55:10  Boot                 Prior downtime:0d 0h:8m:33s
    3/7/02    16:59:21  Bluescreen           STOP 0x0000001e
    3/8/02     9:08:22  Shutdown             Prior uptime:0d 16h:13m:12s
    3/8/02     9:11:00  Boot                 Prior downtime:0d 0h:2m:38s
    3/8/02     9:11:48  Shutdown             Prior uptime:0d 0h:0m:48s
    3/8/02     9:14:22  Boot                 Prior downtime:0d 0h:2m:34s
    3/8/02    10:47:40  Shutdown             Prior uptime:0d 1h:33m:18s
    3/8/02    10:51:58  Boot                 Prior downtime:0d 0h:4m:18s
    3/8/02    13:44:46  Shutdown             Prior uptime:0d 2h:52m:48s
    3/8/02    13:49:04  Boot                 Prior downtime:0d 0h:4m:18s
              (Multiple Lines Deleted)
   6/14/02     9:54:51  Boot                 Prior downtime:0d 9h:51m:41s
   6/14/02    19:54:51  Abnormal Shutdown    Prior uptime:0d 10h:0m:0s
   6/17/02     8:24:21  Boot                 Prior downtime:2d 12h:29m:30s
```


© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

```
   6/18/02     9:53:27  Shutdown             Prior uptime:1d 1h:29m:6s
   6/18/02     9:58:19  Boot                 Prior downtime:0d 0h:4m:52s
   6/18/02    10:00:59  Shutdown             Prior uptime:0d 0h:2m:40s
   6/18/02    10:03:35  Boot                 Prior downtime:0d 0h:2m:36s

Current System Uptime: 2 day(s), 23 hour(s), 34 minute(s), 27 second(s)


-------------------------------------------------------------------------------


Since 3/7/02:


           System Availability: 96.2629%
                  Total Uptime: 101d 17h:31m:53s
                Total Downtime: 3d 22h:47m:3s
                 Total Reboots: 38
     Mean Time Between Reboots: 2.78 days
             Total Bluescreens: 1
    Total Application Failures: 11
```

© SANS Institute 2003, Author retains full rights

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Bibliography

"Configure Computers for Secure Remote Administration", CERT® Coordination Center, URL: http://www.cert.org/security-improvement/practices/p073.html

U.S. National Security Agency. Guide to Securing Microsoft Windows 2000 Terminal Services, [by Vincent J. DiMaria, et al] Ft. Meade, MD, July 2, 2001. URL: http://www.nsa.gov/snac/win2k/guides/w2k-19.pdf

U.S. National Security Agency. Guide to Securing Microsoft Windows 2000 File and Disk Resources, [McGovern, Owen R., EDS, DISA, and Haney, Julie M., NSA] Ft. Meade, MD, April 19, 2001. URL: http://www.nsa.gov/snac/win2k/guides/w2k-8.pdf

U.S. National Security Agency. Guide to Securing Microsoft Windows 2000 Active Directory, [Sanderson, Mark J. and Rice, David C.] Ft. Meade, MD, December, 2000. URL: http://www.nsa.gov/snac/win2k/guides/w2k-5.pdf

U.S. National Security Agency. Guide to Securing Microsoft Windows NT Networks, [Bartock, Paul F. et al.] Ft. Meade, MD, September 18, 2001. URL: http://www.nsa.gov/snac/winnt/guides/wnt-1.pdf

Brooke, Paul. "Building an In-Depth Defense", Network Computing, July 9, 2001. URL: http://www.networkcomputing.com/1214/1214ws1.html

Benson, Christopher. "Best Practices: Security Planning", URL: http://www.microsoft.com/technet/security/bestprac/bpent/sec1/secplan.asp?frame=true

Boswell, Bill. "Remote Operator", Microsoft Certified Professional Magazine Online, June 2002. URL: http://mcpmag.com/Features/article.asp?EditorialsID=275

Harwood, Ted. Windows NT Terminal Server and Citrix MetaFrame. New Riders Publishing, 1998.

Jones, Don. "Must-Have Remote Administration Tools", Windows and .Net Magazine Network, May 2002. URL: http://www.winnetmag.com/Articles/Index.cfm?ArticleID=24536

Mathers, Todd W.. Windows NT/2000 Thin Client Solutions: Implementing Terminal Services and Citrix MetaFrame. New Riders Publishing, 2000.

Mackey, David, "Securing Windows 2000 Terminal Services", URL: http://www.microsoft.com/technet/prodtechnol/win2kts/maintain/optimize/secw2kts.asp?frame=true

Smith, Randy Franklin. "Terminal Services, Part 1", Windows and .Net Magazine Network, February 1, 2001. URL: http://www.windowsitsecurity.com/Articles/Index.cfm?ArticleID=19791

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Smith, Randy Franklin. "Terminal Services, Part 2", Windows and .Net Magazine Network, February 15, 2001. URL: http://www.windowsitsecurity.com/Articles/Index.cfm?ArticleID=19936

Tannenbaum, Todd. "15 Minutes to Remote Administration", Network Computing, March 6, 2000. URL: http://www.networkcomputing.com/1104/1104ws1.html

Zimmerman, Scott, C. "Secure Infrastructure Design", CERT Coordination Center, URL: http://www.cert.org/archive/pdf/Secure_Infrastructure_Design.pdf

Microsoft Windows 2000 Professional Resource Kit, Microsoft Press; March 15, 2000.

"Best Practices: Authentication for Administrative Authority", URL: http://www.microsoft.com/technet/security/bestprac/authent.asp?frame=true

Denial of Service Vulnerability in Windows 2000 RunAs Service, URL: http://www.securiteam.com/windowsntfocus/6K00F1535I.html

"Retiring Windows NT Server 4.0: Changes in Product Availability and Support", URL: http://microsoft.com/ntserver/ProductInfo/Availability/Retiring.asp

"Using Terminal Services for Graphical Remote Administration of the Windows 2000 Server Family", URL http://www.microsoft.com/technet/prodtechnol/win2kts/maintain/monitor/tsremote.asp?frame=true

"Windows 2000 Server Training: Security Services", URL: http://www.microsoft.com/technet/prodtechnol/windows2000serv/training/w2khost/w2ktsec.asp?

"Windows 2000 Server: Resource Kits: Windows 2000 Deployment Guide: Part 3– Active Directory Infrastructure: Chapter 11 – Planning Distributed Security", URL: http://www.microsoft.com/technet/prodtechnol/windows2000serv/reskit/deploy/part3/chapt-11.asp?

"Windows XP Professional: Resources Kit Documentation: Part 3– Security: Chapter 15 – Logon and Authentication", URL: http://www.microsoft.com/technet/prodtechnol/winxppro/reskit/prdp_log_gwno.asp?frame=true

© SANS Institute 2003, As part of the Information Security Reading Room. Author retains full rights.

Last Updated: June 19th, 2013

Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

Digital Forensics & Incident Response Summit 2013	**Austin, TXUS**	**Jul 09, 2013 - Jul 16, 2013**	**Live Event**
SANS London Summer 2013	**London, GB**	**Jul 09, 2013 - Jul 16, 2013**	**Live Event**
SANS Rocky Mountain 2013	**Denver, COUS**	**Jul 14, 2013 - Jul 20, 2013**	**Live Event**
SEC528 SANS Training Program for the CompTIA® New Advanced Security Practitioner (CASP) Certification	**Washington, DCUS**	**Jul 22, 2013 - Jul 26, 2013**	**Live Event**
SANS Mumbai 2013	**Mumbai, IN**	**Jul 22, 2013 - Jul 27, 2013**	**Live Event**
SANS San Francisco 2013	**San Francisco, CAUS**	**Jul 29, 2013 - Aug 03, 2013**	**Live Event**
SANS Boston 2013	**Boston, MAUS**	**Aug 05, 2013 - Aug 10, 2013**	**Live Event**
SANS SEC 560: Network Penetration Testing @ Bangalore 2013	**Bangalore, IN**	**Aug 05, 2013 - Aug 10, 2013**	**Live Event**
Critical Security Controls Summit	**Washington, DCUS**	**Aug 12, 2013 - Aug 18, 2013**	**Live Event**
Industrial Control Systems Security Training - DC	**Washington, DCUS**	**Aug 12, 2013 - Aug 16, 2013**	**Live Event**
SANS Virginia Beach 2013	**Virginia Beach, VAUS**	**Aug 19, 2013 - Aug 30, 2013**	**Live Event**
SANS Thailand 2013	**Bangkok, TH**	**Aug 19, 2013 - Aug 31, 2013**	**Live Event**
SANS Melbourne 2013	**Melbourne, AU**	**Sep 02, 2013 - Sep 07, 2013**	**Live Event**
SANS Capital City 2013	**Washington, DCUS**	**Sep 03, 2013 - Sep 08, 2013**	**Live Event**
Network Security 2013	**Las Vegas, NVUS**	**Sep 14, 2013 - Sep 23, 2013**	**Live Event**
SANS Canberra 2013	**OnlineAU**	**Jul 01, 2013 - Jul 13, 2013**	**Live Event**
SANS OnDemand	**Books & MP3s OnlyUS**	**Anytime**	**Self Paced**