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.

Using Terminal Services to Remotely Administer Windows2000 Servers Securely

This paper will focus primarily on the security issues of using Terminal Services to remotely administer Windows 2000 Servers. A general discussion of Terminal Services clients, licensing, the Remote Desktop Protocol (RDP) and Terminal Services encryption methods will provide the reader with a fundamental understanding of Terminal Services. A brief discussion of various security and Denial of Service vulnerabilities will demonstrate the need for Terminal Services security. The paper will conclude with the general steps...

Copyright SANS Institute

Author Retains Full Rights

AD

Using Terminal Services to Remotely Administer Windows 2000 Servers Securely

Dave Myhre

October 29, 2002

Abstract

This paper will focus primarily on the security issues of using Terminal Services to remotely administer Windows 2000 Servers. A general discussion of Terminal Services clients, licensing, the Remote Desktop Protocol (RDP) and Terminal Services encryption methods will provide the reader with a fundamental understanding of Terminal Services. A brief discussion of various security and Denial of Service vulnerabilities will demonstrate the need for Terminal Services security. The paper will conclude with the general steps necessary to plan, implement, test and monitor a secure Terminal Services configuration to remotely administer Windows 2000 Servers.

Terminal Services Fundamentals

The Microsoft Windows 2000 Server family operating systems include a remote desktop feature called Terminal Services. Terminal Services extends a Windows graphical desktop to clients over Local Area Network (LAN), Wide Area Networks (WAN) and Internet connections.[1] The Terminal Services client software is supported on a wide variety of hardware platforms and operating systems to include:

- Windows CE-based terminals
- Windows CE-based Handheld Professional devices (H/PC Pro)
- 32-bit Windows-based PCs running Windows 95, Windows 98, Windows NT 3.51, Windows NT 4.0, or Windows 2000 Professional
- 16-bit Windows-based PCs running Windows for Workgroups 3.11 with MS TCP/IP-32[2]
- Mac OS X version 10.1 or later[3]
- Windows XP Home and Windows XP Professional[4]
- PocketPC 2002[5]

Regardless of the hardware platform or operating system used, the user sees the same Windows graphical interface. The images they see on their terminal client are generated on the terminal server. All of the client's application, data processing, and data storage occur on the server.[6] Terminal Services can be configured to run in two different modes; a remote administration mode and application server mode.

[1] Using Terminal Services for Remote Administration of the Windows 2000 Server Family, p. 1.
[2] Windows 2000 Terminal Services: An Integrated, Server-Based Computing Solution, p. 4.
[3] Remote Desktop Connection Client for Mac OS X, p. 1.
[4] Windows XP Technical Overview, p. 8.
[5] PocketPC: Terminal Services Client, p. 1.
[6] Terminal Services Features, p. 1.

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

The remote administration mode is limited to two concurrent connections primarily intended for remote server administration duties. Any Window 2000 server can be configured to support the remote administration mode and has two built-in per-server connections specifically for remote administration. A Terminal Services Client Access License (CAL) is not required to connect to Terminal Services in Remote Administration mode.[7]

The application server mode is intended to support a thin-client computing environment for large numbers of users. This mode requires additional client licenses. A Terminal Services Client Access License (CAL) as well as a Windows 2000 Server CAL is required for each client connection.[8] The licensing requirements can become complex depending on the client used and should be thoroughly researched prior to implementation. Regardless of the Terminal Services licensing mode employed, all clients connect to the server using Microsoft's RDP protocol.

Microsoft's Remote Desktop Protocol (RDP) was based on an existing ITU T.120 family of protocols. It is a multi-channel capable protocol that transports presentation data from the server to the client. The protocol also returns keyboard and mouse inputs from the client to the server.[9] Microsoft best describes the details of how the RDP protocol enables the client-server communication:

> RDP uses its own video driver on the server side to render display output by constructing the rendering information into network packets using RDP protocol and sending them over the network to the client. On the client side, it receives rendering data and interprets them into the corresponding Win32® GDI API calls. On the input path, client mouse and keyboard messages are redirected from the client to the server. On the server side, RDP uses its own virtual keyboard and mouse driver to receive these keyboard and mouse events.[10]

The initial protocol implementation, RDP 4.0, was released with Windows NT Server 4.0 Terminal Server Edition.[11] Since the original release, the RDP protocol has been updated to include new features and performance enhancements. RDP 5.0, an updated version, was released with the Windows 2000 server family and RDP 5.1 was released with Windows XP. Both versions added several new features such as local printer redirection and clipboard mapping.[12] In order to protect the data communication stream carried by the RDP protocol, various levels of encryption are employed.

---

[7] Windows 2000 Terminal Services: An Integrated, Server-Based Computing Solution, p. 2.
[8] Windows 2000 Terminal Services: An Integrated, Server-Based Computing Solution, p. 1.
[9] Remote Desktop Protocol (RDP) Features and Performance, p. 6.
[10] Remote Desktop Protocol (RDP) Features and Performance, p. 6
[11] Remote Desktop Protocol (RDP) Features and Performance, p. 4
[12] Remote Desktop Protocol (RDP) Features and Performance, p 4.

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

The RDP protocol supports three levels of encryption: low, medium and high. All levels of RDP encryption utilize RSA Security's RC4 cipher.[13] Low encryption is uni-directional; it secures the data going from the client to the server but not the data returning from the server. This encryption setting is intended to protect sensitive user information, such as passwords, during transit to the server. Information sent to the client is screen refreshed for protection, which according to Microsoft is difficult to intercept.[14] The medium encryption setting is bi-directional, encrypting data to and from the server.[15] For both low and medium encryption settings, the strength of the RC4 encryption depends on the Terminal Services client used. RDP 4.0 clients connecting to a Windows 2000 server will use 40-bit encryption and RDP 5.0 clients will use 56-bit encryption.[16] The high encryption setting uses 128-bit bi-directional encryption to secure communications between the client and server.[17] (With the removal of the strong encryption export restriction, 128-bit RDP encryption is the only encryption level for the high encryption settings on post Service Pack 2 installations.[18]) In addition to the standard Terminal Services client application, clients can access the terminal server with the Terminal Services Advanced Client (TSAC).

The TSAC is a Win32 ActiveX control that allows users to run a Terminal Services session through an Internet Explorer browser session.[19] When the TSAC package is installed on a Window 2000 server configured with Terminal Services, it installs the downloadable ActiveX component and updates the Internet Information Server to create a web-based entry point. When a client accesses the Terminal Services web entry page, the ActiveX client is downloaded and installed on the client. A terminal session is then launched through the web browser. Even though the clients view their Terminal Services session in a web browser, the actual terminal session is conducted over TCP port 3389 using the RDP Protocol.[20] Now that we have gone over some of the basics of Terminal Services, we can begin to discuss locating terminal servers and exploiting some of the security vulnerabilities.

Security and Denial of Service Vulnerabilities

Before an attacker can exploit terminal server vulnerabilities, he or she must find the terminal server. There are many ways to do this. The choices range from a simple Internet search engine to a sophisticated terminal server probing utility. The tools and techniques vary depending on where the terminal server is located and how it is configured.

[13] Remote Desktop Protocol (RDP) Features and Performance, p 6.
[14] Zimmerman, Maureen Williams, ed., p. 607.
[15] Zimmerman, Maureen Williams, ed., p. 607.
[16] Securing Terminal Server Communication Between Client and Server, p.1.
[17] Zimmerman, Maureen Williams, ed., p. 607.
[18] 128-bit Encryption Becomes the Default in Windows 2000 Service Pack 2 (SP2), p. 1.
[19] Microsoft Terminal Services Advanced Client, p. 1
[20] How Terminal Server Advanced Client Connects to a Terminal Server Computer, p. 1

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

An Internet search engine, such as Google (www.google.com), can be used to locate a terminal server exposed the Internet that is configured to support the TSAC web client.[21] The default Uniform Resource Locator (URL) for the Terminal Services web entry point is http://server/TSWeb/.[22] Using the advanced features on the search engine, the search can be narrowed to only look for URLs containing the string "TSWeb". As an example, on the Google site using the search criteria "inurl:TSWeb" will produce a list of sites containing TSWeb in their URL. Potentially many of these sites will be Terminal Services web entry pages.

Port scanning for a single IP or a range of IP addresses on TCP port 3389 can also identify terminal servers.[23] Since the default TCP port for Terminal Services is 3389, a port that responds open to the scan may be a terminal server. To confirm that an open TCP 3389 port is a terminal server, an attacker can simply connect to the IP address with a Terminal Services client. If the server responds with a login prompt he or she has confirmed that the open TCP 3389 port is a listening terminal server.[24] In addition to port scanning, other tools are available to locate terminal servers on a LAN.

Included with the Windows operating system is a utility called "qappsrv.exe". When run from the command prompt, this application will produce a list of servers running Terminal Services. By default, the application lists servers that are in the workstation's domain; however, the "/domain:<domain name>" switch can be used to list terminal servers in other domains.[25] If the default TCP port 3389 has been changed, the task of locating terminal servers is more difficult and requires other tools.

Tim Mullen (Thor@hammerofgod.com) has developed two utilities that can find terminal servers that are not running on the default TCP port 3389.[26] "ProbeTS.exe" and "TSEnum.exe" both scan the local network looking for terminal servers, but each uses a different method for locating the servers.

"ProbeTS.exe" scans the network querying each computer using Remote Procedure Calls (RPC). In order for the query to work, the user running the scan must be an authenticated Terminal Services user on the target computer.[27] Tim Mullen noted that this was serious limitation and therefore, the tool would typically be limited to users with domain admin privileges.[28] To counter this limitation, he developed another terminal server enumeration tool.

[21] Scambray, p. 312.
[22] Using Remote Desktop Web Connection, p. 1.
[23] Scambray, p. 312.
[24] Scambray, p. 312.
[25] Minasi, p. 1
[26] Scambray, p. 314.
[27] Mullen, p. 1.
[28] Mullen, p. 1.

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

"TSEnum.exe" also scans LANs looking for terminal servers. Instead of using RPC queries to locate terminal servers, it queries the browser service. When a workstation or server joins the domain, it registers itself with the master browser and identifies all its services. "TSEnum.exe" has the target computer query its master browser using the NetServerEnum API call to retrieve the running services.[29] [30] In addition to providing a list of all the terminal servers on the network, "TSEnum.exe" provides a wealth of information without requiring any special user privileges.[31] Now that we have discussed different ways to find terminal servers, we need to address some of the security vulnerabilities and exploits.

While an unauthorized user gaining access to a server is a bad thing, an unauthorized user gaining access through a Terminal Services session is even worse. When users are authenticated to a server through a Terminal Services session they are assigned the Interactive Security Identifier (SID), not the Network SID that is assigned to a user accessing a server over a network share.[32] The difference in SID assignments is a significant matter. The Interactive SID presents two major issues. The first deals with privilege escalation attacks and the second with account lockout policy.

After a user successfully logs into a terminal server session and is assigned the Interactive SID, their ability to run privileged escalation attacks is greatly improved over a user with only the Network SID. Joel Scambray and Stuart McClure said it best, "On unpatched terminal servers, everyone is an Administrator!"[33] In addition to making it easier to run privilege escalation attacks, the Interactive SID also affects the account lockout policy.

It is possible to configure an account lockout policy to counter brute force password guessing attacks by setting the policy to lock the account after "X" number of failed network login attempts. Normally this lockout policy will affect all user accounts but not the built-in Administrator account. The Administrator account can be set to follow the lockout policy by running a utility found in the Windows NT 4.0 Resource Kit called "passprop.exe" with the "/adminlockout" switch set.[34] This utility will allow the Administrator account to lockout after the number of failed network login attempts has been exceeded; however, it will *not* lockout the account as a result of failed interactive logins.[35] Therefore, since the Terminal Services is an Interactive login, the built-in Administrator account is completely exposed to a brute force password guessing attack over a terminal server session.[36] To automate the exploitation of this vulnerability, Tim Mullen is

[29] Mullen, p. 1.
[30] NetServerEnum, p. 1.
[31] Mullen, p. 1.
[32] Zimmerman, Maureen Williams, ed., p. 1526.
[33] Scambray, p. 317.
[34] Windows 2000 Server Baseline Security Checklist, p 4.
[35] Scambray, p. 316.
[36] Scambray, p. 316.

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

currently developing a tool called "TSGrinder.exe". The tool will systematically conduct a brute force dictionary attack until the Administrator password is guessed.[37] If the terminal server is configured to display a logon banner, "TSGrinder.exe" patiently waits to exploit a logic error in the logon banner. The logon banner will close itself if the OK button is not selected after two minutes, then "TSGrinder.exe" continues is attack.[38] If a terminal server is successfully attacked, finding the IP address of the attacker may prove to be difficult.

The terminal server does not come with a built-in logging feature to record the IP address of each connected session. However, there are ways to view the current session connection details in the Terminal Services Management MMC. In some cases the connection details may also be written to the Windows event log.[39] The difficulty arises when the client connection is made from a host behind a Network Address Translation (NAT) device or a proxy server. The IP address displayed in the Terminal Services Management MMC is the private IP address of the client, not the public address of the NAT device or proxy server.[40] Without the public IP address, locating the source of the connection is next to impossible. There is a way to identify the public IP address if the terminal server client is still active by using the "netstat.exe –an" command and looking for established TCP 3389 connections.[41] This will produce a list of established Terminal Services connections; however, if there is more than one connection it may be difficult to prove which public IP address is associated with the private IP address in question. In addition to the attacks directed at the terminal server, there are also vulnerabilities in the RDP protocol.

Depending on the encryption level settings selected, the communication between the terminal server and client is either bi-directionally or uni-directionally encrypted. However, regardless of the encryption setting, not all the communication between the server and client is encrypted. While the actual data is encrypted, some traffic is passed to the server in plain-text. This plain-text traffic includes:

- Information passed through the virtual channel,
- initial packets used to create the RDP connection,
- encryption level negotiation,
- and licensing packets (containing client computer name, client username and client license information).[42]

Although this plain-text information is not critical data such as username/passwords, it is valuable information in regards to network reconnaissance efforts. Recently, another plain-text item was identified in the RDP communication.

---

[37] Scambray, p. 316.
[38] Mullen, p. 1.
[39] Windows 2000 And XP Terminal Services IP Address Spoofing, p. 1.
[40] Windows 2000 And XP Terminal Services IP Address Spoofing, p. 1.
[41] Windows 2000 And XP Terminal Services IP Address Spoofing, p. 1.
[42] High Encryption on a Terminal Services Session Does Not Encrypt All Information, p. 1

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

The RDP implementation in Window 2000 and Windows XP does not encrypt the checksum value used to detect communication errors between the server and client. The plain-text checksum value can be used to mount a brute force cryptographic analysis of the RDP session.[43] Once an attacker has obtained the encryption key, he or she could recover all the session information that used the cracked key. However, since a unique key is used for each RDP session, one cracked key would not result in a compromise of every RDP session.[44] In addition to encryption vulnerabilities, there are numerous Denial of Service (DoS) vulnerabilities in Terminal Services.

Certain types of network scans can result in DoS conditions. For example, when the network scanning tool, "nmap.exe", is used to SYN scan a terminal server (with option -sS -p "3389" xxx.xxx.xxx.xxx) it may cause the server to restart if the scan is conducted before the first Terminal Services client connection.[45] Terminal servers are also susceptible to other DoS attacks. Malformed RDP data packets can also result in a variety DoS conditions.

One type of malformed RDP data packet attack results in small memory leaks on the server. The small memory leaks are insufficient to cause a major problem on the server; however, a large number of packets received over a given time period could render the server non-functional.[46] Other types of malformed RDP data packet attacks can cause the server to fail. Microsoft published two security bulletins MS01-006 and MS01-052 that detailed the DoS conditions. In both cases, the attacker would not be required to start a Terminal Services session with the server. They would only need to send the malformed RDP packet sequence to the server to cause it to fail.[47] [48] With all the security and DoS vulnerabilities how do you take advantage of the benefits of remotely administering servers with Terminal Services while minimizing the risks? Through detailed planning, careful implementation, complete testing, and continuous monitoring.

General Steps for Securing Terminal Services

One of the first steps in making a decision on whether to implement Terminal Services in a remote administration role is to conduct a risk analysis. Regardless of the risk analysis methodology you use, you are really only looking for the answer to the basic question, "Does the benefit of Terminal Services remote administration out weigh the cost?" If your analysis identifies an acceptable risk for remote administration the next step would be to incorporate its use into your security policy.

---

[43] CAN-2002-0863 (under review)
[44] Cryptographic Flaw in RDP Protocol can Lead to Information Disclosure (Q324380), p. 1.
[45] MS Terminal Services Vulnerable to SYN Scan, p. 1.
[46] Invalid RDP Data Can Cause Memory Leak in Terminal Services, p. 2.
[47] Invalid RDP Data can Cause Terminal Service Failure, p. 2.
[48] Invalid RDP Data can cause Terminal Server Failure, p. 1.

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

At a minimum, the security policy should include the following:

- What servers will be configured to support Terminal Services
- What administrators will have access to Terminal Services
- Access time restrictions (work hours, after hours etc.)
- Which client computers will be permitted access to Terminal Services
- Which client software will be supported (TSAC ActiveX, Terminal Services Client, MMC, etc)
- Where the clients will connect from (Internet, VPN, LAN, wireless, etc)
- RDP encryption strength
- TCP port assignment
- Firewall rules and logging
- Network Intrusion Detection System (IDS) rules and logging
- Auditing of Terminal Services activity

This list is by no means all-inclusive; however, this list should cover the basics and serve as a starting point for a policy specific to your environment. After the security policy is established, the next step is configuring the designated servers.

Configuring a server for Terminal Services in the remote administration mode is easy. With a couple of clicks of the mouse, the service is running. The difficult part is configuring the service to run securely. Whether building a server from scratch or adding the service to an existing server, the first step is to make sure the latest service packs and hotfixes are applied to the system. (If you're updating a production server, make sure you have current and verified backups, just in case…) The majority of security vulnerabilities and DoS conditions previously identified in this paper are corrected by service packs and available hotfixes. Service packs even resolve some vulnerabilities that are not widely published. As an example, prior to SP3, repeatedly pressing the F7 and ENTER keys in quick succession from the command prompt could cause the server to restart.[49] Following the application of service packs and hotfixes, a review of published security checklists is in order.

There are numerous security checklists that can aid in server hardening efforts. Most lists are created by a team of knowledgeable experts and do a very good job at providing security guidance. However, before blindly implementing the steps, take time to review the checklist and make sure you understand what each step does prior to implementing it on your server. (Remember, you want to harden the server not break it.) The following are a few security checklists that cover both general and terminal server specific security:

- Windows 2000 Server Baseline Security Checklist (Microsoft)
- Securing Windows 2000: Step-by-Step (SANS)
- Securing Windows 2000 Terminal Services (Microsoft)

[49] List of Terminal Server Fixes in Windows 2000 Service Pack 3, p. 1.

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

- Guide to Securing Microsoft Windows 2000 Terminal Services (National Security Agency)

The complete URLs to the checklists are available in the reference section of this document. Another source of useful tools to secure a terminal server is the Microsoft Windows 2000 Server Resource Kit.

In addition to providing in-depth documentation on Terminal Services, the Resource Kit contains tools to help administer terminal servers. Two notable tools are "Tsver.exe" and "Winsta.exe". The Terminal Services version limiter application (Tsver.exe) limits clients who can connect to a terminal server by the client build number.[50] The WinStation Monitor (Winsta.exe) application monitors the status of all users logged on to a terminal server.[51] Both tools are nice additions to the administrator's terminal server security arsenal. Securing the server is only part of the task, attention needs to be paid to the network itself.

Whether or not the terminal server is accessible from the Internet determines how the network security is configured. The simple solution is to not permit access to Terminal Services from the Internet. Configuring the router and firewall to block TCP port 3389 (assuming the default port on the terminal server has not been changed) will effectively block access to the terminal server. However, if access to the terminal server is required from the Internet, special consideration on how to access the server is required. Obviously, accessing the terminal servers over TCP 3389 is possible but not very secure. It effectively advertises to anyone with a port scanner that the servers are probably running Terminal Services. It also exposes the RDP checksum encryption flaw to any attacker who can sniff the traffic. A more secure method allows terminal server connection over the Internet through a Virtual Private Network (VPN) connection, preferably using IPSec. This option provides several benefits:

- TCP3389 can be blocked,
- terminal server existence is not advertised,
- communication is encrypted by IPSec in addition to RDP RC4 encryption,
- client validation is better controlled during VPN tunnel setup.

These are just a couple of options for controlling access to the terminal server over the Internet. Depending on the network architecture and access requirements, other solutions may be more appropriate. Now that the security policy has been established and the security measures applied, the next step is to test the security of the Terminal Services implementation.

The important aspects of this step are to 1) test the security measures to ensure they enforce the security policy standards, and 2) protect the terminal server. There are many different network penetration testing methods and tools available to test the security of the Terminal Services implementation. Any failures of the security policy or weakness of the terminal server need to be

[50] Windows 2000 Server Resource Kit Tools, p. 19.
[51] Windows 2000 Server Resource Kit Tools, p. 21.

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

corrected. With the security measures thoroughly tested, the next step is to monitor the Terminal Services activity.

Reviewing the server event logs, firewall logs, and IDS logs for unauthorized terminal server access is a critical part of monitoring the terminal server security policy. If you set a high level of detail for the logging services, you can use the information for several purposes such as, identifying unauthorized activity, auditing terminal server access, or updating the security policy. Frequent review of the logs is a critical part to the overall security of the terminal servers.

Conclusion

With a better understanding of the basic fundamentals of Terminal Services RDP protocol and encryption methods, the administrator can appreciate how Terminal Services is vulnerable to attacks. The benefits of remotely administrating Windows 2000 servers are obvious, but the risks can be disastrous is your security is breached. Knowing the various security vulnerabilities and Denial of Service attacks is the first step to protecting against them. With a commitment to detailed planning, careful implementation, complete testing and continuous monitoring, administrators can gain all the benefits of remote administration while minimizing the security risk.

© SANS Institute 2003, Author retains full rights.

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

References

"128-bit Encryption Becomes the Default in Windows 2000 Service Pack 2 (SP2)." March 27, 2001. URL: http://www.microsoft.com/windows2000/server/evaluation/news/bulletins/128bit.asp (June 25, 2002).

"CAN-2002-0863 (under review)" Aug 18, 2002. http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2002-0863 (October 21, 2002).

"Cryptographic Flaw in RDP Protocol can Lead to Information Disclosure (Q324380)" September 18, 2002. URL: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS02-051.asp (October 21, 2002).

"High Encryption on a Terminal Services Session Does Not Encrypt All Information." August 10, 2001. http://support.microsoft.com/default.aspx?scid=kb;en-us;Q275727 (June 25, 2002).

"How Terminal Server Advanced Client Connects to a Terminal Server Computer." August 7, 2000. URL: http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q270897& (October 17, 2002).

"How to Change the Listening Port in the Windows Terminal Server Web Client." July 24, 2002. URL: http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q326945& (October 17, 2002).

"Invalid RDP Data Can Cause Memory Leak in Terminal Services." July 25, 2001. URL: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS01-040.asp (June 25, 2002).

"Invalid RDP Data can cause Terminal Server Failure." January 31, 2001. URL: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS01-006.asp (June 25, 2002).

"Invalid RDP Data can Cause Terminal Service Failure." October 18, 2001. URL: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS01-052.asp (June 25, 2002).

"List of Terminal Server Fixes in Windows 2000 Service Pack 3." June 20, 2002. URL: http://support.microsoft.com/default.aspx?scid=kb;en-us;Q324956 (October 17, 2002).

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

"Microsoft Terminal Services Advanced Client." October 16, 2000. URL: http://www.microsoft.com/windows2000/server/evaluation/news/bulletins/tsac.asp (June 25, 2002).

"MS Terminal Services Vulnerable to SYN Scan." August 1, 2002. http://www.securitytracker.com/alerts/2002/Aug/1004927.html (October 21, 2002).

"NetServerEnum." Platform SDK: Network Management. August 2002. http://msdn.microsoft.com/library/default.asp?url=/library/en-us/netmgmt/netmgmt/netserverenum.asp (October 17, 2002).

"PocketPC: Terminal Services Client." Aug 16, 2002. URL: http://www.microsoft.com/mobile/pocketpc/learnmore/software/terminalservices.asp (Oct 2, 2002).

"Remote Desktop Connection Client for Mac OS X." July 17, 2002. URL: http://www.microsoft.com/mac/DOWNLOAD/MISC/RDC.asp (Oct 2, 2002).

"Remote Desktop Protocol (RDP) Features and Performance." June 27, 2000. URL: http://www.microsoft.com/windows2000/docs/rdpfandp.doc (June 13, 2002).

"Securing Terminal Server Communication Between Client and Server [Q232514]." Jan 14, 2001. URL: http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q232514& (Oct 14, 2002).

"Terminal Services Features." Oct 28, 1999. URL: http://www.microsoft.com/windows2000/server/evaluation/features/terminal.asp (Oct 2, 2002).

"Using Remote Desktop Web Connection." URL: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/winxppro/reskit/pree_rem_iriq.asp (October 17, 2002).

"Using Remote Desktop Web Connection." URL: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/winxppro/reskit/pree_rem_iriq.asp (October 17, 2002).

"Using Terminal Services for Remote Administration of the Windows 2000 Server Family." March 8, 2001. Microsoft Corporation. URL: http://www.microsoft.com/windows2000/docs/TSRemote.doc (October 2, 2002).

"Windows 2000 And XP Terminal Services IP Address Spoofing." November 7, 2001. http://www.xato.net/reference/xato-112001-01.txt (June 25, 2002).

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

"Windows 2000 Server Baseline Security Checklist." http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/chklist/w2ksvrcl.asp (October 17, 2002).

"Windows 2000 Server Resource Kit Tools." URL: http://www.microsoft.com/windows2000/techinfo/reskit/rktour/server/S_tools.asp (October 21, 2002).

"Windows 2000 Terminal Services: An Integrated, Server-Based Computing Solution." September 24, 1999. Microsoft Corporation. URL: http://www.microsoft.com/windows2000/docs/Tssol.doc (Oct 2, 2002).

"Windows XP Technical Overview." May 18, 2001. URL: http://www.microsoft.com/windowsxp/pro/techinfo/planning/techoverview/WindowsXPTechnicalOverview.doc (Oct 2, 2002).

Mackey, David "Securing Windows 2000 Terminal Services." http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/win2kts/maintain/optimize/secw2kts.asp (October 17, 2002).

Minasi, Mark. "9 Undocumented Terminal Server Commands." May 1999. http://www.winnetmag.com/Articles/Print.cfm?ArticleID=5146 (October 17, 2002).

Mullen, Tim. "Downloads and Stuff." http://www.hammerofgod.com/download.htm (October 17, 2002).

Scambray, Joel and Stuart McClure. Hacking Windows 2000 Exposed. New York: Osbourne/McGraw-Hill, 2001.

Zimmerman, Maureen Williams, ed. Microsoft Windows 2000 Server Deployment Planning Guide. Redmond: Microsoft Press, 2000.

Zimmerman, Maureen Williams, ed. Microsoft Windows 2000 Server Distributed Systems Guide. Redmond: Microsoft Press, 2000.

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

Security Checklist Resources

Guide to Securing Microsoft Windows 2000 Terminal Services
http://www.nsa.gov/snac/win2k/download.htm

Securing Windows 2000: Step-by-Step
http://store.sans.org/store_category.php?category=consguides

Securing Windows 2000 Terminal Services
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/win2kts/Default.asp

Windows 2000 Server Baseline Security Checklist
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/chklist/w2ksvrcl.asp

© SANS Institute 2003, Author retains full rights.

Last Updated: June 18th, 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**