



# Global Information Assurance Certification Paper

Copyright SANS Institute  
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permitted without express written permission.

## Interested in learning more?

Check out the list of upcoming events offering  
"Security Essentials: Network, Endpoint, and Cloud (Security 401)"  
at <http://www.giac.org/registration/gsec>

**Windows Remote Desktop Heroes and Villains**

*GSEC Gold Certification*

Author: Greg Farnham

Adviser: Don Weber

Accepted: December 10<sup>th</sup> 2007

## Outline

1. Introduction .....	4
1.1. Scenario.....	4
1.2. Remote Desktop Overview .....	5
1.3. Remote Desktop Tips .....	7
1.3.1. Connect to the console.....	7
1.3.2. Query connected users .....	8
1.4. Network Configuration for Testing .....	9
2. RDP Vulnerability History.....	10
2.1. MS01-006 (Microsoft-MS01-006, 2001).....	10
2.2. MS01-052 (Microsoft-MS01-052, 2004).....	10
2.3. MS02-051 (Microsoft-MS02-051, 2007).....	11
2.4. MS05-041 (Microsoft-MS05-041, 2005).....	11
3. Villains .....	11
3.1. TSGrinder .....	13
3.2. ProbeTS.....	14
3.3. TSCrack.....	15
3.4. rdesktop .....	16
3.5. Cain and Able .....	17
3.6. SPIKE fuzzing tool, used in RDP DOS Advisory.....	17
4. Heroes .....	18
4.1. Policies and Procedures .....	18
4.1.1. Password Policy .....	19
4.1.2. General Policies .....	19
4.2. Windows Server Configuration .....	20
4.2.1. Rename the Administrator Account.....	20
4.2.2. Configure Password Policy.....	21

4.2.3.	Configure RDP Server Settings .....	23
4.2.4.	Change the Remote Desktop port .....	23
4.2.5.	Configure Windows Firewall .....	25
4.3.	2X SecureRDP .....	26
4.3.1.	Server Configuration.....	27
4.3.2.	SecureRDP Summary .....	29
4.4.	IPSec .....	29
4.4.1.	Server Configuration.....	31
4.4.2.	Client Configuration .....	35
4.4.3.	IP Security Monitor .....	37
4.4.4.	IPSec Summary .....	39
4.5.	OpenVPN.....	39
4.5.1.	Server Configuration.....	40
4.5.2.	Client Configuration .....	43
4.5.3.	Firewall Configuration.....	44
4.5.4.	OpenVPN Summary .....	45
4.6.	TLS based authentication .....	46
5.	Future .....	47
6.	Traffic Captures .....	49
7.	Summary .....	51
8.	References.....	52

## 1. Introduction

This paper will focus on a fictitious scenario of a non-profit organization that would like to understand the threats to remote desktop and improve security. This paper will review past vulnerabilities in the Windows Remote Desktop service, review threats, review mitigation techniques and summarize the results.

### 1.1. Scenario

The organization, NPO, has limited funds and cannot afford to maintain an IT infrastructure. NPO rents 6 internet based Windows 2003-SP2 servers. The servers are located in the data center of the server provider. NPO does not have physical access to the servers. They run a Voice over IP (VoIP) application that uses one udp port. The servers are not part of a domain and are managed by a Remote Desktop connection over the Internet. NPO typically has four part time Administrators. NPO would like to know the threats from allowing Remote Desktop access over the internet and identify possible mitigation techniques to those threats.

The NPO System Administrators will access the servers using Windows Remote Desktop. They will run Windows XP Home or Windows XP Pro. They typically access the servers from residential internet connections. The residential connections can have static or

dynamic IP addresses and often include a home router with Network Address Translation (NAT). Any security improvements will have to meet these basic requirements:

- Support Windows XP Home and Professional
- Support dynamic IP addresses
- Support clients using NAT
- Increase Security
- Low Cost
- Low Client Footprint

## 1.2. Remote Desktop Overview

Remote Desktop is a feature built into Windows 2003 Server which allows a user to remotely connect to the server desktop. With the remote desktop the remote user can interact with the server just like they are logged in directly at the console. For this paper, we are focusing on “Remote Desktop for Administration”. The Administration mode allows 2 concurrent connections and is intended primarily for Administration. Remote Desktop can also be used in “Application Mode” which allows multiple users to connect and run applications on the server. Remote Desktop has also been known as “Terminal Services”. This paper will use Remote Desktop and Terminal Services interchangeably. Remote Desktop Protocol (RDP) is the protocol used for remote desktop connections. The default

port used is TCP 3389. The version history of Remote Desktop Protocol is shown in Table 1 (Wikipedia-RDP, 2007)

Remote Desktop Version History

Operating System	Remote Desktop Version
Windows NT 4.0 Server, Terminal Server Edition	4.0
Windows 2000 Server	5.0
Windows XP Pro	5.1
Windows Server 2003	5.2
Windows Vista	6.0
Windows Server 2008	6.1

Table 1

For Windows Server 2003, remote administrators can connect to the console in addition to the two virtual sessions. Remote Desktop Protocol currently will only run over TCP/IP, but it has been designed to be independent of the transport and could be run over other transports in the future (Microsoft-186607, 2007). Remote Desktop offers several features, among them are RC4 stream cipher with 56 or 128 bit encryption, Roaming disconnect, Remote control and Bandwidth reduction (Microsoft-aa383, 2007). With Remote Desktop, client resources such as file systems, printers, and audio can be redirected to the server (Microsoft-techts, 2005). This allows for example, a user to print to their local printer.

There are additional clients available for remote desktop including Windows CE “Remote Desktop Web Connection” and the Linux based rdesktop program.

### 1.3. Remote Desktop Tips

These tips will be useful when using the Remote Desktop functionality.

#### 1.3.1. Connect to the console

The console login allows a remote user to interact directly with the console (Microsoft-278845, 2007). It may be useful for applications that display messages directly to the console. The physical console will be locked when a remote user is connected. The console connection will allow a connection even if the two virtual connections are in use. The console connection will also allow you to connect if someone else is connected to the console remotely. In this situation, it will disconnect the other user. Normally, a user starts remote desktop from the All Programs menu or by running mstsc.exe without any options. The user will then get a “Remote Desktop Connection” window where they can enter the IP address and other options. To connect to the console, a user adds the “-console” to the mstsc.exe command line. With this option, the user will get the same “Remote Desktop Connection” window, but they will be connecting to the console. The console is also known as Session 0.



Start>Run and type in “mstsc.exe -console” and hit Enter.

Log in with the Remote Desktop login window that appears

### 1.3.2. Query connected users

The quser command will display all connected users. This is useful to see which users are logged in with RDP. It can be run by anyone with a Command Prompt on the server. In addition, it can be used to query the RDP connections on a remote computer using the /SERVER option.

Start>Run>CMD

Type **quser** in the Command Prompt window and hit Enter.

In the example shown below, there is one user connected to the console (ID=0) with session rdp-tcp#4 with username npoking. There is another session (ID=3) with the same username that is currently disconnected. The “>” before the username for the first user, indicates the session that ran the quser command.

```
C:\>quser
USERNAME                SESSIONNAME              ID  STATE  IDLE TIME
>npoking                 rdp-tcp#4                0   Active
npoking                  3   Disc      none
```

#### 1.4. Network Configuration for Testing

Testing was performed in various configurations including using the same network segment and using VMWare. The network diagram shown in Figure 1 shows an example configuration for testing. It includes a Router with NAT and the Server on a separate segment from the client.

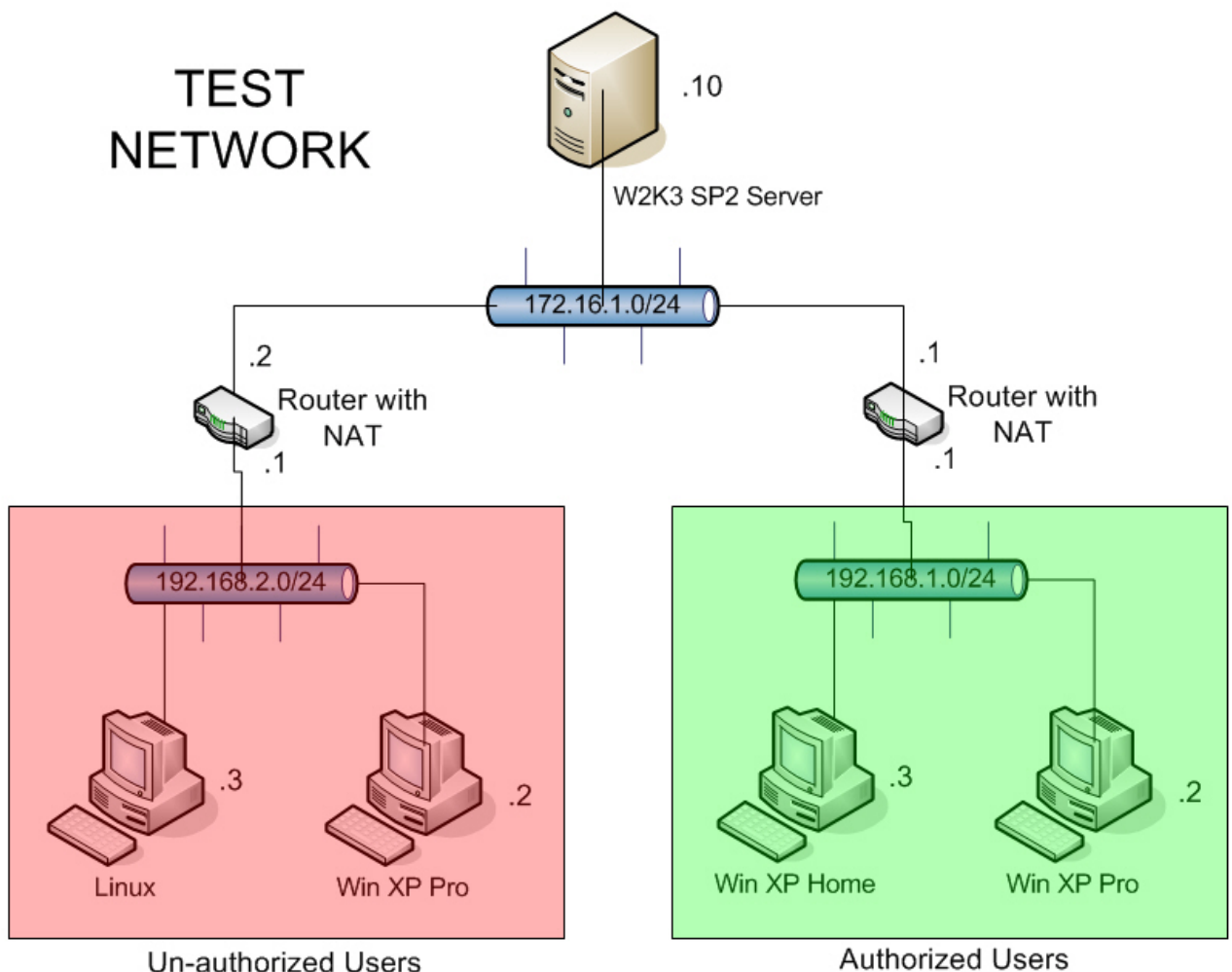


Figure 1

## 2. RDP Vulnerability History

Like most products, Microsoft's Remote Desktop feature using the RDP protocol has had vulnerabilities in the past. Many of these vulnerabilities are Denial of Service (DOS) as shown below. These are not as severe as a remote exploit, but DOS vulnerabilities are sometimes a precursor to a remote exploit. All of the vulnerabilities below have been patched. Likely, there will be additional vulnerabilities in the future.

### 2.1. MS01-006 (Microsoft-MS01-006, 2001)

This vulnerability was reported by Yoichi Ubukata and Yoshihiro Kawabata. A patch has been released (Q286132). A remote user can send malformed RDP packets to the server and cause it to stop responding

### 2.2. MS01-052 (Microsoft-MS01-052, 2004)

This bulletin originated from a DOS vulnerability reported by Luciano Martins in October, 2001 (Martins, 2001). A remote user can send malformed RDP packets to the server and cause it to stop responding. A patch has been released (Q307454).

### 2.3. MS02-051 (Microsoft-MS02-051, 2007)

This bulletin originated from vulnerabilities reported to the bugtraq mailing list by Ben Cohen and Pete Chown in August 2002 (Cohen, 2002). A patch (Q324380) has been released. The denial of service vulnerability allows an attacker to send a specially crafted package and cause the server to reboot. The packet can be sent prior to authentication. The keystroke vulnerability is interesting because it was introduced by a change to increase performance. It was introduced in RDP 5.0. In the original post, the author recommends using the RDP 4.0 client since it was not vulnerable. The checksum vulnerability could allow an attacker with access to the RDP traffic to gather information.

### 2.4. MS05-041 (Microsoft-MS05-041, 2005)

This bulletin originated from a vulnerability reported by Tom Ferris. It was posted in August of 2005. Similar to some of the other RDP DOS vulnerabilities, an attacker can send malformed RDP packets and cause the server to stop responding.

## 3. Villains

There are a number of different threats (Villains) that arise from having a Remote Desktop connection available on the internet. Many of the threats can be categorized as

Information Disclosure, Dictionary, Brute Force, Denial of Service and Man in the Middle (MITM) attacks. Information Disclosure is an attack that results in the disclosure of information that is not intended to be public. The information could be confidential data such as Human Resource records or something less obvious such as knowing when an Administrator is connected to a server. A Dictionary attack can be used to guess a password by trying all the passwords in a list or dictionary. A Brute Force attack can also be used to guess a password. An attacker will repeatedly try all possible passwords until he finds the valid one.

A Denial of Service attack is used to disrupt normal operations. While the attacker will not gain information or access to a system, he is able to deny its use to legitimate users. This can result in significant loss of productivity. A Denial of Services (DOS) can come in several different forms. We saw in the Vulnerability History section that there have been several specific DOS vulnerabilities in RDP. Another type is a Distributed Denial of Service (DDOS). In a DDOS attack, the attacker has a large number of hosts, hundreds or thousands that he uses to send normal requests to the victim. The victim is overwhelmed and cannot service legitimate users.

The Man in the Middle (MITM) attack is one of the more complex attacks. In this attack, an attacker will impersonate a server. The user will unknowingly create an

authenticated session to the attacker allowing the attacker to capture the credentials. The attacker will then create an authenticated session to the real server. The attacker will sit in the middle and pass traffic in both directions between the client and the server. The attacker is able to view all the traffic (unencrypted) between the client and the real server. The user is unaware that he is being monitored. In April of 2003, Erik Forsberg released an advisory describing a man in the middle vulnerability in RDP (Forsberg, 2003). In it he described how there is no verification of the identity of the server. In May of 2005, Massimiliano Montoro released a paper that explains how RDP is still vulnerable to MITM (Montoro, 2005). In it, he explains that the use of a private key hard coded in one of the DLLs allows an attacker to calculate a valid signature. This allows the attacker to successfully impersonate the server without the client knowing.

There are a number of specific tools designed for attacking Remote Desktop. Like many tools, they can be used for good or for bad. The Villain moniker only applies when used for malicious purposes.

### 3.1. TSGrinder

TSGrinder is a tool that can be used to perform a dictionary attack on a Remote Desktop server. It leverages tools available from Microsoft for load testing Terminal Services.

TSGrinder will sequentially try passwords from a dictionary list file. It will also allow the words in the list file to be modified with “1337” substitution. For example, a 3 would be substituted for an E in the list of passwords. It supports multiple threads and can try up to 5 passwords per connection. The Remote Desktop server will drop the connection and log the event on the 6<sup>th</sup> try. TSGrinder is a Windows executable. Executing the command without any options will present a usage page.

```
C:\tsgrinder>tsgrinder
tsgrinder version 2.03
```

Usage:

```
tsgrinder [options] server
```

Options:

```
-w dictionary file (default 'dict')
-l 'leet' translation file
-d domain name
-u username (default 'administrator')
-b banner flag
-n number of simultaneous threads
-D debug level (default 9, lower number is more output)
```

Example:

```
tsgrinder -w words -l leet -d workgroup -u administrator -b -n 2 10.1.1.1
```

### 3.2. ProbeTS

ProbeTS is a tool to find Terminal Services on a network. ProbeTS requires an authenticated connection to the target. This limits its use to be within the same domain. Instead of scanning ports, it uses RPC to determine if Terminal Services is running on the target. This would typically only be useful for scanning an internal network by a Domain

Administrator. Because it requires an authenticated connection, an attacker on the internet could not use this tool to determine if an NPO server is running Terminal Services. This tool is not a threat in the NPO scenario. Executing the command without any options will present a usage page.

```
C:\>probets

ProbeTS v1.1 - thor@hammerofgod.com
Terminal Server Probe

Usage: probets NBIOSName/IP
i.e. probets 192.168.1.1
-or-
Usage: probets CClass [BegIP] [EndIP]
i.e. probets 192.168.1 1 200

Get hammered at HammerofGod.com
```

### 3.3. TSCrack

TSCrack is a tool for performing a dictionary and brute force attack against a Remote Desktop server. It uses screen scraping of the graphical logon to test for success. TSCrack supports two simultaneous connections and can optionally prevent the system from logging failed password attempts by limiting the number of tries per connection. Executing the command without any options will present a usage page.

```
C:\>tscrack
terminal services cracker (tscrack.exe) v2.0.55 2002-13-10 04:13 AM
(c) 2002 by gridrun [TNC] - All rights reserved -
http://softlabs.spacebitch.com
Usage help:
    tscrack [switch] [switch [arg]] ... <Host/IP[:port]>

Parameters:
    <Host/IP[:port]> : DNS name or IP address of target server, optional port
```



### Switches:

```
-h : Print usage help and exit
-V : Print version info and exit
-s : Print chipher strenght info and exit
-b : Enable failed password beep
-t : Use two simultaneous connections [EXPERIMENTAL]
-N : Prevent System Log entries on targeted server
-U : Uninstall tscrack and remove components
-f <number> : Wordlist entry to start cracking with
-F <delay> : Sampling Frequency (Delay between samples in ms)
-l <user> : Account name to use, defaults to Administrator
-w <wordlist> : Wordlist to use; tscrack tries blank passes if omitted
-p <password> : Use <password> to logon instead of wordlist/blank pass
-D <domain> : Specify domain to attempt logon to
```

### 3.4. rdesktop

The rdesktop application is an open source client that runs on Unix/Linux based systems. It can be used for example by a linux user to connect to a Windows 2003 Remote Desktop. There is a patch available that allows it to be used to perform a dictionary attack (Gates, 2007). The usage information with some common options is shown below. With the brute force patch, the `-p` option will accept a file name with a dictionary list.

```
Usage: rdesktop [options] server[:port]
Description
-u <username>
    Username for authentication on the server.
-d <domain>
    Domain for authentication.
-n <hostname>
-p <password>
    The password to authenticate with. Note that this may have no effect if
    "Always prompt for password" is enabled on the server. WARNING: if you specify a
    password on the command line it may be visible to other users when they use tools
    like ps. Use -p - to make rdesktop request a password at startup (from standard
    input).
    Client hostname. Normally rdesktop automatically obtains the hostname of
    the client.
-f
    Enable fullscreen mode. This overrides the window manager and causes the
    rdesktop window to fully cover the current screen. Fullscreen mode can be toggled
```

at any time using Ctrl-Alt-Enter.

```
-0      Attach to the console of the server (requires Windows Server 2003 or
newer).
-4      Use RDP version 4.
-5      Use RDP version 5 (default).
```

### 3.5. Cain and Able

Cain and Able is a multi featured tool for Windows. It includes many password related features such as brute force, dictionary and cryptanalysis. It also has features for sniffing, recording voip conversations and wireless. One of the features related to Remote Desktop is the ability to do a MITM attack against RDP using Arp Poison Routing (APR).

### 3.6. SPIKE fuzzing tool, used in RDP DOS Advisory

In August of 2005 Tom Ferris released an advisory on the Remote Desktop DOS (Ferris, 2005) identified in Microsoft Bulletin MS05-041. The advisory includes the SPIKE script and usage information for causing a denial of service on a Remote Desktop server. SPIKE is a linux based Fuzzer Creation Kit. Security Researchers can use SPIKE to test how applications respond to malformed packets. Fuzzing is an automated technique where valid input is repeatedly modified creating fuzzed input. Each variation of the input data is sent to the application to see if it causes a problem such as a crash. For the RDP DOS, the generic tcp fuzzer (generic\_send\_tcp) program included with SPIKE is used. A SPIKE script

remoteass.spk defines the input for the generic tcp fuzzer. Below is abridged output from running the RDP DOS with SPIKE.

```
$ ./generic_send_tcp 192.168.1.101 3389 remoteass.spk 1 0
Total Number of Strings is 681
Fuzzing
Fuzzing Variable 1:0
Fuzzing Variable 1:1
Variablesize= 5004
Fuzzing Variable 1:2
Variablesize= 5005
Fuzzing Variable 1:3
Variablesize= 21
```

## 4. Heroes

We have seen that there are many threats to Remote Desktop. Now we will look at techniques to improve security, the Heroes. In this scenario, some techniques such as an external firewall or external VPN device had to be ruled out due to cost. The primary area researched was improving security on the Server. A focus was put on limiting access to the Remote Desktop on the server. The goal is to control access to the Remote Desktop login screen. Specifically, only allowing access to authorized Administrators and denying access to the rest of the internet.

### 4.1. Policies and Procedures

Policies and Procedures are a valuable component of security solutions. Policies are used to define the required configuration of systems and behavior of personnel. Procedures

are used to define how a repeatable task should be performed. Procedures are embedded in the latter sections describing solutions for improving security.

#### 4.1.1. Password Policy

For this scenario, a password policy is needed. With the Remote Desktop port available to the general internet, the servers can easily be subjected to a brute force password attack. A strong password policy will help to mitigate the brute force threat. Users are encouraged to think in terms of a pass phrase. Using a phrase typically has a large number of characters and is easier to remember than random sequences of letters. The following was decided for the password policy.

- Passwords must be 14 characters or more in length.
- Passwords must be changed every 6 months.
- Passwords must meet Windows default complexity requirements.
- Passwords must contain at least 3 types from: lower case, upper case, number, special
- Passwords must be securely communicated.
- Passwords must be securely stored.

#### 4.1.2. General Policies

A few general policies have been defined to help keep operations secure and ensure that client machines meet a minimum standard.

- Servers must have Auto Update turned with automatic installation.
- Server must automatically run an up to date Anti-Virus program.
- All Administrators must run Windows XP SP2 or higher.

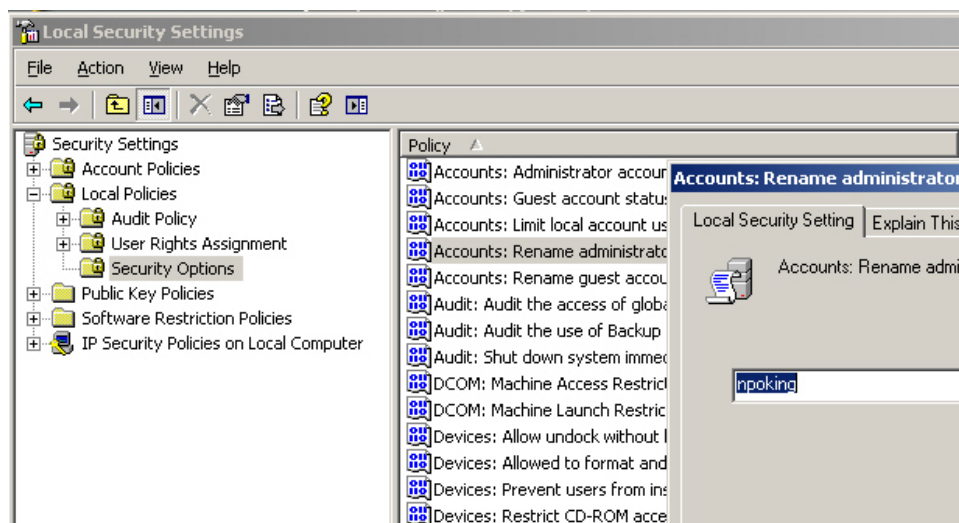
## 4.2. Windows Server Configuration

There are a few things on the Windows Server configuration that directly relate to securing Remote Desktop Access. This is not intended to be a complete Windows Server hardening guide. This section will cover Rename the Administrator account, Configure Password Policy, Configure RDP Server Settings, Change the Remote Desktop Port and Configure Windows Firewall.

### 4.2.1. Rename the Administrator Account

Renaming the Administrator Account will help to prevent a brute force attack on the Administrator account. Most brute force attacks will use the account name "Administrator". This is the default name and this account is not subject to account lockout. This configuration change is done by editing the Local security policy (Microsoft-2230, 2005). For the example shown, we are changing the Administrator account name to npoking.

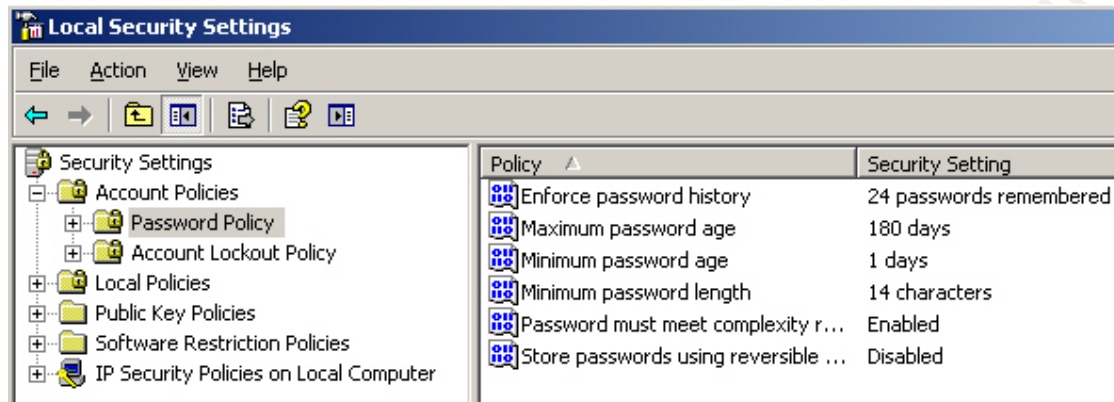
- Start>Settings>Administrative Tools>Local Security Policy
- Local Policies>Security Options>Accounts: Rename administrator account
- Change the value to npoking



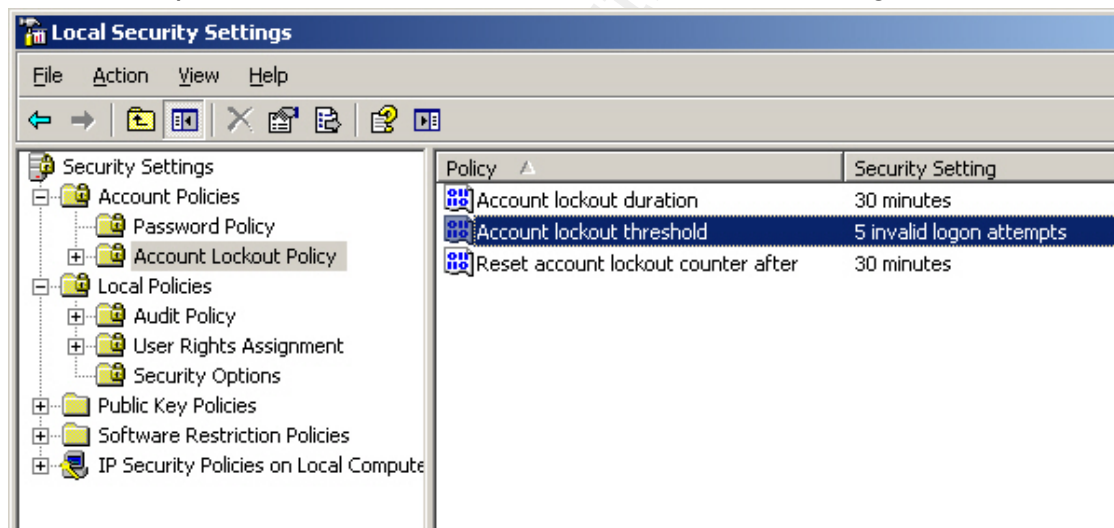
### 4.2.2. Configure Password Policy

Windows Server has a very robust password policy enforcement capability. This will be used to enforce our password policy from above. The password policy is also configured by making changes to the Local Security Policy.

- Start>Settings>Administrative Tools>Local Security Policy
- Select Account Policies>Password Policy
- Change settings per password policy.



- Select Account Policies Lockout Policy
- Change the Account lockout threshold to 5
- Accept the defaults of 30 minutes for the other 2 settings.



Local Security Settings can be exported and used to automate the configuration of another server.

- Select the Security Settings in the left pane

- Select Action>Export Policy...
- Type in localpol as the file name and click OK.

#### 4.2.3. Configure RDP Server Settings

The RDP Server settings can be used to increase security. Changes will only allow high encryption and limit some of the functionality. Limiting functionality will lower the attack surface available to an attacker.

- Start>Settings>Administrative Tools>Terminal Services Configuration
- Select Connections, Double Click RDP-Tcp
- Make the following changes
- General Tab: Encryption level: High
- Environment: Toggle on Do not allow an initial program...
- Remote Control: Toggle on Do not allow remote control
- Client Settings: Disable Drive mapping, Windows printer mapping, LPT port mapping, COM port mapping and Audio mapping.

#### 4.2.4. Change the Remote Desktop port

Changing the Remote Desktop port lowers the visibility of the server. It will require an attacker to do more than a port scan of common ports to find the RDP listening port. It could also help avoid a possible future worm that only propagates on the default port. This change is accomplished by changing a registry key (Microsoft-306759, 2007). Note: A reboot is

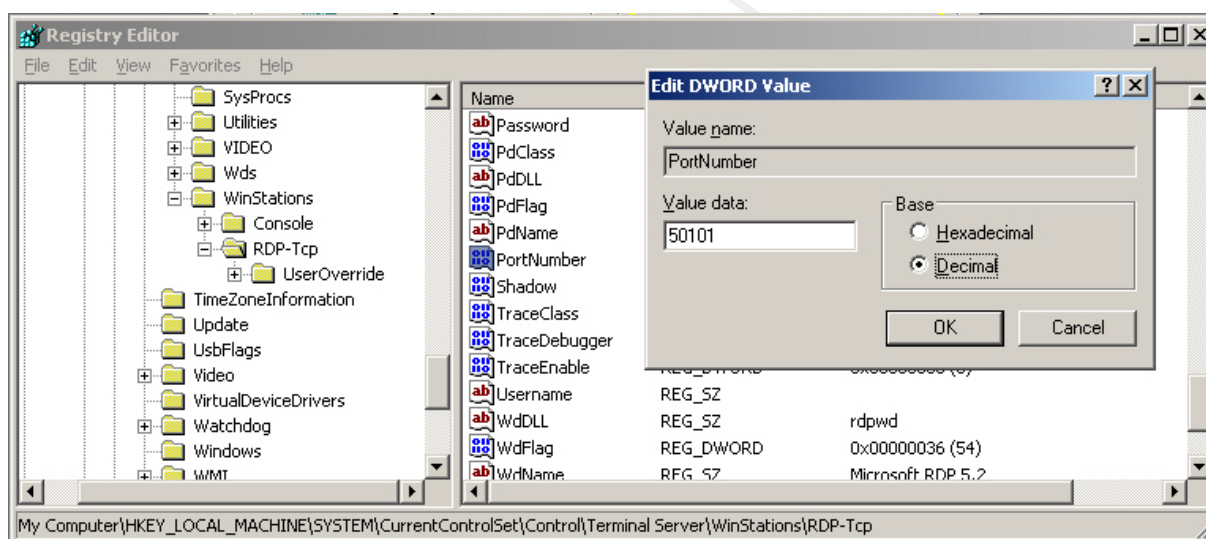


required for this change to take affect.

To change the registry key:

- Start>Run>regedit
- HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Control\TerminalServer\WinStations\RDP-Tcp\PortNumber

For the example shown, we are changing it to 50101 (decimal).



The RDP settings are stored in the registry. These settings can be exported as a registry file and used to automate the configuration on other servers.

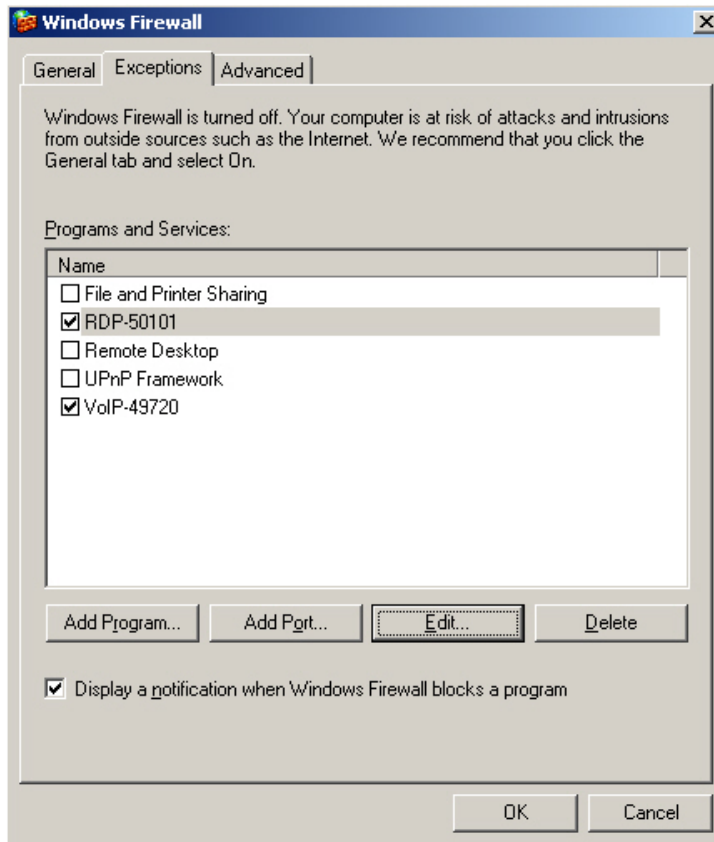
- Select the RDP-Tcp folder in the left pane
- Select File>Export
- Enter rdp for the filename and click OK.

#### 4.2.5. Configure Windows Firewall

The host firewall will be configured to allow only 2 exceptions. One for Remote Desktop access (TCP 50101) and one for the VoIP applications (UDP 49720).

- Start>Settings>Control Panel>Windows Firewall
- Note: You may need to start the Windows Firewall/ICS service.
- Toggle the FW on.
- Use Add Port to create custom services for TCP 50101 and UDP 49720.
- Uncheck all other Exceptions.

Note: If accessing remotely, be careful not to lock yourself out of the server. Configure the exceptions for the RDP port before turning the firewall on.



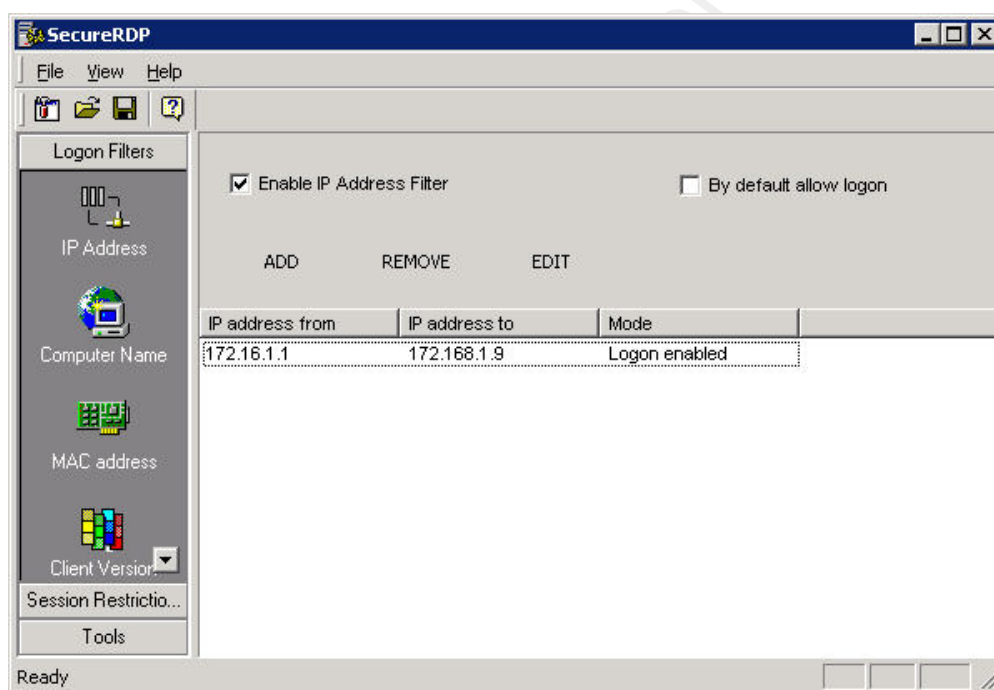
### 4.3. 2X SecureRDP

SecureRDP looked like a promising solution. SecureRDP is provided by 2X. 2X has a number of thin client computing products. SecureRDP is freeware available for download at no charge. It is an application that runs on the server and is specifically designed to control access to the RDP Service. SecureRDP allows configuring access to RDP based on IP Address, Mac Address, Computer Name, RDP Client version and time of day (SecureRDP, 2007). It also allows limiting the number of RDP sessions based on IP Address or User

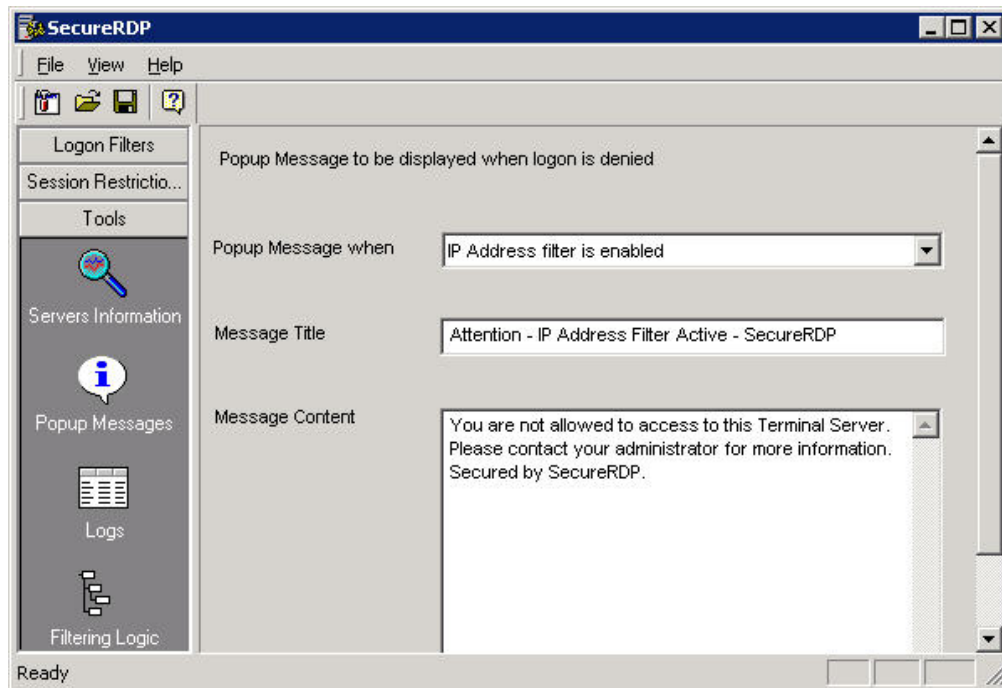
Name.

### 4.3.1. Server Configuration

Installation on the server was easy with a standard setup program. Nothing is required on the client. The configuration was also easy. The IP Address Logon Filter allows for simple adding and removing of IP addresses or ranges.



The application has customizable pop up windows for when someone is denied login.



The standard message does give away the fact that SecureRDP is being used. It is desirable to avoid this kind of information leak. Testing showed that if the Message Title and Message Content are deleted, no pop up window is displayed. There is one major short coming of the IP Address Logon Filter. It filters based on the local IP address of the client. In our test configuration shown in Figure 1, the local IP address for the client 192.168.1.2 had to be entered, not the NAT'd IP address 172.16.1.1.

The MAC Address Logon filter will only work if the client and server are on the same network segment. A client MAC Address will only be visible to the server if it is on the same network segment. For a connection that goes through a router, the MAC Address of the client will not be visible. The NPO Servers are in a data center on the internet and all clients will be

connecting through a routed connection. The MAC Address Logon filter is not applicable for the NPO scenario.

The Client version Logon filter might be useful in denying access to some attackers, but it would also put a burden on our administrators to have a specific client version.

The Computer name Logon filter might be useful. One issue with this option is that Administrators may sometimes use different computers, for example at a friend's house. It would be possible to change the name of the client computer to match one on the allowed list. Changing a computer name is a burden and does require a reboot.

#### 4.3.2. SecureRDP Summary

After testing, it was determined that this solution is not applicable to the NPO scenario. This is mainly due to the fact that it does not handle NAT'd IP addresses. Also, if IP address filtering was the approach, the Windows firewall would be sufficient by using a custom scope. The other filtering mechanisms do not meet the requirements.

#### 4.4. IPSec

IPSec is an internet standard protocol suite that provides encryption and authentication. It is built into Windows XP and Windows Server 2003. Initially, I did not think

this was viable in our scenario due to the Windows XP Home requirement and NAT requirement. After doing the research, I learned that these requirements could be met. Regarding Windows XP Home, there is a misconception that IPSec is not supported. The IPSec GUI is not available on Windows XP Home. However, the IPSec functionality is available. It can be configured using the ipseccmd command line tool. The ipseccmd command is available as part of the Windows XP Service Pack 2 Support Tools (Microsoft-49ae, 2004). For the NAT requirement, this had been a problem for IPSec. If a layer 4 header is protected by IPSec, then if the traffic gets NAT'd, the header cannot be updated to reflect the new IP address. This issue was resolved with the NAT-T standard and Microsoft released an update to support it (Microsoft-818043, 2006). This update is included in Windows XP Service Pack 2.

For Windows IPSec, peer authentication can be done using Kerberos, Pre-shared Key or Certificates. In the NPO scenario, peers are not part of a domain, so Kerberos can't be used. Pre-shared key was selected over Certificates to avoid the overhead of installing and maintaining a Windows Certificate Server. A draw back of using pre-shared keys is that they are stored in clear text in the registry (Microsoft-ipsecfaq, 2006). This is an acceptable risk for the NPO scenario. IPSec will be configured with pre-shared keys to authenticate the server and client with each other. To implement IPSec, configuration is required on the server and

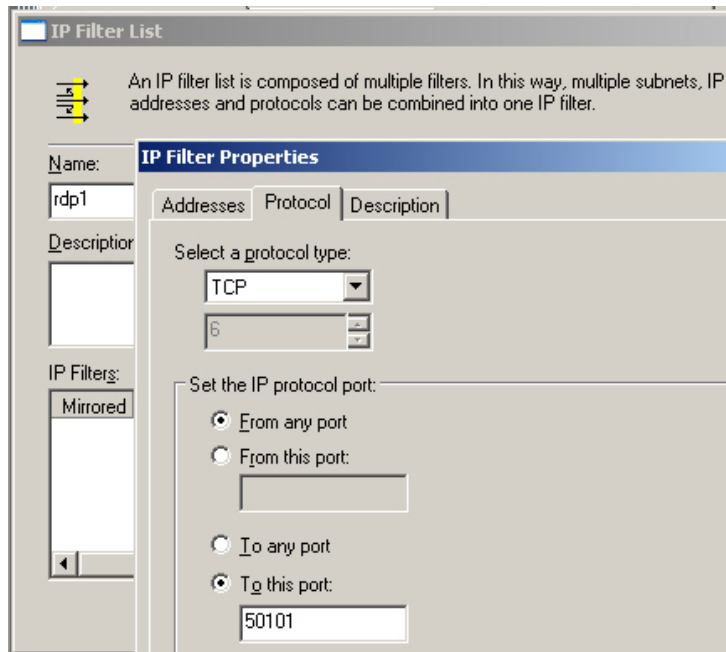
the client.

#### 4.4.1. Server Configuration

These instructions were created using “IPSec to secure Terminal Services” (Microsoft-816521, 2007) as a reference.

- Start>Settings>Administrative Tools>Local Security Policy
- Right click on IP Security Policies on Local Computer and select “Manage IP filter...”
- Click the Add button
- Enter the name rdp1
- Uncheck the “Use Add Wizard” button
- Click the Add button
- Select “Any IP Address” for the Source address.
- Select “My IP Address” for the Destination address.
- Verify that the Mirror box is checked.
- Select the Protocol tab
- Set the Protocol type to TCP.
- Select From any port
- Select To this port and enter 50101.

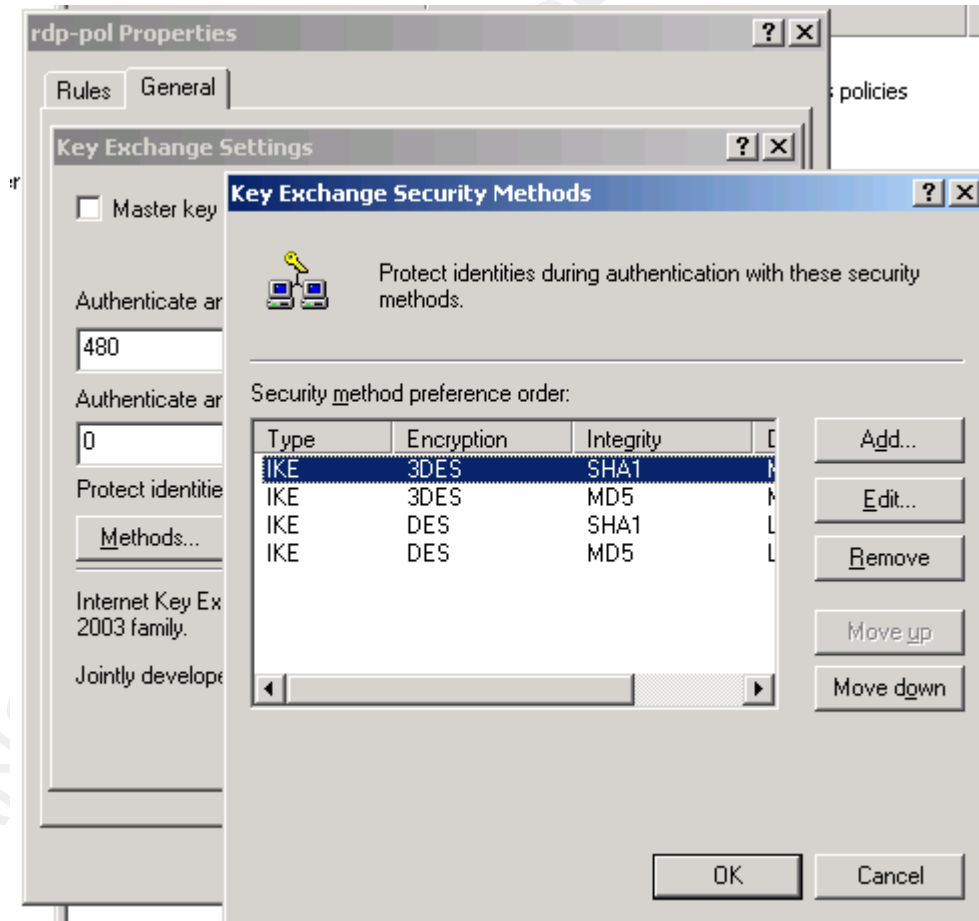




- 
- Click OK, Click OK,
- Click on the Manage Filter Actions Tab
- Uncheck the Use Wizard check box
- Click on the Add... button
- Click the General Tab
- Enter rdp-filteraction for the name
- Click the Security Methods Tab
- Select Negotiate Security
- Click Add...
- Select Integrity and encryption
- Click OK
- Verify that Security Method is ESP[3DES,SHA]
- Click OK
- Click Close
- Now create the Policy
- In the right pane of the Local Security Settings window, right click and select

“Create IPsec Security Policy”.

- Click Next
- Enter rdp-pol for the name. Click Next
- Uncheck the “Activate the default response rule”, click Next
- Click Next
- Click Finish
- Click the General Tab
- Click the Settings or Advanced button
- Click the Methods button
- Verify the IKE, 3DES, SHA1, Medium(2) is top in the list.



- Click OK

- Click OK
- Click the Rules Tab
- Uncheck the Use Add Wizard check box
- Click Add...
- Select the rdp1 filter list.
- Click on the Filter Action tab.
- Select the rdp-filteraction radio button.
- Click on the Authentication Methods tab.
- Click Add
- Select Use this string (preshared key):
- Enter the string npotest [a stronger key would be used in actual deployment]
- Click OK
- Select Kerberos, click Remove, click Yes
- Click OK, Click OK
- Right click rdp-pol and select Assign

The server can also be configured using the netsh command line utility. In Windows Server 2003, the XP functionality from ipseccmd was moved into netsh. The following commands will configure IPSec on the Server.

```
:IPSec Policy Definition
netsh ipsec static add policy name="rdp-pol" description="Remote Desktop
policy" activatedefaulttrule=no assign=no

:IPSec Filter List Definitions
netsh ipsec static add filterlist name="rdp-filter1" description="All
Connections to RDP"

:IPSec Filter Definitions
netsh ipsec static add filter filterlist="rdp-filter1" srcaddr=any dstaddr=me
description="RDP connections" protocol=TCP
mirrored=yes srcport=0 dstport=50101
```

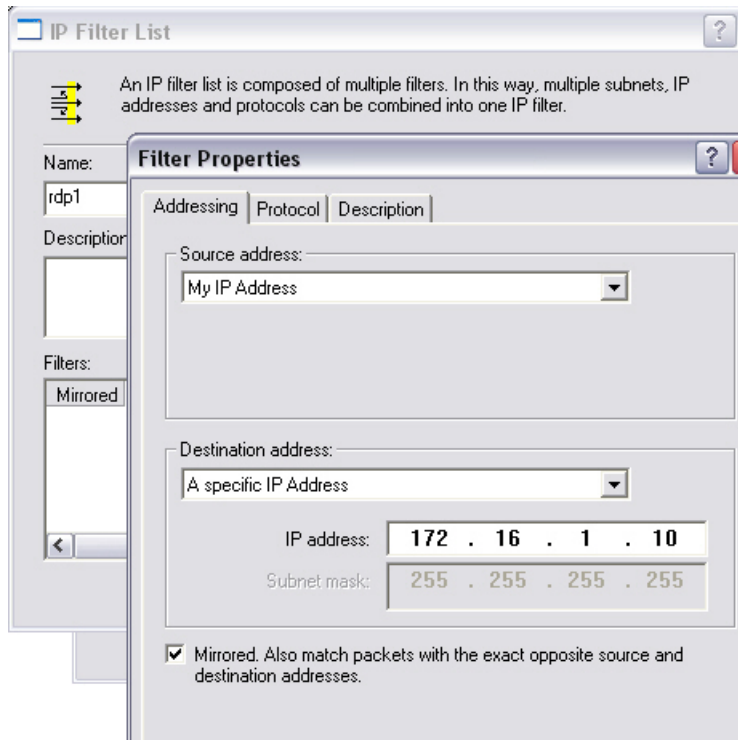
```
:IPSec Filter Action Definitions
netsh ipsec static add filteraction name="rdp-filteraction1"
description="encrypt" qmpfs=no inpass=no soft=no action=negotiate
qmsecmethods="ESP[3DES,SHA1]"

:IPSec Rule Definitions
netsh ipsec static add rule name="rdp-rule" policy="rdp-pol" filterlist="rdp-
filter1" psk="npotest" filteraction="rdp-filteraction1"
```

For definitions of each command, see (Microsoft-netsh, 2005).

### 4.4.2. Client Configuration

The configuration on the client is similar to the server. The main difference is when configuring the IP Filter list, use “My IP Address” for the Source and the specific server IP address for the destination.



The IPSec client configuration needs to be repeated for each server a user will connect to.

The client can also be configured using the ipseccmd command line utility. For Windows XP Home, the only option for configuring IPSec is to use ipseccmd. The ipseccmd utility requires installation of Windows XP Support tools (Microsoft-49ae, 2004). During installation, the “complete” option must be selected instead of the default “typical”. Otherwise, ipseccmd will not be installed.

Once ipseccmd is installed, the client can be configured with a one line command which could be stored in a batch file.

The following command should be entered on one line:

```
ipseccmd.exe -f 0+172.16.1.10/255.255.255.255:50101:TCP -n ESP[3DES,SHA] -a  
PRESHARE:"npotest" -ls 3DES-SHA-2 -w reg -p rdppol -r rdprule -x
```

Below is a summary of each parameter:

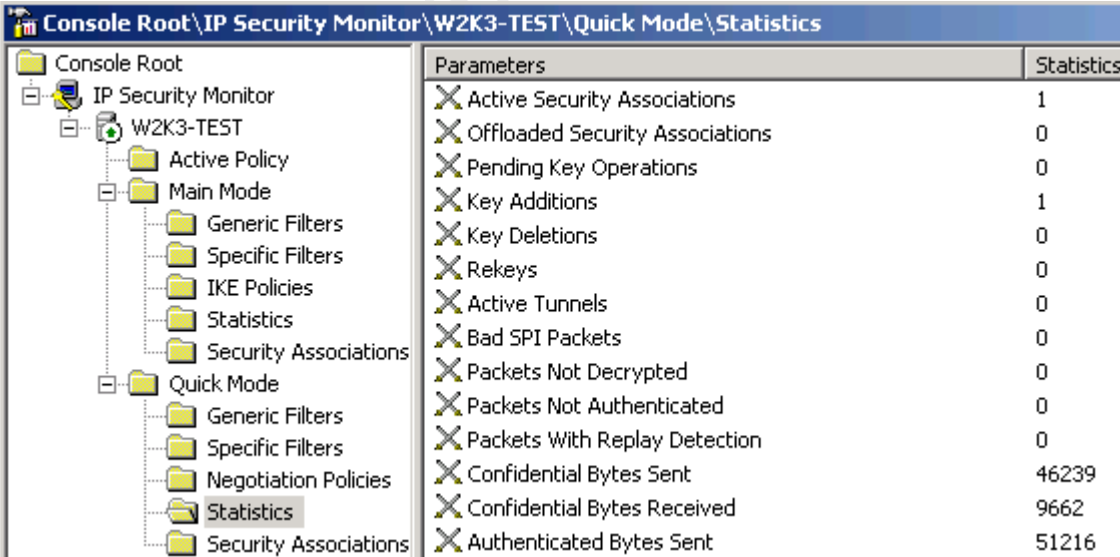
- **-f 0+172.16.1.10/255.255.255.255:50101:TCP** This is the filter definition. The 0 indicates a source of “My IP Address” which is the IP address of the client. The 172.16.1.10/255.255.255.255 is the server address with a 32 bit mask. The 50101:TCP indicates the destination port.
- **-n ESP[3DES,SHA]** This is the security method to be used for securing the traffic identified by the filter.
- **-a PRESHARE:"npotest"** This defines the authentication method as pre-shared key and the key as npotest.
- **-ls 3DES-SHA-2** This defines the security method for key exchange.
- **-w reg** This parameter specifies that the policies and rules will be written to the local registry.
- **-p rdppol** This parameter specifies the name of the IPsec Policy
- **-r rdprule** This parameter specifies the name to use in the IP Filter List and the Filter Action.
- **-x** This parameter specifies that the new policy “rdppol” is assigned. In other words active. Note: To disable a policy, run the same command with a -y instead of a -x.

### 4.4.3. IP Security Monitor

IP Security Monitor is a tool built in to Windows Server 2003 and Window XP Pro. It can be used to monitor the status of any IPsec connections. IP Security Monitor is provided

as an MMC Snap-In. It can be accessed by running MMC and adding the Snap-In.

- Start>Run>mmc
- File>Add/Remove Snap-In
- Click Add...
- Select IP Security Monitor
- Click Add
- Click Close
- Click OK
- Expand the items in the left pain.
- Select Statistics under Quick Mode



Parameters	Statistics
Active Security Associations	1
Offloaded Security Associations	0
Pending Key Operations	0
Key Additions	1
Key Deletions	0
Rekeys	0
Active Tunnels	0
Bad SPI Packets	0
Packets Not Decrypted	0
Packets Not Authenticated	0
Packets With Replay Detection	0
Confidential Bytes Sent	46239
Confidential Bytes Received	9662
Authenticated Bytes Sent	51216

On Windows XP Home with the support tools installed, the following command will display information similar to IP Security Monitor.

```
ipseccmd localhost show gpo filters policies auth stats sas all
```

#### 4.4.4. IPSec Summary

This section has described the basics of IPSec and how it will be configured to work in our scenario. Once configured, each user can be given a batch file with the one line required to configure IPSec to connect to a server. Only clients configured to use IPSec with the pre-shared key will be able to connect to Remote Desktop and get to the login screen. Authorized users can carry around a USB flash drive that has the Windows XP Support tools and batch files to configure IPSec for the servers they need to access. They would then be able to install ipseccmd and execute the batch script. Alternatively, if they have the pre shared key, they could configure Windows XP Pro client manually. This solution meets all of our requirements.

#### 4.5. OpenVPN

OpenVPN is an Open Source project by James Yonan and is licensed under the GPL (Wikipedia-OpenVPN, 2006). OpenVPN uses SSL/TLS protocol to provide VPN Services on multiple platforms including Linux, Windows, Mac and others. OpenVPN is very flexible. There are over 100 different configuration settings for meeting various needs. OpenVPN supports features such as, client/server VPNs, pre-shared keys, certificates, bridged VPNs, routed vpns, dhcp server and nat traversal (Yonan, 2003).



When installed on Windows, OpenVPN creates a TAP-Win32 virtual adapter. This adapter will show up in the Network Connections form and the output of the ipconfig command. The virtual adapter can be used in tap mode to create bridged VPNs or in tun mode to create a routed VPN.

For the NPO scenario, a routed VPN will be used. The Remote Desktop server will be configured with OpenVPN in server mode and the Remote Desktop client will be configured with OpenVPN in client mode. Peer authentication can be done with Pre-shared keys or Certificates. OpenVPN installation includes an easy to use certificate server (easy-rsa). Since the certificate server is already available, certificates will be used for peer authentication in the NPO scenario. The server will listen on the standard port UDP 1194. Once the VPN tunnel is established, each host will have an IP address on the VPN network (10.8.0.0/24). For the Remote Desktop client to connect to the server, it will use the servers VPN network IP (10.8.0.1) instead of its native IP (172.16.1.10) shown in Figure 1. To setup OpenVPN, the application must be installed on the server and the client.

### 4.5.1. Server Configuration

The server installation requires running the installer, generating certificates, editing the configuration file, start OpenVPN and setting the Service to Auto.

- **Run installer** – This is similar to most Windows installation programs. All the defaults work fine for this scenario.
- **Generate Certificates** – This step warrants some additional discussion. For authentication, we will run a certificate authority (CA) on our server. For the server and each client, we will generate a private key and a Certificate Signing Requests (CSR). The CA will be used to sign the CSR and generate a Certificate for the server and each client. The Public Key Infrastructure (PKI) for OpenVPN is included in the easy-rsa folder. This folder includes a README.txt file which outlines the steps. Also included are several batch files which execute the steps we need to perform. These batch files will run openssl with the correct parameters to complete that step in the process. Note: Files in the OpenVPN have linux style end of line characters. Wordpad will correctly display the files whereas Notepad will not.

build-ca.bat will generate the CA certificate file ca.crt and the CA private key file ca.key.

build-dh.bat will generate the DH file dh1024.pem (assuming default of 1024 bits).

build-key-server.bat will first generate a Server Certificate Signing Request server.csr and a Server private key server.key. Next, it will use the CA private key to sign the server.csr resulting in the Server Certificate file server.crt.

build-key.bat will first generate a Client Certificate Signing Request client.csr

and a Client private key client.key. Next, it will use the CA private key to sign the client.csr resulting in the Client Certificate file client.crt. It is important when generating certificates for different clients to use a different common name. The common name is one of the prompts when executing the batch file.

After generating the certificates and keys, copy the server files, dh1024.pem and ca files to the config folder under the OpenVPN installation on the server. The client files and ca.crt should be securely transferred to the client.

- **Edit the config file** – The config file contains settings that are used when starting OpenVPN. These settings could also be applied on the command line, but for our scenario we will use the config file. The OpenVPN installation includes a sample-config folder. There is a server.ovpn file that will be copied to the config folder as the starting point for the server configuration. Important settings are shown below.

```
# SSL/TLS root certificate (ca), certificate
# (cert), and private key (key). Each client
# and the server must have their own cert and
# key file. The server and all clients will
# use the same ca file.
ca ca.crt
cert server.crt
key server.key # This file should be kept secret

# Configure server mode and supply a VPN subnet
# for OpenVPN to draw client addresses from.
# The server will take 10.8.0.1 for itself,
# the rest will be made available to clients.
```

```
# Each client will be able to reach the server
# on 10.8.0.1. Comment this line out if you are
# ethernet bridging. See the man page for more info.
server 10.8.0.0 255.255.255.0
```

- **Start OpenVPN** – OpenVPN can be started by right-clicking on the server.ovpn file and selecting “Start OpenVPN on this config file”. A command tool style window will appear. If everything starts successfully, you will see an “Initialization Sequence Completed” message.
- **Set Service to Auto** – Once everything is working and tested, the service should be set to auto start. Select Start>Run>services.msc. Right-click the OpenVPN Service and select Properties. Change the Startup Type to Automatic and click OK. Right-click the OpenVPN Service and select Start.

### 4.5.2. Client Configuration

The client installation requires running the installer, copying certificate files from server, editing the configuration file, start OpenVPN and Set Service to Auto.

The client installation is similar to the server installation and uses the same installation executable. For the client, the certificate can be generated on the server and securely copied to the client. The main difference is the config file.

- **Edit the config file** – The config file contains settings that are used when starting

OpenVPN. The OpenVPN installation includes a sample-config folder. There is a client.ovpn file that will be copied to the config folder as the starting point for the client configuration. Important settings are shown below.

```
# Specify that we are a client and that we
# will be pulling certain config file directives
# from the server.
client

# The hostname/IP and port of the server.
# You can have multiple remote entries
# to load balance between the servers.
remote 172.16.1.10 1194

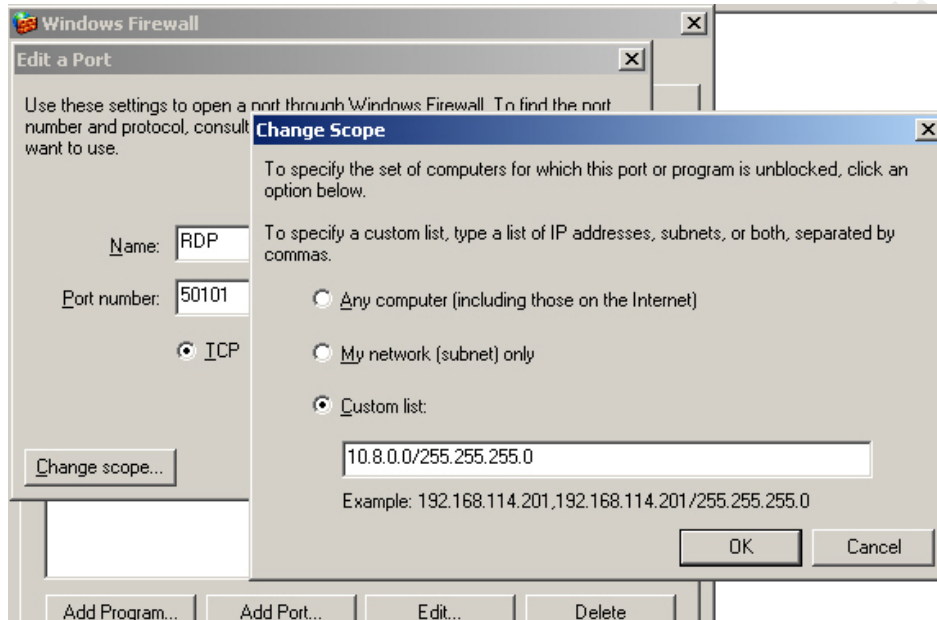
# SSL/TLS parms.
# See the server config file for more
# description. It's best to use
# a separate .crt/.key file pair
# for each client. A single ca
# file can be used for all clients.
ca ca.crt
cert client.crt
key client.key
```

### 4.5.3. Firewall Configuration

With the OpenVPN connection, our authorized clients will always be connecting to the Remote Desktop port using the VPN Network (10.8.0.0/24). Therefore, we can change the scope of the rule allowing access to the RDP port to only allow the VPN Network. This is done with the following steps on the Server.

- Start>Control Panel>Windows Firewall
- Click on the Exceptions Tab.
- Select our custom RDP Exception.

- Click on Edit, Click on Change Scope
- Select Custom list: and enter 10.8.0.0/255.255.255.0
- Click Ok, Click Ok, Click Ok.



This configuration will block unauthorized clients from connecting to the Remote Desktop Port while still allowing authorized clients through the OpenVPN connection.

#### 4.5.4. OpenVPN Summary

This section has described the basics of OpenVPN and how it will be configured to work in our scenario. Once configured, each user can be given a client certificate that is signed by our servers CA server. Only clients that have a certificate signed by our CA Server will be allowed to connect to open VPN. Since the firewall is configured to only allow RDP connections from the VPN Network, only users connected to OpenVPN will be allowed to

connect to Remote Desktop. Authorized users can carry around a USB flash drive that has OpenVPN, their client certificate files and their client.ovpn config file. They would then be able to install OpenVPN and connect to the Remote Desktop server from any PC that they have Administrator right to. This solution meets all of our requirements.

#### 4.6. TLS based authentication

TLS authentication is a solution provided by Microsoft to mitigate the Man in the Middle attack. It works the same as a web based TLS authentication. A server has a certificate that is signed by a trusted Certificate Authority. The client trusts the Certificate Authority, so it knows that the server is the correct one and not an imposter. This solution requires Windows 2003 SP1 or higher on the server side and RDP 5.2 or higher on the client side. For the server certificate, it can be obtained one of three ways (Kiaer, 2006). The SelfSSL.exe tool in the IIS 6.0 resource kit can be used. An SSL certificate could be signed by a 3<sup>rd</sup> party CA. Or, an organization can use an existing Public Key Infrastructure (PKI) such as Microsoft Certificate Services. After obtaining the certificate, Terminal Services needs to be configured to use it. On the client side, users can configure one of three options for the Remote Desktop Connection. They can configure “No authentication”, “Attempt authentication” or “Require authentication”. “Require authentication” would not allow a connection unless the server’s identity has been authenticated. The client also needs to load the CA’s certificate or

otherwise already trust the CA.

While this solution helps to mitigate the MITM attack, it does not help prevent unwanted connections to the RDP port. An attacker can configure their client for “No authentication” and connect whether TLS authentication is used or not.

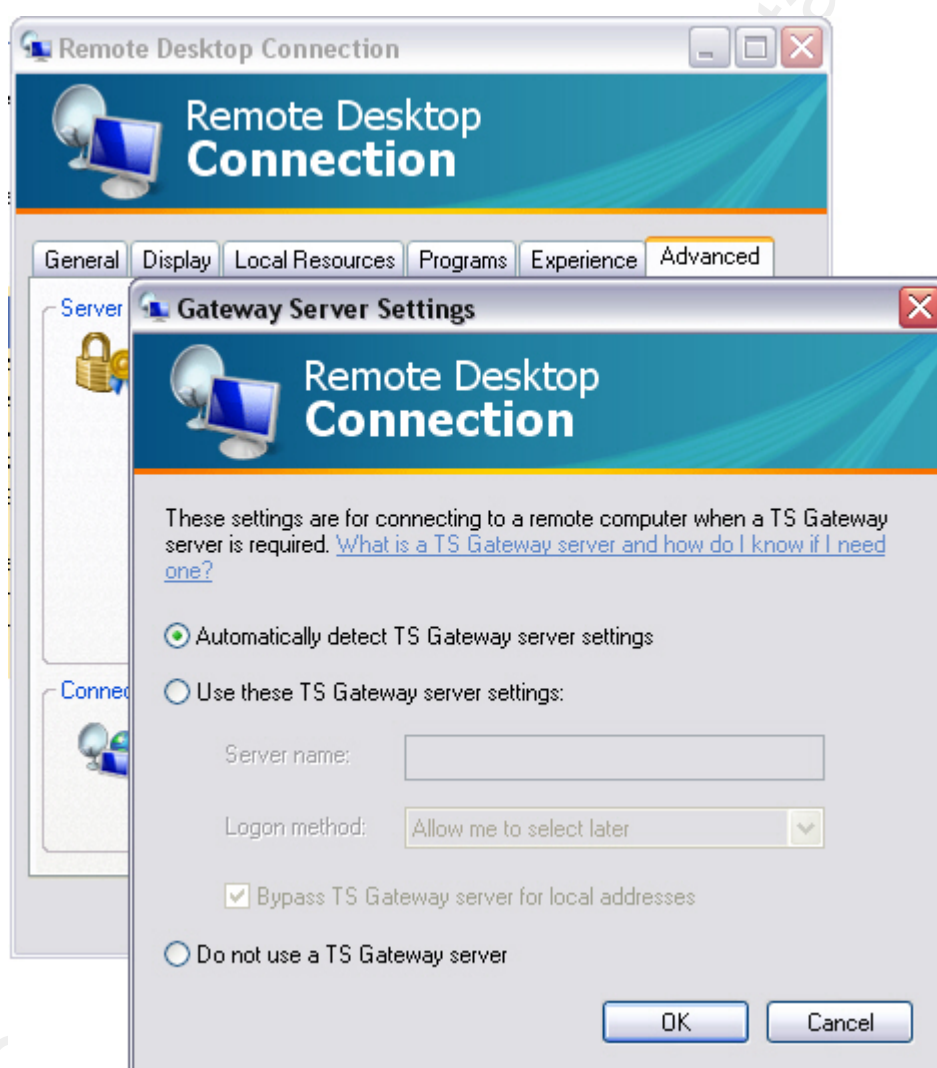
## 5. Future

Microsoft released version 6.0 of the Remote Desktop client with Vista (Microsoft-925876, 2007). The new features in 6.0 will be available with Server 2008 as well. The 6.0 client can also be installed on Windows XP SP2, Server 2K3 SP1 and Server 2K3 SP2. Remote Desktop client 6.0 was released as an automatic update to XP.

There are new security related features in RDP 6.0. The first is Terminal Services Gateways. A Terminal Services Gateway functions similar to a VPN appliance. Users will connect to it on port 443 using the RDP 6.0 client. Once connected, they will be able to access the internal network. The next is “Network Level Authentication” (NLA). NLA completes user authentication before providing a Remote Desktop connection. This reduces the resource used by an unauthorized user trying to connect and thus helps mitigate DOS attacks. The other feature is “Server authentication” (Microsoft-92586, 2007) which helps prevent MITM attacks. Server authentication is performed using Kerberos or Certificates.



Server authentication uses the same client settings described above in TLS based authentication. The selections have been reworded. In 6.0, they are “Always connect, even if authentication fails”, “Warn me if authentication fails” and “Don’t connect if authentication fails”.



Microsoft is making security improvements to Remote Desktop. It is also adding a lot

of functionality which means opportunity for bugs and vulnerabilities.

## 6. Traffic Captures

Looking at traffic captures for different connection methods demonstrates the different ports and protocols used. These captures show what an attacker would see if they were able to sniff traffic between the client and the server.

The data in Figure 2 shows a traffic capture for a normal RDP connection with no additional security. In this capture, the client (192.168.1.100) is connecting from port 1234 to the server (192.168.1.103) on the custom port of 50101.

No.	Time	Source	Destination	Protocol	Info
17	6.736053	192.168.1.100	192.168.1.103	TCP	1234 > 50101 [SYN] Seq=0 Len=0 MSS=1260
18	6.736213	192.168.1.103	192.168.1.100	TCP	50101 > 1234 [SYN, ACK] Seq=0 Ack=1 win=64240 Len=0 MSS=1460
19	6.736518	192.168.1.100	192.168.1.103	TCP	1234 > 50101 [ACK] Seq=1 Ack=1 win=65535 Len=0
20	6.736956	192.168.1.100	192.168.1.103	TCP	[TCP segment of a reassembled PDU]
21	6.739076	192.168.1.103	192.168.1.100	TCP	[TCP segment of a reassembled PDU]
22	6.739588	192.168.1.100	192.168.1.103	TCP	1234 > 50101 [ACK] Seq=38 Ack=12 win=65524 Len=0
23	6.740011	192.168.1.100	192.168.1.103	TCP	[TCP segment of a reassembled PDU]
24	6.734364	192.168.1.103	192.168.1.100	TCP	[TCP segment of a reassembled PDU]
25	6.741395	192.168.1.100	192.168.1.103	TCP	1234 > 50101 [ACK] Seq=450 Ack=349 win=65187 Len=0
26	6.741412	192.168.1.100	192.168.1.103	TCP	[TCP segment of a reassembled PDU]
27	6.741419	192.168.1.100	192.168.1.103	TCP	[TCP segment of a reassembled PDU]
28	6.741559	192.168.1.103	192.168.1.100	TCP	50101 > 1234 [ACK] Seq=349 Ack=470 win=63771 Len=0
29	6.741654	192.168.1.103	192.168.1.100	TCP	[TCP segment of a reassembled PDU]
30	6.741822	192.168.1.100	192.168.1.103	TCP	1234 > 50101 [ACK] Seq=470 Ack=360 win=65176 Len=0
31	6.742246	192.168.1.100	192.168.1.103	TCP	[TCP segment of a reassembled PDU]
32	6.742329	192.168.1.103	192.168.1.100	TCP	[TCP segment of a reassembled PDU]
33	6.742674	192.168.1.100	192.168.1.103	TCP	1234 > 50101 [ACK] Seq=482 Ack=375 win=65161 Len=0
34	6.742684	192.168.1.100	192.168.1.103	TCP	[TCP segment of a reassembled PDU]
35	6.742754	192.168.1.103	192.168.1.100	TCP	[TCP segment of a reassembled PDU]
36	6.743104	192.168.1.100	192.168.1.103	TCP	1234 > 50101 [ACK] Seq=494 Ack=390 win=65146 Len=0
37	6.743113	192.168.1.100	192.168.1.103	TCP	[TCP segment of a reassembled PDU]
38	6.743179	192.168.1.103	192.168.1.100	TCP	[TCP segment of a reassembled PDU]

Figure 2

The data in Figure 3 shows a traffic capture for an RDP connection using IPsec with

NAT-T. In this capture, the client (192.168.1.100) is connecting to the server (172.16.1.10) using IPsec. The ISAKMP protocol is used to negotiate the tunnel parameters and then encrypted traffic flows via the ESP protocol. The bottom pane shows a source port of UDP 4500 and a destination port of UDP 4500. This is due to the packets being encapsulated in UDP to traverse the NAT per the NAT-T standard.

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.1.100	172.16.1.10	ISAKMP	Identity Protection (Main Mode)
2	0.036117	172.16.1.10	192.168.1.100	ISAKMP	Identity Protection (Main Mode)
3	0.073056	192.168.1.100	172.16.1.10	ISAKMP	Identity Protection (Main Mode)
4	0.163473	172.16.1.10	192.168.1.100	ISAKMP	Identity Protection (Main Mode)
5	0.176906	192.168.1.100	172.16.1.10	ISAKMP	Identity Protection (Main Mode)
6	0.192002	172.16.1.10	192.168.1.100	ISAKMP	Identity Protection (Main Mode)
7	0.194290	192.168.1.100	172.16.1.10	ISAKMP	Quick Mode
8	0.209388	172.16.1.10	192.168.1.100	ISAKMP	Quick Mode
9	0.209963	192.168.1.100	172.16.1.10	ISAKMP	Quick Mode
10	0.211027	172.16.1.10	192.168.1.100	ISAKMP	Quick Mode
11	0.211210	192.168.1.100	172.16.1.10	ESP	ESP (SPI=0xab9fc471)
12	0.211761	172.16.1.10	192.168.1.100	ESP	ESP (SPI=0x3c9946f6)
13	0.211868	192.168.1.100	172.16.1.10	ESP	ESP (SPI=0xab9fc471)
14	0.212226	172.16.1.10	192.168.1.100	ESP	ESP (SPI=0x3c9946f6)

Frame 11 (102 bytes on wire, 102 bytes captured)

- Ethernet II, Src: GigaFast\_05:d9:14 (00:90:47:05:d9:14), Dst: Cisco-Li\_b1:e4:0e (00:1c:11:00:00:00)
- Internet Protocol, Src: 192.168.1.100 (192.168.1.100), Dst: 172.16.1.10 (172.16.1.10)
- User Datagram Protocol, Src Port: 4500 (4500), Dst Port: 4500 (4500)
- UDP Encapsulation of IPsec Packets
- Encapsulating Security Payload

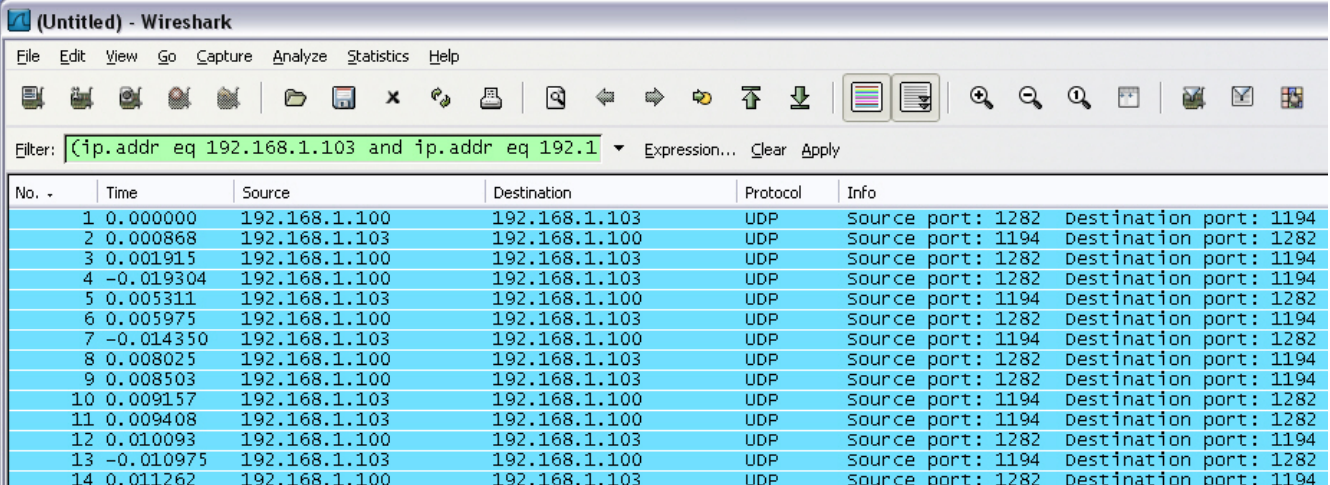
Figure 3

The data in Figure 4 shows a traffic capture for an RDP connection using OpenVPN.

In this capture, the client (192.168.1.100) is connecting from port 1282 to the server

(192.168.1.103) on the UDP port 1194. This is the standard port for OpenVPN connections.

All of the traffic is encrypted and transported over UDP.



The image shows a Wireshark packet capture window titled "(Untitled) - Wireshark". The filter bar contains the expression "(ip.addr eq 192.168.1.103 and ip.addr eq 192.168.1.100)". The packet list shows 14 packets, all of which are UDP. The source and destination IP addresses are 192.168.1.100 and 192.168.1.103. The source and destination ports are 1282 and 1194. The packet details pane shows the source and destination ports for each packet.

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.1.100	192.168.1.103	UDP	Source port: 1282 Destination port: 1194
2	0.000868	192.168.1.103	192.168.1.100	UDP	Source port: 1194 Destination port: 1282
3	0.001915	192.168.1.100	192.168.1.103	UDP	Source port: 1282 Destination port: 1194
4	-0.019304	192.168.1.100	192.168.1.103	UDP	Source port: 1282 Destination port: 1194
5	0.005311	192.168.1.103	192.168.1.100	UDP	Source port: 1194 Destination port: 1282
6	0.005975	192.168.1.100	192.168.1.103	UDP	Source port: 1282 Destination port: 1194
7	-0.014350	192.168.1.103	192.168.1.100	UDP	Source port: 1194 Destination port: 1282
8	0.008025	192.168.1.100	192.168.1.103	UDP	Source port: 1282 Destination port: 1194
9	0.008503	192.168.1.100	192.168.1.103	UDP	Source port: 1282 Destination port: 1194
10	0.009157	192.168.1.103	192.168.1.100	UDP	Source port: 1194 Destination port: 1282
11	0.009408	192.168.1.103	192.168.1.100	UDP	Source port: 1194 Destination port: 1282
12	0.010093	192.168.1.100	192.168.1.103	UDP	Source port: 1282 Destination port: 1194
13	-0.010975	192.168.1.103	192.168.1.100	UDP	Source port: 1194 Destination port: 1282
14	0.011262	192.168.1.100	192.168.1.103	UDP	Source port: 1282 Destination port: 1194

Figure 4

## 7. Summary

This scenario started with NPO's goals to know the threats from allowing Remote Desktop access over the internet and identify possible mitigation techniques. Several different threats and mitigation techniques were analyzed.

Some of the mitigation techniques did not meet the requirements or were insufficient. SecureRDP has some interesting capabilities, but it does not provide features to meet NPO's requirements. "TLS Authentication" provides server authentication, but it does not provide

any features to stop unwanted RDP connections. These two techniques were dropped from consideration.

A number of mitigation techniques will help reduce the risk to accessing Remote Desktop over the internet. The following actions are recommended.

- Implement the Policies and Procedures described in Section 4.1. These steps are easy to implement and will help to reduce risk.
- Implement the Windows Server Configuration changes described in Section 4.2. These steps are also easy to implement and will help to reduce risk.
- Implement Host Based VPN. There are two viable options considered in this paper, IPSec and OpenVPN. Given the small size of the organization, IPSec is recommended. It has a simple implementation and does not require the use of a different IP address for connection. OpenVPN is also an acceptable option. It may be a better fit for larger organizations or if support for Linux clients is required.

Implementing these recommendations will significantly reduce the risk NPO faces using Remote Desktop over the internet.

## 8. References

- Aitel, Dave (2007). SPIKE. Retrieved November 22, 2007, from immunitysec.com Web site: <http://www.immunitysec.com/resources-freesoftware.shtml>
- Cohen, B (2002, September 16). Microsoft Windows XP Professional Remote Desktop Denial Of Service Vulnerability. Retrieved August 21, 2007, from securityfocus.com Web site: <http://www.securityfocus.com/bid/5713/info>
- Cohen, B (2002, September 16). Microsoft Windows Encrypted RDP Packet Information Leakage Vulnerability. Retrieved August 21, 2007, from securityfocus.com Web site: <http://www.securityfocus.com/bid/5711/info>
- Cohen, B (2002, September 16). Microsoft Windows RDP Keystroke Injection Vulnerability. Retrieved August 21, 2007, from securityfocus.com Web site: <http://www.securityfocus.com/bid/5712/info>
- Ferris, T (2005, August 9). Microsoft Windows RDP 'rdpwd.sys' Remote Kernel DoS. Retrieved November 22, 2007, from security-protocols.com Web site: <http://security-protocols.com/sp-x16-advisory.php>
- Forsberg, E (2003, April 1). Microsoft Terminal Services vulnerable to MITM-attacks.. Retrieved August 12, 2007, from securityfocus.com Web site: <http://www.securityfocus.com/archive/1/317244>
- Gates, C (2007, January 4). Tutorial: MS Terminal Server Cracking. Retrieved November 17, 2007, from ethicalhacker.net Web site: <http://www.ethicalhacker.net/content/view/106/24/>
- Kiaer, M (2006, November 1). How to secure remote desktop connections using TLS/SSL based authentication. Retrieved September 2, 2007, from WindowsSecurity.com Web site: <http://www.windowsecurity.com/articles/Secure-remote-desktop-connections-TLS-SSL-based-authentication.html>
- Martins, L (2001, October 18). Microsoft Windows 2000/NT Terminal Server Service RDP DoS Vulnerability. Retrieved November 17, 2007, from securityfocus.com Web site: <http://www.securityfocus.com/bid/3445/info>
- Microsoft-816521, (2007, February 28). HOW TO: Use IPSec Policy to Secure Terminal Services Communications in Windows Server 2003. Retrieved September 23, 2007, from microsoft.com Web site: <http://support.microsoft.com/kb/816521>
- Microsoft-49ae, (2004, August 10). Windows XP Service Pack 2 Support Tools. Retrieved September 24, 2007, from microsoft.com Web site: <http://www.microsoft.com/downloads/details.aspx?FamilyID=49ae8576-9bb9-4126-9761-ba8011fabf38&displaylang=en>



- Microsoft-818043, (2006, October 26). L2TP/IPsec NAT-T update for Windows XP and Windows 2000. Retrieved September 23, 2007, from microsoft.com Web site: <http://support.microsoft.com/kb/818043>
- Microsoft-ipseccmd, (2007). Ipseccmd. Retrieved September 29, 2007, from microsoft.com Web site: <http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/ipsecmd.mspix?mfr=true>
- Microsoft-Bb742429, (2000, February 17). Step-by-Step Guide to Internet Protocol Security (IPSec). Retrieved September 29, 2007, from microsoft.com Web site: <http://technet.microsoft.com/en-us/library/Bb742429.aspx> ZZZ not ref
- Microsoft-816514, (2006, October 30). How To Configure IPsec Tunneling in Windows Server 2003. Retrieved September 29, 2007, from microsoft.com Web site: <http://support.microsoft.com/kb/816514> ZZZ not ref
- Microsoft-bb45, (2005, November 3). Configuring Remote Desktop. Retrieved November 3, 2007, from microsoft.com Web site: <http://technet.microsoft.com/en-us/library/bb457106.aspx> ZZZ not ref
- Microsoft-925876, (2007, October 11). Remote Desktop Connection (Terminal Services Client 6.0). Retrieved November 3, 2007, from microsoft.com Web site: <http://support.microsoft.com/?kbid=925876>
- Microsoft-278845, (2007, February 28). How to Connect to and Shadow the Console Session with Windows Server 2003 Terminal Services. Retrieved November 3, 2007, from microsoft.com Web site: <http://support.microsoft.com/kb/278845>
- Microsoft-306759, (2007, January 31). How to change the listening port for Remote Desktop. Retrieved November 3, 2007, from microsoft.com Web site: <http://support.microsoft.com/kb/306759>
- Microsoft-2230, (2005, January 21). Accounts: Rename administrator account. Retrieved November 3, 2007, from microsoft.com Web site: <http://technet2.microsoft.com/windowsserver/en/library/2230ece2-b4f9-4dc9-b08f-7d29338c374b1033.mspix?mfr=true>
- Microsoft-186607, (2007, March 27). Understanding the Remote Desktop Protocol (RDP). Retrieved November 11, 2007, from microsoft.com Web site: <http://support.microsoft.com/kb/186607>
- Microsoft-92586, (2007, October 11). Remote Desktop Connection (Terminal Services Client 6.0). Retrieved December 3, 2007, from microsoft.com Web site: <http://support.microsoft.com/kb/925876>
- Microsoft-aa383, (2007, July 20). Remote Desktop Protocol (RDP). Retrieved August 24, 2007, from Microsoft Developer Network Web site: <http://msdn2.microsoft.com/en-us/library/aa383015.aspx>
- Microsoft-netsh, (2005, January 21). Netsh commands for Internet Protocol security. Retrieved December 3, 2007, from microsoft.com Web site:

<http://technet2.microsoft.com/windowsserver/en/library/c3ae0d03-f18f-40ac-ad33-c0d443d5ed901033.msp?mfr=true>

- Microsoft-techts, (2005, January). Technical Overview of Terminal Services. Retrieved November 11, 2007, from microsoft.com Web site: <http://download.microsoft.com/download/7/b/3/7b3aa957-4865-427d-9650-789179a5d666/TerminalServerOverview.doc>
- Microsoft-MS01-006, (2001, January 31). Invalid RDP Data can cause Terminal Server Failure. Retrieved November 17, 2007, from microsoft.com Web site: <http://www.microsoft.com/technet/security/Bulletin/MS01-006.msp>
- Microsoft-MS02-051, (2002, September 18). Microsoft Security Bulletin MS02-051. Retrieved November 16, 2007, from microsoft.com Web site: <http://www.microsoft.com/technet/security/bulletin/MS02-051.msp>
- Microsoft-MS05-041, (2005, August 9). Microsoft Security Bulletin MS05-041. Retrieved November 16, 2007, from microsoft.com Web site: <http://www.microsoft.com/technet/security/Bulletin/MS05-041.msp>
- Microsoft-MS01-052, (2001, October 18). Microsoft Security Bulletin MS01-052. Retrieved November 17, 2007, from microsoft.com Web site: <http://www.microsoft.com/technet/security/bulletin/MS01-052.msp>
- Microsoft-ra, (2003 March 24). Remote Administration of Windows Servers Using Remote Desktop for Administration. Retrieved November 16, 2007, from microsoft.com Web site: <http://www.microsoft.com/windowsserver2003/techinfo/overview/tsremoteadmin.msp>
- Microsoft-ipsecfaq, (2006, February 13). IPSec : Frequently Asked Questions. Retrieved October 18, 2007, from microsoft.com Web site: <http://www.microsoft.com/technet/network/ipsec/ipsecfaq.msp>
- Montoro, Massimiliano (2005, May 28). Remote Desktop Protocol, the Good the Bad and the Ugly. Retrieved August 12, 2007
- SecureRDP, (2007, July 28). Secure RDP of Windows Terminal Services with 2X SecureRDP. Retrieved September 15, 2007, from 2x.com Web site: <http://www.2x.com/securerdp/windows-terminal-services.html>
- Wikipedia-OpenVPN, (2006, December). OpenVPN. Retrieved September 6, 2007, from wikipedia.org Web site: <http://en.wikipedia.org/wiki/OpenVPN>
- Wikipedia-RDP, (2007, November 07). Remote Desktop Protocol. Retrieved November 11, 2007, from Wikipedia Web site: [http://en.wikipedia.org/wiki/Remote\\_Desktop\\_Protocol](http://en.wikipedia.org/wiki/Remote_Desktop_Protocol)
- Yonan , J (2003). Understanding the User-Space VPN: History, Conceptual Foundations, and Practical Usage. Retrieved September 6, 2007, from openvpn.net Web site: <http://openvpn.net/papers/BLUG-talk/>