Global Information Assurance Certification Paper

Taufiq Chilwan
GSEC Practical req v1.4b option 1 (Feb 2002)

Increasing Terminal Services Security as Remote Management of Windows 2000 Server

Abstract
Windows 2000 Server family can be administered remotely using built in feature of Terminal Services in remote administration mode. Although terminal services offer several level of encryption to secure the traffic between client and server, however, terminal services only offer user/domain authentication to access the server. There is no obvious way to limit access to terminal services login information, so once the terminal services is enabled on server, every one can access login page windows 2000 server using terminal services.
This paper tries to develop a solution to increase terminal services security with no additional cost. To achieve this result, windows terminal services will be used, combine with free software utility, stunnel and openssl. A filter list and filter action on local security setting of windows 2000 server will be created to block default terminal services communication, to limit access of terminal services. Brief explanation how to configure and use this software to increase the security of terminal services and how it affect terminal services will be given.

Introduction

This paper will not discuss about how to configure and set the default security when installing terminal services. Information on this subject could be found at the following address:
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/win2kts/maintain/optimize/secw2kts.asp and also
http://www.nsa.gov/snac/win2k/download.htm

Terminal Services in Windows 2000 Server family could become the easiest way to perform desktop management remotely, because it comes as a built in feature of Windows 2000 Server. It is easy to setup, uses built in windows/domain account for authentication, and also offer different encryption level as standard security requirement. But Terminal Services also have some limitation. It doesn't offer the ability to limit where the connection may originate, so once the terminal services is enabled on server, everyone will have access to login page of windows 2000 server. Another weakness is Terminal services also do not offer an option to prove the identity of client/computer that is allowed to access the login page of windows 2000 server.

Based on those limitations, this paper tries to develop a solution that might overcome those problems and gives a terminal services as a total remote desktop solution for managing windows 2000 server.

Define the Solution

- Windows Terminal Services.

In this paper, we will focus Terminal Services term as a component of Windows 2000 Server family, Windows 2000 Server, Windows 2000 Advanced Server, and Windows 2000 Data Center Server. We are not going to discuss Terminal Services as a feature of Windows NT4. For the implementation, the configuration that is presented in this paper may also be applied to the new coming Windows Server 2003.

- Local Security Policy

Terminal services can be activated either as an application server mode or as a remote administration mode. When configured as an application server, everyone need to have access to login page of your server, so they can login and have any application installed on server as if it is installed on their local pc. But when enabling terminal services as a remote administration mode, the goal is only to remote and perform an administrative task of maintaining your server by authorized people. You still need to keep access to your server limited.
Some security papers write the needs of securing your server physically. One important goal by doing this is to limit access to your server only by authorized person. When enabling Windows Terminal Services as remote administration, this security point is actually compromises, because now anyone can have access to your server login screen, leaving a considerable risk especially for an internal intruder.
Terminal services does not offer an option to limit where the connection may originate when enabled in remote administration mode. So we will create a rule using local security policy of windows 2000 server to block all default terminal services coming to server. By utilizing this feature, only the designed computer (have certificate on their computer), with authorized people (have their user name and password), and stunnel installed, able to initiate terminal services connection, while connection from others computers will be blocked.

- Stunnel

Stunnel is a program that allows you to encrypt arbitrary TCP connections inside SSL (Secure Sockets Layer) available on both Unix and Windows. Stunnel main purpose is to give the ability to create SSL tunnels to transmit terminal services communication. It assures the confidentiality and integrity of the transmitted data. Besides that, Stunnel also offer the opportunity to authenticate terminal services client and the remote server by using certificates.

- OpenSSL

OpenSSL is a library that can provide cryptographic functionality to application. OpenSSL will be used to create, sign, and revoke certificate that will be used by terminal services server and client to authenticate each other.

The following figure will briefly summarize what the connection will look like.

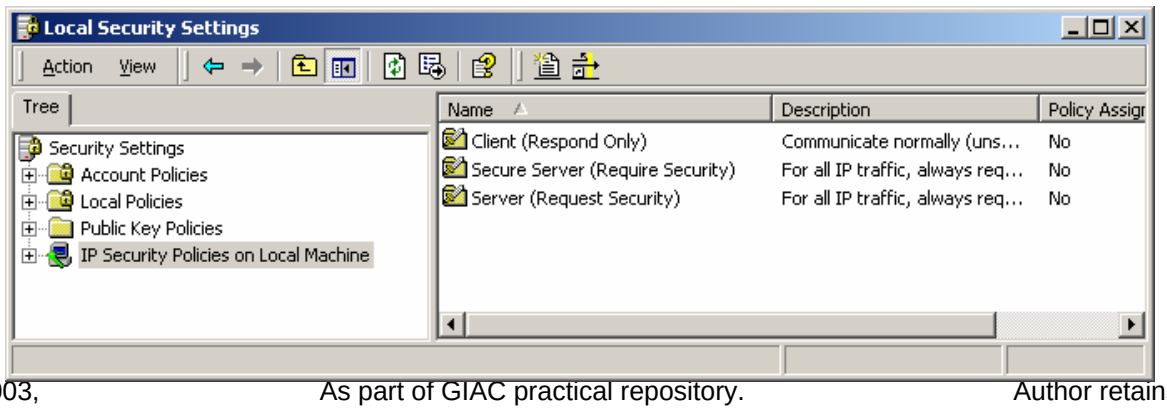
Implementation

1. Installing Terminal Services.

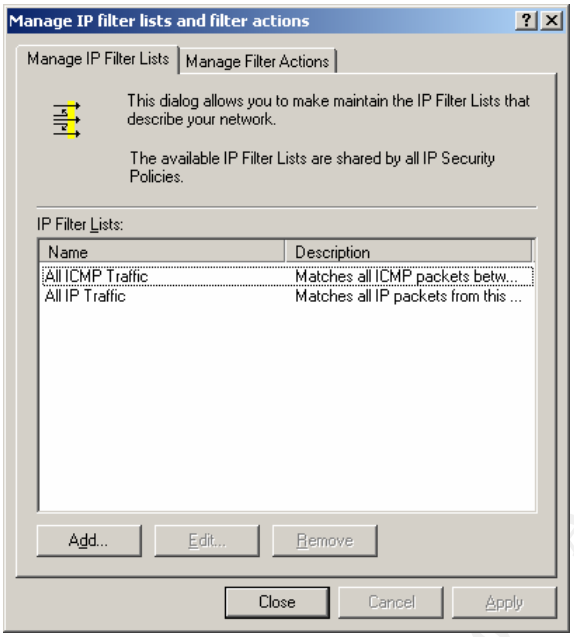
Terminal Services has two modes of installation, remote administration, and application server. This paper will need to install the terminal services in remote administration mode. Since Terminal services comes as a built in feature of Windows 2000 Server, the installation can be done easily. National Security Agencies (NSA) Network Security Evaluation and Tools Division of National Security Agencies (NSA) gives the detail installation step and standard security configuration on the following link: http://www.nsa.gov/snac/win2k/download.htm

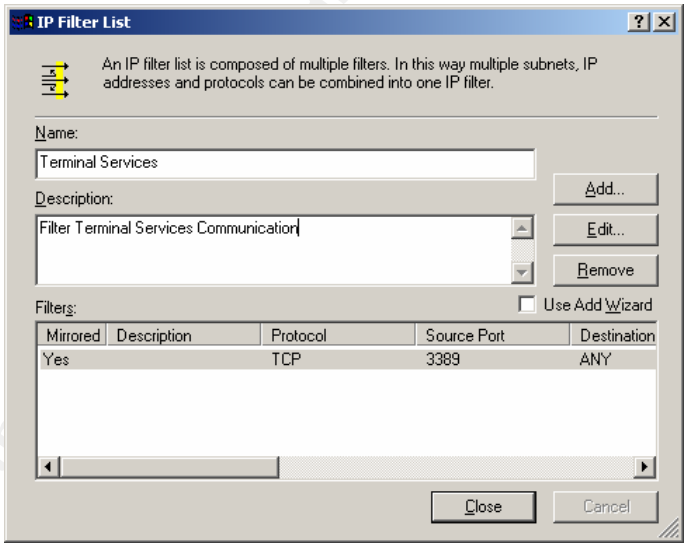
2. Limiting the connection from localhost only.

In the proposed configuration, as a security requirement, Terminal services should only be accessible by locally installed stunnel utility, so it is important to block all default terminal services communication on port 3389/tcp coming to or from windows 2000 server. On windows 2000 Server, this goal can be achieved by using local security policy, and creating filter access list.

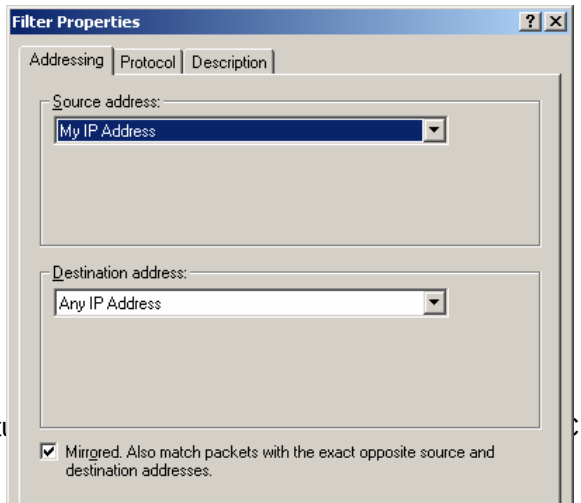


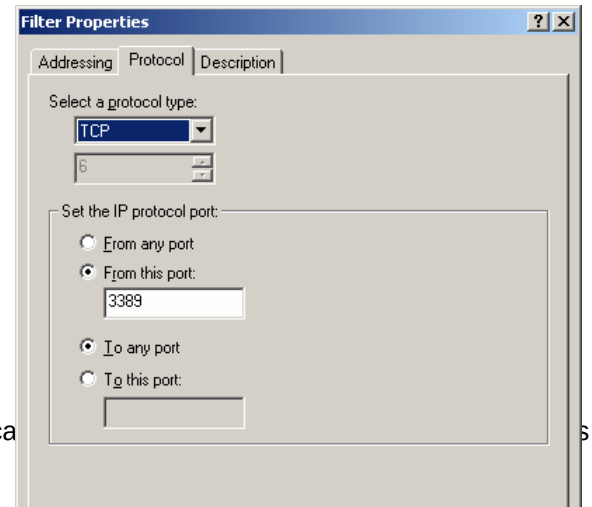
On **Local Security Settings** window, expand **Security Settings**, Right Click **IP Security Policies on Local Machine**, and choose **Manage IP filter lists and filter actions**.



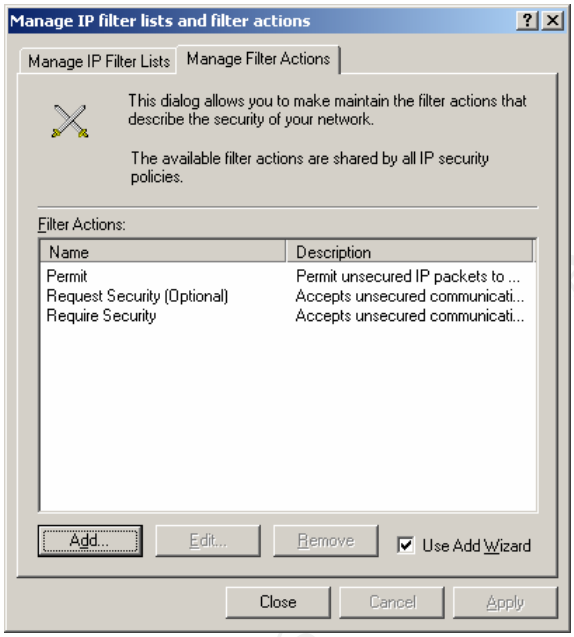


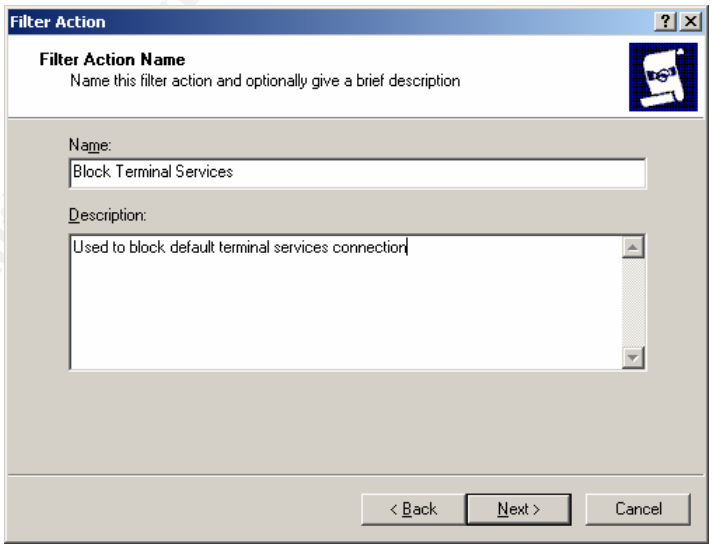
Click **Add** to create new IP filter lists. Type *Terminal Services* for IP filter list name. Clear **Use Add Wizard**, and click **Add**.
In the **Addressing** tab of **Filter properties**, choose **My IP address** for Source Address, and **Any IP address** as Destination address.
Click **Protocol** tab of **Filter properties** and choose **TCP** as a protocol type. Type **3389** in the **From this port** box, and choose **to any port**. Click **OK** to close the window.



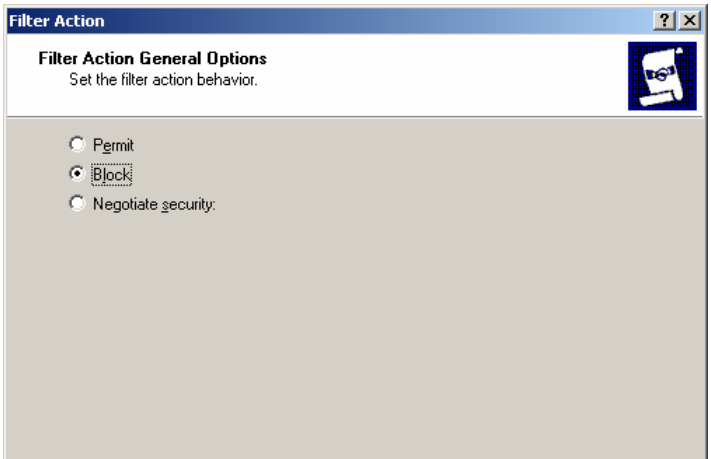


The next step is to create a new filter action. Click **Manage Filter Action** tab at the **Manage IP filter lists and filter actions** window, and click **add** to add new filter action.

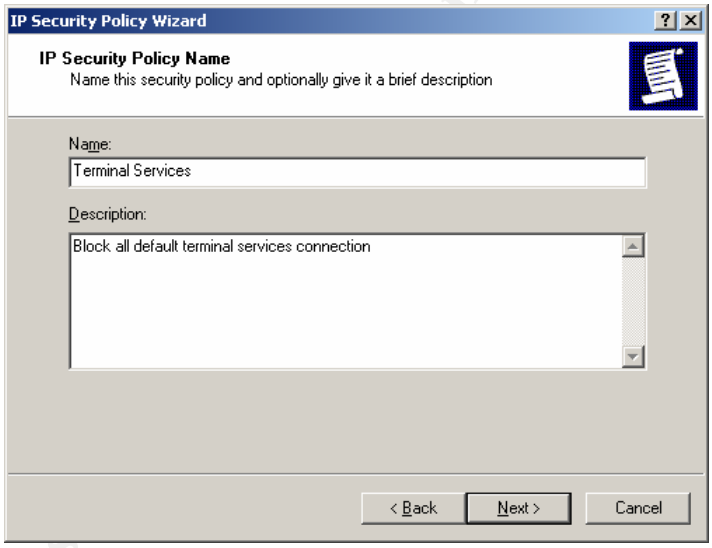




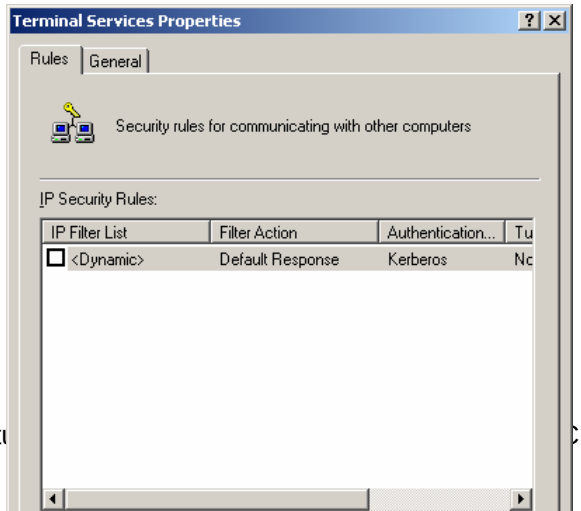
Type `Block Terminal Services` as the new filter action name, press **next** and choose **Block** for the **Filter Action General Options**. Press **Nex**t and close all window.

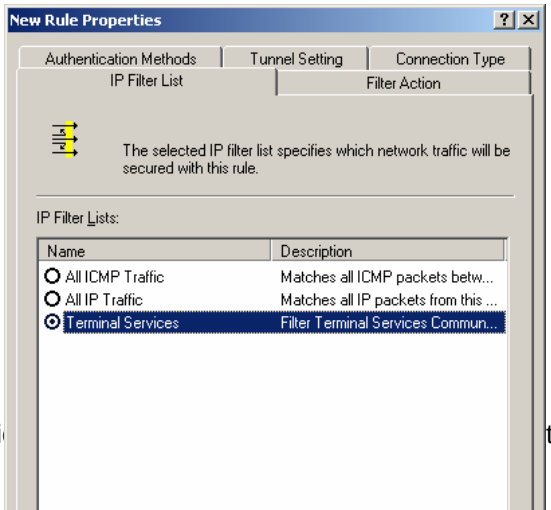


After define the terminal services filter list and filter action, now we need to create new IP Security policy that will be used to block all default terminal services connection. At the **Local Security Settings**, right click **IP Security Policies on Local Machine**, and choose **Create IP Security Policies**. Give the new policies name `Terminal Services` and put a short description.

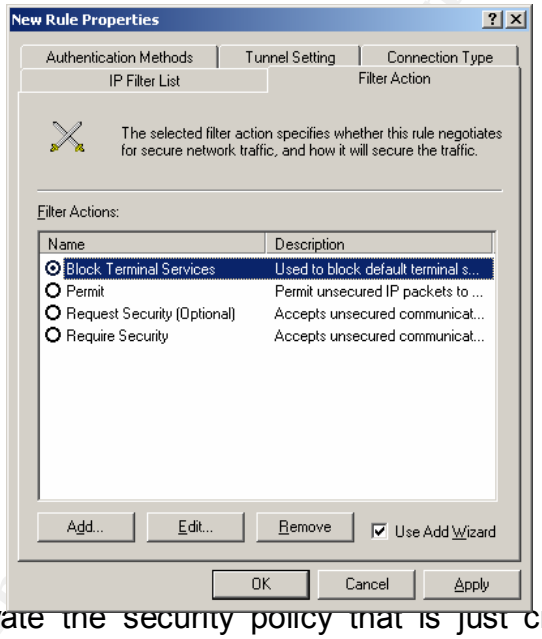


Press **Next**. Clear the **Activate the default response rule** check box, press **next** again, and now tick the **Edit properties** check box. Then click on **Finish**.





Click on **Add** at the Terminal Services policy security properties, and choose **Terminal Services** in the IP Filter lists. At the **Filter Action** tab, choose **Block Terminal Services**. Press **OK** and close all windows until we get back to **Local Security Settings** window.



Last Step is to activate the security policy that is just created. At the **Local Security Settings** window, right click new **Terminal Services** policy, and choose **assign**.
After this policy is assigned, all default terminal services connection coming to server will be blocked. Now terminal services can only be accessed by stunnel utility, so it runs as if the connection is coming only from localhost.

3. Installing Stunnel

Stunnel utility can be download from stunnel website (http://www.stunnel.org). This application must be installed both on server and client so they can communicate each other. Here I installed it under C:\Program Files\Stunnel.
In order to work, another library files, libeay32.dll, libssl32.dll must also be downloaded and place on the same folder.

4. Installing Open SSL

OpenSSL will be used to generate, sign, and revoke certificate. Because of its important role, it's recommended to install OpenSSL on a stand-alone computer, not connected from the network. A binary windows version of OpenSSL can also be downloaded from stunnel website. Two others library files, libeay32.dll and libssl32.dll, should also be downloaded and installed on the same folder.
To generate certificates, the following configuration files should also be downloaded, openssl.conf (http://www.securityfocus.com/data/tools/openssl.conf) and the ca.bat script (http://www.securityfocus.com/data/tools/ca.bat), which will be modified and used to easily generate certificates. The above files should be placed in C:\Program Files\OpenSSL directory.

5. Configure the application

After all the software is installed, the first thing that needs to do is to create a certificate both for client and server, so they can authenticate each other.

Certificate Authorities

The process of generating certificates should be started by generating a private/public key pair and certificate for the trusted third party, or CA (Certification Authority). The CA's private key will be used later to sign the terminal services server's and client's certificates. The CA's certificate will be placed on all terminal services servers and clients. Because the CA's private key is one of the most important elements of every PKI implementation, the key must be protected by strong pass phrase and kept away from regular users.

The public/private key pair of CA can be generated by using the modified ca.bat file that we already download before, in the following manner:

```
C:\Program Files\OpenSSL\ca genca
```

After performing the above steps, the CA's certificate will be stored in the C:\CA\CAcert.pem file, and the private/public key pair will be stored in the C:\CA\private\CAkey.pem file.
PEM is stand for privacy enhanced mail which is now liberally used as a key format.

Create Server Certificate

Generating server certificate can be done by executing the modified ca.bat in the following manner:

```
C:\Program Files\OpenSSL\ca server
```

Following the script and giving the information needed, will result the following files are created in the C:\CA\Temp\ts_server directory.

- Server.key -- private/public key pair
- Server.crt – server certificate
- Server.pem – server.key + server.crt (used by stunnel)

Create Client Certificate
Client certificate can be obtained by executing the modified ca.bat file in the following manner:

C:\Program Files\OpenSSL\ca client

Similar when generating the server certificate, completing this step will result the following files are created in the C:\CA\Temp\ts_client directory:

- client.key -- private/public key pair
- client.crt – client certificate
- client.pem – client.key + client.crt (used by stunnel)

Configure Stunnel
Stunnel must be configured first before it can be used to secure communication between server and client, and authenticate each of them using certificates.

Stunnel on Server
Stunnel should be configured on server by creating file stunnel.conf at the C:\Program Files\Stunnel folder, with the following content:

```
CAfile = CAcert.pem
CApath = certificates
cert = server.pem
client = no
verify = 3

[TS]
accept = 443
connect = 127.0.0.1:3389
```

The default configuration of stunnel port used here is port 443/tcp. This value can be changed with other value. The above configuration will cause all incoming connections to the 443/tcp port to be forwarded to the local port 3389/tcp. This should only happen after the client proves his identity by showing a valid, signed certificate, which also present in the local certificates directory ("verify = 3" require and verify certificate authentication of both sides against locally installed certificates).

The next step is to place both the CA's certificate (C:\CA\CAcert.pem) and terminal services server's private/public key pair and certificate (C:\CA\temp\ts_server\server.pem) in the C:\Program Files\Stunnel directory.
Finally we must also load the terminal services client's certificate. In order for the Stunnel utility to find the certificate during the authentication process, we must change its name as follows (the following commands must be run on the server on which the certificates was generated; the *value* should be replaced by the result of the "openssl x509" command):

```
cd C:\CA\temp\ts_client
C:\progra~1\openssl\openssl x509 -hash -noout -in client.crt
value
copy client.crt value.0
```

The file *value*.0 should be placed in the C:\Program Files\Stunnel\certificates directory

Stunnel on Client
Similar to server, stunnel configuration file stunnel.conf should be created on C:\Program Files\Stunnel folder on client, with the following content:

```
CAfile = CAcert.pem
CApath = certificates
cert = client.pem
client = yes
verify = 3

[TS]
accept = 127.0.0.1:3389
connect = Terminal_Services_Server_IP_address:443
```

The next step is to store both the CA's Certificate (C:\CA\CAcert.pem) and terminal services client's private/public key pair and certificate (C:\CA\temp\ts_client\client.pem) in the C:\Program Files\Stunnel directory.
Finally, we must change the name of the terminal services server's certificate file in the way as follows:

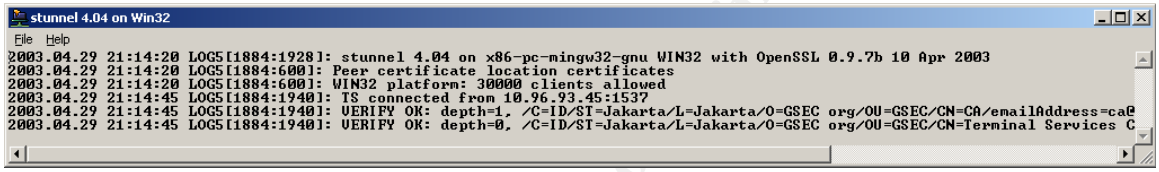
```
cd C:\CA\temp\ts_server
C:\progra~1\openssl\openssl x509 -hash -noout -in server.crt
value
copy server.crt value.0
```

The file *value*.0 should be placed in the C:\Program Files\Stunnel\certificates directory

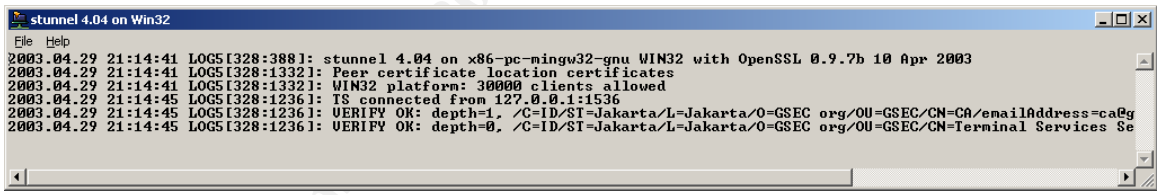
Installing Terminal Service Client
Client may start terminal services by either using small application software, created by windows 2000 server terminal services client creator, or by using Terminal Services advanced client using Microsoft Internet Explorer 5.0 or higher.

6. Testing the application
Once all configurations have been completed, terminal services can be tested from client either using client software or using internet explorer. As terminal services address, type 127.0.0.1. If everything is setup correctly, the terminal services login page should be appear, which mean, the connection is already established. On the stunnel window, the following information will be showed:

On the stunnel server:



On the stunnel client:



Summary
Right now there are a lot of option for remote desktop management of Windows server. But, sometime, not all of them give a reliable security that can be count on, or if they give, sometime it will cost you with an extra additional budget. Terminal Services as a built in feature of Windows 2000 Server family, can be configured and made reliable secure with additional free software utility, such as Stunnel and OpenSSL, to give the administrator more than enough security need. Utilizing this solution will offer extra security because now you have more option to limit where the connection may originate and the ability of using certificate to prove the identity of the client.

Appendix A
Modified ca.bat file

Reference:

1. Remote Desktop Management Solution for Microsoft
http://www.securityfocus.com/infocus/1677

2.

Last Updated: July 27th, 2017

Upcoming Training

SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
Community SANS Omaha SEC401*	Omaha, NE	Aug 14, 2017 - Aug 19, 2017	Community SANS
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style	Virginia Beach, VA	Aug 21, 2017 - Aug 26, 2017	vLive
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
Community SANS Pasadena SEC401 @ NASA	Pasadena, CA	Aug 23, 2017 - Aug 30, 2017	Community SANS
Mentor Session - SEC401	Minneapolis, MN	Aug 29, 2017 - Oct 10, 2017	Mentor
SANS San Francisco Fall 2017	San Francisco, CA	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS Tampa - Clearwater 2017	Clearwater, FL	Sep 05, 2017 - Sep 10, 2017	Live Event
Mentor Session - SEC401	Edmonton, AB	Sep 06, 2017 - Oct 18, 2017	Mentor
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
Community SANS Albany SEC401	Albany, NY	Sep 11, 2017 - Sep 16, 2017	Community SANS
Mentor Session - SEC401	Ventura, CA	Sep 11, 2017 - Oct 12, 2017	Mentor
Community SANS Columbia SEC401	Columbia, MD	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS Dallas SEC401	Dallas, TX	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS Boise SEC401	Boise, ID	Sep 25, 2017 - Sep 30, 2017	Community SANS
Baltimore Fall 2017 - SEC401: Security Essentials Bootcamp Style	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
Community SANS New York SEC401	New York, NY	Sep 25, 2017 - Sep 30, 2017	Community SANS
Rocky Mountain Fall 2017	Denver, CO	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Copenhagen 2017	Copenhagen, Denmark	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Sacramento SEC401	Sacramento, CA	Oct 02, 2017 - Oct 07, 2017	Community SANS
SANS DFIR Prague 2017	Prague, Czech Republic	Oct 02, 2017 - Oct 08, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Oct 02, 2017 - Oct 07, 2017	Community SANS
Mentor Session - SEC401	Arlington, VA	Oct 04, 2017 - Nov 15, 2017	Mentor
Community SANS Indianapolis SEC401	Indianapolis, IN	Oct 09, 2017 - Oct 14, 2017	Community SANS