



Fight crime.
Unravel incidents... one byte at a time.

Copyright SANS Institute
Author Retains Full Rights

This paper is from the SANS Computer Forensics and e-Discovery site. Reposting is not permitted without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Advanced Incident Response, Threat Hunting, and Digital Forensics (FOR508)"
at <http://digital-forensics.sans.org><http://digital-forensics.sans.org/events/>

Forensic Analysis of a Windows 95 System
GIAC Practical Assignment
Version 1.0

Completed in partial fulfillment of
GIAC Forensic Analyst Certification (GCFA)

Gregory Leibolt
SANS – Orlando, FL
April 3 – 7, 2002

© SANS Institute 2000 - 2002
Author retains full rights.

Table of Contents

| | |
|--|----|
| Part 1, Option 1 – Perform Forensic Analysis of a System..... | 3 |
| Synopsis of Case Facts..... | 3 |
| System Description, Hardware and Evidence Handling..... | 3 |
| Setting Up the Forensic Toolkit..... | 5 |
| Imaging of the Media and Media Analysis of a System | 7 |
| Image Mounting..... | 8 |
| Autopsy Software Setup..... | 9 |
| Operating System Identification..... | 11 |
| Y2K Compliance..... | 13 |
| MACTime Analysis | 14 |
| Operating System Installation Date..... | 14 |
| The Registry | 17 |
| Interesting Registry Information..... | 18 |
| HKEY_CURRENT_USER/ Entries..... | 18 |
| HKEY_LOCAL_MACHINE/SOFTWARE/ Entries..... | 20 |
| Recover Deleted Files | 20 |
| Interesting Files | 22 |
| String Searches | 28 |
| Conclusions | 28 |
| User Activity | 28 |
| Software | 30 |
| User Files | 31 |
| Viruses | 31 |
| Network..... | 33 |
| Summary | 35 |
| Part 2 – Analyze an Unknown Binary..... | 37 |
| Binary Details Summary | 37 |
| Program Identification..... | 37 |
| Strings Analysis..... | 38 |
| Binary Duplication | 38 |
| Program Description..... | 41 |
| Forensic Details..... | 42 |
| Legal Issues..... | 43 |
| Proof that ADMsniff was run..... | 43 |
| No Proof that ADMsniff was run | 45 |
| Interview Questions | 46 |
| Additional Information..... | 47 |
| Summary | 47 |
| Part 3 – Legal Issues of Incident Handling – Wiretap Statute | 49 |
| List of References | 55 |
| Appendix A: Windows 95 file attribute tests | 56 |
| Appendix B: Y2K Windows 95 Issues | 58 |

Part 1, Option 1 – Perform Forensic Analysis of a System

This section of the document is a step-by-step description of the forensic evaluation of a laptop computer. The proprietary information contained in this “real world” case within a corporation has been sanitized by replacing sensitive information with the letter “X” or a descriptive string such as “girl’s first name.”

Synopsis of Case Facts

The president of an employee club within a large company suspected the treasurer of taking money from the club’s funds. As a result, the president demanded that the treasurer return the club’s laptop on October 12, 2001. The president received the laptop on October 23rd, 2001. At that time, he locked it up securely for several months. In February, the president turned on the laptop and took two actions. He looked at the “start menu” for “RECENT” document listings and started up Excel to look at Excel’s “RECENT” listings. He stated that he took no other actions. He had the computer turned on for about 15 minutes and then shut it down properly. Once again he locked it up securely. In April, 2002 the president shipped the laptop to the forensics team for review. Since the president suspected misuse of funds, he hoped the forensics team would find files related to funds and lists of contributors or sponsors.

System Description, Hardware and Evidence Handling

The system was shipped to the forensic lab via Airborne Express. The sanitized Airborne Express shipping information read as follows:

Airborne Express Tracking Number: XXXXXXXXXXX

Shipment Summary:

Current Status: Delivered.
Delivered on: Apr XX 2002 9:59 am
Delivered to: Left at Shipping Dock
Signed for by: Greg Leibolt

Shipment History:

| Date | Time* | Activity* | Location** |
|-------------|---------|----------------------|-------------|
| Apr XX 2002 | 9:59 am | Delivered. | Orlando, FL |
| | 6:52 am | Arrived at Airborne. | Orlando, FL |

| | | | |
|-------------|---------|------------------------|----------------|
| Apr YY 2002 | 5:00 pm | Left Airborne. | Tempe, AZ |
| | 3:36 pm | Picked up by Airborne. | Shipper's Door |

* status times reflect the time zone where the update took place.

** cities reflect the Airborne terminal servicing the area.

Shipper:

XXXXXX
MESA, AZ 85201

Receiver:

GREG LEIBOLT

ORLANDO, FL 12345

Shipment Detail:

Service: Second day Ship Type: Package
Weight: 18 Description:
Pieces: 1 Shipper's Reference: XXXXXXXXXXX

Tracking request generated on 4/25/02 at 1:21:17 PM.

If you have any further questions, please contact Customer Service at
1-800-AIRBORNE (1-800-247-2676) or write to us.

The exact dates and times are noted for each step performed while handling evidence and making an image of the hard disk. These time frames may be of some value to the reader for planning and preparation purposes.

DATE: 4/24/02

09:59 AM EDT on 4/24/02

The evidence was delivered, signed for and remained in the locked forensics lab while preparations were made to begin analysis.

11:00 AM EDT on 4/24/02

The system was packed in a cardboard box, which was reviewed for signs of tampering. None were found. The box was opened and another taped up box was inside. This box was also reviewed for signs of tampering. None were found. This second box was opened and the contents were removed while wearing latex gloves to limit contamination of the physical evidence.

14:14 PM EDT on 4/24/02

In order to follow proper procedures for evidence handling, the different components were logged as follows:

- Logged 1 Acer Laptop model no. 950C serial no. 1600017384.
- Logged 1 Delta electronics battery/power supply unit S/N A9730005202.
- Logged 1 Nickel Metal Hydride rechargeable Battery S 063124.
- Logged one [EMPTY] floppy drive (plugged into the Laptop).
- Noted 2 [EMPTY] PCMCIA ports.

14:16 PM EDT on 4/24/02

- The hard drive caddy was removed from the laptop by releasing a latch and sliding the unit out.

14:20 PM EDT on 4/24/02

- The hard drive was extracted from the caddy to allow it to be placed in a different caddy for attachment to the forensic system. The following information about the hard drive was recorded from the unit label:
Hard drive: Seagate model ST9816AG, Serial #: 1600017384
810 MB, 1571 CYL, 16 HEADS, 63 SECTORS.

14:22 PM EDT on 4/24/02

- **Established case number 42402.**
- Tagged all evidence components as follows:
Tag # 42402-1
Acer Laptop Model No. 950C Serial #: 160001738.
Tag # 42402-2
Delta Electronics Battery / Power Adapter SN: A9730005202.

Tag # 42402-3

Seagate IDE hard drive, Model: ST9816AG, Serial Number:
1600017384 Size:810 MB.

Tag # 42402-4

Nickel Metal Hydride rechargeable battery, S 063124.

Tag # 42402-5

Floppy Drive (installed in the laptop pictured below).

The following picture shows latex gloves and the different computer components with evidence tags attached:



Setting Up the Forensic Toolkit

14:34 PM EDT on 4/24/02

- The forensic laptop used to image the evidence hard drive was a Toshiba Tecra 8000 with 256 MB RAM and an 18 GIG Hard Drive using RedHat 7.0 operating system with the 2.2.16-22 kernel. The hda11 partition on the forensic laptop was repartitioned and reformatted to correct a partition

problem created by accidentally writing to the superblock using a `dd if=/dev/zero` command. This partition was considered sterile and mounted as `/mnt/image`. The disk image collected from the imaging process was collected in the `/mnt/image` directory.

- The evidence hard disk was connected to the forensic laptop using an EZ-GIG expansion PCMCIA card by Apricorn.
- The evidence hard disk device was identified as `/dev/hde` by looking in the `/proc/partitions` file for newly added partitions.

The following picture shows the forensic laptop connected to the evidence hard disk with the EZ-GIG expansion PCMCIA card by Apricorn:



15:12 PM EDT on 4/24/02

- To ensure that a valid copy was obtained from the evidence hard disk, an MD5 sum was obtained from the original evidence by performing `md5sum /dev/hde1`. The calculation completed at 15:27 PM.
MD5sum=07C33BC04D8A5700495892C5EC555CB8.

15:30 PM EDT on 4/24/02

- The file system type was identified by using the `fdisk -l /dev/hde`. The command reported `/dev/hde1` bootable FAT16, with this one partition using all of the drive.

Output of the `fdisk` command:

Disk `/dev/hde`: 32 heads, 63 sectors, **785** cylinders
Units = cylinders of 2016 * 512 bytes

| Device | Boot | Start | End | Blocks | Id | System |
|------------------------|------|-------|-----|---------|----|--------|
| <code>/dev/hde1</code> | * | 1 | 784 | 790240+ | 6 | FAT16 |

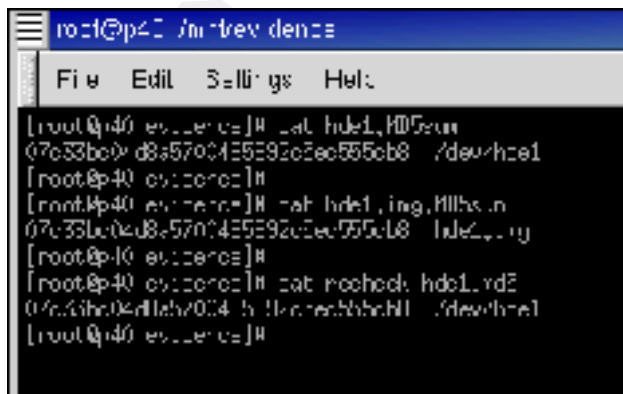
Imaging of the Media and Media Analysis of a System

15:40 PM EDT on 4/24/02

- The `dd` utility is a long established and powerful tool used to do bit by bit copies of media. Its reputation for accuracy and reliability has also been established in the courts. An image of the `/dev/hde1` partition was obtained using the `dd if=/dev/hde1 of=hde1.img` command. This process completed at 15:56 PM.

16:09 PM EDT on 4/24/02

- An MD5 sum of the captured image, `hde1.img` was obtained and compared against the original hard disk image to verify an exact copy. This calculation completed at 16:13 PM.
MD5sum=07C33BC04D8A5700495892C5EC555CB8.
- **It was noted that both of the MD5 sums were the same.**
- A final re-check of the MD5 sum of the original hard disk, `/dev/hde1`, was performed after obtaining a copy of the image to prove that imaging the hard disk did not alter the original in any way. The re-check test completed successfully as shown in the following screen shot:



```
root@p40 ~# cat /dev/hde1,MD5sum
07c33bc04d8a5700495892c5ec555cb8 /dev/hde1
root@p40 evidence1#
root@p40 evidence1# cat hde1.img,MD5sum
07c33bc04d8a5700495892c5ec555cb8 hde1.img
root@p40 evidence1#
root@p40 evidence1# cat recheck_hde1.md5
07c33bc04d8a5700495892c5ec555cb8 /dev/hde1
root@p40 evidence1#
```


Image Mounting

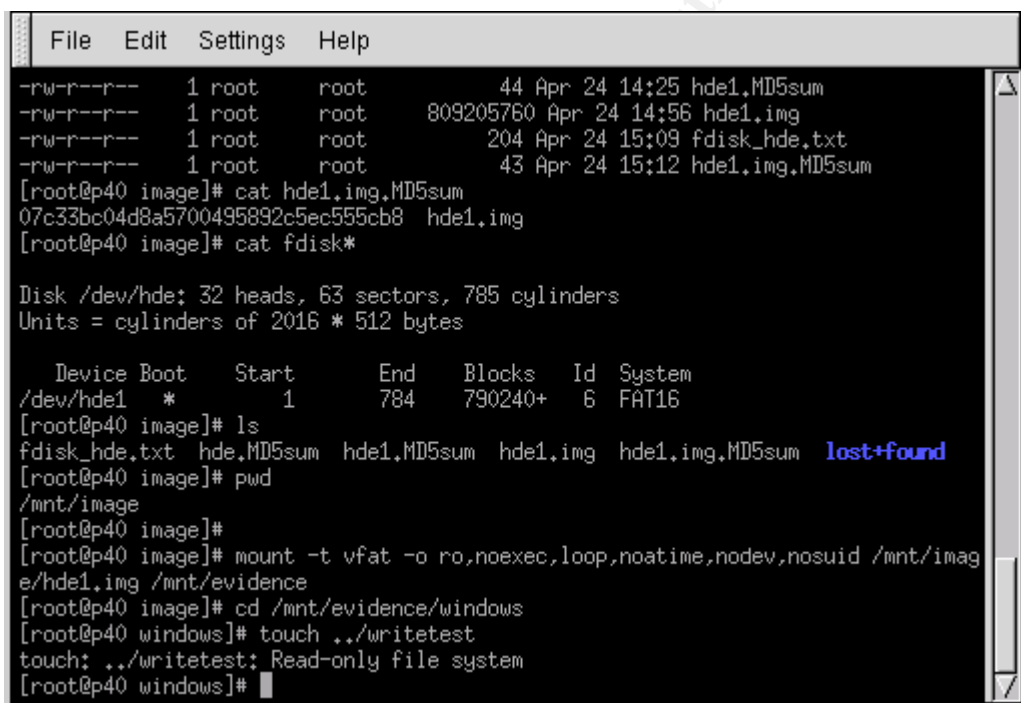
16:39 PM EDT on 4/24/02

- The image, hde1.img was mounted in read-only mode, not allowing execution of binaries, not allowing the update of inode access time, not interpreting character or block special devices and not allowing set-user-identifier or set-group-identifier bits to take effect. These mount options were used to ensure safe access to the evidence hard drive. The command used was:
mount -t vfat -o -ro ,noexec,loop,noatime,nodev,nosuid /mnt/image/hde1.img /mnt/evidence.

16:40 PM EDT on 4/24/02

- A read-only validation test to verify that the hard drive could not be altered in any way was performed by using the touch command to write a file. The test completed successfully.

Fdisk information, mount command and read-only test verification:



```
File Edit Settings Help
-rw-r--r-- 1 root root 44 Apr 24 14:25 hde1.MD5sum
-rw-r--r-- 1 root root 809205760 Apr 24 14:56 hde1.img
-rw-r--r-- 1 root root 204 Apr 24 15:09 fdisk_hde.txt
-rw-r--r-- 1 root root 43 Apr 24 15:12 hde1.img.MD5sum
[root@p40 image]# cat hde1.img.MD5sum
07c33bc04d8a5700495892c5ec555cb8 hde1.img
[root@p40 image]# cat fdisk*

Disk /dev/hde: 32 heads, 63 sectors, 785 cylinders
Units = cylinders of 2016 * 512 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/hde1  *            1           784     790240+    6   FAT16
[root@p40 image]# ls
fdisk_hde.txt hde.MD5sum hde1.MD5sum hde1.img hde1.img.MD5sum lost+found
[root@p40 image]# pwd
/mnt/image
[root@p40 image]#
[root@p40 image]# mount -t vfat -o ro,noexec,loop,noatime,nodev,nosuid /mnt/image/hde1.img /mnt/evidence
[root@p40 image]# cd /mnt/evidence/windows
[root@p40 windows]# touch ../writetest
touch: ../writetest: Read-only file system
[root@p40 windows]#
```

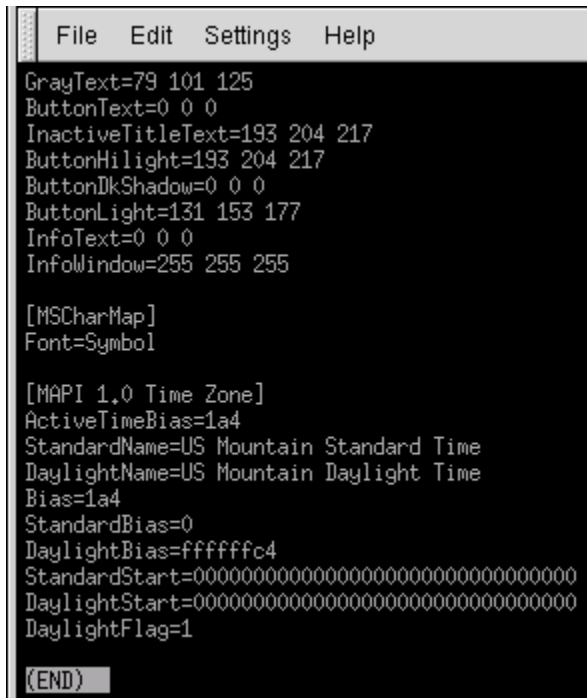
16:45 PM EDT on 4/24/02

- The time zone used on the evidence computer needed to be identified so analysis tools, specifically “mactime,” would report times in the correct time zone. The file windows\win.ini contained the time zone setting as Mountain Time.
- Each Linux window used for analysis work was set to use Mountain Time. The set command was used: *SET TZ=MST7MDT export TZ.*

16:49 PM EDT on 4/24/02

- The forensic laptop was set to the correct Florida date and time using the date command: `date -set '2002-04-24 16:49:00'`.

Time zone verification:



```
File Edit Settings Help
GrayText=79 101 125
ButtonText=0 0 0
InactiveTitleText=193 204 217
ButtonHighlight=193 204 217
ButtonDkShadow=0 0 0
ButtonLight=131 153 177
InfoText=0 0 0
InfoWindow=255 255 255

[MSCharMap]
Font=Symbol

[MAPI 1.0 Time Zone]
ActiveTimeBias=1a4
StandardName=US Mountain Standard Time
DaylightName=US Mountain Daylight Time
Bias=1a4
StandardBias=0
DaylightBias=fffffc4
StandardStart=00000000000000000000000000000000
DaylightStart=00000000000000000000000000000000
DaylightFlag=1

(END)
```

At this point the forensics work stopped for the day. All of the equipment was locked up securely in the forensics lab. This was the standard procedure followed at the end of each day's work.

DATE: 04/25/02

Autopsy Software Setup

09:20 AM EDT on 4/25/02

The next morning the forensics work resumed. Autopsy version 1.5 software was the primary tool used in the analysis phase. Basic Linux tools such as vi, grep, sort and others provided assistance on occasion. Autopsy is a powerful, easy to use HTML-based graphical interface to the suite of forensic tools called "Task, version 1.0." These tools have been widely used for forensic analysis and have also shown their value and reliability in the courts.

It should be pointed out that Autopsy uses a particular convention that might be confusing if it was not explained. The standard way to refer to a Windows operating system file is with a backwards slash, for example: `\WINDOWS\SYSTEM.DAT`. However, Autopsy reports its output files using a forward slash. As a result, this same file would be shown, if it were output from Autopsy, as: `/WINDOWS/SYSTEM.DAT`. In contrast to the Windows operating systems, Linux uses the convention of a forward slash to refer to a file, such as `/etc/passwd`. Since both Autopsy and Linux tools were used in the analysis, the reader will see files referred to with both backward and forward slashes.

- The Autopsy software was installed in `/home/install`. Because Autopsy expects the image to be in the local directory, the link command was used to link the `/mnt/image/hde1.img` to `/home/install/hde1.img` (`/home/install` is the working directory of Autopsy software. The image file was in `/mnt/image`). The command used was:
`ln -s /mnt/image/hde1.img /home/install/hde1.img.`
- Autopsy uses a configuration file called `fsmorgue`. It was edited to include the appropriate file system and time zone information:
Fsmorgue entry:
`hde1.img fat16 / MST7MDT`

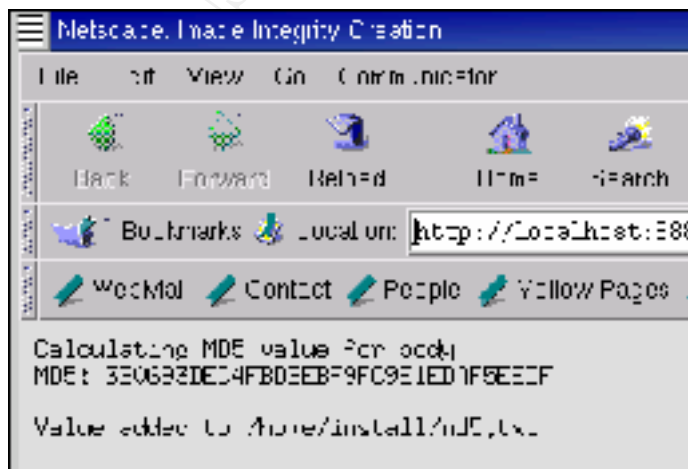
09:22 AM EDT on 4/25/02

- Autopsy was used to create the body file, which contains file attribute data for analysis. The Autopsy/Create, data file: `/home/install/body` created the file.

09:25 AM EDT on 4/25/02

- An MD5 sum of the body file was obtained using Autopsy:
Autopsy/Calculate MD5 sum of `/home/install/body`
MD5 sum = `3B0693DEC4FBD3EBF9FC9E1EDAF5EEDF`
Recorded in `/home/install/md5.txt`.

MD5 value of `/home/install/body`:



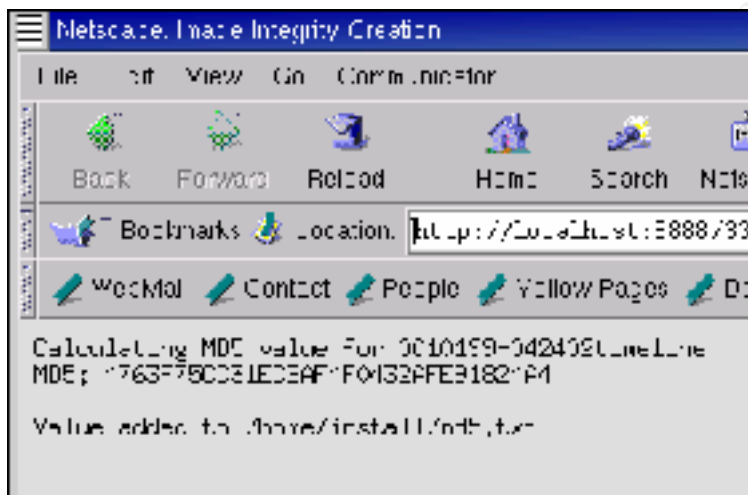
09:32 AM EDT on 4/25/02

- A MAC timeline was then created using Autopsy by using the Autopsy/Create timeline function. This used /home/install/body as the source file and output was placed in /home/install/010199-042402timeline using the MST5MDT time zone.

09:36 AM EDT on 4/25/02

- Next, an MD5 sum was obtained for the timeline file:
Autopsy/Calculate MD5 sum of /home/install/010199-042402timeline
MD5 sum = 4763F75CC31ECBAF4F0432AFE91824A4
Recorded in /home/install/md5.txt.

Timeline MD5 value:

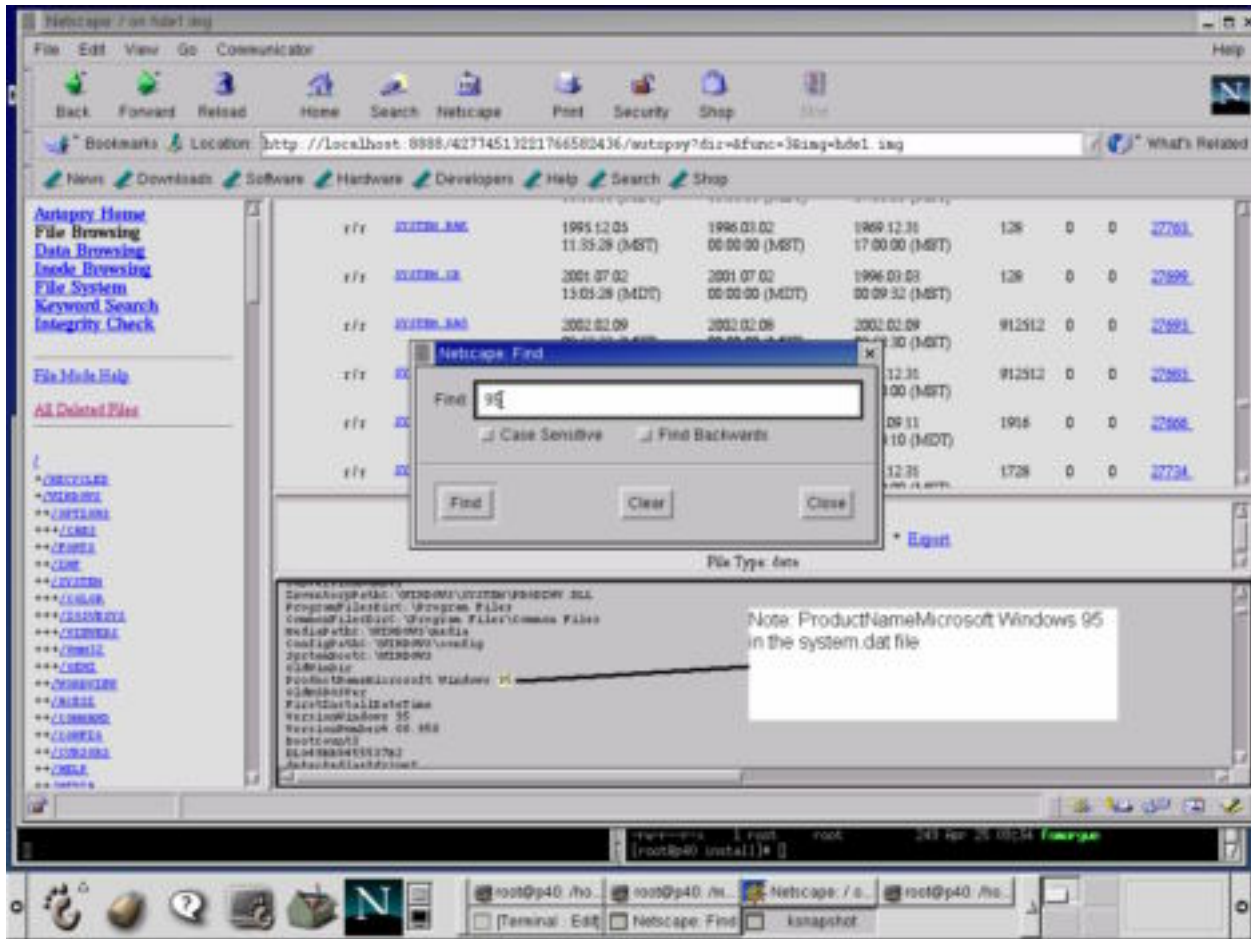


Operating System Identification

09:50 AM EDT on 4/25/02

- The operating system of the evidence laptop was identified by using "strings" on the \Windows\system.dat file. This showed the following strings:
 - ProductNameMicrosoft Windows 95
 - VersionWindows 95
 - VersionNumber4.00.950

Strings output of Windows\system.dat showing Windows 95 as the operating system:



At this point, the original physical evidence was locked up securely in the forensics lab and not used any further.

The remainder of the analysis proceeded over a period of several weeks. The steps taken are presented and discussed in the same order as the outline shown in the GIAC Practical Assignment available at: http://www.giac.org/GCFA_assignment.php. Note that the outline does not, in some cases, match the order in which the analysis steps were actually performed.

The following information from http://www.theosfiles.com/os/windows/osp_g_w95.htm was used to verify the Operating System as Windows 95 first edition. Note that the web site suggested looking at a file called "GUI.EXE." This file was not found on this system. The text saying "Version 4.00.950" was the only identifier on the evidence image. The following Windows 95 Version identification information was found on the website listed above:

Windows 95 version identification:

First edition 11-Jul-1995
Service Pack 1 (update) 31-Dec-1995
OSR2 26-Aug-1996
OSR2.5

Three major releases of Windows 95 were made. The first edition does not support FAT32, nor enhanced FAT (for support beyond 8 GB). OSR2 is a separate version, but was never provided as a free upgrade. OSR2.5 added several Y2K fixes, and is provided as a free update. All Windows 95 editions are referred to as v4.0.

1) Identification of Windows 95 version: Look at the file date and time of the GUI.EXE file in the Windows\system directory (or where you installed windows).

File time: 11-Jul-1995 9:50am = Windows 95 first edition
File date: 24-Aug-1996 11:11am = Windows 95 OSR2 (Windows 95 B)
File date: 19-Oct-1998 3:00pm (or later) = Windows 95 OSR2.5 (Windows 95 C)

2) Go to "Start", "Settings", "Control Panel". Select the "System" icon. In the General tab (the default), it shows the product "Windows 95" and the version:

Version 4.00.950 = Windows 95 first edition
Version 4.00.950 B = Windows 95 OSR2 or later

Windows is a GUI interface, but supports both a text-based DOS window, and can boot from a single diskette into a "DOS" prompt.

Windows 16-bit programs (Windows 3.x type programs)
Windows 32-bit programs
Windows VxD device drivers
DOS programs, 100% native, while Windows is not running
DOS programs in a "DOS box" while Windows is running, compatibility excellent
DOS device drivers

Y2K Compliance

If the evidence system was not patched for Y2K issues, the validity of the MAC times could come into question. It was determined that the Windows 95 operating system on the evidence computer did not appear to be Y2K compatible. This was determined by locating the following files:

```
Jul 11 1995 09:50:00 27961 m.. -/rwxrwxrwx 0 0 53219 /WINDOWS/SYSTEM/VDHCP.386  
Sep 29 1995 00:00:00 922384 m.c -/rwxrwxrwx 0 0 53135 /WINDOWS/SYSTEM/MFC40.DLL
```

These files should have been updated with the Y2K upgrade. If they had been updated, they would have had more recent MAC times. It is, however, still possible, but not likely, that the system was Y2K patched and then the older versions of these files were re-installed from the original media. In view of this, there is no way to conclusively identify this system as unpatched. The safest assumption is that this

system was not Y2K patched. As a result, some research was performed to see if Y2K problems would affect the accuracy of file MAC times. See Appendix B for details about the Y2K issues. The final conclusion was that the Y2K concerns would not affect MAC times.

MACTime Analysis

Operating System Installation Date

The Windows 95 operation system was installed on July 11, 1995 at 16:50:00. This is based on the time stamps of key operating system files and the many files installed in the \WINDOWS and \WINDOWS\SYSTEM directories. The Modify times and often the Create times are set to the date and time of the installation. Research could not identify a specific method to determine exactly when the system was installed, so an installation was performed in the lab to identify installation behavior. Certain key files and most of the files in the \WINDOWS\SYSTEM directory share the time stamp of the installation date/time. Some of the key operating system files are displayed below:

Autopsy string Report (ver 1.50)

```
-----  
File: /WINDOWS/SYSTEM/KERNEL32.DLL  
MD5 of file: 2afbc67420b9c636166e49dd0cf23a5b  
MD5 of strings: 187409a85cbe719946cee7f3d3480d94  
Image: /home/install/hde1.img  
Image Type: fat16  
Date Generated: Tue Jun 4 11:47:01 2002  
Investigator: Gregory Leibolt  
-----
```

```
inode: 52836  
Allocated  
uid / gid: 0 / 0  
mode: -rwxrwxrwx  
size: 411136  
num of links: 1  
Written: 07.11.1995 16:50:00 (MDT)  
Accessed: 02.08.2002 00:00:00 (GMT)  
Created: 12.31.1969 17:00:00 (MST)  
Name: KERNEL32.DLL
```

Autopsy string Report (ver 1.50)

```
-----  
File: /COMMAND.COM  
MD5 of file: ea923fa01468598ff2c54c90620fd450  
MD5 of strings: 6abbafb8d072352c19c1c03958b4f6e2  
Image: /home/install/hde1.img  
Image Type: fat16  
Date Generated: Tue Jun 4 11:49:24 2002  
Investigator: Gregory Leibolt
```

inode: 5
Allocated
uid / gid: 0 / 0
mode: --wx-wx-wx
size: 92870
num of links: 1
Written: **07.11.1995 16:50:00** (MDT)
Accessed: 06.17.1996 00:00:00 (GMT)
Created: 12.31.1969 17:00:00 (MST)
Name: COMMAND.COM

Autopsy string Report (ver 1.50)

File: /WINDOWS/SYSTEM/USER.EXE
MD5 of file: 792a3017d2967929187b151e1d6dc7d8
MD5 of strings: 61de333a6d3649bbfe4e71fc9345e2c2
Image: /home/install/hde1.img
Image Type: fat16
Date Generated: Tue Jun 4 11:48:22 2002
Investigator: Gregory Leibolt

inode: 52984
Allocated
uid / gid: 0 / 0
mode: -rwxrwxrwx
size: 462112
num of links: 1
Written: **07.11.1995 16:50:00** (MDT)
Accessed: 02.08.2002 00:00:00 (GMT)
Created: 12.31.1969 17:00:00 (MST)
Name: USER.EXE

Autopsy string Report (ver 1.50)

File: /WINDOWS/SYSTEM/USER32.DLL
MD5 of file: f88228bf978d0bee09e34b7f63a960ed
MD5 of strings: a93e1492ebae3854aae61c37c1a32447
Image: /home/install/hde1.img
Image Type: fat16
Date Generated: Tue Jun 4 11:48:01 2002
Investigator: Gregory Leibolt

inode: 52985
Allocated
uid / gid: 0 / 0
mode: -rwxrwxrwx
size: 44544
num of links: 1
Written: **07.11.1995 16:50:00** (MDT)
Accessed: 02.08.2002 00:00:00 (GMT)
Created: 12.31.1969 17:00:00 (MST)
Name: USER32.DLL

Note that there were **119** system files with the same time stamp.
This is a sampling of them:

```

Jul 11 1995 16:50:00 17920 m.. -/rwxrwxrwx 53199 /WINDOWS/SYSTEM/MSPP32.DLL
375962 m.. -/rwxrwxrwx 27960 /WINDOWS/NET.EXE
42080 m.. -/rwxrwxrwx 27943 /WINDOWS/WINSOCK.DLL
403 m.. -/rwxrwxrwx 235112 /WINDOWS/HELP/WINPOPUP.CNT
109229 m.. -/rwxrwxrwx 27961 /WINDOWS/NET.MSG
728 m.. -/rwxrwxrwx 27955 /WINDOWS/HOSTS.SAM
23776 m.. -/rwxrwxrwx 27947 /WINDOWS/NETSTAT.EXE
33371 m.. -/rwxrwxrwx 27946 /WINDOWS/NBTSTAT.EXE
61952 m.. -/rwxrwxrwx 53196 /WINDOWS/SYSTEM/MSAB32.DLL
38352 m.. -/rwxrwxrwx 53193 /WINDOWS/SYSTEM/CE2NDIS3.VXD
38444 m.c -/rwxrwxrwx 235733 /WINDOWS/MEDIA/In the Hall of the Mountain King.rmi (INTHEH~1.RMI)
407 m.. -/rwxrwxrwx 27957 /WINDOWS/NETWORKS
800 m.. -/rwxrwxrwx 27958 /WINDOWS/PROTOCOL
144902 m.c -/rwxrwxrwx 235737 /WINDOWS/MEDIA/Bach's Brandenburg Concerto No. 3.rmi (BACH'S~1.RMI)
15522 m.. -/rwxrwxrwx 53218 /WINDOWS/SYSTEM/W SOCK.VXD
19129 m.. -/rwxrwxrwx 53215 /WINDOWS/SYSTEM/VNETSUP.VXD
5687 m.. -/rwxrwxrwx 53223 /WINDOWS/SYSTEM/VTDI.386
37520 m.. -/rwxrwxrwx 27945 /WINDOWS/FTP.EXE
7584 m.. -/rwxrwxrwx 53202 /WINDOWS/SYSTEM/RPCLTC3.DLL
20861 m.c -/rwxrwxrwx 235525 /WINDOWS/MEDIA/CANYON.MID
20240 m.c -/rwxrwxrwx 4733959 /PROGRA~1/MICROS~1/EXCHNG32.EXE
24099 m.. -/rwxrwxrwx 27954 /WINDOWS/TELNET.HLP
8192 m.. -/rwxrwxrwx 53201 /WINDOWS/SYSTEM/RPCLTC1.DLL
42080 m.. -/rwxrwxrwx 240171 /WINDOWS/SYSBCKUP/WINSOCK.DLL
25088 m.. -/rwxrwxrwx 53204 /WINDOWS/SYSTEM/SECUR32.DLL
5816 m.. -/rwxrwxrwx 53217 /WINDOWS/SYSTEM/WSHTCP.VXD
42080 m.. -/rwxrwxrwx 240170 /WINDOWS/SYSBCKUP/_EMP.000 (deleted)
50512 m.. -/rwxrwxrwx 27942 /WINDOWS/INETMIB1.DLL
3691 m.. -/rwxrwxrwx 27956 /WINDOWS/LMHOSTS.SAM
99084 m.. -/rwxrwxrwx 53212 /WINDOWS/SYSTEM/NDIS.VXD
13456 m.. -/rwxrwxrwx 240166 /WINDOWS/SYSBCKUP/WFM0200.ACV
13312 m.. -/rwxrwxrwx 53205 /WINDOWS/SYSTEM/SVRAPI.DLL
60416 m.. -/rwxrwxrwx 53197 /WINDOWS/SYSTEM/MSNET32.DLL
27961 m.. -/rwxrwxrwx 53219 /WINDOWS/SYSTEM/VDHCP.386
147968 m.. -/rwxrwxrwx 240133 /WINDOWS/SYSBCKUP/MPLAYER.EXE
23025 m.. -/rwxrwxrwx 53210 /WINDOWS/SYSTEM/FILESEC.VXD
11591 m.. -/rwxrwxrwx 235113 /WINDOWS/HELP/WINPOPUP.HLP
6960 m.. -/rwxrwxrwx 53207 /WINDOWS/SYSTEM/WSASRV.EXE
9056 m.. -/rwxrwxrwx 27951 /WINDOWS/TRACERT.EXE
21657 m.. -/rwxrwxrwx 53211 /WINDOWS/SYSTEM/MSSP.VXD
47377 m.. -/rwxrwxrwx 53222 /WINDOWS/SYSTEM/VTCP.386
995 m.. -/rwxrwxrwx 53208 /WINDOWS/SYSTEM/LMSCRIP.T.PIF
27600 m.. -/rwxrwxrwx 27953 /WINDOWS/WINPOPUP.EXE
66672 m.. -/rwxrwxrwx 27950 /WINDOWS/TELNET.EXE
19536 m.. -/rwxrwxrwx 27944 /WINDOWS/ARP.EXE
25402 m.. -/rwxrwxrwx 53209 /WINDOWS/SYSTEM/AFVXD.VXD
4785 m.. -/rwxrwxrwx 53206 /WINDOWS/SYSTEM/LMSCRIP.T.EXE
12128 m.. -/rwxrwxrwx 27948 /WINDOWS/PING.EXE
95969 m.. -/rwxrwxrwx 53221 /WINDOWS/SYSTEM/VNBT.386
42080 m.. -/rwxrwxrwx 240170 <hde1.img-_EMP.000-dead-240170>
106960 m.. -/rwxrwxrwx 53192 /WINDOWS/SYSTEM/NETAPI.DLL
398 m.. -/rwxrwxrwx 52777 /WINDOWS/SYSTEM/KBDUS.KBD
46000 m.. -/rwxrwxrwx 240168 /WINDOWS/SYSBCKUP/SB16SND.DRV
23744 m.. -/rwxrwxrwx 53213 /WINDOWS/SYSTEM/NDIS2SUP.VXD
23165 m.c -/rwxrwxrwx 235526 /WINDOWS/MEDIA/PASSPORT.MID
23696 m.. -/rwxrwxrwx 27949 /WINDOWS/ROUTE.EXE
20906 m.c -/rwxrwxrwx 235698 /WINDOWS/MEDIA/Dance of the Sugar-Plum Fairy.rmi (DANCEO~1.RMI)
38912 m.. -/rwxrwxrwx 27952 /WINDOWS/WINIPCFG.EXE
147968 m.c -/rwxrwxrwx 27772 /WINDOWS/MPLAYER.EXE
6496 m.. -/rwxrwxrwx 53195 /WINDOWS/SYSTEM/ICMP.DLL
27221 m.. -/rwxrwxrwx 53214 /WINDOWS/SYSTEM/VNETBIOS.VXD
67584 m.. -/rwxrwxrwx 53198 /WINDOWS/SYSTEM/MSNP32.DLL
42496 m.. -/rwxrwxrwx 52992 /WINDOWS/SYSTEM/VLB32.DLL
22016 m.. -/rwxrwxrwx 53194 /WINDOWS/SYSTEM/CHOOSUSR.DLL

```

```

17776 m.. -/rwxrwxrwx 240167 /WINDOWS/SYBCKUP/CSPMAN.DLL
592896 m.. -/rwxrwxrwx 52850 /WINDOWS/SYSTEM/MAPI32.DLL
62614 m.. -/rwxrwxrwx 53220 /WINDOWS/SYSTEM/VIP.386
143914 m.c -/rwxrwxrwx 235629 /WINDOWS/MEDIA/Jungle Close.wav (JUNGL~15.WAV)
21312 m.c -/rwxrwxrwx 235581 /WINDOWS/MEDIA/Beethoven's Fur Elise.rmi (BEE THO~2.RMI)
140343 m.. -/rwxrwxrwx 53216 /WINDOWS/SYSTEM/VREDIR.VXD
44304 m.. -/rwxrwxrwx 51754 /WINDOWS/FONTS/DOSAPP.FON
9168 m.. -/rwxrwxrwx 53203 /WINDOWS/SYSTEM/RPCLTS3.DLL
6007 m.. -/rwxrwxrwx 27959 /WINDOWS/SERVICES
73275 m.. -/rwxrwxrwx 27962 /WINDOWS/NETH.MSG
26608 m.. -/rwxrwxrwx 53200 /WINDOWS/SYSTEM/PMSPL.DLL
Jul 11 1995 16:50:00 5632 m.c -/rwxrwxrwx 401926 /WINDOWS/SHELLNEW/EXCEL.XLS
2118 m.c -/rwxrwxrwx 27846 /WINDOWS/Bubbles.bmp (BUBBLES.BMP)
39744 m.. -/rwxrwxrwx 240141 /WINDOWS/SYBCKUP/OLE2.DLL
88544 m.. -/rwxrwxrwx 240152 /WINDOWS/SYBCKUP/COMMDLG.DLL
3200 m.. -/rwxrwxrwx 240135 /WINDOWS/SYBCKUP/WIN32S16.DLL
5584 m.. -/rwxrwxrwx 240163 /WINDOWS/SYBCKUP/MCIOLE.DLL
59392 m.c -/rwxrwxrwx 27669 /WINDOWS/CALC.EXE
24064 m.. -/rwxrwxrwx 240145 /WINDOWS/SYBCKUP/OLESVR.DLL
5532 m.. -/rwxrwxrwx 240137 /WINDOWS/SYBCKUP/STDOLE.TLB
116144 m.. -/rwxrwxrwx 240149 /WINDOWS/SYBCKUP/SHELL.DLL
169440 m.. -/rwxrwxrwx 240140 /WINDOWS/SYBCKUP/OLE2DISP.DLL
53552 m.. -/rwxrwxrwx 240162 /WINDOWS/SYBCKUP/MSACM.DLL
57328 m.. -/rwxrwxrwx 240138 /WINDOWS/SYBCKUP/OLE2CONV.DLL
23696 m.. -/rwxrwxrwx 240153 /WINDOWS/SYBCKUP/LZXPAND.DLL
5632 m.. -/rwxrwxrwx 240154 /WINDOWS/SYBCKUP/DCIMAN32.DLL
32240 m.. -/rwxrwxrwx 240144 /WINDOWS/SYBCKUP/DDEML.DLL
4208 m.. -/rwxrwxrwx 240142 /WINDOWS/SYBCKUP/STORAGE.DLL
109424 m.. -/rwxrwxrwx 240158 /WINDOWS/SYBCKUP/AVIFILE.DLL
177856 m.. -/rwxrwxrwx 240136 /WINDOWS/SYBCKUP/TYPELIB.DLL
30976 m.. -/rwxrwxrwx 240143 /WINDOWS/SYBCKUP/COMPOBJ.DLL
4608 m.c -/rwxrwxrwx 27726 /WINDOWS/PBRUSH.EXE
12112 m.. -/rwxrwxrwx 240151 /WINDOWS/SYBCKUP/TOOLHELP.DLL
153040 m.. -/rwxrwxrwx 240139 /WINDOWS/SYBCKUP/OLE2NLS.DLL
16976 m.. -/rwxrwxrwx 240157 /WINDOWS/SYBCKUP/MIDIMAP.DRV
6992 m.. -/rwxrwxrwx 240160 /WINDOWS/SYBCKUP/DISPDIB.DLL
253952 m.. -/rwxrwxrwx 240134 /WINDOWS/SYBCKUP/MSVCRT20.DLL
82944 m.. -/rwxrwxrwx 240146 /WINDOWS/SYBCKUP/OLECLI.DLL
113664 m.. -/rwxrwxrwx 240159 /WINDOWS/SYBCKUP/MSVIDEO.DLL
6928 m.. -/rwxrwxrwx 240155 /WINDOWS/SYBCKUP/DCIMAN.DLL
21872 m.. -/rwxrwxrwx 240161 /WINDOWS/SYBCKUP/MSACM.DRV
67520 m.. -/rwxrwxrwx 240165 /WINDOWS/SYBCKUP/MCIAVI.DRV
2416 m.. -/rwxrwxrwx 240150 /WINDOWS/SYBCKUP/WINHELP.EXE
2416 m.c -/rwxrwxrwx 27751 /WINDOWS/WINHELP.EXE
1264 m.. -/rwxrwxrwx 240156 /WINDOWS/SYBCKUP/MSMIXMGR.DLL
92870 m.c -/rwxrwxrwx 27674 /WINDOWS/COMMAND.COM
72272 m.. -/rwxrwxrwx 240164 /WINDOWS/SYBCKUP/AVICAP.DLL
154880 m.. -/rwxrwxrwx 240148 /WINDOWS/SYBCKUP/COMMCTRL.DLL
34304 m.c -/rwxrwxrwx 27724 /WINDOWS/NOTEPAD.EXE
12144 m.. -/rwxrwxrwx 240147 /WINDOWS/SYBCKUP/VER.DLL
Jul 14 1995 16:00:38 31104 m.. -/rwxrwxrwx 11339362 /PROGRA~1/INTUIT/QUICKB~1/qct13d.dll (QCTL3D.DLL)

```

The Registry

The registry contains information which might be useful in determining or correlating user activity. Since it is a hive structure that is a bit confusing to analyze with “strings,” a useful way to view the information is with a registry viewer. To accomplish this the registry data needed to be imported into a Windows 95 registry.

1. Used Autopsy to export c:\windows\system.dat and c:\windows\user.dat, the two files that make up the registry on a Windows 95 system. These two files were taken to the lab’s Windows 95 system and loaded into the registry for viewing by

using the following steps.

2. Copied the evidence system's system.dat and user.dat files to C:\temp on the lab's system.
3. Backed up the lab's Windows 95 registry using the "Export Registry File" option in regedit.
4. Rebooted the system with a Windows 95 boot disk, went to C:\temp, then used `A:\regedit /L:system.dat /R:user.dat -e Evidence.reg` to create a registry file.
5. Updated the lab's Windows 95 test system's registry file using `a:\regedit /L:system.dat /R:user.dat /C Evidence.reg` to IMPORT the evidence system's registry. NOTE: THIS DOES **MIX** REGISTRY DATA, BUT IT IS A HELP IN VIEWING THE EVIDENCE REGISTRY INFO. THIS INFORMATION WOULD NOT HOLD UP IN COURT IF REFERENCING ONLY THE IMPORTED REGISTRY DATA. CORRELATION MUST BE MADE BACK TO ORIGINAL "IMAGED" DATA. VIEWING REGISTRY INFORMATION THROUGH THE REGISTRY EDITOR DOES HELP ORGANIZE WHAT ONE IS VIEWING, HOWEVER.
6. Rebooted the forensic lab's test Windows 95 system.
7. Viewed the registry for potentially useful information. Gathered history information from registry data. Unfortunately, there is now way to know when "recent" events occurred.
8. Restored the original Windows 95 test system registry using the following step:

Re-booted test Windows 95 system with the Windows 95 boot disk.
C: cd \temp A:\regedit /L:system.bak /R:user.bak /C backup.reg.

Interesting Registry Information

HKEY_CURRENT_USER/ Entries

NOTE: This is a list of recent files edited with Microsoft Paint.

```
[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Applets\Paint\Recent File List]
"File1"="C:\Program
Files\Accessories\DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
DDDDDESSSSSSSSSSSI.bmp"
"File2"="A:\FLAG XXXX.bmp"
"File3"="C:\WINDOWS\Bubbles.bmp"
"File4"="C:\CSW\PICTURES\STREETS.BMP"
```

NOTE: This is a list of strings used to search for a file START/FIND/FILES.

```
[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Doc Find Spec MRU]
```

```
"a"=""
"MRUList"="aedcbjihgf"
"b"="corporate table"
"c"="corporate "
"d"="mail list "
"e"="CorporateList1997"
"f"="0196 "
"g"="corporate"
"h"=" list"
"i"="tables"
"j"="table"
```

NOTE: This is a list of recent commands issued through the START/RUN option.

```
[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\RunMRU]
```

```
"a"="a:/setup\1"
"MRUList"="dcab"
"b"="a:/scenery1.exe\1"
"c"="a:\setup\1"
"d"="regedit\1" (NOTE: This one line was added by the forensic investigator when regedit was used to view the registry)
"e"=""
```

NOTE: This is a list of recent files edited with Microsoft WordPad.

```
[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Applets\WordPad\Recent File List]
```

```
"File1"="C:\\TEMP\\HKYsoftw.txt" (NOTE: This one line was added by the forensic investigator when an export of the registry was performed)
"File2"="C:\\Program Files\\Accessories\\Thecampingtrip.doc"
"File3"="C:\\Program Files\\Accessories\\The family trip.doc"
"File4"="A:\\Hawaii 2001 agenda.doc"
```

NOTE: This is a list of recent files edited with Microsoft PowerPoint.

```
[HKEY_CURRENT_USER\Software\Microsoft\Office\PowerPoint\7.0\Recent File List]
```

```
"File1"="A:\\To XXXX From YYYY.ppt"
"File2"="A:\\XXXX PRESENT.ppt"
"File3"="C:\\My Documents\\DDDEEESSIII.ppt"
"File4"="A:\\Hawaii back ground.ppt"
"File5"="C:\\My Documents\\XXXX PRESENT.ppt"
"File6"="C:\\My Documents\\U'I XXXX.ppt"
"File7"="C:\\MSOFFICE\\POWERPNT\\WIZARDS\\BADNEWS.PPT"
```

NOTE: This is a list of recent files edited with Microsoft Excel.

```
[HKEY_CURRENT_USER\Software\Microsoft\Excel\7.0\Recent File List]
```

```
"File1"="A:\\time trials.xls"
"File2"="C:\\My Documents\\Nominations9999.xls"
"File3"="C:\\WINDOWS\\Desktop\\My Briefcase\\1998MONIES.xls"
"File4"=""
```

NOTE: This is a list of recent files edited with Microsoft Network.

```
[HKEY_CURRENT_USER\Network\Recent\././AZ10FD8W./F]
```

```
"ConnectionType"=dword:00000001
"UserName"="XXXXXXX"
"ProviderName"="Microsoft Network"
[HKEY_CURRENT_USER\Network\Persistent]
```

NOTE: This is a list of recent installation source directories.

```
[HKEY_CURRENT_USER\InstallLocationsMRU]
"a"="A:\\"
"MRUList"="ba"
"b"="C:\WINDOWS\OPTIONS\CABS\"
"c"="D:\WIN95"
"d"="D:\WIN95\"
"e"="D:\ADMIN\NETTOOLS\PRAGENT\"
```

HKEY_LOCAL_MACHINE/SOFTWARE/ Entries

NOTE: QuickBooks registration information.

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Intuit\QuickBooksRegistration6.0]
"QBMode"="ThisWasAUsername"
"InstallNumber"="0260-270-656-6206"
"VersionNumber"=""
"RegistrationNumber"="UNREG"
"GroupNumber"=""
"AOIndex"="0000"
"LA"="YES"
>WelcomeVersionNumber"="Version 6.0D"
```

NOTE: Internet Explorer configuration information.

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\Main]
"Enable_Disk_Cache"="yes"
"History_Directory"="C:\Program Files\Plus!\Microsoft Internet\history"
"History_Num_Places"=hex:2c,01,00,00
"Cache_Directory"="C:\Program Files\Plus!\Microsoft Internet\cache"
"Cache_Percent_of_Disk"=hex:0a,00,00,00
"Delete_Temp_Files_On_Exit"="yes"
"History_Expire_Days"=hex:ff,ff,ff,ff
"Anchor_Visitation_Horizon"=hex:01,00,00,00
"Use_Async_DNS"="yes"
"Placeholder_Width"=hex:1a,00,00,00
"Placeholder_Height"=hex:1a,00,00,00
"Default_Page_URL"="http://www.home.msn.com"
"Cache_Update_Frequency"="Once_Per_Session"
```

Recover Deleted Files

Autopsy's "File Activity Time Line Analysis" functions were used to create a timeline of all the files in the image (deleted and current). Autopsy identifies deleted files in the timeline and can be used to view and recover deleted files.

- Created a list of deleted files by grep'ing "delete" from the body file and piping it through mactime: `grep delete body | task-1.00/bin/mactime | tee deleted_files.`
- Began exploring the image with Autopsy.

The focus was within a time frame of 1/1/1998 to 10/23/01. Since the misuse of funds was suspected, the goal was to find anything related to funds and any list of contributors or sponsors. A concerted effort was placed in reviewing deleted files because deleted files might show an effort to cover up information. The thought that the user might try

to “clean up” incriminating evidence before returning it to the president of the organization gave credence to focusing within a date range.

It is important to note that the deleted files listed below were created, accessed or modified within the time frame of 1/1/1998 to 10/23/2001. Files that were created, accessed or modified prior to the time frame of the investigation may have been deleted within the time frame of interest, but would not be listed in the following list. That is because the Windows 95 FAT 16 file system does not change any of the MAC times when a file is deleted (see Appendix A). For example, a file which was created on 1/1/1997 and deleted on 2/1/1998 would not have appeared in the list below. That is because the MAC date associated with the file would remain 1/1/1997.

As stated above, the deleted files listed below were created, accessed or modified within the time frame of 1/1/1998 to 10/23/2001. Also, the files listed below were deleted sometime after they were created, accessed or modified. Again, as stated above, since the Windows 95 FAT 16 file system does not modify any of the MAC times when a file is deleted, there is no way to know from the MAC times of the files exactly when the deletion occurred. A forensics investigator must work with this limitation. It is also impossible to say for sure who deleted the files. For the sake of argument, however, a reasonable assumption can be made. The president assured the forensics team that he did not delete any files, and also, that the laptop was locked up securely at all times while it was in his possession. Therefore, it is likely that the files were deleted while the laptop was under the control of the treasurer. Other methods of identifying when the files were deleted need to be explored so some validation can be obtained.

These are the deleted files that looked interesting:

```
Sep 16 1998 02:00:00 11264 .a -/rwxrwxrwx 19138 /RECYCLED/_C855.DOC (deleted)
295424 .a -/rwxrwxrwx 11148305 /MSOFFICE/WINWORD/WORKSH-1/_EBELUNG.DOC (deleted)
295424 .c -/rwxrwxrwx 19148 /RECYCLED/_C865.WBK (deleted)
295424 ..c -/rwxrwxrwx 11148305 /MSOFFICE/WINWORD/WORKSH-1/_EBELUNG.DOC (deleted)
295424 .c -/rwxrwxrwx 19145 /RECYCLED/_C862.WBK (deleted)
295424 ..c -/rwxrwxrwx 11148326 /MSOFFICE/WINWORD/WORKSH-1/Backup of Neb (_ACKUP~3.WBK) (deleted)
295936 ..c -/rwxrwxrwx 11148307 /MSOFFICE/WINWORD/WORKSH-1/Lopez.doc (_OPEZ.DOC) (deleted)
Sep 16 1998 22:10:14 295936 m.. -/rwxrwxrwx 11148307 /MSOFFICE/WINWORD/WORKSH-1/Lopez.doc (_OPEZ.DOC) (deleted)
Sep 17 1998 02:00:00 295936 .a -/rwxrwxrwx 11148307 /MSOFFICE/WINWORD/WORKSH-1/Lopez.doc (_OPEZ.DOC) (deleted)
295424 .a -/rwxrwxrwx 11148309 /MSOFFICE/WINWORD/WORKSH-1/~WRD1294.tmp (_WRD1294.TMP) (deleted)
295424 .a -/rwxrwxrwx 11148318 /MSOFFICE/WINWORD/WORKSH-1/Backup of Bes (_ACKUP~1.WBK) (deleted)
2681 .a -/rwxrwxrwx 3499533 /WINDOWS/DESKTOP/MYBRIE~1/_F414000 (deleted)
Sep 26 1998 00:09:14 2681 ..c -/rwxrwxrwx 3499533 /WINDOWS/DESKTOP/MYBRIE~1/_F414000 (deleted)
Sep 26 1998 00:09:16 2681 m.. -/rwxrwxrwx 3499533 /WINDOWS/DESKTOP/MYBRIE~1/_F414000 (deleted)
Sep 26 1998 00:32:32 2681 ..c -/rwxrwxrwx 3499532 /WINDOWS/DESKTOP/MYBRIE~1/1998MONIES.xls (_998MO~1.XLS) (deleted)
2681 ..c -/rwxrwxrwx 18959 /RECYCLED/_C870.XLS (deleted)
Nov 28 1999 01:00:00 268 .a -/rwxrwxrwx 4932661 /SIDEKI~1/DATA/openlist.lst (_PENLIST.LST) (deleted)
626 .a -/rwxrwxrwx 4931630 /SIDEKI~1/USERDATA/_CTASK.SKW (deleted)
210182 .a -/rwxrwxrwx 5734954 /MSOFFICE/POWERPNT/_OWERPNT.TMP (deleted)
0 .a -/rwxrwxrwx 236600 /WINDOWS/TEMP/_DFC0B3.TMP (deleted)
Jun 25 2001 19:49:48 210182 m.. -/rwxrwxrwx 5734954 /MSOFFICE/POWERPNT/_OWERPNT.TMP (deleted)
Jun 27 2001 16:15:40 353 m.c -/rwxrwxrwx 367641 /WINDOWS/RECENT/Hey there bab (_EYTHE~1.LNK) (deleted)
Jun 27 2001 16:25:40 431 m.c -/rwxrwxrwx 367666 /WINDOWS/RECENT/Ddeardjhjdufudjtuthuftr (_DEARD~1.LNK) (deleted)
Jun 27 2001 16:42:04 8628 m.c -/rwxrwxrwx 19477 /CSW/streets.TMP (_TREETS.TMP) (deleted)
Jun 29 2001 15:28:22 8628 m.c -/rwxrwxrwx 5734956 /MSOFFICE/POWERPNT/ppttod.TMP (_PTTOD.TMP) (deleted)
Jul 02 2001 18:23:54 25201 m.c -/rwxrwxrwx 235115 /WINDOWS/HELP/wordpad.TMP (_ORDPAD.TMP) (deleted)
Jul 04 2001 16:30:50 568 m.c -/rwxrwxrwx 367680 /WINDOWS/RECENT/DDDDDDDDDDDDDDDDDDDDDDDDDDDD (_DDDDD~1.LNK)
(deleted)
0 m.c -/rwxrwxrwx 4734497 /PROGRA~1/ACCESS~1/_DDDDD~1.BMP (deleted)
```

```

Jul 07 2001 02:00:00 568 .a. -/rwxrwxrwx 367680 /WINDOWS/RECENT/DDDDDDDDDDDDDDDDDDDDDDDDDDDDDD (_DDDDD~1.LNK)
(deleted)
Oct 20 2001 02:00:00 353 .a. -/rwxrwxrwx 367641 /WINDOWS/RECENT/Hey there bab (_EYTHE~1.LNK) (deleted)
53 .a. -/r-xr-xr-x 5739052 /MSOFFICE/WINWORD/_$HX1295.DOC (deleted)
53 .a. -/r-xr-xr-x 6038157 /MYDOCU~1/~$y there baby.doc (_$YTHE~1.DOC) (deleted)
2681 .a. -/rwxrwxrwx 3499532 /WINDOWS/DESKTOP/MYBRIE~1/1998MONIES.xls (_998MO~1.XLS) (deleted)
431 .a. -/rwxrwxrwx 367663 /WINDOWS/RECENT/Ddeardjhjdufufdujtuthfutftuefhurdm u.doc.lnk (_DEARD~1.LNK)
(deleted)
Oct 20 2001 08:57:14 2681 m. -/rwxrwxrwx 18959 /RECYCLED/_C870.XLS (deleted)
2681 m. -/rwxrwxrwx 3499532 /WINDOWS/DESKTOP/MYBRIE~1/1998MONIES.xls (_998MO~1.XLS) (deleted)
Oct 20 2001 09:12:26 53 m.c -/r-xr-xr-x 6038154 /MYDOCU~1/_$EARD~1.DOC (deleted)
431 ..c -/rwxrwxrwx 367663 /WINDOWS/RECENT/Ddeardjhjdufufdujtuthfutftuefhurdm u.doc.lnk (_DEARD~1.LNK)
(deleted)
Oct 20 2001 09:16:18 53 m.c -/r-xr-xr-x 5739052 /MSOFFICE/WINWORD/_$HX1295.DOC (deleted)

```

Interesting Files

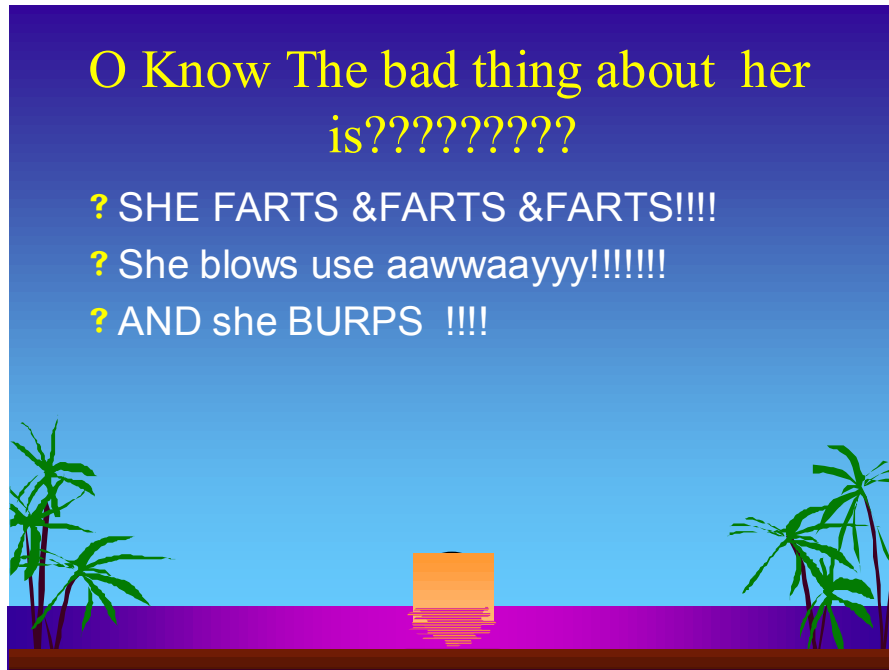
Windows applications create many temporary files which are often copies of working documents, spreadsheets, etc. These temporary files were all reviewed with the “strings” command to see if they might contain useful information. These and some of the other interesting files are discussed below. File names have been changed to protect proprietary information.

The next step taken was to extract the files named c:\exchange\mailbox.pst and c:\exchange\mailbox.pab to see if mail data could provide information. Both files were attached to MS Outlook on a lab system and were found to be empty with the exception of the Microsoft welcome message that is part of the installation.

There were several interesting files in the \MyDOCUMENTS\ directory. The last name of the treasurer was found in \MYDOCUMENTS\ DDDEESSSIII.PPT, with Modify, Access and Create times dated 06/29/2001. This is a very strange file name which prompted additional investigation. The file was extracted through Autopsy and viewed on a lab system. One slide of the presentation is displayed below:

© SANS Institute - Author retains full rights

Slide from \MYDOCUMENTS\ DDDEESSIII.PPT:



Well! Surprise! Surprise! It looked like a presentation that a child created. Could the treasurer have children that used the computer?

Further investigation identified two deleted files. The first was \TEMP\~WRA3033.asd 6/25/2001 (deleted) which showed information about a young girl: (girl's first name) (treasurer's last name), age 9. The second was \TEMP\~WRS0260.tmp 6/25/2001 (deleted) which showed information about her sister: (sister's first name) (treasurer's last name), age 11.

Other documents also appeared to be written by one of the children and contained information related to the PowerPoint presentation. Specifically, these were the files named \PROGRAM FILES\ACCESSORIES\The family trip.doc and \PROGRAM FILES\ACCESSORIES\The campingtrip.doc. It was later confirmed with the president that these were the daughters of the treasurer.

On to other leads....

MYDOCUMENTS/ Nominations9999.xls with Modify, Access and Create dates of 09/10/1999 contained a list of people nominated for various positions. The treasurer's name was not in this spreadsheet. There did not appear to be any evidence value to this file.

MYDOCUMENTS/ OC4.doc with Modify date of 4/6/1997 and Access and Create dates of 4/5/1997 was a deleted file, which appeared to be a list of organization supporters.

This was information that could be used to obtain financial information regarding donations. MS Word could not open the file, but the data could be read with the "strings" command. A sanitized portion of the Autopsy report of OC4.Doc is shown below.

Autopsy string Report (ver 1.50)

File: /My Documents (MYDOCU~1)/_OC4.DOC
MD5 of file: 0976c419d2fd44e6b28592949b1dd500
MD5 of strings: 54ff99f6427344e3119064497e1c4140
Image: /home/install/hde1.img
Image Type: fat16
Date Generated: Mon April 25 11:23:16 2002
Investigator: Gregory Leibolt

inode: 6038076
Not Allocated
uid / gid: 0 / 0
mode: -rwxrwxrwx
size: 29696
num of links: 0
Written: 04.06.1997 15:03:26 (MDT)
Accessed: 04.05.1997 00:00:00 (GMT)
Created: 04.05.1997 16:20:10 (MST)
Name: _OC4.DOC
Sectors:
426147 426148 426149 426150 426151 426152 426153 426154
426155 426156 426157 426158 426159 426160 426161 426162
426163 426164 426165 426166 426167 426168 426169 426170
426171 426172 426173 426174 426175 426176 426177 426178
File Type: data

U.S. XXXXXXXX Chamber of Commerce
XXXX Xth St. NW, Suite XXX
Washington, DC 20005
National Institute For XXXXXX Development
(The data continues but is cut short for brevity and sanitizing.)

\FILE0000.CHK - \FILE0005.CHK are lost clusters found during a scandisk operation. These were viewed for possible evidence. These restored clusters contained information related to organizational meetings, but did not appear to be of direct interest to this case. They all had Modify dates of 7/5/1998 and this correlated with \SCANDISK.LOG which reported scandisk activity at 10:06 7/5/1998.

[Microsoft ScanDisk for Windows](#)

[NOTE: If you use an MS-DOS program to view this file, some of the characters may appear incorrectly. Use a Windows program such as Notepad instead.](#)

[Log file generated at 10:06 on 7/5/1998.](#)

ScanDisk used the following options:

Standard test
Automatically fix errors

Drive Windows95 (C:) contained the following errors:

ScanDisk found 131072 bytes of data in 6 lost file fragment(s).

Resolution: Convert the lost file fragment(s) into file(s)

Results: Error was corrected as specified above.

ScanDisk found errors on this drive and fixed them all

MYDOCUMENTS\Sponsortablelistpage6.dot 4/5/1997 was outside the target time frame of 1/1/1998 to 10/23/2001. It contained a table of company addresses, which could be a list of sponsors as the name suggests. It was saved as backup information relating to sponsors.

MYDOCUMENTS\Sponsortablelistpage7.doc 10/4/1998 was within the target time frame. It contained a table of company addresses, which could be a list of sponsors as the name suggests.

WINDOWS\DESKTOP\MYBRIE~1\F414000 9/25/1998 (deleted) was also within the target time frame. It looked like a spreadsheet but was not readable by Excel. The suggested procedure for recovering corrupted Excel files did not work. The procedure from Microsoft Excel Help, Recover information from a damaged workbook file, did not help. Attempts to open the file with Word or WordPad did not succeed. Strings output currently provided the following information.

A sanitized portion of the Autopsy report of F41400 is shown below.

Autopsy string Report (ver 1.50)

File: /WINDOWS/Desktop (DESKTOP)/My Briefcase (MYBRIE~1)/_F414000

MD5 of file: 11a0f6dcc5b69e99fcf53ed906598e5b

MD5 of strings: fc70ab22865cd5cb278c4f7cfaf85e27

Image: /home/install/hde1.img

Image Type: fat16

Date Generated: Mon April 25 14:05:56 2002

Investigator: Gregory Leibolt

inode: 3499533

Not Allocated

uid / gid: 0 / 0

mode: -rwxrwxrwx

size: 2681

num of links: 0

Written: 09.25.1998 22:09:16 (MDT)

Accessed: 09.25.1998 00:00:00 (GMT)

Created: 09.25.1998 22:09:14 (MDT)

Name: _F414000

Sectors:
810499 810500 810501 810502 810503 810504
File Type: data

#C:\
Program Files
PROGRA~1
Accessories
ACCESS~1
Wordpad.exe
WINDOWS95
C:\Program Files\Accessories\WORDPAD.EXE
WordPad1
..\..\..\Program Files\Accessories\WORDPAD.EXE
C:\Progra~1\Access~1
###0_);[Red](###0\
###0.00_);(###0.00\
###0.00_);[Red](###0.00\
"\$#,#0_);("\$#,#0\
"\$#,#0_);[Red]("\$#,#0)

WINDOWS\DESKTOP\MYBRIE~\1998MONIES.xls 10/20/2001 (Deleted) was a readable Excel file. A sanitized version is shown below.

| RECEIVED | AMOUNT | BALANCE |
|----------|------------|-----------|
| 16-Jun | \$4,975.00 | |
| 22-Jun | 507.00 | 5,482.00 |
| 9-Jul | 1,333.00 | 6,815.00 |
| 29-Jul | 2,000.00 | 8,815.00 |
| 7-Aug | 700.00 | 9,515.00 |
| 11-Aug | 800 | 10,315.00 |
| 17-Aug | 1,200.00 | 11,515.00 |
| 21-Aug | 400 | 11,915.00 |
| 25-Aug | 200 | 12,115.00 |
| 4-Sep | 1,200.00 | 13,315.00 |
| 9-Sep | 100.00 | 13,415.00 |
| 11-Sep | 1,000.00 | 14,415.00 |
| 18-Sep | 2,000.00 | 16,415.00 |

A sanitized portion of the Autopsy report of 1998MONIES.xls is shown below. It is important to note that "strings" does not display any of the integer data (monetary value). Excel files need to be viewed via Excel or other capable utility.

Autopsy string Report (ver 1.50)

File: /WINDOWS/DESKTOP/MYBRIE~1/1998MONIES.xls (_998MO~1.XLS)
MD5 of file: dfda7243bb6b55d13c4474bbe867a0ca
MD5 of strings: f44fcc0401ad2d204231d4bbcef7cced
Image: /home/install/hde1.img
Image Type: fat16
Date Generated: Mon April 25 13:42:11 2002
Investigator: Gregory Leibolt

inode: 3499532
Not Allocated
uid / gid: 0 / 0
mode: -rwxrwxrwx
size: 2681
num of links: 0
Written: 10.20.2001 06:57:14 (MDT)
Accessed: 10.20.2001 00:00:00 (GMT)
Created: 09.25.1998 22:32:32 (MDT)
Name: _998MO~1.XLS
Sectors:
810563 810564 810565 810566 810567 810568
File Type: data

XX & XXXXXX XXXXXX
MbP?_
Arial1
Arial1
Arial1
Arial1
Arial
Page &P
General
0.00
###0
###0.00
###0_);\(#,##0\
###0_);[Red]\(#,##0\
###0.00_);\(#,##0.00\
###0.00_);[Red]\(#,##0.00\
"\$"#,##0_);\""\$"#,##0\
"\$"#,##0_);[Red]\("\$"#,##0\
"\$"#,##0.00_);\""\$"#,##0.00\
""\$"#,##0.00_);[Red]\("\$"#,##0.00\
0.00%
0.00E+00
#\ ??/?
#\ ??/??
m/d/yy
d\-mmm\-yy
d\-mmm
mmm\-yy
h:mm\ AM/PM
h:mm:ss\ AM/PM
h:mm

```

h:mm:ss
m/d/yy\ h:mm
##0.0E+0
mm:ss
2_("$* #,##0_);_("$* \(\#,##0\);_("$* "-"_);_(@_)
)_(* #,##0_);_(* \(\#,##0\);_(* "-"_);_(@_)
:("$* #,##0.00_);_("$* \(\#,##0.00\);_("$* "-"??_);_(@_)
1_(* #,##0.00_);_(* \(\#,##0.00\);_(* "-"??_);_(@_)
RECEIVED
AMOUNT
BALANCE~

```

String Searches

When a search for data containing the treasurer's last name was performed with Autopsy's Keyword Search, 24 sectors were identified. These sectors were viewed for potentially useful evidence. They did not appear to be useful to the case. For the most part, they contained information from organization minutes and other notes. Other key words such as money, monies, dollar, corporate, sponsor and budget were used.

Regular expressions were used in the key word search field to reduce the time needed to search for several key words. One example of a regular expression used in the search field is "money|monies|dollar|corporate|sponsor|buget." All the words in this search string were examined on the same pass.

This search turned up 658 occurrences. The key word search is one way to find information in files that may have otherwise been ignored because they might have been renamed to nonstandard naming conventions. For example, a document file could be renamed "acmod.dll" or some other innocuous name. All these sectors were reviewed and, for those that looked promising, the associated inode and file name were identified for further analysis. It turned out that most of these sectors were related to application help files, application templates, installation files and meeting minutes.

Conclusions

User Activity

Viewing the timeline data showed that the computer had more MAC time activity in 1996 to 1998 and much less between 1999 to 2002. It was confirmed that the president of the organization did use the computer briefly on February 9, 2001 from 8:53 AM to 9:08 AM, to look at the recent files in Start/Documents and Excel:

```

Feb 09 2002 08:53:28    1916 m.. -/rwxrwxrwx 0    0    27666 /WINDOWS/SYSTEM.INI
Feb 09 2002 09:01:02    328 m.c -/rwxrwxrwx 0    0    367652 /WINDOWS/RECENT/XXXXA .doc.lnk
Feb 09 2002 09:01:58    334 m.c -/rwxrwxrwx 0    0    367636 /WINDOWS/RECENT/Excel.xls.lnk (EXCELX~1.LNK)
Feb 09 2002 09:08:40  131336 m.. -/--x-x-x 0    0    27764 /WINDOWS/USER.DAT

```

Activity on the computer, prior to the president accessing it, occurred on October 20, 2001. This date was after October 12th, which was the day that the president demanded the computer be sent to him. Also, it was just three days before the president received the computer. On October 20th several key files were accessed.

The files named \WINDOWS\DESKTOP\MYBRIE~1\1998MONIES.xls, \MYDOCU~1_\$_EARD~1.DOC, \MYDOCU~1\American XXXXX of the United States.doc, \MSOFFICE\WINWORD\PHX1295.DOC and \MYDOCU~1\U.doc were viewed and possibly modified. The file named \MYDOCU~1\1997 BUSINESS PLAN.doc (1997BU~1.DOC) was possibly viewed.

The timeline output is shown below:

```
Oct 20 2001 00:00:00 13312 .a. -/rwxrwxrwx 6038130 /MYDOCU~1/Backup of U.wbk (BACKUP~6.WBK)
12288 .a. -/rwxrwxrwx 6038143 /MYDOCU~1/~WRD3362.tmp (_WRD3362.TMP) (deleted)
14336 .a. -/rwxrwxrwx 6038063 /MYDOCU~1/American XXXXXXXXX of the United States.doc (AMERIC~1.DOC)
53 .a. -/r-xr-xr-x 6038157 /MYDOCU~1/~$y there baby.doc (_$YTHE~1.DOC) (deleted)
15360 .a. -/rwxrwxrwx 6038037 /MYDOCU~1/1997 BUSINESS PLAN.doc (1997BU~1.DOC)
12288 .a. -/rwxrwxrwx 6038135 /MYDOCU~1/Ddeardjhjdufufdujtuthfutretftuefhurdm u.doc
(DDEARD~1.DOC)
21504 .a. -/rwxrwxrwx 6038075 /MYDOCU~1/Professional Report.doc (PROFES~1.DOC)
199380 .a. -/r-xr-xr-x 18440 /RECYCLED/_NFO (deleted)
2681 .a. -/rwxrwxrwx 3499532 /WINDOWS/DESKTOP/MYBRIE~1/1998MONIES.xls
12288 .a. -/rwxrwxrwx 6038117 /MYDOCU~1/U.doc (U.DOC)
Oct 20 2001 06:57:14 2681 m.. -/rwxrwxrwx 3499532 /WINDOWS/DESKTOP/MYBRIE~1/1998MONIES.xls (_998MO~1.XLS)
(deleted)
Oct 20 2001 06:58:18 199380 m.. -r-xr-xr-x 18440 <hde1.img-_NFO-dead-18440>
199380 m.. -/r-xr-xr-x 18440 /RECYCLED/_NFO (deleted)
Oct 20 2001 06:59:18 65 m.c -/r-xr-xr-x 18438 /RECYCLED/desktop.ini (DESKTOP.INI)
Oct 20 2001 07:11:56 14336 m.. -/rwxrwxrwx 6038063 /MYDOCU~1/American XXXXX of the United States.doc
(AMERIC~1.DOC)
53 m.c -/r-xr-xr-x 6038154 /MYDOCU~1/_$_EARD~1.DOC (deleted)
Oct 20 2001 07:16:40 13824 m.. -/rwxrwxrwx 5739038 /MSOFFICE/WINWORD/PHX1295.DOC
Oct 20 2001 07:16:12 12288 m.. -/rwxrwxrwx 6038117 /MYDOCU~1/U.doc (U.DOC)
Oct 20 2001 07:17:20 12288 m.. -/rwxrwxrwx 6038135 /MYDOCU~1/Ddeardjhjdufufdujtuthfutretftuefhurdm u.doc
(DDEARD~1.DOC)
431 m.c -/rwxrwxrwx 367649 /WINDOWS/RECENT/Ddeardjhjdufufdujtuthfutretftuefhurdm u.doc.lnk
(DDEARD~1.LNK)
```

To be thorough, every “*.doc, *.xls, *.ppt and *.txt” file was identified by grep’ing them out of the timeline created by Autopsy. These files were reviewed for potential evidence.

The registry provided a list of the “RECENT” files from WordPad, Excel and PowerPoint. Reviewing these files suggested that the computer was used infrequently because of the large range between the dates associated with the files. The oldest file had an access time dating back to 1997. Not all the files recorded as “RECENT” files were listed in the timeline created by Autopsy. This suggested that the file was never saved.

WordPad had two recent files listed that could be analyzed. These were “C:\Program Files\Accessories\Thecampingtrip.doc” and “C:\Program Files\Accessories\The family trip.doc.” Timeline information on the files is shown below:

```

Nov 29 1999 14:39:40 9728 ..c -/rwxrwxrwx 4734486 /PROGRA~1/ACCESS~1/Thecampingtrip.doc
Nov 29 1999 16:13:12 9728 m.. -/rwxrwxrwx 4734486 /PROGRA~1/ACCESS~1/Thecampingtrip.doc
Jul 02 2001 00:00:00 9728 .a. -/rwxrwxrwx 4734486 /PROGRA~1/ACCESS~1/Thecampingtrip.doc

Nov 30 1999 16:09:36 5120 ..c -/rwxrwxrwx 4734489 /PROGRA~1/ACCESS~1/The family trip.doc
Nov 30 1999 16:09:38 5120 m.. -/rwxrwxrwx 4734489 /PROGRA~1/ACCESS~1/The family trip.doc
Jul 02 2001 00:00:00 5120 .a. -/rwxrwxrwx 4734489 /PROGRA~1/ACCESS~1/The family trip.doc

```

WordPad also had "A:\\Hawaii 2001 agenda.doc" listed as a recent document which showed that the user did occasionally use the floppy drive.

PowerPoint also displayed recent files. Not all of these were found on the system.

```

[HKEY_CURRENT_USER\Software\Microsoft\Office\PowerPoint\7.0\Recent File List]
"File1"="A:\\To XXXX From YYYY.ppt"
"File2"="A:\\XXXX PRESENT.ppt"
"File3"="C:\\My Documents\\DDDEEESSIII.ppt" FOUND
"File4"="A:\\Hawaii back ground.ppt"
"File5"="C:\\My Documents\\XXXX PRESENT.ppt" NOT FOUND
"File6"="C:\\My Documents\\U'I XXXX.ppt" NOT FOUND
"File7"="C:\\MSOFFICE\\POWERPNT\\WIZARDS\\BADNEWS.PPT" FOUND

```

Excel's RECENT file list contained two files on drive C:

```

[HKEY_CURRENT_USER\Software\Microsoft\Excel\7.0\Recent File List]
"File1"="A:\\time trials.xls"
"File2"="C:\\My Documents\\Nominations9999.xls" FOUND
"File3"="C:\\WINDOWS\\Desktop\\My Briefcase\\1998MONIES.xls" FOUND
"File4"=""

```

Timeline information on the files is shown below.

```

Jun 29 2001 17:11:54 135168 ..c -/rwxrwxrwx 6038138 /MYDOCU~1/DDDEEESSIII.ppt
Jun 29 2001 18:02:54 135168 m.. -/rwxrwxrwx 6038138 /MYDOCU~1/DDDEEESSIII.ppt
Jun 29 2001 00:00:00 135168 .a. -/rwxrwxrwx 6038138 /MYDOCU~1/DDDEEESSIII.ppt

Jul 08 1994 00:00:00 27648 m.. -/rwxrwxrwx 7516165 /MSOFFICE/POWERPNT/WIZARDS/BADNEWS.PPT
May 01 1996 13:31:40 27648 ..c -/rwxrwxrwx 7516165 /MSOFFICE/POWERPNT/WIZARDS/BADNEWS.PPT
Mar 03 1997 23:00:00 27648 .a. -/rwxrwxrwx 7516165 /MSOFFICE/POWERPNT/WIZARDS/BADNEWS.PPT

Sep 10 1999 00:00:00 17408 .a. -/rwxrwxrwx 6038115 /MYDOCU~1/Nominations9999.xls
Sep 10 1999 11:05:12 17408 ..c -/rwxrwxrwx 6038115 /MYDOCU~1/Nominations9999.xls
Sep 10 1999 15:59:10 17408 m.. -/rwxrwxrwx 6038115 /MYDOCU~1/Nominations9999.xls

Sep 25 1998 22:32:32 2681 ..c -/rwxrwxrwx 3499532 /WINDOWS/DESKTOP/MYBRIE~1/1998MONIES.xls (deleted)
Oct 20 2001 00:00:00 2681 .a. -/rwxrwxrwx 3499532 /WINDOWS/DESKTOP/MYBRIE~1/1998MONIES.xls (deleted)
Oct 20 2001 06:57:14 2681 m.. -/rwxrwxrwx 3499532 /WINDOWS/DESKTOP/MYBRIE~1/1998MONIES.xls (deleted)

```

Software

The ACER laptop had several applications installed on the system. These applications were determined through registry entries and file system analysis. There were Microsoft applications installed and they are listed below.

MSOffice Applications:

- WinWord
- PowerPoint
- Excel
- Access (never used)
- Exchange

In addition to the Microsoft applications, the system had two other applications that were never used. The applications are listed below.

Other Applications:

- SideKick95 (never used)
- QuickBooks Pro (never used)
- CSW (Map software)

User Files

Documents were found in six different directories:

| | |
|------------------------------------|-------------------------------|
| MSOFFICE\WINWORD\Workshop Letters\ | 10 .doc files |
| \WINDOWS\Favorites | 1 .doc file |
| \WINDOWS\Desktop\My Briefcase | 2 .xls files |
| \Program Files\Accessories | 1 .bmp file, 2 .doc files |
| \WINDOWS\RECENT | References 13 links to files |
| \My Documents | 14 .doc files and 1 .xls file |

Note: There were no user text (.txt) files.

Viruses

During the analysis it was noticed that there were numerous copies of a file called: \MYDOCU~1\Hey there baby.doc (HEYTHE~1.DOC). This raised questions and led to a strings analysis to look for potential virus signatures. Since viruses frequently infect the \MSOFFICE\Templates\normal.dot file, this file was inspected for unusual strings. A sanitized and shortened Autopsy string report follows with some of the suspicious strings shown in bold. These strings were searched for on the McAfee site, http://vil.mcafee.com/dispVirus.asp?virus_k=9931, which returned information about a virus called the WM/CONCEPT.AJ virus, also known as the WW6-Concept virus.

Autopsy string Report (ver 1.50)

File: /MSOFFICE/Templates (TEMPLA~1)/NORMAL.DOT
MD5 of file: f7426d7a838b2bbf8c57f027201ebcff
MD5 of strings: 8dabd5206a5b993479399e98600e9984

Image: /home/install/hde1.img
Image Type: fat16
Date Generated: Wed Jun 12 15:14:39 2002
Investigator: Gregory Leibolt

inode: 9505286
Allocated
uid / gid: 0 / 0
mode: -rwxrwxrwx
size: 16896
num of links: 1
Written: 02.09.2002 08:57:04 (MST)
Accessed: 02.08.2002 00:00:00 (GMT)
Created: 10.06.1995 00:00:00 (MDT)
Name: NORMAL.DOT

Sectors:

537155 537156 537157 537158 537159 537160 537161 537162
537163 537164 537165 537166 537167 537168 537169 537170
537171 537172 537173 537174 537175 537176 537177 537178
537179 537180 537181 537182 537183 537184 537185 537186
537187

File Type: data

Note: The following text, particularly what is in bold, are strings that looked questionable. They related directly to the virus.

see if we're already installedR#i

iMacroCountdo

PayLoad

bInstalled

FileSaveAs

bTooMuchTrouble

k5add FileSaveAs and copies of AutoOpen and FileSaveAs.do

PayLoad is just for fun.do

iWW6IInstance

WW6Infector

WW6I

iWW6IInstance

iWW6IInstance

Abort

MAINdk3this becomes the FileSaveAs for the global templated/i

Global:AAAZAO

MAINdRp That's enough to prove my pointd

Global:autoOpendRi

MacroFile\$

NORMAL.DOT

globMacro\$

fileMacro\$do

Global:AAAZAO

Global:AAAZFS

:PayLoaddg

Global:PayLoad

C:\MSOFFICE\WINWORD\TEMPLATE\NORMAL.DOT

PayLoad

FileSaveAs

AAAZFS

AAAZAO

AutoClose
autoOpen
Microsoft Word Document
MSWordDoc
Word.Document.6
Normal.dot
Microsoft Word for Windows 95

The McAfee site explained that this virus is propagated by infecting documents in Word Versions 6 and 7. It contains the macros, AUTOOPEN, AAAZFS, AAAZAO, and PAYLOAD, which change their names to AAAZAO, AAAZFS, FILESAVEAS and PAYLOAD in Word's Global Template NORMAL.DOT. Using Auto and System Macros activates the virus. These strings were clearly shown in the Normal.dot string output.

Network

Autoexec.bat, config.sys and bootlog.txt did not contain entries suggesting network devices. Since use of the computer would require network devices to be installed at boot-up, this computer, in its current state, could not be connected to a network.

Contents Of File: /AUTOEXEC.BAT

Autopsy string Report (ver 1.50)

File: /AUTOEXEC.BAT
MD5 of file: bf317b36c0cbc868217b820d21e0197a
MD5 of strings: 0f26d67779602855846d142984c2de59
Image: /home/install/hde1.img
Image Type: fat16
Date Generated: Mon Jun 3 15:37:29 2002
Investigator: Gregory Leibolt

inode: 23
Allocated
uid / gid: 0 / 0
mode: -rwxrwxrwx
size: 207
num of links: 1
Written: 08.26.1998 13:53:42 (MDT)
Accessed: 02.08.2002 00:00:00 (GMT)
Created: 08.26.1998 13:53:42 (MDT)
Name: AUTOEXEC.BAT
Sectors:
406243
File Type: ASCII text

@Echo Off
Path C:;\C:\Windows;C:\Windows\System;C:\Windows\Command
Set BLASTER=A240 I5 D1 T4
C:\DOSDRVRS\MIXERSET /p /q
Set DIRCMD=/OEN/P

```
Set WinTemp=C:\Temp
Set Temp=C:\Temp
LoadHigh Doskey > nul
```

Contents Of File: /CONFIG.SYS

Autopsy string Report (ver 1.50)

```
-----
File: /CONFIG.SYS
MD5 of file: b3250a1c45bdf4d4bde61b13a1d7754a
MD5 of strings: 5cc3900960f2a42fe8c1c98a552d2e3a
Image: /home/install/hde1.img
Image Type: fat16
Date Generated: Mon Jun  3 15:37:56 2002
Investigator: Gregory Leibolt
-----
```

```
inode: 8
Allocated
uid / gid: 0 / 0
mode: -rwxrwxrwx
size: 247
num of links: 1
Written: 08.26.1998 14:03:00 (MDT)
Accessed: 02.08.2002 00:00:00 (GMT)
Created: 08.26.1998 14:03:00 (MDT)
Name: CONFIG.SYS
Sectors:
698339
File Type: ASCII text
-----
```

```
Device=C:\Windows\HiMem.Sys /TestMem:Off
Device=C:\Windows\EMM386.Exe NoEms X=D000-DFFF
Rem Device=C:\Windows\EMM386.Exe NoEms X=D000-DFFF I=E000-EFFF FRAME=E000
Buffers=40
Files=20
DOS=High,UMB
Shell=C:\Windows\Command.Com C:\Windows /p
```

Some review of dial-up network activity was performed to determine if the system was connected to the an ISP or other dial-up service.

The Internet Explorer executable, \Program Files\PLUS!\Microsoft Internet\iexplore.exe, had an access time of 7/5/1997.

The \Program Files\PLUS!\Microsoft Internet\history directory only contained two entries that pointed to a file, A:\BARNEY.AU.

Registry entries pointed to some original Microsoft network Dialup configuration.

\Program Files\The Microsoft Network directory did not show any signs of use. The file named ccei.dll showed an access time of 2/9/2002. This is an application extension dll

which means that another application used it, which makes sense because the ccdialer.exe had a 6/17/1996 timestamp.

Access times of AT&T Mail, CompuServe and MCI Mail in the Hyper Terminal entries showed access times of 6/22/ 2001. This appeared to be an attempt to use them or a configuration change.

The final analysis was that the computer was not used to access an ISP or other dial-up service.

Summary

The final results of the forensic analysis showed that the user in question definitely performed a last review of documents on the system. Even though the FAT16 file system does not modify any MAC times when a file is deleted, it was still possible to prove when files were deleted by looking at the time stamps of \RECYCLED_NFO and \RECYCLED\desktop.ini. When a file is deleted to the recycle bin, an entry is made in the \RECYCLED_NFO file. When Files are removed from the recycle bin, the Modify and Create times change on the \RECYCLED\desktop.ini file. In the case of the \WINDOWS\DESKTOP\MYBRIE~1\1998MONIES.xls file, the following steps occurred:

The file was created:

```
Sep 25 1998 22:32:32 2681 .c -/rwxrwxrwx 3499532 /WINDOWS/DESKTOP/MYBRIE~1/1998MONIES.xls (deleted)
```

The file was last accessed:

```
Oct 20 2001 00:00:00 2681 .a -/rwxrwxrwx 3499532 /WINDOWS/DESKTOP/MYBRIE~1/1998MONIES.xls (deleted)
```

The file was last modified:

```
Oct 20 2001 06:57:14 2681 m.. -/rwxrwxrwx 3499532 /WINDOWS/DESKTOP/MYBRIE~1/1998MONIES.xls (deleted)
```

The /RECYCLED/_NFO file was last accessed:

```
Oct 20 2001 00:00:00 199380 .a -/r-xr-xr-x 0 0 18440 /RECYCLED/_NFO (deleted)
```

The \RECYCLED_NFO file was last modified:

```
Oct 20 2001 06:58:18 199380 m.. -r-xr-xr-x 18440 <hde1.img- _NFO-dead-18440>  
199380 m.. -/r-xr-xr-x 18440 /RECYCLED/_NFO (deleted)
```

The \RECYCLED\desktop.ini file was last modified:

```
Oct 20 2001 06:59:18 65 m.c -/r-xr-xr-x 18438 /RECYCLED/desktop.ini (DESKTOP.INI)
```

The \RECYCLED\desktop.ini file was last accessed:

```
Feb 09 2002 00:00:00 65 .a -/r-xr-xr-x 0 0 18438 /RECYCLED/desktop.ini (DESKTOP.INI)
```

This clearly showed that someone using the computer, while it was in the control of the treasurer, deleted the 1998MONIES.xls file on October 20th, 2001. This is because the Modify and Create time stamps changed on the file named \RECYCLED_NFO and the file named \RECYCLED\desktop.ini. The president's activity only changed the access time of \RECYCLED\desktop.ini file when he used the computer on February 9, 2002. Therefore, it was not possible that the president deleted this file.

The deleted 1998MONIES.xls file, which was completely recovered, contained information about funds received from sponsors. Several documents containing lists of possible sponsors were also identified. Armed with the information from these files, corporate investigators would be better able to determine conclusively whether or not the treasurer of the club mismanaged funds.

© SANS Institute 2000 - 2002, Author retains full rights.

Part 2 – Analyze an Unknown Binary

Binary Details Summary

The following initial information related to the binary artifact is as follows:

Name of the program/file found on the system:

Extracted sn.dat and sn.md5 from the zip file named sn.zip.

File/MACTime information (last modified, last accessed, and last create time):

The artifact, sn.dat and the related MD5 sum information in a file called sn.md5 were received in a zip file. The only date associated with the artifact is a modified time stamp of 4/11/2002 09:29 AM displayed in the zip archive file. Due to the fact that the artifact was obtained in a zip file, there was no way to determine the last time the binary was accessed.

File owner(s) – (user and/or group):

The artifact, sn.dat and the related MD5 sum information in a file called sn.md5 were received in a zip file. No owner or group information could be determined. However, this binary was determined to be a sniffer. For a sniffer program to run on a Linux system in promiscuous mode, it must be run as root.

File size (in bytes):

399,124 bytes

MD5 hash of the file (the screen shot is printed later in this section):

0e954f43fd73f56e812a7285f32ef1d3

Program Identification

After downloading the zip file and extracting sn.dat and sn.md5, the following steps were taken:

- Verified that the MD5 sum of the downloaded sn.dat matched what was in sn.md5. The command used was: `md5sum sn.dat`.
- The `file` command was used to try to determine the file type of the artifact, sn.dat. The actual command was: `file sn.dat`. The output of this command is shown below:

sn.dat: ELF 32-bit LSB executable, Intel 80386, version 1, statically linked, stripped

The result of the file command showed that sn.dat was probably a Linux [LSB (Linux Standard Base)] 386 binary, statically linked (so no library dependencies were required) and it was stripped to remove comments.

Strings Analysis

The next step taken was to perform a strings analysis on sn.dat:
strings sn.dat > sn.strings

Viewed strings output with vi. Interesting clues:

```
\* The END */
priv 1.0
ADMsniff %s <device> [HEADERSIZE] [DEBUG]
ex : admsniff le0
..ooOO The ADM Crew OOoo..
cant open pcap device :<
init_pcap : Unknown device type!
ADMsniff %s in libpcap we trust !
credits: ADM, mel , ^pretty^ for the mail she sent me
The_l0gz
@(#) $Header: pcap-linux.c,v 1.15 97/10/02 22:39:37 leres Exp $ (LBL)
@(#) $Header: pcap.c,v 1.29 98/07/12 13:15:39 leres Exp $ (LBL)
@(#) $Header: savefile.c,v 1.37 97/10/15 21:58:58 leres Exp $ (LBL)
@(#) $Header: bpf_filter.c,v 1.33 97/04/26 13:37:18 leres Exp $ (LBL)
```

“ADMsniff %s <device> [HEADERSIZE] [DEBUG]” appeared to be a usage string for a “ADMsniff. ” ..ooOO The ADM Crew OOoo.. “ and the other references suggested that this, or part of this binary was ADMsniff by the ADM Crew.

Key words associated with this program were ADM, ADMsniff, Crew, priv 1.0, The_l0gz.

Binary Duplication

The following steps were taken:

- Downloaded ADMsniff from <http://adm.freelsd.net/ADM/>.
- Compiled ADMsniff with the Compression Log File Option (COMPFLAGS = -DCOMPRESS, COMPLIB = -lz) because it would be the most optimal way to operate a sniffer and a logical choice for the hacker to use. Log files, on a busy network can grow quickly. Someone who is trying to hide sniffer activity would wisely use compression on the log file to reduce exposure of the activity. The -static option was added to the CFLAGS section of the Makefile to statically link the libraries. This reduced potential problems of trying to run the program on a system that did not have the required libraries.

After compiling, the ADMsniff-1 the file size was 1,485,617 bytes. This was 1,086,493 bytes larger than sn.dat, which was 399,124 bytes. The output of the file command said that sn.dat was stripped, so this was also done on ADMsniff-1 and the file size was reduced to 389,912 bytes. This, however, was smaller than the artifact by 9,212 bytes.

Obviously, something was not the same. The “strings” command was used to compare the strings from sn.dat with those from the ADMsniff-1 binary.

Strings of the ADMsniff-1 file looked VERY much like those in the artifact. Much of the same information was in the strings output, but one noticeable difference was that the ADMsniff-1 file contained comments about compiling it with the Compression support. The sn.dat binary did not! The specific strings from the ADMsniff-1 binary are shown in blue below:

```
*-----*
\*   The END       */
priv 1.0
ADMsniff %s <device> [HEADERSIZE] [DEBUG]
ex  : admsniff le0
..ooOO The ADM Crew OOoo..
cant open pcap device :<
init_pcap : Unknown device type!
ADMsniff %s  in libpcap we trust !
credits: ADM, mel , ^pretty^ for the mail she sent me
You compiled ADMsniff with compression support, don't
forget about the log flushing tricks (see README).
The_l0gz
@(#) $Header: pcap-linux.c,v 1.15 97/10/02 22:39:37 leres Exp $ (LBL)
read: %s
SIOCGSTAMP: %s
malloc: %s
socket: %s
bind: %s: %s
SIOCGIFHWADDR: %s
unknown physical layer type 0x%x
SIOCGIFMTU: %s
SIOCGIFFLAGS: %s
SIOCSIFFLAGS: %s
linux socket: %s
linux SIOCSIFFLAGS: %s
@(#) $Header: pcap.c,v 1.29 98/07/12 13:15:39 leres Exp $ (LBL)
%s: %s
@(#) $Header: savefile.c,v 1.37 97/10/15 21:58:58 leres Exp $ (LBL)
```

The next step was to compile ADMsniff-1 without the logfile compression support. Recompiling without the compression and stripping the binary created ADMsniff-1 with a size of 346,128 bytes, which was 52,996 bytes smaller than the artifact. This was very close, but not the same, so a study of different compiling options was done to see if other compilations should be tried. The -g (debugging option) was used in another compilation to see if the byte sizes were any better. This didn't produce any better results.

A careful review of the strings from both binaries could possibly provide some information as to what the differences were. This was performed by sorting and uniq'ing the strings of each binary. This output was then compared by using the UNIX “comm” utility.

The exact commands are listed below:

```
Sort sn.strings | uniq > sn.sort
Sort ADMsniff.strings | uniq > ADMsniff.sort
comm -3 SN.sort ADMsniff.sort
```

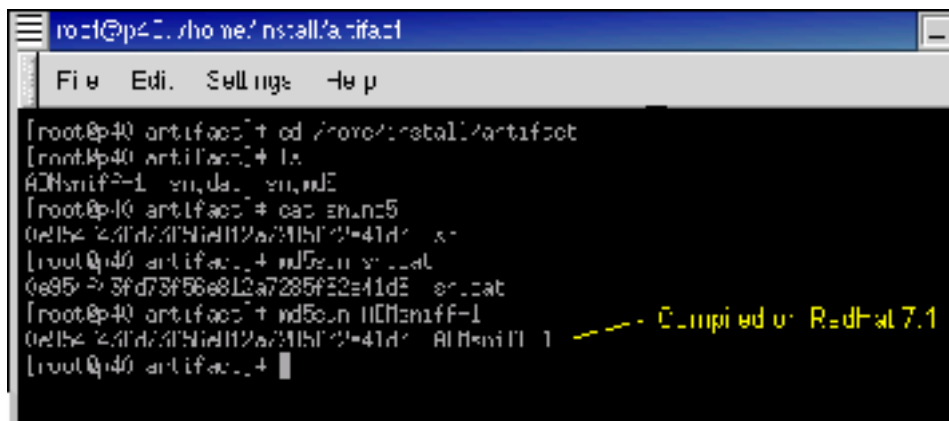
The following output was a small subset of the string differences only showing some of the more interesting differences. The indented (blue) strings were in ADMsniff.sort and the strings on the left side were in the artifact sn.sort.

Some of the more identifiable strings were noted (highlighted in bold) in case they could be used in future string searches of binaries. The ANSI references and some of the other strings appeared to be from libraries.

```
%a %b %e %H:%M:%S %Z %Y
ANSI_X3.4-1986// ANSI_X3.4-1968//
ANSI_X3.4// ANSI_X3.4-1968//
cannot load auxiliary '%s' because ofempty dynamic string token substitution
CP367// ANSI_X3.4-1968//
CSASCII// ANSI_X3.4-1968//
                                ELF file class not 32-bit
                                ELF file class not 64-bit
                                ELF file machine architecture does not match
Filters not supported with LD_TRACE_PRELINKING
                                glibc-ld.so.cache
glibc-ld.so.cache.1.1
                                _GNU_nonoption_argv_flags_=
HOSTALIASES
IBM367// ANSI_X3.4-1968//
ISO_646.IRV:1991// ANSI_X3.4-1968//
ISO646-US// ANSI_X3.4-1968//
ISO-IR-6// ANSI_X3.4-1968//
LD_AOUT_LIBRARY_PATH
LD_AOUT_PRELOAD
LOCALDOMAIN
only ET_DYN and ET_EXEC can be loaded
RESOLV_HOST_CONF
```

Assuming that different libraries or library versions were used, ADMsniff was compiled on a different system running Redhat 7.2 using the 2.4.7-10 kernel. The Makefile was edited to use the “-static” option on the CFLAGS section to statically link the libraries. Log file compression was not used. After compiling, the binary was stripped, which produced an exact copy of the artifact, sn.dat, the same size and MD5 sum. Finally, it was confirmed that the artifact, sn.dat was indeed, ADMsniff-1!

MD5 sum values for SN.DAT and ADMsniff-1:



```
root@p40:~/home/install/artifact
File Edit Settings Help
[root@p40 artifact]# cd /home/install/artifact
[root@p40 artifact]# ls
ADMsniff-1  sn.dat  sn.md5
[root@p40 artifact]# cat sn.md5
0e1b4 24fd73f56e812a7285f32e41de  sn.dat
[root@p40 artifact]# md5sum sn.dat
0e1b4 24fd73f56e812a7285f32e41de  sn.dat
[root@p40 artifact]# md5sum ADMsniff-1
0e1b4 24fd73f56e812a7285f32e41de  ADMsniff-1
[root@p40 artifact]#
```

Program Description

As stated before, sn.dat and the related MD5 sum information in a file called sn.md5, were received in a zip file from the following URL: <http://www.giac.org/gcfa/sn.zip>. Due to the fact that the artifact was obtained in a zip file, there was no way to determine the last time it was used. Since the file was not on an actual system when it was obtained, no actual MAC times were available for the file. The only date associated with the artifact was a modified time stamp displayed in the zip file of 4/11/2002 at 09:29 AM.

Neither owner nor group information could be determined. However, for a sniffer program such as ADMsniff-1 to run on a Linux system in promiscuous mode, it must be run as root. This could be done by setting the SUID bit on the file, which would require that the file be owned by root. It could also be run directly by the root ID. The permission flags of sn.dat were not available from the zip archive file, so there was no way to know exactly how the program was set up to run. However, it must have had the execute bit set to run on the system.

The sole purpose of the sn.dat program was to gather TCP network packet data, specifically, when the source port or destination port matches one of the ports listed in a list that the author calls “coolports.” The actual services that usually run on these ports all use IDs and password authentication mechanisms. In other words, an ID and a password must be supplied in order to use the service. By logging the data packets from these ports, IP addresses, IDs and passwords can be obtained and used on the associated systems. The IP addresses listed in the log file used by sn.dat, must be investigated as leads to potentially compromised systems.

List of “coolports” used by ADMsniff:

| Port Number | Usual service provided on the port |
|-------------|---|
| 21 | FTP |
| 23 | Telnet |
| 109 | pop-2 |
| 110 | pop-3 |
| 143 | Imap |
| 512 | Exec |
| 513 | Login |
| 514 | Shell |
| 1521 | Oracle |
| 31337 | BackOrifice and many other hacker tools |

Forensic Details

Analysis of the ADMsniff-1 source code showed that when run, a log file called “The_l0gz” was created in the working directory of the user who started the program. This means that the user could be in an obscure directory when sn.dat was started. The whole system must be searched to find the log file(s). The user could have started and stopped sn.dat numerous times, placing the log file in different locations. On a live system, the “find” command could be used. On a forensic image, a “body” file, or better yet, a MAC timeline file could be *grep*'d. Since sn.dat was compiled without log file compression, the log file would be readable using *cat* or *vi*.

If the artifact, sn.dat used shared libraries in any way, it would be important to know. Since an ADMsniff-1 was made to be an exact copy of sn.dat, it was used to evaluate interaction with shared libraries. To verify that ADMsniff-1 did not use dynamically linked libraries, the *gdb* debugging tool was used. ADMsniff-1 was first compiled with the *-g* and *-static* options to support debugging statically linked libraries. After compilation, the program was not stripped. *Gdb* was started and the file to run was specified at the *gdb* prompt using the command “file ADMsniff-1.” ADMsniff-1 required an interface name as an argument. This was specified using the command, “set args eth0.” The command, “show confirm” was then issued to confirm potentially dangerous operations. A breakpoint at the function “main” was set up using the command “break main.” The program was then run using the command “run,” which started ADMsniff-1 and stopped it at the break point, *main()*. The command “info functions” was then used to see what functions were loaded into memory. This clearly showed all the library functions and files that were incorporated into ADMsniff-1 at compile time. The command “show sources” was also very helpful in displaying similar information. To verify that no shared libraries were loaded, the command “info sharedlibrary” was run, which stated “No shared libraries loaded at this time.” To be 100 percent sure that this meant no shared libraries were used, the ADMsniff-1 program was recompiled using shared libraries (without the *-static* option).

The exact same procedure was followed and when the command “info sharedlibrary” was run, it stated:

| From | To | Syms Read | Shared Object Library |
|------------|------------|-----------|-----------------------|
| 0x40020000 | 0x4002d330 | Yes | /usr/lib/libz.so.1 |
| 0x4002e000 | 0x401522e8 | Yes | /lib/libc.so.6 |
| 0x40000000 | 0x40016c50 | Yes | /lib/ld-linux.so.2 |

The tests with the gdb debugger confirmed that ADMsniff-1 did not interact with or modify any system files including shared libraries.

Legal Issues

There is no administrative reason to have ADMsniff running. It was designed to gather data specifically to help a “hacker” gain entry into systems. It is also important to note that the artifact was named sn.dat, not ADMsniff-1, the default name used when compiling the program. This suggested some attempt was made to hide the true nature of the sniffer program.

Proof that ADMsniff was run

If proof was obtained that the artifact was run, for example, a snapshot of the process table (ps) or the actual log file, “The_l0gz” with associated MAC times was collected as evidence, then this proof could be used in court to possibly prosecute under both federal and state laws.

The federal laws that apply in this case are covered under the Computer Fraud and Abuse Act, 18 U.S.C. § 1030 available at:

http://www.usdoj.gov/criminal/cybercrime/1030_new.html

This law discusses issues (all of which may pertain to the activity surrounding the use of ADMsniff) such as:

1. Exceeding authorized access
The user gained access to the system to run sn.dat. More than likely, this would have been unauthorized access and punishable by law.
2. Causing loss or damage
Damage could be caused by running sn.dat. This could be the filling up of partitions because the log file grew so large, or someone else may have found and used the program as well to cause other damage.
3. Reckless disregard of a substantial and unjustifiable risk from the action if the hacker was an outsider (no authority to access)

The hacker may not have intended to cause problems with the use of the program, but the process of placing the program on the system and using it without thought of the consequences could be proved reckless.

4. Negligent conduct if the hacker was an outsider (no authority to access)
The hacker could possibly be charged with negligent conduct because the use of a tool like sn.dat, even if its purpose was unknown to the hacker, would be negligent action.
5. Causing the loss of \$5,000 or more within a 1 year period
The costs of forensic investigation of the system running sn.dat and all the systems possibly identified in log files, along with the clean-up or restoring from backups could possibly cost \$5,000 or more. If the cost was not at least \$5,000 dollars, legal charges would not be filed.
6. Traffics in any password or similar information
Finally, the hacker may have shared ID and password information, thus adding on another charge.

The Wiretap Act, 18 U.S.C. § 2511 available at:

<http://www.usdoj.gov/criminal/cybercrime/usc2511.htm>

covers real-time interception of electronic communications and applies to the installation and running of a sniffer such as ADMsniff without permission.

A unauthorized, non employee, hacker caught running sn.dat would definitely be charged under 18 U.S.C. § 2511 because there are no exceptions that could apply. If the hacker was an employee or had other legal permission for being on the system, then certain exceptions may apply.

The Electronic Communications Privacy Act (ECPA), 18 U.S.C. § 2701 available at:

<http://www.usdoj.gov/criminal/cybercrime/usc2701.htm>

also pertains to sniffers being installed.

Similar to the Wiretap Act, if the hacker did not have legal access to the system, they may also be charged under the ECPA.

Note that the Wiretap Act has many exceptions that allow use of sniffers and must be taken into account when reviewing the actions and intent of someone using sniffers such as ADMsniff. Particularly, if the hacker had legal access to the system, three key exceptions that may apply in this case are:

- The Provider Exception, 18 U.S.C. § 2511 (2)(a)(i),
- The Consent Exception, 18 U.S.C. § 2511 (2)(c), and
- The Computer Trespasser Exception, 18 U.S.C. § 2511 Chapter 1119 (21).

The (ECPA) also has exceptions which must also be taken into account, but do not appear as directly related in this case.

Florida law also has statutes, which would apply and also has exceptions to be addressed. It is available under chapter 934 at:

http://www.leg.state.fl.us/Statutes/index.cfm?mode=View%20Statutes&SubMenu=1&App_mode=Display_Statute&Search_String=&URL=CH0934/Ch0934.HTM

No Proof that ADMsniff was run

Without proof that ADMsniff was run on the system the hacker could still be prosecuted, depending on the circumstances, under the Computer Fraud and Abuse Act, 18 U.S.C. § 1030 available at:

http://www.usdoj.gov/criminal/cybercrime/1030_new.html.

This law discusses the following:

1. Exceeding authorized access
Even though the program was not run, the hacker had to gain access to the system, which in itself, is possibly illegal.
2. Causing loss or damage
Though it is not likely in the case of sn.dat, it might have been possible for the hacker to have caused damage in the process of implementing the program.
3. Reckless disregard of a substantial and unjustifiable risk from the action if the hacker was an outsider (no authority to access)
Once again, the attempt to install a program such as sn.dat is likely to fall under reckless disregard of the risks involved in this action.
4. Negligent conduct if the hacker was an outsider (no authority to access)
Even if the hacker did not know the purpose of the sn.dat program, the attempt to install the program could be viewed as negligent conduct.
5. Causing the loss of \$5,000 or more within a 1 year period
It may be hard to imagine that the installation of the sn.dat program could cause \$5,000 in damage if the program was not used. However, a fair amount of investigation may need to be performed to determine that the program was not used. This could be costly.

Employee use of ADMsniff

If the person who installed ADMsniff was an employee and no laws were broken, then the company would have to handle the case according to company policy. In our

company, the policy documentation clearly allows “monitoring” by system or network administrators under certain conditions.

Sanitized excerpts of the corporate policy that permits monitoring is shown below:

Audit section:

.....as well as session and network activity, is critical in preventing and detecting intrusions that could disrupt business operations on Company networks. This monitoring process also allows administrators to retrace an intruder's activity and may help correct any damage caused by the intrusion.

Network Administration section:

Network administrators who monitor data traffic as part of an assigned job responsibility are not to disclose the content of that traffic to anyone who is not authorized to have it.

Banners are also used extensively in the company, which explicitly state that monitoring is performed for administrative and security reasons. An example is shown below:

Warning: This system is restricted to “Company Name” authorized users for business purposes. Unauthorized access is a violation of the law. This service may be monitored for administrative and security reasons. By proceeding, you consent to this monitoring.

If the person who installed ADMsniff was not allowed to, per job function, the company could administer appropriate punishment. In this case, ADMsniff is not a very useful tool for most administrative purposes. An employee would have a difficult time arguing that it was used for a valid purpose.

Interview Questions

During an interview with the person suspected of installing and running sn.dat, the following questions might lead to proof of culpability.

1. How did you manage to identify where the network problems were coming from?
The suspect may divulge information only the actual user of ADMsniff would know.
2. Do you have a favorite sniffer utility?
The suspect may mention ADMsniff.
3. Do you know what kind of sniffers hackers use?
The suspect may mention ADMsniff which would be suspicious, since there are so many sniffer utilities available on the Internet. Any mention of ADMsniff would be warrant further questioning.

4. How did you know that running this program would not adversely affect the system?
The suspect may divulge information about the log file: "The_I0gz."
5. What file contains the information you used to evaluate the network problems?
The suspect may divulge information about the log file: "The_I0gz."
6. When were you last on the computer and what did you do?
The suspect may provide information which gives clues as to IDs and times the person was on the system, thus placing the person on the system at specific dates and times that could possibly correlate to the use of ADMsniff.
7. Have you downloaded any files from the Internet recently? What were they?
The suspect may divulge incriminating information.
8. Do you have any idea what this sn.dat file is and would it be safe to run?
The suspect may divulge information that only the actual user of ADMsniff would know.

Additional Information

For additional information the reader may wish to access the following sites:

ADMsniff available at:
<http://adm.freelsd.net/ADM/>

Federal Law 1030 available at:
http://www.usdoj.gov/criminal/cybercrime/1030_new.html

Federal Law 2511 available at:
<http://www.usdoj.gov/criminal/cybercrime/usc2511.htm>

Federal Law 2701 available at:
<http://www.usdoj.gov/criminal/cybercrime/usc2701.htm>

Florida State Law available at:
http://www.leg.state.fl.us/Statutes/index.cfm?mode=View%20Statutes&SubMenu=1&App_mode=Display_Statute&Search_String=&URL=CH0934/Ch0934.HTM

Summary

In cases where artifacts are found on systems, the only way to truly know the exact nature of the actions performed by the hacker, is to know exactly what the programs in question do. In the example provided for this exercise, proper binary analysis identified the artifact as a sniffer, which used log files to store captured ID and password data. This fact would automatically lead one to look for other compromised systems in the

likelihood that the hacker used the authentication information to further exploit the environment. Had the binary analysis not been performed, the intent and methodology of the hacker may not have been realized and compromised systems may not have been identified.

© SANS Institute 2000 - 2002, Author retains full rights.

Part 3 – Legal Issues of Incident Handling – Wiretap Statute

This section is a discussion of federal law, definitions of the word “incident,” the use of banners, the 2001 Patriot’s Act, and state law related to the monitoring of network traffic. Every time a network analyzer or “sniffer” tool is used, the Wiretap Statute governs their legal use. Misuse could have dire consequences.

Federal Law

The wiretap statute, Wiretap Act, 18 U.S.C. § 2511 covers real-time interception of electronic communications. When system administrators monitor network communications with sniffers, this federal law (as well as related state laws) govern them.

The federal law, available at:

<http://www.usdoj.gov/criminal/cybercrime/usc2511.htm> states that unless an exception applies, the interception of electronic communication is illegal. Excerpts from the law are as follows:

Interception and disclosure of wire, oral, or electronic communications prohibited

(1) Except as otherwise specifically provided in this chapter any person who—

(a) intentionally intercepts, endeavors to intercept, or procures any other person to intercept or endeavor to intercept, any wire, oral, or electronic communication;...

The law also explains that disclosing information knowingly obtained through interception of electronic communication is illegal, again, unless an exception applies. Subsections (c), (d), and (e) of this same section 1 of the law says that anyone who:

(c) intentionally discloses, or endeavors to disclose, to any other person the contents of any wire, oral, or electronic communication, knowing or having reason to know that the information was obtained through the interception of a wire, oral, or electronic communication in violation of this subsection;

(d) intentionally uses, or endeavors to use, the contents of any wire, oral, or electronic communication, knowing or having reason to know that the information was obtained through the interception of a wire, oral, or electronic communication in violation of this subsection; or

(e)

(i) intentionally discloses, or endeavors to disclose, to any other person the contents of any wire, oral, or electronic communication, intercepted by means authorized by sections 2511(2)(a)(ii), 2511(2)(b) to (c), 2511(2)(e), 2516, and 2518 of this chapter, (ii) knowing or having reason to know that the information was obtained through the interception of such a communication in connection with a criminal investigation, (iii) having obtained or received the information in connection with a criminal investigation, and (iv) with intent to improperly obstruct, impede, or

interfere with a duly authorized criminal investigation, shall be punished as provided in subsection (4) or shall be subject to suit as provided in subsection (5).

It must be noted that for those in positions as system administrators, certain exceptions do apply! Section 2 of this law, subsection (a) says:

(2)

(a)

*(i) It shall not be unlawful under this chapter for an operator of a switchboard, or an officer, employee, or agent of a provider of wire or electronic communication service, whose facilities are used in the transmission of a wire or electronic 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 his service or to the protection of the rights or property of the provider of that service**, except that a provider of wire communication service to the public shall not utilize service observing or random monitoring except for mechanical or service quality control checks.*

It is very important to note the conditions in which the exceptions apply. The defining words that are used are: **“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 his service or to the protection of the rights or property of the provider of that service.”** This means that there must be a specific reason to intercept and/or use intercepted data. This reason must be related to a necessary incident pertaining to a service or in order to “protect” rights or property. In particular, this language permits interception and disclosure during the ordinary course of business when the interception is unavoidable. The example given in the SANS Institute Track 8.4 course text is a switchboard operator or a repairman hearing snippets of conversation during normal operations. This would suggest that a system administrator would see snippets of intercepted electronic communications during normal maintenance or repair work.

Definitions of the Word “Incident”

It would be useful to discuss the meaning of the word “incident.” The word “incident” is defined by the online Merriam-Webster dictionary available at:

<http://www.m-w.com/cgi-bin/dictionary> as:

*Main Entry: ¹**in-ci-dent***

Pronunciation: 'in(t)-s&d&nt, -"dent

Function: noun

Etymology: Middle English, from Middle French, from Medieval Latin incident-, incidens, from Latin, present participle of incidere to fall into, from in- + cadere to

fall -- more at CHANCE

Date: 15th century

1 : something dependent on or subordinate to something else of greater or principal importance

2 a : an occurrence of an action or situation that is a separate unit of experience :

HAPPENING b : an accompanying minor occurrence or condition :

CONCOMITANT

3 : an action likely to lead to grave consequences especially in diplomatic matters <a serious border incident>

It is the definition “2a” above that applies the best to what is discussed in this paper. The only computer related definition of “incident” is from the CERT web site available at: http://www.cert.org/tech_tips/incident_reporting.html#1.A, which states:

The CERT/CC's incident definition

The CERT Coordination Center is interested in receiving reports of security incidents involving the Internet. A good but fairly general definition of an incident is:

The act of violating an explicit or implied security policy.

Unfortunately, this definition relies on the existence of a security policy that, while generally understood, varies between organizations. We have attempted to characterize below the types of activity we believe are widely recognized as being in violation of a typical security policy. These activities include but are not limited to:

- o attempts (either failed or successful) to gain unauthorized access to a system or its data*
- o unwanted disruption or denial of service*
- o the unauthorized use of a system for the processing or storage of data*
- o changes to system hardware, firmware, or software characteristics without the owner's knowledge, instruction, or consent*

Naturally, there are a wide variety of “incidents” that might occur on a network or in a computer that could warrant the use of interception of electronic communication to obtain information important to resolution. For example, network congestion, performance analysis, network tuning and statistical reporting are some of the many valid “incidents.” It should be kept in mind, however, that in court a prosecutor might challenge the definition of the word “incident.” Basically, the wording means that the “provider exception 18 U.S.C. § 2511 (2)(a)(i), does not permit providers to perform unlimited monitoring. Monitoring must be tailored to minimize the interception and disclosure of private communications unrelated to the investigation.

The Use of Banners

Many companies use a technique called a “banner” to notify users of the legal implications related to the use of various computer services. Typically, a banner is displayed to the user, and the user is required to validate that the messages was seen by pressing the enter key. It is usually a message which specifically states that constant monitoring can be performed for administrative and security reasons. The banner advises the user that by proceeding, the user is consenting to this monitoring, and the user has given permission to be monitored.

This situation is covered under the “consent” exception in the federal law 18 U.S.C. § 2511 (2)(c) available at: <http://www.usdoj.gov/criminal/cybercrime/usc2511.htm>, which states:

(2)

(c) It shall not be unlawful under this chapter for a person acting under color of law to intercept a wire, oral, or electronic communication, where such person is a party to the communication or one of the parties to the communication has given prior consent to such interception.

The use of banners provides expanded abilities for authorized system administrators and security specialists to monitor network and system activity. The use of these banners comes with some technical difficulties, which must be considered. Specifically, difficulties relate to the ability to place banners on all appropriate ports and services. There are 65,535 TCP ports and another 65,535 UDP ports. Technically and logistically, it is impossible to place banners on all these ports.

For example, the REXECD service listens on port 512. It would be difficult to place a banner on this service and even more difficult to require acknowledgment. Therefore, if this was the only means by which a hacker accessed a system, it could possibly be argued that no banner was seen and that access to the system was “open to the public.” This could raise related legal questions and should be evaluated with corporate attorneys.

The 2001 Patriot’s Act

The new 2001 USA Patriot’s Act, which was passed in the fall of 2001, expanded federal law 18 U.S.C. § 2511 to include SEC. 217. Interception of Computer Trespasser Communications. It is available at:

<http://thomas.loc.gov/cgi-bin/query/D?c107:1:./temp/~c1073E99HG:e64245:>

The section deals with what is termed a “computer trespasser” and states:

Chapter 119 of title 18, United States Code, is amended--

‘(21) ‘computer trespasser’--

`(A) means a person who accesses a protected computer without authorization and thus has no reasonable expectation of privacy in any communication transmitted to, through, or from the protected computer; and

`(B) does not include a person known by the owner or operator of the protected computer to have an existing contractual relationship with the owner or operator of the protected computer for access to all or part of the protected computer.'

Therefore, a system administrator may monitor all activity related to a "trespasser." Law enforcement may also be called in to assist in this activity.

18 U.S.C. § 2511(2)(i) further states:

`(i) It shall not be unlawful under this chapter for a person acting under color of law to intercept the wire or electronic communications of a computer trespasser transmitted to, through, or from the protected computer, if--

`(I) the owner or operator of the protected computer authorizes the interception of the computer trespasser's communications on the protected computer;

`(II) the person acting under color of law is lawfully engaged in an investigation;

`(III) the person acting under color of law has reasonable grounds to believe that the contents of the computer trespasser's communications will be relevant to the investigation; and

`(IV) such interception does not acquire communications other than those transmitted to or from the computer trespasser.'

State Law

In addition to the federal laws, systems administrators must be sure that their activity complies with state laws. In the state of Florida, for example, the laws that apply are under chapter 934. These laws read very much like the federal laws as shown by descriptions of two of the exception clauses displayed below from 934.03 available at: http://www.leg.state.fl.us/Statutes/index.cfm?mode=View%20Statutes&SubMenu=1&App_mode=Display_Statute&Search_String=&URL=CH0934/Ch0934.HTM

(2)(a)1. It is lawful under ss. 934.03 - 934.09 for an operator of a switchboard, or an officer, employee, or agent of a provider of wire or electronic communication service whose facilities are used in the transmission of a wire or electronic communication, to intercept, disclose, or use that communication in the normal course of his or her employment while engaged in any activity which is a necessary incident to the rendition of his or her service or to the protection of the rights or property of the provider of that service, except that a provider of wire

communication service to the public shall not utilize service observing or random monitoring except for mechanical or service quality control checks.

(d) It is lawful under ss. 934.03 - 934.09 for a person to intercept a wire, oral, or electronic communication when all of the parties to the communication have given prior consent to such interception.

Summary

Administrators definitely need tools such as network sniffers to help debug network problems and to ensure that systems are operating correctly. This being said, it becomes the responsibility of each administrator to understand when it is legal to use these tools. Sniffing on the wrong network interface could easily be an illegal act. The same applies to constant or long term network monitoring. It is recommended that system administrators do not share private information that they obtain through monitoring. They should only perform network monitoring with sniffers when absolutely necessary. The interpretation of the specific wording or meaning of many laws is often challenging to understand and frequently argued in court. It, therefore, behooves administrators to seek legal counsel when questionable situations arise. By doing so, system administrators would reduce their exposure to potentially breaking the law.

List of References

PART 1:

GIAC Certified Forensic Analyst (GCFA) Practical Assignment Version 1.0:
http://www.giac.org/GCFA_assignment.php.

Windows OS version available at:
http://www.theosfiles.com/os_windows/ospq_w95.htm

WM/CONCEPT.AJ Virus available at:
http://vil.mcafee.com/dispVirus.asp?virus_k=9931

How the Recycle Bin Stores Files (Q136517) available at:
<http://support.microsoft.com/default.aspx?scid=kb;EN-US;q136517>

Windows 95 Y2K available at:
<http://www.microsoft.com/windows/downloads/bin/w95/y2kw95.txt>

PART 2:

Assignment files sn.dat and sn.md5 available in a zip file at:
<http://www.giac.org/gcfa/sn.zip>

ADMSniff available at:
<http://adm.freelsd.net/ADM/>

Federal Law 1030 available at:
http://www.usdoj.gov/criminal/cybercrime/1030_new.html

Federal Law 2511 available at:
<http://www.usdoj.gov/criminal/cybercrime/usc2511.htm>

Federal Law 2701 available at:
<http://www.usdoj.gov/criminal/cybercrime/usc2701.htm>

Florida State Law available at:
http://www.leg.state.fl.us/Statutes/index.cfm?mode=View%20Statutes&SubMenu=1&App_mode=Display_Statute&Search_String=&URL=CH0934/Ch0934.HTM

PART 3:

Federal Law 2511 available at:
<http://www.usdoj.gov/criminal/cybercrime/usc2511.htm>

SANS Institute Track 8.4 Textbook:
Forensics Frameworks and Best Practices – Managerial and Legal Issues, p. 1-31.

Definition of Incident available at:
<http://www.m-w.com/cgi-bin/dictionary>

Computer Related Definition of Incident available at:
http://www.cert.org/tech_tips/incident_reporting.html#1.A

2001 Patriot's Act available at:
<http://thomas.loc.gov/cgi-bin/query/D?c107:1:./temp/~c1073E99HG:e64245:>

Florida State Law available at:
http://www.leg.state.fl.us/Statutes/index.cfm?mode=View%20Statutes&SubMenu=1&App_mode=Display_Statute&Search_String=&URL=CH0934/Ch0934.HTM

Appendix A: Windows 95 file attribute tests

In order to understand and correlate the behavior of MAC times in the fat16 filesystem and Windows 95, a test system was created by installing Windows 95 and performing some tests creating and deleting files. The table below shows various steps at various times and one can see how Windows 95 and the file system behave during these operations.

| TIME STAMP | FILE ACTIVITY | ACTION |
|----------------------|---|---|
| May 08 2002 17:28:00 | May 08 2002 17:28:00 268 m. -/rwxrwxrwx 897542 /WINDOWS/STARTM~1/PROGRAMS/ACCESS~1/Notepad .lnk (NOTEPAD.LNK) | 5/8/02 Started NotePad at 17:28:00 PM EDT |
| May 08 2002 17:29:00 | May 08 2002 17:29:00 245 ..c -/rwxrwxrwx 2509322 /WINDOWS/RECENT/testfile2.lnk (TESTFI~1.LNK) | 5/8/02 Saved testfile2.txt at 17:29:00 PM EDT |
| | 52 ..c -/rwxrwxrwx 23 /testfile2.txt (TESTFI~1.TXT) | 5/8/02 Related activity to saving testfile2.txt at 17:29:00 PM EDT |
| May 08 2002 17:29:02 | May 08 2002 17:29:02 52 m. -/rwxrwxrwx 23 /testfile2.txt (TESTFI~1.TXT) | 5/8/02 Related activity to saving testfile2.txt at 17:29:00 PM EDT |
| | 245 m. -/rwxrwxrwx 2509322 /WINDOWS/RECENT/testfile2.lnk (TESTFI~1.LNK) | 5/8/02 Related activity to saving testfile2.txt at 17:29:00 PM EDT |
| May 08 2002 17:30:02 | May 08 2002 17:30:02 61 ..c -/rwxrwxrwx 25 /testfile3.txt (<u>ESTFI~2.TXT</u>) (deleted) | 5/8/02 Saved testfile3.txt at 17:30:00 PM EDT (To delete later) (Note: On 5/9/02 , testfile3.txt was deleted to the recycle bin at 11:36:00 AM EDT, BEFORE THIS IMAGE WAS TAKEN ON 5/9/02) |
| | 245 ..c -/rwxrwxrwx 2509324 /WINDOWS/RECENT/testfile3.lnk (TESTFI~2.LNK) | 5/8/02 Related activity to saving testfile3.txt at 17:30:00 PM EDT Note Link created here. |
| | 61 ..c -/rwxrwxrwx 256519 /RECYCLED/_C0.TXT (deleted) | 5/9/02 Deleted testfile3.txt to recycle bin at 11:36:00 AM EDT 5/9/02 Emptied recycle bin at 11:37:00 |
| | 61 ..c -/rwxrwxrwx 256519 <testwin95.img-_C0.TXT-dead-256519> | Related activity to deleting file to recycle bin at 11:36:00 AM EDT |
| | 61 ..c -/rwxrwxrwx 25 <testwin95.img-_ESTFI~2.TXT-dead-25> | Related activity to deleting file to recycle bin at 11:36:00 AM EDT |
| May 08 2002 17:30:04 | May 08 2002 17:30:04 61 m. -/rwxrwxrwx 256519 <testwin95.img-_C0.TXT-dead-256519> | Related activity to deleting file to recycle bin at 11:36:00 AM EDT |
| | 61 m. -/rwxrwxrwx 256519 /RECYCLED/_C0.TXT (deleted) | Related activity to deleting file to recycle bin at 11:36:00 AM EDT |
| | 245 m. -/rwxrwxrwx 2509324 /WINDOWS/RECENT/testfile3.lnk (TESTFI~2.LNK) | Related activity to saving testfile3.txt at 17:30:00 PM EDT |
| | 61 m. -/rwxrwxrwx 25 /testfile3.txt (<u>ESTFI~2.TXT</u>) (deleted) | Related activity to saving testfile3.txt at 17:30:00 PM EDT |
| | 61 m. -/rwxrwxrwx 25 <testwin95.img-_ESTFI~2.TXT-dead-25> | Related activity to saving testfile3.txt at 17:30:00 PM EDT |
| May 08 2002 17:33:58 | May 08 2002 17:33:58 4608 ..c -/rwxrwxrwx 27 /testfile4.doc (TESTFI~1.DOC) | 5/8/02 Started WordPad at 17:32:00 PM EDT |

| | | |
|----------------------|---|---|
| | | 5/8/02 Created testfile4.doc and closed WordPad 17:34:00 PM EDT |
| May 08 2002 17:34:00 | May 08 2002 17:34:00 1 ..c -rwxrwxrwx 1542 <testwin95.img-_S3401.TMP-dead-1542> | 5/8/02 Created testfile4.doc and closed WordPad 17:34:00 PM EDT |
| | 4608 m.. -/rwxrwxrwx 27 /testfile4.doc (TESTFI~1.DOC) | |
| | 1 ..c -/rwxrwxrwx 1542 /WINDOWS/TEMP/ms3401.tmp (_S3401.TMP) (deleted) | |
| May 08 2002 17:34:02 | May 08 2002 17:34:02 1 m.. -rwxrwxrwx 1542 <testwin95.img-_S3401.TMP-dead-1542> | |
| | 1 m.. -/rwxrwxrwx 1542 /WINDOWS/TEMP/ms3401.tmp (_S3401.TMP) (deleted) | |
| May 09 2002 00:00:00 | May 09 2002 00:00:00 271 .a. -/rwxrwxrwx 2509318 /WINDOWS/RECENT/Readme.lnk (README.LNK) | |
| | 245 .a. -/rwxrwxrwx 2509322 /WINDOWS/RECENT/testfile2.lnk (TESTFI~1.LNK) | |
| | 257 .a. -/rwxrwxrwx 2509326 /WINDOWS/RECENT/testfile4.lnk (TESTFI~3.LNK) | |
| | 23696 .a. -/rwxrwxrwx 2587 /WINDOWS/SYSBCKUP/LZEXPAND.DLL | |
| | 300 .a. -/r-xr-xr-x 256520 /RECYCLED/_NFO (deleted) | |
| | 698868 .a. -/--x-x-x-x 409144 /WINDOWS/SYSTEM.DAT | |
| | 300 .a. -r-xr-xr-x 256520 <testwin95.img-_NFO-dead-256520> | |
| | 65 .a. -/r-xr-xr-x 256518 /RECYCLED/desktop.ini (DESKTOP.INI) | |
| | 83662 .a. -/r-xr-xr-x 409219 /WINDOWS/ShellIconCache (SHELLI~1) | |
| | 12144 .a. -/rwxrwxrwx 2581 /WINDOWS/SYSBCKUP/VER.DLL | |
| | 242 .a. -/rwxrwxrwx 2509320 /WINDOWS/RECENT/testfile.lnk (TESTFILE.LNK) | |
| | 293 .a. -/rwxrwxrwx 896524 /WINDOWS/STARTM~1/PROGRAMS/Windows Explorer.lnk (WINDOW~1.LNK) | |
| | 245 .a. -/rwxrwxrwx 2509324 /WINDOWS/RECENT/testfile3.lnk (TESTFI~2.LNK) | |
| | 5859 .a. -/rwxrwxrwx 409202 /WINDOWS/WIN.INI | |
| May 09 2002 11:36:00 | May 09 2002 11:36:00 300 ..c -r-xr-xr-x 256520 <testwin95.img-_NFO-dead-256520> | Deleted testfile3.txt to recycle bin at 11:36:00 AM EDT |
| | 300 ..c -/r-xr-xr-x 256520 /RECYCLED/_NFO (deleted) | Deleted testfile3.txt to recycle bin at 11:36:00 AM EDT Note, the RECYCLED/INFO file was created at this time to hold the testfile3.txt information. |
| May 09 2002 11:36:02 | May 09 2002 11:36:02 300 m.. -/r-xr-xr-x 256520 <testwin95.img-_NFO-dead-256520> | Related to deleting testfile3.txt to recycle bin at 11:36:00 AM EDT |
| | 300 m.. -/r-xr-xr-x 256520 /RECYCLED/_NFO (deleted) | Related to deleting testfile3.txt to recycle bin at 11:36:00 AM EDT |
| May 09 2002 11:37:00 | May 09 2002 11:37:00 65 ..c -/r-xr-xr-x 256518 /RECYCLED/desktop.ini (DESKTOP.INI) | Emptied recycle bin at 11:37:00 Note that the RECYCLED/desktop.ini file is created to record the deletion of the files in the recycle bin. |
| May 09 2002 11:37:06 | May 09 2002 11:37:06 65 m.. -/r-xr-xr-x 256518 /RECYCLED/desktop.ini (DESKTOP.INI) | Related to emptying recycle bin at 11:37:00 |
| May 09 2002 11:39:52 | May 09 2002 11:39:52 83662 m.. -/r-xr-xr-x 409219 /WINDOWS/ShellIconCache (SHELLI~1) | |
| May 09 2002 11:39:58 | May 09 2002 11:39:58 698868 m.. -/--x-x-x-x 409144 /WINDOWS/SYSTEM.DAT | Shut down system at 11:39:58 AM EDT |

Appendix B: Y2K Windows 95 Issues

This document on the Microsoft Web site, discusses issues relating to Y2K problems and patches for Windows 95.

The complete document is available at:

<http://www.microsoft.com/windows/downloads/bin/w95/y2kw95.txt>

The following information is just the highlights of the document, which explains the Y2K issues with Windows 95.

The issues listed below are resolved by installing this Update.

1. Find "File or Folders" Dialog (shell32.dll)
2. Windows File Manager (winfile.exe)
3. Command Interpreter (command.com)
4. Date/Time Picker (comctl32.dll).
5. Phone Dialer applet (dialer.exe)
6. Time and Date Control Panel applet (timedate.cpl)
7. DHCP Virtual Driver (vdhcp.386)
8. Microsoft Foundation Class Library file (mfc40.dll)
9. DOS Xcopy (xcopy.exe, xcopy32.exe)
10. Microsoft Run Time Library file (msvcrt40.dll)
11. OLE AUTOMATION (oleaut32.dll, olepro32.dll, stdole2.tlb, asycfilt.dll)

Upcoming SANS Forensics Training



CLICK HERE TO
REGISTER NOW!

| | | | |
|---|---------------------------------|-----------------------------|------------|
| SANS Northern VA - Reston Spring 2020 | Reston, VA | Mar 02, 2020 - Mar 07, 2020 | Live Event |
| SANS Secure Japan 2020 | Tokyo, Japan | Mar 02, 2020 - Mar 14, 2020 | Live Event |
| SANS Munich March 2020 | Munich, Germany | Mar 02, 2020 - Mar 07, 2020 | Live Event |
| SANS St. Louis 2020 | St. Louis, MO | Mar 08, 2020 - Mar 13, 2020 | Live Event |
| SANS Dallas 2020 | Dallas, TX | Mar 09, 2020 - Mar 14, 2020 | Live Event |
| Dallas 2020 - FOR500: Windows Forensic Analysis | Dallas, TX | Mar 09, 2020 - Mar 14, 2020 | vLive |
| SANS vLive - FOR610: Reverse-Engineering Malware: Malware Analysis Tools and Techniques | FOR610 - 202003, | Mar 09, 2020 - Apr 22, 2020 | vLive |
| SANS Paris March 2020 | Paris, France | Mar 09, 2020 - Mar 14, 2020 | Live Event |
| SANS Secure Singapore 2020 | Singapore, Singapore | Mar 16, 2020 - Mar 28, 2020 | Live Event |
| SANS London March 2020 | London, United Kingdom | Mar 16, 2020 - Mar 21, 2020 | Live Event |
| SANS Norfolk 2020 | Norfolk, VA | Mar 16, 2020 - Mar 21, 2020 | Live Event |
| SANS San Francisco Spring 2020 | San Francisco, CA | Mar 16, 2020 - Mar 27, 2020 | Live Event |
| SANS Seattle Spring 2020 | Seattle, WA | Mar 23, 2020 - Mar 28, 2020 | Live Event |
| SANS Madrid March 2020 | Madrid, Spain | Mar 23, 2020 - Mar 28, 2020 | Live Event |
| SANS Secure Canberra 2020 | Canberra, Australia | Mar 23, 2020 - Mar 28, 2020 | Live Event |
| SANS Oslo March 2020 | Oslo, Norway | Mar 23, 2020 - Mar 28, 2020 | Live Event |
| Mentor Session - FOR508 | Sao Paulo, Brazil | Mar 25, 2020 - Mar 28, 2020 | Mentor |
| SANS Abu Dhabi March 2020 | Abu Dhabi, United Arab Emirates | Mar 28, 2020 - Apr 02, 2020 | Live Event |
| SANS FOR585 Rome March 2020 (In Italian) | Rome, Italy | Mar 30, 2020 - Apr 04, 2020 | Live Event |
| SANS Frankfurt March 2020 | Frankfurt, Germany | Mar 30, 2020 - Apr 04, 2020 | Live Event |
| SANS vLive - FOR508: Advanced Incident Response, Threat Hunting, and Digital Forensics | FOR508 - 202003, | Mar 31, 2020 - May 07, 2020 | vLive |
| SANS 2020 | Orlando, FL | Apr 03, 2020 - Apr 10, 2020 | Live Event |
| SANS Riyadh April 2020 | Riyadh, Kingdom Of Saudi Arabia | Apr 04, 2020 - Apr 16, 2020 | Live Event |
| SANS Minneapolis 2020 | Minneapolis, MN | Apr 14, 2020 - Apr 19, 2020 | Live Event |
| SANS Bethesda 2020 | Bethesda, MD | Apr 14, 2020 - Apr 19, 2020 | Live Event |
| SANS Brussels April 2020 | Brussels, Belgium | Apr 20, 2020 - Apr 25, 2020 | Live Event |
| SANS London April 2020 | London, United Kingdom | Apr 20, 2020 - Apr 25, 2020 | Live Event |
| SANS Baltimore Spring 2020 | Baltimore, MD | Apr 27, 2020 - May 02, 2020 | Live Event |
| SANS Bucharest May 2020 | Bucharest, Romania | May 04, 2020 - May 09, 2020 | Live Event |
| SANS Security West 2020 | San Diego, CA | May 06, 2020 - May 13, 2020 | Live Event |
| Security West 2020 - FOR610: Reverse-Engineering Malware: Malware Analysis Tools and Techniques | San Diego, CA | May 08, 2020 - May 13, 2020 | vLive |