Global Information Assurance Certification Paper

GIAC Certified Forensic Analyst (GCFA)
Practical Assignment
Version 1.0 (April 3, 2002)

Denis E. Brooker

Part 1 - "Restorer 2000 Pro as a Forensic Tool"
Part 2 – "Binary Analysis"
Part 3 – "Legal Issues of Incident Handling – The Wiretap Statute"

Table of Contents

Introduction

The following pages represent my submission to GIAC for consideration as the practical certification requirements for the GIAC Certified Forensic Analyst certification.

Part 1 of the paper is an in-depth study of the tool "Restorer 2000 Pro" from Bitmart.Net Part 2 of the paper is a binary analysis of a file downloaded from the practical assignment. Part 3 is a paper on "The Wiretap Statute."

As an administrative note, due to the large number of computer screen shots that I have included in this paper, I segregated them into appendices instead of imbedding them within the paper. This has an effect of shortening the overall length of the paper and providing for a better paper format.

Part 1
"Restorer 2000 Pro as a Forensic Tool"

Introduction

Microsoft Windows 2000 is quickly becoming the standard operating system in the business world. Based upon the structure of Microsoft's earlier version, Windows NT, Windows 2000 uses as its primary file system NTFS, which has several key security features. Like Windows NT, Windows 2000 does not have an integrated method of restoring files that have been accidentally or purposefully deleted. Likewise, it has no inherent method of recovering NTFS partitions that have been formatted, deleted, or repartitioned. The ability to recover files and deleted partitions is important to the administrators of Windows systems, but it is absolutely critical to the Forensic Investigator in investigating systems that have been compromised as the intruder may have and probably did delete files in an attempt to hide his/her tracks.[1]

As the Manager of Data Security Architecture for a large financial institution, I was recently confronted with a problem concerning deleted files on an NTFS partition. In this particular incident, the system in question had not been compromised, but numerous employees had purposefully deleted information from the network servers in order to cover up behavior that was detrimental to the company. The server was confiscated and the hard drives from the system were transported to Data Security for analysis, archiving, and evidence gathering. The resulting information was to be used in a civil court case against several of the (now) ex-employees. The dilemma that I faced, was how to successfully recover the deleted files and make that evidence available to the court in a manner that was reliable, reproducible, and beyond reproach. Research into the problem resulted in several products that had the potential to accomplish the job, but Restorer 2000 Pro was the product that stood out as the most likely to efficiently meet the required objectives.

This report will investigate and validate the use of Restorer 2000 Pro as a forensic tool suitable for gathering, preserving, and presenting evidence in a court of law.

Scope

Restorer 2000 Pro has several key features that may prove useful to a Forensic Investigator, including undelete capability, data recovery on damaged partitions and formatted NTFS drives, recover compressed files on NTFS partitions, restoring files and directories which names contain national language characters, NTFS file recovery with alternative data streams, creation and use of drive images, and copying of system locked files such as registry hives[2]. This

[1] John B, maruti sunil, "how to hack a web site?"
URL: http://www.faqts.com/knowledge_base/view.phtml/aid/11815/fid/118, (20 August 2002)
[2] "NTFS Data Recovery & Undelete Software for Windows 2000. NT. XP. Unformat Utility. " 8 Mar 2002" URL: http://www.bitmart.net/r2k.htm (20 August 2002)

paper will investigate each of these key features and relate the feature as to its usefulness in a forensics investigation. Testing will be conducted on a hard drive removed from a system and simulated to have challenges that are likely to be encountered in a forensic investigation. The testing will be conducted in a manner as to simulate a forensic investigation to ensure the validity of the product as a forensic tool.

Tool Description

Restorer 2000 Pro Version 1.1 (Build 110621) is primarily a data recovery tool designed to recover data that has been deleted or lost due to drive formatting, partitioning, or damage. It has additional functionality in that it can produce drive images that can be stored and reopened at a later time and it can recover files with alternative data streams.

The software is developed and distributed by Bitmart.Net, located in Richmond Hill, Ontario Canada. At the time of this writing, the Professional version of the software cost $49.99 U.S. A Standard version is also available at a cost of $29.99, but lacks the ability to recover from damaged or erased partitions. They also have a free demo version of the software that is so limited in its functionality as to have no practical purpose other than to demonstrate the GUI user interface.[3] All versions of the software may be downloaded from the Bitmart.Net site at http://www.bitmart.net/r2kfull.htm.

The Forensic Investigator will find this tool very useful in the collection of evidence from a machine that has been compromised and files deleted, hard drives erased, partitions erased, or hard drives damaged as it will give him/her the ability to recover those files that may provide clues as to the identity of the intruder. This is a likely scenario as files used to install malware[4] are routinely deleted either by the intruder or by an automated process designed to cover the tracks of the intruder. In addition, other capabilities of the software, such as the ability to create and read images of drive systems may be very useful in the investigation. A tool that has the capability to restore and recover files from Windows NT/2000 systems is an absolute requirement of any forensic toolkit where NTFS is used. One weakness of the program is its inability to work with any file system other than NTFS. The basic FAT and FAT32 partitions cannot be recovered using this system, meaning that the forensic investigator will be required to carry another tool in his toolkit to address files on those files systems.

According to Christopher Spera, Restorer 2000 Pro has "what the doctor ordered".[5] In his article, he gives the software rave reviews, although he shows no research or testing to support his conclusions.

---

[3] "NTFS Data Recovery & Undelete Software for Windows 2000. NT. XP. Unformat Utility. URL: http://www.bitmart.net/r2kfull.htm (8 Aug 2002)

[4] "Webopedia. The #1 Online Encyclopedia Dedicated to Computer Technology" URL: http://www.webopedia.com/TERM/M/malware.html (9 Aug 2002)

[5] Spera, Christopher, "Restorer 2000 Pro", User to User Reviews URL: http://www.wugnet.com/csreviews/software/Restorer2000Pro/ (15 Aug 2002)

Installation of the software is very straightforward. Once the software is purchased, you will receive an email from Bitmart.Net that contains a zipped copy of the software as shown in Figure A-1. The file is an executable installation file that can be run from inside the zip file. Once executed, you can accept the default installation path or designate a path of your choosing. The copying of files and final installation take only a few seconds. Once finished, you can start the software immediately as no reboot is required. The first time the program is started, you have to register the software using the information contained in a separate email from Bitmart.Net. The registration screen is shown in Figure A-2. Once the registration information is successfully entered, the program starts normally with the main screen as shown in Figure A-3.

As you can see from Figure A-3, Restorer 2000 Pro is able to see those drives that are local to the machine on which it is installed. It is not network aware, meaning that network connections will have no affect on the operation of the software. While there is no threat of interference or inaccurate results due to a network connection, the inability of the program to work across a network limits its functionality. During an investigation the drives must be removed from the victim machine and placed on a different computer or the Restorer 2000 Pro program will have to be installed on the victim machine. The fact that installing software on a victim machine is likely to cause problems in the investigation, perhaps even permanently overwriting the data that is being sought, means that the only real option is to remove the drives. This could entail considerable downtime on the victim machine.

Restorer 2000 Pro installs into its own directory and takes few files to be operational as shown by the file listing in Figure A-4. The file "Restorer2000.exe" is the main file for the program, while "Restorer2000.cnt", "FAQ.htm, and "Restorer2000.hlp" are the help files for the program. Included in the installation is an uninstall program to remove the files installed during a normal installation. Restorer 2000 Pro's executable file is 844kb in size and will fit on a floppy drive. It can be run without installation from a floppy or CD, but the registration screen (Figure A-2) will pop up and the User Name, Company, and Serial Number must be provided to continue. The serial number is of significant length as to make the typing of it a slow process. In addition, there is a registry entry that will be added to the machine even though the program is being run from removable media. This, in effect, makes the program unusable from removable media.

Using the "Depends" program from the Microsoft NT Resource kit, we are able to see that Restorer 2000 Pro uses a plethora of .dll files from the Windows operating system as shown in Figure A-5. In addition, the installation adds a registry entry containing the registered name, company, and serial number as shown in Figure A-6. This registry key is added during the installation program, but will also be added if you run the program directly from a CD or floppy drive as described in the previous paragraph.

System dll files and the Restorer2000 executable file are the only files that are used by the program. It is a commonly known fact that dll files are not written to or changed during the course of their use. According to Webopedia.com dll files are either statically or dynamically linked. Static files remain the same during

program execution while dynamic dll links are created by the program execution.[6] In either case, the dll files themselves do not change. The only other file that could change during execution would be the Restorer2000.exe file itself. In order to check this, an MD5 hash value was taken of the file prior to execution and then again while the program was running. As you can see from the before and during execution MD5 hash values in Figure A-7, the file does not change during execution. This shows, without doubt, the program is static and evidentiary sound.

Test Apparatus

The computer used for testing Restorer 2000 Pro was a Forensic Air-Lite III from Forensic-Computers.com. This particular model sports a Pentium III 1.2 Ghz processor, 512MB of RAM, and a built in 37.2Gb IDE hard drive. This system is designed for forensic investigations and includes cabling and power to attach external IDE and SCSI drives. It also has an IEEE 1394 (Firewire) bus with Forensic-Computers.com Firewire to IDE Write-Protection boxes allowing connection to IDE drives through Firewire without the worry of accidentally writing files to a drive under investigation. The bios on the system is Phoenix – AwardBIOS v6.00PG.

The operating system used for this test is Microsoft Windows 2000 Professional Version 5.0.2195 with Service Pack 3 build 2195. Microsoft Internet Explorer version 6.0.2600.0000 with 128-bit Cipher was also installed. Security Patches MS02-009 concerning XML, MS02-009 concerning VBScript, and MS02-023 concerning Internet Explorer are all installed as well. The patch for outstanding Security Bulletin MS02-027 was not available for Internet Explorer as of this writing. Otherwise, all outstanding service packs and security patches recommended by Microsoft have been installed.[7] System time is set to Central Daylight Time (GMT – 6:00).

In addition to Restorer 2000 Pro being tested, several other forensic tools were used to validate the testing. These include MD5Sum[8] to obtain hash values used to verify and compare before and after images of files, KillDisk[9] used to clean the test drive before testing, and DD[10] used to image the entire hard drive for verification no changes occurred.

Environmental Conditions

---

[6] **DLL**,
URL: http://www.webopedia.com/TERM/D/DLL.html, (20 August 2002)
[7] Microsoft Corporation, "Hotfix & Security Bulletin Service"
URL: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/current.asp (20 August 2002)
[8] **MD5Sum**
URL: http://www.etree.org/md5com.html, (20 August 2002)
[9] **KillDisk**, Lsoft Technologies, Version 1.1
URL: http://www.lsoft.net, (20 August 2002)
[10] **DD**,
URL: http://www.redhat.com/swr/i386/fileutils-4.1-10.i386.html, (20 August 2002)

As Restorer 2000 Pro is not network aware and does not have the capability to see network drives anyway, all testing will be done on the stand-alone machine without network connectivity. All drives were cleared using KillDisk and setting all bits on the drive to zero. This effectively eliminates all outside forces that could affect the results of the tests.

As verification of the evidentiary quality of the program, a DD image was taken of the test drive after the test files were placed and deleted and an MD5 hash value of the image was taken. After the testing was completed, the process was repeated to determine the testing process had not changed the "victim" drive.

Description of the Procedures

There were four hard drives used for this test. The first drive was an IDE Western Digital Caviar 11000 with 2046 cylinders, 16 heads, 63 sectors, and 1055.9 MB of storage. The second drive was a SCSI Seagate Barracuda model ST15150N. The third drive was an IDE IBM Travelstar 18GB drive from a Dell Notebook computer. The fourth drive was an IDE Seagate Model ST34313A with 8944 cylinders, 15 Heads, 63 Sectors, and 4.3 GB of storage.

TEST PLAN

The original plan for this test was to perform a series of seven tests to verify the functionality of Restorer 2000 Pro. These test will accomplish a simple file recovery after deletion from an IDE drive, a file recovery after deletion and the deletion of the partition from an IDE drive, a file recovery after deletion of the file from a SCSI drive, a file recovery after the deletion of the file and partition and a repartition and reformat with FAT32 on a SCSI drive, reading a non-NTFS partition, create and read images, and read DD images.

Flexibility has been built into this test plan. If additional information and testing is needed, additional tests will be added at the end to find the best answers possible.

After all testing is completed; they will all be repeated two additional times (including tests that have been added on) to confirm the tests are repeatable and reproducible.

As stated earlier a DD image and MD5 hash value of that image was taken on the victim drives at the beginning and ending of testing on that drive to confirm the drive was not changed by the testing operations.

PRE-TEST, Program Operations

In this section, I will cover all of the operating procedures, parameters, and options of using the program. These procedures will be repeated numerous times through the remainder of the test project and do not need to be written in

this report each time. Refer back to this section for information pertaining to any general aspect of the program mentioned during the testing phase.

Starting the program is accomplished by pressing the START button, choosing Programs, Restorer 2000 Pro, and Restorer 2000 Pro as depicted in Figure B-1. Once started, the main screen appears as depicted in Figure B-2. The screen can be divided into four sections for purposes of discussion. The first section is the Menu Section. As you can see from Figure B-3, the menu is comprised of four line items: Drive, Tools, View, and Help. The second section, depicted in figure B-4, is the Device/Disk section where you can see the drives the program can access. The third section, depicted by Figure B-5 is the files section. The fourth section, depicted by Figure B-6, is the status section.

The menus available are the Drive, Tools, View, and Help menus. The Drive menu has several menu functions depending on what is selected on the Device/Disk section. In Figure B-7, the menu is open while a physical drive is selected. You can see that you can create or open an image file, you can scan the physical drive or an area of the physical drive you designate, you can create a region, or you can set read attempts. According to the help file for the program a "Drive region is a custom defined area on the drive. It can be used for scanning as well as storage of regular drives."[11]

In Figure B-8, the Drive Menu is shown while a Logical Drive is selected. In addition to the options you had with a physical drive, you can also open and recover files from the logical partition.

If you open the Drive Menu while the Image Folder is highlighted, your only option is to open the image file as shown in Figure B-9.

The Tools Menu, shown in Figure B-10, has two entries to control the log file. The first, "Log Filter" allows you to set what you want to be logged. The second, "Clear Log" clears the log file so you can start fresh. The other items that are grayed out in the figure are "Stop", which is used to stop an action in progress, "Delete", which is used to delete image files, and "Edit", which is used to edit the image files.

The View Menu allows you to select the configuration of what you see on the main menu, the view, as shown in Figure B-11. The other option (Info…) is to see the Information about a logical or physical drive as shown in Figure B-12.

The Help Menu, Figure B-13, allows you to open help, which has limited information about how to use the program. You can also get information about the program by selecting "About Restorer2000" as shown in Figure B-14.

There is also a set of icons that perform some of the functions available in the menus. These are shown in Figure B-15.

Once a scan of a physical drive, logical drive, partition, or part thereof is started, the program reads each bit of the drive to find the data. It does not manipulate the data (evidence) in any way while being run. This is verified by the fact that it runs fine with the "Read-Only" adapter attached to the workstation. It would be safe to run without the "Read-Only" adapter if it were not an available option. The location to which restored files are written is users selectable.

[11] Restorer 2000 Pro Help File, "Using Regions"

Therefore, caution should be used when performing tests without the "Read-Only" device as the program could write recover files to the original drive if not configured properly.

Overall, the program is very user friendly and intuitive to operate. It only takes a few minutes for the user to get comfortable with the program and the locations of its tools and features. While the Help files for the program are somewhat lacking in detail and clarity, the layout of the program and its tools make the program easy to use and negate the weak Help system.

TESTING

The following section provides detail about the individual tests that were conducted. A total of nine tests were conducted, seven from the original plan and two additional that were added to clarify the results of other testing. These tests were designed to demonstrate the capabilities of the program with its use as a forensic tool in mind. As mentioned earlier in this report, each series of test was conducted a total of three times to confirm and verify that the results are repeatable and reproducible.

TEST 1 – IDE Drive, Standard File Deletion

Purpose - The purpose of this test was to check the basic functionality of the program by creating and deleting files on an NTFS partition to see that they could be restored using Restorer 2000 Pro. This particular test was on an IDE drive. Later test will compare the functionality on IDE drives against the functionality on SCSI drives.

Expected Results - The deleted files should be fully restored to a new location. MD5 hash values should verify the restored files are exactly the same as the original. In addition, the drive from which the files were restored should not be changed at all by the testing process.

Test Procedures

1. KillDisk was used to sanitize the IDE hard drive with parameters specifying that all sectors be overwritten with zeros.
2. The drive was installed on the test computer. At this point, the drive has not been added to the Windows 2000 structure, has not been partitioned or formatted with NTFS.
3. Restorer 2000 Pro was then used to verify that no files could be seen on the drive by the program. Figure C-1 shows there were no files on the drive.
4. Five test files were developed and placed in five separate directories on the drive as shown in Figure C-2.
5. An MD5 hash value was obtained on each of the files with the results shown in Figure C-3. (Figure C-3 also shows the MD5 hash value of the files that

were restored for comparison. Notice they match showing perfect recovery of the files.)

6. The directories and files were then all deleted. It was confirmed that neither the files nor directories could be accessed using Windows 2000. In addition, the Recycle Bin was emptied. Screens are shown in Figure C-4 and C-5.
7. A DD image was taken of the IDE hard drive and an MD5 hash value was taken of the image as shown in Figure C-6. (Figure C-6 also shows the MD5 hash value of an image taken after this test was completed. Notice they match showing that the drive was not affected by the operation of Restorer 2000 Pro.)
8. Restorer 2000 Pro was then used to scan the drive to see if the files could be found, recognized, and recovered. Figure C-7 shows the files were found and recognized on the drive, deleted from the Recycle Bin. This is because the files were first deleted normally, which moves them to the Recycle Bin, and then deleted from the Recycle Bin.
9. A directory was created on the Forensic Workstation to receive the recovered files. The files were then restored to that directory using Restorer Pro 2000. See Figure C-8 for the restore dialog box.
10. MD5 hash values were obtained for all five files for comparison against the MD5 hash values recorded earlier. The comparisons shows the files were successfully restored and are absolutely identical in content. See Figure C-3.
11. DD was used to make an image of the drive after the tests. MD5sum was used to retrieve a hash value of the image and was compared to the pretest image as shown in Figure C-6.

Results – In test 1, the program worked flawlessly. The files were created on an NTFS partition and then an MD5 hash value was obtained for verification. The files were deleted using Windows and then restored using Restorer 2000 pro. After recovery, an MD5 hash value was again obtained and the results were compared to the MD5 hash value of the original file. They were identical, proving the restored files are the originals with no changes or other undesirable affects to the data. In addition images of the hard drive taken before and after the testing confirmed that Restorer 2000 Pro did not affect the original drive in any way.

TEST 2 – IDE Drive, Deleted Partition

Purpose – This test will determine the program's ability to recover data even after the logical partition that held the data has been deleted. This particular test was accomplished using the same drive and test files created and deleted in Test 1. Remember that Restorer 2000 pro does not affect the drive during restore (unless you restore to that drive) so the files should be in their condition prior to the start of Test 1.

Expected Results – In this test, Restorer 2000 Pro should be able to recover the test files with no problems. MD5 hash values should verify the original files and the recovered files are absolutely identical.

Test Procedures

1. The drive used in the previous test was reused for this test. The test computer was booted to DOS from a floppy drive. FDISK was then used to simply delete the Non-DOS partition.
2. Restorer 2000 Pro was used to scan the disk. The files were still available to be seen on the drive as shown in Figure D-1
3. The files were restored to the test directory.
4. MD5 hash values were obtained for all five files for comparison against the MD5 hash values obtained earlier (Figure C-3 Top). The comparisons show the files were successfully restored and are absolutely identical in content. See Figure D-2.

Results – Once again, the program worked flawlessly. It did not matter that the partition was no longer available, Restorer 2000 Pro scanned the physical drive and was able to find and recover the test files. The MD5 hash value again proved the validity of the restore.

TEST 3 – SCSI Drive, Standard File Deletion

Purpose - The purpose of this test was to check the basic functionality of the program by creating and deleting files on an NTFS partition to see that they could be restored using Restorer 2000 Pro. In this case, the functionality of the program on SCSI drives was the focus.

Expected Results - The deleted files should be fully restored to a new location. MD5 hash values should verify the restored files are exactly the same as the original. The testing process should not change the SCSI drive. This test is identical to Test 1, except it is performed on a SCSI drive instead of an IDE drive. The results of the testing should be the same.

Test Procedures

1. KillDisk was used to sanitize the SCSI hard drive with parameters specifying that all sectors be overwritten with zeros.
2. The drive was installed on the test computer. At this point, the drive has not been added to the Windows 2000 structure, partitioned, or formatted with NTFS.
3. Restorer 2000 Pro was then used to verify that no files could be seen on the drive by the program as illustrated by Figure E-1.
4. Five test files were developed and placed in five separate directories on the drive as shown in Figure E-2.

5. MD5 hash values were obtained for each of the files with the results shown in Figure E-3. (Figure E-3 also shows the results of the MD5 hash values on the recovered files. Notice they are identical proving the ability of the program to restore files to their original condition.)
6. The directories and files were then all deleted. It was confirmed that neither the files nor directories could be accessed using Windows 2000. In addition, the Recycle Bin was emptied. Screens are shown in Figure E-4 and E-5.
7. A DD image was taken of the drive prior to the testing. An MD5 hash value of the image file was taken as shown in Figure E-6. (Figure E-6 also shows the MD5 hash value of the drive taken after all testing was completed. Notice that they are the same proving that the program does not change the victim drive.)
8. Restorer 2000 Pro was then used to scan the drive to see if the files could be found, recognized, and recovered. Figure E-7 shows the files were found and recognized on the drive, deleted from the Recycle Bin. This is because the files were first deleted normally, which moves them to the Recycle Bin, and then deleted from the Recycle Bin.
9. A directory was created on the Forensic Workstation to receive the recovered files. The files were then restored to that directory using Restorer Pro 2000. See Figure E-8 for the restore dialog box.
10. MD5 hash values were obtained for all five files for comparison against the MD5 hash values recorded earlier. The comparisons shows the files were successfully restored and are absolutely identical in content. See Figure E-3.
11. DD was again used to make an image of the drive followed by the recording of an MD5 hash value of the image as shown in Figure E-6.

Results – In this test, the program again worked flawlessly. The files were created on an NTFS partition and then an MD5 hash value was obtained for verification. The files were deleted using Windows and then restored using Restorer 2000 pro. After recovery, an MD5 hash value was calculated and the results were compared to the MD5 hash value of the original file. They were identical, proving the restored files are the originals with no changes or other undesirable affects to the data. Images taken of the test drive before and after test prove, via the MD5 hash values, that the program did not change the contents of the test drive.

TEST 4 – SCSI Drive, Deleted Partition, Created Partition, and Format

Purpose – This test was designed to see what would happen if the drive in question were subjected to deletion of the partition as was done in Test 2, and the creation of a new partition, and the formatting of the drive with FAT32, which Restorer 2000 Pro cannot read.

Expected Results – The files should be fully recovered by Restorer 2000 Pro as the data on the drive is not actually destroyed by the format or partition actions.

MD5 hash values should verify the original files and the recovered files are absolutely identical.

Test Procedures

1. Working from the drive and files used in Test 3, Figure F-1 shows the starting disk configuration according to Windows 2000.
2. The partitions on the drive were then removed, a new partition was installed in its place, and the partition was formatted using FAT32.
3. The system was booted into MS-Dos with a boot desk and the partition was removed using fdisk.
4. Restorer 2000 Pro was then run in an attempt to recover the original files from the NTFS partition. Figure F-2 shows the results of this test.

Results – In this particular case, it is apparent that Restorer 2000 Pro could not recover the original files. This bears further investigation as the process of creating a partition and formatting the drive should not affect any of the actual data. As a result of this testing, additional tests were added to the suite of tests in order to determine the limits of Restorer 2000 Pro capability.

TEST 5 – Read Non-NTFS Partitions

Purpose – The purpose of this test was twofold, first, to determine if Restorer 2000 Pro would have any problems reading a drive from a Notebook computer and second, to determine how Restorer 2000 pro would respond to a non-NTFS partition.

Expected Results – Restorer 2000 Pro should be able to accurately read the notebook hard drive. While the program can read the drive and make an image of the drive, it will not likely be able to read the actual files on either the drive or the image since it is a tool for NTFS only.

Test Procedures

1. The notebook hard drive was connected to the test computer and started up. This particular hard drive contains a partition that has Windows 2000 installed, but the partition was formatted with FAT32 and not NTFS.
2. A DD image was made of the drive with MD5 hash values made of the image as shown in Figure G-1. (Also shown in Figure G-1 are the MD5 hash values results from an image taken after the test was completed. The results match showing no change in the drive as a result of the test.)
3. Restorer 2000 Pro was ran against the hard drive with the results shown in Figure G-2. Files from the non-NTFS partition did now show up.
4. The drive was again imaged with DD and an MD5 hash value taken as shown in Figure G-1.

Results – Restorer 2000 Pro was able to recognize and scan the drive with no problems. As expected, it was unable to read the data on the FAT32 partition or to even recognize the FAT32 partition was present.

TEST 6 – Create and Read Images

Purpose – The purpose of this test is to determine whether Restorer Pro 2000 can produce a forensic quality image of a hard drive. If the program can produce the image, this test will determine if it can then use the image to restore as though it were a real hard drive. Lastly, the test will determine if the images produced by the program can be used as forensic evidence.

Expected Results – Restorer 2000 Pro should be able to create an image of the drive and then read the image of the drive. We will be able to determine if the image and the original drive are identical to meet the requirements of using the image as forensic evidence. MD5 hash values of the drive itself and of the image or images taken will verify the results.

Test Procedures

1. The fourth test drive was setup as a Windows 2000 Professional installation with NTFS. It was connected to the test workstation.
2. An MD5 hash value of the physical drive was captured to set the baseline for the system. The results are found in Figure H-1.
3. Since Restorer 2000 Pro can capture images of the Physical Drive or of a partition, two images, one of each, were built. An MD5 hash value of both images was obtained with the results shown in Figures H-2 and H-3.
4. Since none of the MD5 hash values were the same, it would be hard to prove that the images were identical to the drive. Further testing is called for and has been added to the testing plan.
5. The images were opened using Restorer 2000 Pro. All of the files on the NTFS partition were accessible on the file that was imaged from the partition, and could easily be recovered using the recover option. See Figure H-4. However, the image file that was built from the physical drive could not be accessed. The error message “Can not find NTFS on …” was displayed.

Results – In this test, we were able to show that Restorer 2000 Pro could create and read image files and recover files from within the images. It was noted that the images taken directly from the Physical Drive were not usable by Restorer 2000 Pro while the images taken from the volume was usable. We were not able to prove that the image files are sufficient to be considered adequate as forensic evidence. As the result of this test, another test has been added to further investigate the use of the images in a forensic investigation for presentation in a court of law.

TEST 7 – Read DD Images

Purpose – This test was conducted to determine if Restorer 2000 Pro could read and extract files from images produced with DD.

Expected Results – Restorer 2000 Pro should be able to read files from DD and allow access to them in much the same way as mounting the files in the Unix environment does.

Test Procedures

1. The same hard drive used in Test 6 was used for this test.
2. DD was used to create an image of the physical drive as shown in Figure I-1.
3. Restorer 2000 Pro was then used to open the image file. An error was given that said, ""Can not find NTFS on …".
4. DD was used to create an image of the D: volume of the drive using the syntax as shown in Figure I-2.[12]
5. Restorer 2000 Pro was used to read the image created by DD.

Results – Restorer 2000 Pro can read image files built by DD on the volume of the hard drive, but cannot read images that are built from the physical drive. This is consistent with earlier test from images created by Restorer 2000 Pro.

TEST 8 – Retrieve Files From a Drive That Has Been Re- Formatted FAT

Purpose – Due to the fact Restorer 2000 Pro could not retrieve files from a drive that was NTFS formatted, but had been repartitioned and formatted with FAT32, this test was added on after the initial testing plan was set. This will either validate the earlier testing (Test 4) or will show capability to recover files in this situation. The earlier failure in Test 4 may have been because of the small number of files that were used for the testing. If these files were overwritten by partition and volume information, they would not be retrievable by any program. If there are files that can be retrieved on this drive during this testing, it shows the capability of seeing through the FAT32 partitioning.

Expected Results – Restorer 2000 Pro should be able to recover at least some of the files from the old NTFS permission, if not the entire directory structure. We know from earlier testing that some files will probably be missing, but the bulk of the files should be available.

Test Procedures

1. The drive used in Test 6, which had a full NTFS partition and information on it, was prepared by using fdisk to delete the partition and create a new partition. Format was then used to format the new partition in FAT32.

[12] SANS Institute, Track 8 – System Forensics, Investigation, and Response, 8.1 Investigative Incident Response and Basic Forensic Windows Principles – Hands On, Sans Institute, 2002, 134-138

2. The drive was then placed on the test workstation and Restorer Pro 2000 was used in an attempt to recover files. Results are shown in Figure J-1.

Results – The results for this test were somewhat mixed. While Restorer 2000 Pro was able to restore some of the files from the NTFS partition, it was only able to recover a small number of them. Certainly, its use would be minimal in a situation like this, but it could recover the file the investigator needs to prove a case.

TEST 9 Recovering a Drive with a Restorer 2000 Pro Image

Purpose – This test was added on to see if a hard drive could be restored by copying the image back to the drive using Restorer 2000 Pro. If this is successful, the next step will be to do an MD5 hash value of the drive to see if that sum matches the sum of the original drive. If they match, this would prove the image capability could be used as forensic evidence.

Expected Results – If the image is to be considered valid for evidentiary purposes, proof of the ability to restore a drive to its absolute original condition, verified by the MD5 hash value, must be demonstrated. This test will either verify that ability or show that ability is not present.

Test Procedures

1. The hard drive and imaged in Test 6 was cleaned using KillDisk to overwrite every bit on the device.
2. Restorer 2000 Pro does not have a "Restore" feature for its imaging, so an attempt was made to copy the image onto the raw drive in hopes it would set the partition as it went. It was found that the image could not be restored until the drive was made a part of the operating system by partitioning and formatting as NTFS. It is highly unlikely we will be able to prove the image is viable as evidence at this point.
3. The drive was made part of the OS by partitioning and formatting as NTFS.
4. The image was recovered to the new drive using the recover option of Restorer 2000 Pro as shown in Figure K-1.
5. During the recover, a message box appeared that identified an existing file. This was because a file of the same name had been deleted on the original drive and the program was attempting to restore both of them. As shown in Figure K-2, the only options were to overwrite or rename. Either one will prevent the drive from being identical.
6. An MD5 hash value of the physical drive (Figure K-3) was obtained and compared to the MD5 hash value accomplished earlier and shown in Figure H-1. As expected, they do not match.

Results – Restorer 2000 Pro does not have the capability to create images suitable as evidence in a court of law. Images can be produced and files from within the images could probably used in some circumstances.

Analysis

After spending several weeks testing Restorer 2000 Pro, I can say without hesitation that this is a good product that will have many uses for a Forensic Investigator with requirements to analyze Windows NTFS partitions.

During the course of the extensive testing conducted, we verified that the program quickly and easily recovered files that had been deleted and files from drives where the partition had been deleted. The program could not read files from non-NTFS partitions, which is a significant limitation, but NTFS partitions that have been overwritten by FAT32 may be read to some degree. Our testing showed the capability existed, but is very limited in its reliability.

It was also determined that the program could read images created by DD. This is one of the most significant findings of the testing as it establishes a capability of taking a forensic quality image from DD and loading into the program for the capability of reviewing the contained files and restoring them as necessary. This gives Windows investigators a tool that can be used to research and manipulate files is much the same way as can be done in the Linux world by mounting a DD image. Unfortunately, the investigator will still have to restore the file before it can be used in this manner.

The ability to create images in the program has little significant appeal, as there is no way to confirm the validity of the image. While images can be made, they can't be used to fully restore a drive. This is due to the fact the program deals only with the contained files and not with the actual partition information. In contrast, programs like DD and Norton Ghost will copy the partition information and can restore that information on the fly, making the image an identical copy of the original.

Regardless of the additional functionality and the usefulness the Forensic Investigator will find, the basic purpose and strength of this program is in its ability to restore files from NTFS partitions that have been deleted. This is crucial to the ability of the investigator to determine the details of the intrusion and to track down the perpetrator.

The following is a listing of the findings of this analysis in a bullet format:

1. Deleted files are easily and accurately restored from NTFS partitions without affecting the status of the original drive.
2. Files are also easily and accurately restored from drives where the NTFS partition has been deleted.
3. Restoration of files from an NTFS partition that has been repartitioned with FAT32 is questionable. Some files can be restored, but many cannot.
4. Restorer 2000 Pro works equally well on IDE or SCSI drives and has no problems recognizing IDE drives from notebook computers.

5. Restorer 2000 Pro cannot read or recover files from Non-NTFS partitions. This is a big weakness in the program, but is overcome by the use of other programs designed to recover files from those partition types.
6. Restorer 2000 Pro can create and read images, but the quality of the images cannot be verified from an evidentiary standpoint. These images should not be used if the information gathered may be used in a court of law.
7. Restorer 2000 Pro can read DD images.
8. The program is user friendly and easy to use.

Presentation

Data obtained by using Restorer 2000 Pro is easily presented to a court and a jury. This tool simply makes the file available for use again, despite the fact it was deleted. Testing has confirmed that the files recovered are absolutely identical as verified by the MD5 hash values. "Calculating the MD5 hash value for the original and the copy is one way of showing that the copying did not alter the data" according to Yoshinori H. T. Himel.[13]

This fact would need to be presented in court as proof of the validity of the file and then the contents of the file would provide a great source of information in court. There are presentation aids available that can help a witness explain the value of the hash. One of these is by NTI and shows exactly how the has is obtained.[14]

Conclusion

The tests performed during the analysis of Restorer 2000 Pro were comprehensive and detailed, designed to prove that the information obtained by using the program is reliable, forensically sound, and suitable for presentation in a court of law. The program performed very well and indeed was proven to meet all of these requirements. This tool is very suitable to a forensic investigation role and should be included in every toolkit of investigators that may be called upon to complete a forensic investigation on a Windows NTFS based system.

Due to the nature of the program and the fact it is not network aware, the only reliable means of using the program is to remove the drive from the victim machine and attach it to a workstation for analysis. This workstation should have the program installed on the system hard drive. Under no circumstances should the program be run on the victim machine as, at a minimum, it would alter the registry of the victim machine thereby polluting the evidence. This is true even if the program is run from a floppy drive or a CD-Rom, either of which would require the registration key anyway.

---

[13] Yoshinori H. T. Himel, "Getting Digital Data Into Evidence", January 2002
URL: http://www.sacbar.org/archives/slugdec01.html

[14] NTI, "Trial Illustration Posters"
URL: http://www.forensics-intl.com/trialill.html, (28 August 2002)

This program could be improved as a forensic tool by adding the ability to read other file systems besides NTFS. While the chances of needing forensic quality evidence off of a NTFS drive that had been repartitioned with FAT32 are low, it is always a possibility and one of the primary limitations of Restorer 2000 Pro. In addition, the program would be improved greatly by the ability to create forensic quality images.

Part 2
Binary Analysis

Investigative Results – Report

The following section presents a synopsis of the information gathered through the investigative process. For detailed information on how the information was obtained and how the conclusions were reached, refer to the "Investigative Process" section below.

Binary Details

- Name of the program/file found on the system. The program found is ADMSniff version Priv 1.0. It had been renamed to "sn.dat".
- File/MACTime Information (last modified, last access, and last changed time). Last modified and accessed time were both identified as April 11, 2002 at 09:29:58. The Created time was unavailable due to the method the file was obtained. The time shown by the debugfs process was the time the file was downloaded to the local hard drive.
- File owner(s) –(user and/or group) Both the owner and group were identified as "root".
- File size (in bytes). The File size was 399124 bytes.
- MD5 hash of the file. The MD5 hash value was calculated as 0e954f43fd73f56e812a7285f32e41d3 as shown in Figure L-2. This matched the MD5 hash value in the file sn.md5 exactly.
- Key words found that are associated with the program/file. Using the "strings" command the following words were found that led to many of the conclusions. "ADMsniff", "priv 1.0", "ADM", "Libpcap", plus the name of the program author.

Program Description

The program found on the system was named "sn.dat". The dat extension is a generic extension and does not indicate what the program may actually be. Analysis confirmed the program was ADMSniff, a "Sniffer" program that uses Libpcap to place the Ethernet network interface card into promiscuous mode so that all traffic on the collision domain can then be captured for analysis. Systems administrators routinely use this type of program for network troubleshooting and monitoring for malicious traffic. Hackers will use Sniffer programs on compromised systems in order to "harvest" usernames and passwords that are traversing the network in the clear.

According to the MAC time information retrieved, the program was last accessed and modified on April 11, 2002 at 9:29:58. It is not possible to definitively conclude whether or not the program has ever been executed without

further investigation beyond the binary file received for this assignment. It is very possible that this date and time indicate the last use of the program.

Since the program is statically linked, the Libpcap libraries and pcap libraries needed for program implementation are included in the compilation. The file had version .4 of Libpcap compiled statically within it.

Please refer to the "Investigative Process" section of this report for detailed information on how these conclusions were reached.

According to testing and research accomplished for this assignment, the program is very self contained and efficient. It appears to accomplish the following actions when executed:

1. The program executes Libpcap and places the Ethernet adapter in promiscuous mode, allowing it to "see" all of the traffic traversing that network segment.
2. It creates a file in the same directory the binary is stored called "The_l0gz" that appears to be the log file.

Other details about the file that are significant are what it does not do. First, it was noted that the program does not automatically determine the Ethernet network adapter card (NIC) in the system. This means that a parameter must be passed to the program on execution or it will fail. An example of the appropriate startup command would be "./sn.dat eth0" with "eth0" being the parameter. Second, the program has no methodology of passing the information it obtains outside of the compromised system. This means the attacker must return to the scene of the crime to retrieve the spoils of his act.

Forensic Details

In this case, the program is a Sniffer that uses Libpcap to set the Ethernet network adapter card into promiscuous mode. Therefore one of the footprints that can indicate the program is operating on a system is to detect the NIC card in this mode. According to the Phrack Magazine article Interface Promiscuity Obscurity article[15] a promiscuous mode interface is recognizable by a flag in the device interface structure that shows the interface status as promiscuous. The article further gives the code necessary to change this flag and hide the fact a card is in promiscuous mode. This code was thoroughly searched for indications that this code was present. It was not, meaning we should be able to see the card in promiscuous mode when this program is running or has been run. The program leaves the card in promiscuous mode after termination.

Another big footprint that is left is the log file that is created by the program. Located within the same directory as the binary file, "The_l0gz" is created to store the captured data. Depending upon the traffic volume of the network that is connected to the computer, this log file may grow to a huge size

[15] Phrack Magazine, "Interface Promiscuity Obscurity", 8 July 98
URL: http://www.phreak.org/archives/exploits/unix/network-sniffers/interface-promiscuity-obscurity.txt, (20 August 2002)

in a relatively short period of time. It is feasible that this could cause the hard drive space to be totally consumed causing a system failure and resulting denial of service.

The program does not appear to start any additional processed or manipulate any other system files. Investigation showed a lot of files that had been modified after the program execution in the "/proc/" directory, but these files were most likely manipulated by the operating system itself. Certainly the execution of the program prompted the operating system to access and modify these files in the normal course of its activity.

The fact that the program has no mechanism to enter the NIC parameters or transfer the log file from the compromised computer to another location provides a very significant "lead" that should immediately be followed up. There are only two ways this program could have been utilized. Either a person who has authorization to the computer and is using that authorization for access installed the program or the computer has been totally compromised, meaning there is other evidence on the computer that may lead to the identification of the culprit. Regardless of which of these cases are true, this system needs to be immediately checked and action taken to identify the culprit.

Program Identification

The program was found at the site that was advertised as the ADM Crew official ftp site, http://adm.freelsd.net/ADM/. This site contained the file that was downloaded and tested for comparison against the "sn.dat" file.

In order to prove beyond any doubt that the two files are from the same source, the goal was to match MD5 hash values and show them identical. Due to the seemingly infinite number of variables that could affect the ability to make the programs match this closely, a realistic expectation was that the MD5 hash values would not match, but that other comparisons would prove beyond a reasonable doubt that the files were from the same source.

The testing yielded strong indications that this file was, in fact, from the same source as "sn.dat". This was verified by a search of the strings results of the binary files. These files, while not identical, were almost identical. It was determined that the two files had been compiled by different versions of GCC, which could easily account for the differences found.

Attempts to find the same GCC compiler to recompile the ADMsniff file were unsuccessful.

Please refer to the "Investigative Process" section of this report for detailed information on how these conclusions were reached.

Legal Implications

The program that was discovered is a "Sniffer" program that is used to gather information off of the network. The use of this program by anyone other than an authorized system administrator for the purpose of maintenance and operations or service provider protection would be in direct violation of 18 USC

2701, also referred to as the "Wiretap Statute".[16] This law prohibits the reception of any communication that is transiting a wire by anyone other than the recipient that is designated by the sender. There are exemptions to this rule for system administrators for specific purposes. Unless these exemptions are met, the person or persons who installed and started this program are in violation of the law, regardless of how the program was installed on the machine.

The second legal implication is that of unauthorized access into the machine. The law that applies here is 18 USC 1030, the Computer Fraud and Abuse Act.[17] Is specifically says: "(a) Whoever - (1) having knowingly accessed a computer without authorization or exceeding authorized access, and by means of such conduct having obtained information that has been determined by the United States Government pursuant to an Executive order or statute to require protection against unauthorized disclosure for reasons of national defense or foreign relations, or any restricted data, as defined in paragraph y. of section 11 of the Atomic Energy Act of 1954, with reason to believe that such information so obtained could be used to the injury of the United States, or to the advantage of any foreign nation willfully communicates, delivers, transmits, or causes to be communicated, delivered, or transmitted, or attempts to communicate, deliver, transmit or cause to be communicated, delivered, or transmitted the same to any person not entitled to receive it, or willfully retains the same and fails to deliver it to the officer or employee of the United States entitled to receive it;"[18] The law goes on to specify which computers are those that "require protection". The section of the law that most often applies is "(C) information from any protected computer if the conduct involved an interstate or foreign communication".[19] This stipulation makes practically every computer that is attached to the Internet a protected computer.

The third law that may be involved in this situation is the "Pen Register Statute" that requires court orders be obtained before any interception of traffic on electronic media. It is similar in scope and function to the other wiretap laws.

Whoever installed this Sniffer program on the victim computer is certainly in violation of the wiretap laws if the program was ever executed. As described earlier, it is impossible to determine without an inspection of the host that contained the file. If the file were executed on the host, that host would have the footprints left by the program, namely the log file would be present and the NIC card would be in promiscuous mode unless the program had terminated and the host had been restarted.

If the program has been started the perpetrator of this crime could expect a fine and/or imprisonment, depending upon the motive behind his/her act. If the

---

[16] "Laws: Cases and Codes: U.S. Code: Title 18: Section 2701 (a)", 23 Jan 2000 URL: http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/i/chapters/121/sections/section_2701.html , (22 August 2002)

[17] "Laws: Cases and Codes: U.S. Code: Title 18: Section 1030", 23 Jan 2000 URLhttp://caselaw.lp.findlaw.com/scripts/ts_search.pl?title=18&sec=1030, (22 August 2002)

[18] "Laws: Cases and Codes: U.S. Code: Title 18: Section 1030", 23 Jan 2000 URLhttp://caselaw.lp.findlaw.com/scripts/ts_search.pl?title=18&sec=1030, (22 August 2002)

[19] "Laws: Cases and Codes: U.S. Code: Title 18: Section 1030 (a)(2)(c)", 23 Jan 2000 URLhttp://caselaw.lp.findlaw.com/scripts/ts_search.pl?title=18&sec=1030, (22 August 2002)

purpose was commercial advantage, malicious destruction, or private commercial gain, the maximum imprisonment is not more than one year for the first offense and not more than 2 years for subsequent offenses. Otherwise, the maximum prison sentence is 6 months.

The perpetrator is likely in violation of the Computer Fraud and Abuse act, regardless of whether or not the program was executed. Even if he/she were authorized to access the computer, it could be argued that he/she exceeded authorization by installing the unauthorized program for illegal purposes. Conviction of this act could lead to a fine and imprisonment of not more than 10 years.

Interview Questions

The following questions are generic and were derived from my experiences in the law enforcement field as well as information obtained from the textbook from Track 8 of the Sans Course.[20]

Question 1: You seem to have some highly developed computer skills - I am very impressed. Where did you learn about Linux and how to program in it?
Explanation: On the first question, I am trying to get my foot in the door. I am complimenting the subject and showing him some respect while attempting to validate his skills.

Question 2: I was most impressed with a file I found on one of our computers. It was a commonly found program, but had been altered. It appears that the library files it needed had been added to the main program. I have never seen anything like that before, have you?

Explanation: Here I am continuing to complement his skills with a computer, the weakness of every computer savvy person I have met. I am feigning ignorance on an issue where he has knowledge. I expect he will tell me what it takes to build the file that was found on the system.

Question 3: The only thing about that file is that I don't know how it got there and that is presenting me with a big problem. If I can't figure this out on my own, I'll have to call in the Police to do an investigation. Do you think you could help me out here and avoid the outside interference.

Explanation: Now that he thinks I respect his skills and am on his side and he should be getting comfortable, I am throwing another factor into

[20] SANS Institute, Track 8 – System Forensics, Investigation, and Response, 8.4 – Forensic Frameworks and Best Practices – Managerial and Legal Issues,
Sans Institute, 2002, 2-52 through 2-69

the mix. I am betting he is going to want to avoid the Police coming into the scene and will give up some more information on the file.

Question 4: Network sniffers are commonly used for maintenance or operations reasons so maybe there is a good reason for it to be there. Do you see any reason to have a Sniffer on this particular computer?

Explanation: At this point, I am trying to give him a lifeline or an excuse for the existence of the program. This may give him incentive to come clean about the program. If not, the next question combined with this one will help push him in that direction.

Question 5: If I can't find out on my own why this program is here and what its purpose is, legitimate or not, I will have no choice but to call the Police in for a criminal investigation. If this happens, the person who put it there could well end up going to jail and I don't want that to happen. Why don't you help me out here and let's get this thing closed and move on to something else.

Explanation: Now I am putting the pressure on. I have given him a lifeline, an excuse as to why the program could be there, now I am letting him know that if anyone but me has to figure this out that the consequences may be harsh and I will have no control of it.

Additional Information

http://www.phreak.org/archives/exploits/unix/network-sniffers/interface-promiscuity-obscurity.txt

http://www.cet.nau.edu/~mc8/Socket/Tutorials/section1.html

http://www.phreak.org/archives/exploits/unix/network-sniffers/The-Sniffer-FAQ.txt

Investigative Process

Preparation and Setup

The computer used for this analysis was a Forensic Air-Lite III from Forensic-Computers.com. This particular model sports a Pentium III 1.2 Ghz processor, 512MB of RAM, and a built in 37.2Gb IDE hard drive. This system is designed for forensic investigations and includes cabling and power to attach external IDE and SCSI drives. It also has an IEEE 1394 (Firewire) bus with Forensic-Computers.com Firewire to IDE Write-Protection boxes allowing connection to IDE drives through Firewire without the worry of accidentally writing files to a drive under investigation. The bios on the system is Phoenix –

AwardBIOS v6.00PG. The operating system used for the analysis was Redhat Linux version 7.3. In addition VMWare for Linux was installed in order to simulate a network environment within the test computer. The analysis computer was connected to a closed network to ensure no damage could be done outside of the test system.

The binary file will be downloaded to a clean drive to ensure no contamination from previous files occurs. The drive to be used in this case is a Seagate Model ST34313A IDE drive. It has 8944 cylinders, 15 heads, and 63 sectors, with a capacity of 4.3 Gbytes. The serial number of the drive is 7DI07XV3.

Running KillDisk[21] on the drive with three passes cleaned the drive to ensure no contamination was present prior to the download. The file "sn.zip" was then downloaded from the assignments page directly to the clean drive. The next challenge was to extract the file from the zip archive while preserving the original MAC times for the "sn.dat" file. This was accomplished by using the UNZIP[22] Utility with the syntax shown in Figure L-1. The goal of this part of the exercise was to extract the subject file from the archive and back to its original state. The –X parameter was used to restore the original UID/GID and timestamp information.

Two files were extracted onto the test drive, "sn.dat" and "sn.md5". The "sn.md5" file contained an MD5 hash value[23] of "0e954f43fd73f56e812a7285f32e41d3" for the "sn.dat" file. An MD5 has value of the restored file was retrieved as shown in Figure L-2, which matched "sn.md5" exactly. This serves as proof the extracted file is an exact duplicate of the original file.

Since an original copy of the file is stored in the zipped file, that file was archived to another location for safekeeping. If it appears criminal proceedings may be in order, this file will be downloaded to a mobile media, verified as original, and then handled as evidence utilizing a "Chain of Custody" document. In addition, to ensure the working file is not changed causing inaccurate information to be extracted, CHMOD was used to make the file Read Only as shown in Figure L-3.

The Investigation

The "ls –i" command followed by the "debugfs –R 'stat <1655010>' /dev/hda2 " was then used to determine the MAC information for the "sn.dat" file as shown in Figure L-4. The results of these commands show that the file was last modified and accessed on Thursday April 11, 2002 at 09:29:58. The time

[21] KillDisk, Lsoft Technologies, Version 1.1
URL: http://www.lsoft.net, (20 August 2002)
[22] Unzip Utility Reference
URL: http://www.info-zip.org/pub/infozip/UnZip.html, (20 August 2002)
[23] MD5Sum
URL: http://www.etree.org/md5com.html, (20 August 2002)

shown for the created time is the time that the file was placed on the drive for analysis.

The next step in the process is to determine the type of file that we are dealing with. In order to do this, we simply use the FILE command as shown in figure L-5. The additional information that we have as a result is:

1. This is an ELF file. "Elf is a constraint logic programming language based on the LF Logical Framework", according to "The Elf Meta-Language" website[24]
2. The file is LSB executable.[25]
3. It is designed for the Intel 80386 processor.
4. It is statically linked, meaning all of the required functions are included in the binary.[26]
5. The file has been stripped, meaning the symbols have been discarded from the file.[27]

A "strings" command was executed on the file using the syntax "Strings sn.dat |less" in order to see what clues may lie inside the binary itself. Several key pieces of evidence were retrieved as shown in Figures L-6 and L-7. The information contained within the program, available for viewing by strings, was:

1. The program is ADMSniff, version 1.0. It uses pcap and Libpcap to control the functioning of the Ethernet NIC and to allow the program to capture (sniff) network traffic.
2. Keld Simonsen, a well-known programmer that is active in the ISO standards, wrote the program.[28] This does not mean there is evidence that he installed the program, as ADMSniff is available for download from hundreds of different locations. This particular version proved difficult to find, however.
3. The program was compiled on a Red Hat Linux version 7.1. This will be significant information if the source of the file can be found. Confirmation of the source will require the code be compiled and compared on the same operating system.

The next step was to run "gdb" against the program. It did see the program, helping to prove the theory that the file is an executable program.

---

[24] Pfenning, Frank "The Elf Meta-Language",
URL: http://www-2.cs.cmu.edu/~fp/elf.html, (20 August 2002)
[25] Pfenning, Frank "The Elf Meta-Language",
URL: http://www-2.cs.cmu.edu/~fp/elf.html, (20 August 2002)
[26] SANS Institute, "Track 8 – System Forensics, Investigation, and Response, Book 8.3" Pg 2-114
[27] SANS Institute, "Track 8 – System Forensics, Investigation, and Response, Book 8.3" Pg 2-114
[28] Simonsen, Keld, "News About Standardization", 21 Nov 2000
URL: http://www.usenix.org/publications/login/standards/37.simonsen.html, (20 August 2002)

Running several different gdb tests against the file did not prove useful in finding out more about the program than was already known.

Then "objdump" was run against the program with results shown in Figure L-8. The gconv libraries are also used with this program.

Next, since the program is an ELF program, "readelf" was run to get more details on how the program operates. Results of this are shown in Figures L-9, L-10, and L-11. The following facts were noted:

1. The entry point of the program was a normal entry point of 0x80480e0, which is a standard entry point.
2. The Libpcap libraries are indeed compiled inside this binary as the headers point to them as shown in Figure L-10.
3. The program has 19 headers.

At this point in the analysis, there was not much more that could be done without more in-depth testing, including the execution of the binary file. Before that was done, additional groundwork was laid to make sure we retrieved all the data needed for the analysis. The data needed was to find what was happening on the Ethernet interface when the program was running, what processes were spawned, what files were created, and what files were opened. It was determined that the best way to get this information was to get a before, during, and after snapshot of the system using "lsof", "netstat", "ps", and "ifconfig".

In addition, there is the possibility that a Trojan of some type may be imbedded in the binary file in order to advertise that it was successful in infiltrating and running on the victim machine. According to Symantec "A trojan horse (or "trojan") is simply a program that purports to do one thing but does something else that you do not know about. It has, so to speak, a public agenda that is harmless, and a private agenda that is not. One particular sub-category of trojans makes it possible for someone else to access your computer over the Internet."[29] Specifically, the concern here would be that a Trojan like program was included in the binary to send out a TCP connection to allow someone to get through a firewall to access the compromised machine. Many firewalls will block all traffic coming inbound, but will allow inbound connections and their replies to flow through the firewall. We will look for that by sniffing the network traffic coming from the victim machine with another computer with a Sniffer installed.

In order to setup the tests, several actions were taken. First, two VMWare[30] sessions were installed; one in Linux 7.3 and one in Linux 7.1 to match the system the file was compiled on. An internal "network" was established between the VMWare sessions using custom networking. This is defined by VMWare as "Any type of network connection between virtual machine and the host that is not bridged or host-only networking. For instance, different

[29] Symantec Anti-Virus Site, "Explanation of Trojan Horses" 29 Jul 2002
URL:http://service2.symantec.com/SUPPORT/nip.nsf/1b078893dcd782a985256771004dfaa5/4b119f1de20fb66188256862007b3a5e?OpenDocument

[30] VMWARE Workstation 3.1, VMWARE, Inc.
URL: http://www.vmware.com/company/ (August 27, 2002)

virtual machines can be connected to the host by separate networks or connected to each other and not the host. Any network topology is possible."[31] In this case, two virtual connections were set up networked together and not to the host, using a virtual hub. The VM session with Linux 7.3 was setup with Ethereal in order to monitor traffic from the other session, which had the "sn.dat" binary executed.

Second, a baseline of the "7.2" system was obtained in order to have an after the fact comparison between the system prior to execution, during execution, and post execution. This was accomplished by :

1. "IFCONFIG –a" was run to see if any changes to the Ethernet interfaces were made by the execution of the program.
2. Netstat was run to see any pre-existing connections to the system.
3. Running "lsof" to get a list of all open files prior to execution. It was piped to a text file that will later be used with the diff command.
4. Running "ps –a" and piping the results to a text file for later analysis using the diff command.

Third, a Sniffer program "Ethereal" was installed on the "7.3" virtual computer in order to monitor traffic from the test virtual computer. This will detect traffic caused by the ADMSniff program to see if it was modified to send out notifications that it is active, to make a connection to allow a hacker back through a firewall, or other anomalous traffic not normally associated with ADMsniff.

First the program was started using the command "strace –f ./sn.dat eth0", which started the program using the strace functionality. As shown in Figure L-12, the strace results show some interesting information. First, the program sets up the environment; there is nothing substantial or unusual there. Second, the program then opens up a network socket and identifies an IP subnet. This subnet was researched and it belongs to IANA (Internet Corporation for Assigned Names and Number), the company that controls IP addresses, and it appears to be in a reserved range.[32] Third, the program runs the "fstat" command. "*fstat* obtains information about an Open file known by the file descriptor *fildes*, obtained from a successful *creat*, *open*, *dup*, *fcntl*, *pipe*, or *ioctl* system call."[33] It appears to be determining the status of the "sn.dat" program. Fourth, the program prints to screen a banner showing "ADMsniff priv 1.0" along with other greeting type information for the user. Lastly, and most importantly, the program creates or opens a file named "The_l0gz", probably the log file.

The next step was to fully run the program by using the command "./sn.dat eth0 &". Once the "sn.dat" program was running, the ethereal program, which was still running, was checked to see if any traffic was emanating from the test VMWare system. Figure L-13 shows that no traffic was outbound from that

[31] VMWARE Help Program
[32] ARIN Whois, "Search Results for 104.48.0.0"
URL: http://ws.arin.net/cgi-bin/whois.pl
[33] Unix Manpages, "stat(2)",
URL: http://www.mcsr.olemiss.edu/cgi-bin/man-cgi?fstat+2

"computer". There does not appear to be any type of Trojan program that started transmitting when "sn.dat" was executed. While it is possible that such a program could be time delayed, there are no indications that this is so.

Next, "lsof", "ps", "ifconfig", and "netstat" were run to see what in the system had been modified while the "sn.dat" file was executed. According to the before and during netstat checks, there were no new connections established by the program. Checking the ps results show that the "sn.dat" process had been executed, but nothing else new. The ifconfig results show that the Ethernet card had been placed in promiscuous mode as shown in Figure L-14. It should also be noted that after "sn.dat" was killed, the card remained in promiscuous mode. Detection of this mode may indicate that this program has been executed on other devices. The lsof results show that the "sn.dat" program has opened several system files as shown in Figure L-15. It should be noted here that lsof confirms that the "The_l0gz" file was created and opened by "sn.dat".

Next, with "sn.dat" process killed, a search was done on the file system for all files that had been modified since the "sn.dat" file was copied to its current location. This was done by using the Find command as follows: "find /* -newer /gcfa/sn.dat –type f –printf "%T@ %k %h/%f\n" | sort".

While checking for files on the file system that had been modified since the "sn.dat" file was transferred, a file called "/proc/kcore" was showing an unusual owner as shown in Figure L-16. Research revealed this file was an alias for the physical memory in the computer. According to Unixguide.Net, "/proc/kcore is like an "alias" for the memory in your computer. Its size is the same as the amount of RAM you have, and if you read it as a file, the kernel does memory reads."[34] The file "/proc/kcore" was searched using the command "strings /proc/kcore | grep sn.dat > /gcfa/kcore.txt". The kcore.txt file had a lot of references to "sn.dat", but most were commands that had been used in the preparation for and testing of the "sn.dat" file. The only useful information obtained was a lack of information; further confirming that "sn.dat" did not execute other processes.

The last test that was conducted was to determine if any further information could be obtained about the running process of "sn.dat" by searching through a process dump by running a kill –5 command on the "sn.dat" process. The "sn.dat" file was first executed and then the process of getting the core dump information was accomplished. The commands for this was "ps –ef | grep sn.dat" in order to obtain the proper process id, which was 866. The next step was to issue the kill –5 process as "kill –5 866", which caused the core dump. The core dump was then accessed by "strings –a core | less". Unfortunately, there was no information in the dump that was not already known.

Finding the File on the Internet

A comprehensive search of the Internet was accomplished using several search engines with the search string of "ADMsniff Private 1.0". The only search

[34] UnixGuide.Net, "What is /proc/kcore?"
URL: http://www.unixguide.net/linux/faq/04.16.shtml , (August 27, 2002)

engine that returned links was Google.[35] The only site that advertised the document was www.unixhq.org. Attempts at downloading from that site resulted in a 404 "File could not be found" error. A follow up search for the string "ADMsniff" was accomplished. There was a plethora of sites that advertised ADMsniff as a download, however, the majority of them was version .8 and were obviously not the file that was being sought. The site that was advertised as the ADM Crew official ftp site, http://adm.freelsd.net/ADM/, contained the ADMsniff file. It was downloaded and a "strings" was run on the "thesniff.c" file located within the archive. There were keywords found that indicated the file was the correct one. Additional searching resulted in also finding the file at another site, http://packetstormsecurity.nl/groups/ADM/indexsize.shtml. This site advertised an MD5 hash value of **352e5e3a460ded8917c27114713dd794** for the file, useful in ensuring the download was not tampered with.

The modified dates of the downloaded archive file would seem to indicate this particular file was created before the suspect file, "sn.dat" had been run last. The dates on the download from Packetstormsecurity were identical, making this the best opportunity to identify the file. The archive contains another archive file for Libpcap version 0.4. Reviews of the modified dates on the files within that archive show the latest modified date to be 7/25/1998. Chances are therefore good that the Libpcap file is the same version as the one that was used in "sn.dat".

In order to confirm the file is the same file that was found on the victim machine, it is absolutely necessary to set up the system as near to the original configuration as possible. The version of the Kernel of the install machine and the version of the compiler must all be the same to achieve an absolute match. Even then, the chances are rather slim that we will achieve an MD5 hash value match, considering that a single bit that is different will cause different hash results.

We know two things about the binary that we can start setup correctly from the beginning. First, the binary is statically linked. This means that in our attempts to make the downloaded file identical we have to static link it as well. This was accomplished by editing the "Makefile" by adding the "-static" entry to the "CFLAGS =" line. Second, the binary is stripped. This is accomplished by using the command "Strip <binary name>" after the file has been compiled. This in mind, we setup the downloaded file.

The file was copied to our "/gcfa" directory and was unarchived by using the tar command as: "tar –zxvf ADMsniff.tar.gz". This created a new directory "/gcfa/ADMsniff". In this directory, the Makefile was edited as described earlier using the VI editor. The "Make" command was then executed. This created a binary file called "ADMsniff-1". The command "strip ADMsniff-1" was executed to strip the file as described earlier. The new binary file was examined and was found to be slightly smaller than "sn.dat". Since that is the case, it would be pointless to try and compare md5 hash values for the two files. Instead, the

[35] Google Search Engine
URL: http://www.google.com, (20 August 2002)

strings command was run against each of the files with the results piped to a text file. These text files were then compared using the diff utility.

The information gathered from the diff utility alone strongly indicates that these two programs are from the same source. It was noted that the GCC compiler for the two binary files was slightly different, which may account for the differences found in the two programs. Otherwise, the great majority of lines from the strings command were identical between the two files.

Next, the results of "readelf", Figure L-17, between the two files were compared and were found to be almost identical. Even the "Magic" number was identical, the starting point was identical, as was the headers. This is very compelling evidence that these two files have the same source, despite the inability to match MD5 hash values.

Part 3
"Legal Issues of Incident Handling – The Wiretap Statute"

The Wiretap Statute

The Wiretap Act is derived from "18 USC 2511", which is the identifier for Title 18 of the United States Code, section 2511. Specifically, Chapter 19 of that act concerns wiretapping and is officially labeled as "Wire and Electronic Communications Interception and Interception of Oral Communications".

Section 2511 (1) specifies the following: "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; (b) intentionally uses, endeavors to use, or procures any other person to use or endeavor to use any electronic, mechanical, or other device to intercept any oral communication when - (i) such device is affixed to, or otherwise transmits a signal through, a wire, cable, or other like connection used in wire communication; or (ii) such device transmits communications by radio, or interferes with the transmission of such communication; or (iii) such person knows, or has reason to know, that such device or any component thereof has been sent through the mail or transported in interstate or foreign commerce; or (iv) such use or endeavor to use (A) takes place on the premises of any business or other commercial establishment the operations of which affect interstate or foreign commerce; or (B) obtains or is for the purpose of obtaining information relating to the operations of any business or other commercial establishment the operations of which affect interstate or foreign commerce; or (v) such person acts in the District of Columbia, the Commonwealth of Puerto Rico, or any territory or possession of the United States; (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)-(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)."[36]

---

[36] "Laws: Cases and Codes : U.S. Code : Title 18 : Section 2511 (1)", 23 Jan 2000
URL: http://caselaw.lp.findlaw.com/scripts/ts_search.pl?title=18&sec=2511 , (22 August 2002)

This code forms the basis for the wiretapping laws in the United States. While it applies to any electronic communications, per Section 2511 (1)(b) "any wire, oral, or electronic communication", the law was originally used for wiretapping of telephone devices.

It is clear from this law that it is illegal to intercept communications over a wire without the senders express authorization. There are exceptions to this law whereby "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."[37]

This section of the law allows service providers to monitor and intercept traffic for certain purposes.

This law applies to the interception of computer communications and gives the system administrator certain exemptions to the law. A newer legislation was passed that was specifically written for computer communications and is easier to understand in the context of computer communications. The Electronic Communications Privacy Act that is derived from "18 USC 2701", which is the identifier for Title 18 of the United States Code, section 2701.[38] This code sets forth specific prohibitions against the unauthorized interception of communications, including computer communications, and establishes rules that must be followed to avoid violation of this code.

Section 2701 (a) of the code specifies what constitutes a violation of this law. It specifically says "Offense. - Except as provided in subsection (c) of this section whoever -(1) intentionally accesses without authorization a facility through which an electronic communication service is provided; or (2) intentionally exceeds an authorization to access that facility; and thereby obtains, alters, or prevents authorized access to a wire or electronic communication while it is in electronic storage in such system shall be punished as provided in subsection (b) of this section." Section 2702 of the code further defines the offense as: "(a) Prohibitions. - Except as provided in subsection (b) - (1) a person or entity providing an electronic communication service to the public shall not knowingly divulge to any person or entity the contents of a communication while in electronic storage by that service; and (2) a person or entity providing remote computing service to the public shall not knowingly divulge to any person or entity the contents of any communication which is carried or maintained on that

---

[37] "Laws: Cases and Codes : U.S. Code : Title 18 : Section 2511 (2)(a) (1)", 23 Jan 2000
URL: http://caselaw.lp.findlaw.com/scripts/ts_search.pl?title=18&sec=2511 , (22 August 2002)

[38] "Laws: Cases and Codes : U.S. Code : Title 18 : Section 2701 (a)", 23 Jan 2000
URL: http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/i/chapters/121/sections/section_2701.html , (22 August 2002)

service - (A) on behalf of, and received by means of electronic transmission from (or created by means of computer processing of communications received by means of electronic transmission from), a subscriber or customer of such service; and (B) solely for the purpose of providing storage or computer processing services to such subscriber or customer, if the provider is not authorized to access the contents of any such communications for purposes of providing any services other than storage or computer processing."[39]

As you can see Title 18, Sections 2701 and 2702 clearly outline what the law prohibits. Fortunately, it was recognized by the lawmakers that the restrictions that would be imposed by the code we have seen so far would literally cripple the capability of network administrators to operate and maintain the networks. They inserted specific exclusions to the law that allow persons or entities that operate these networks to access the data crossing the wires. Specifically, the exclusion says in Section 2701: "(c) Exceptions. - Subsection (a) of this section does not apply with respect to conduct authorized - (1) by the person or entity providing a wire or electronic communications service; (2) by a user of that service with respect to a communication of or intended for that user; or (3) in section 2703, 2704 or 2518 of this title."[40] The exceptions are further defined in Section 2702 (c) where it says: "(b) Exceptions. - A person or entity may divulge the contents of a communication - (1) to an addressee or intended recipient of such communication or an agent of such addressee or intended recipient; (2) as otherwise authorized in section 2517, 2511(2)(a), or 2703 of this title; (3) with the lawful consent of the originator or an addressee or intended recipient of such communication, or the subscriber in the case of remote computing service; (4) to a person employed or authorized or whose facilities are used to forward such communication to its destination; (5) as may be necessarily incident to the rendition of the service or to the protection of the rights or property of the provider of that service; or (6) to a law enforcement agency - (A) if the contents - (i) were inadvertently obtained by the service provider; and (ii) appear to pertain to the commission of a crime; or (B) if required by section 227 of the Crime Control Act of 1990."[41]

The challenge now is to understand exactly what this law means to the system administrators as they encounter traffic on their networks. It is clear that the intent of the law is to protect the privacy of the users of the network. Traffic that traverses the network is intended for a specific recipient or group of recipients and it is unlawful for other persons or entities to intercept this traffic. In the case of the system administrator, however, there is an exception "as may be

---

[39] "Laws: Cases and Codes : U.S. Code : Title 18 : Section 2702 (a)", 23 Jan 2000 URL: http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/i/chapters/121/sections/section_2702.html , (22 August 2002)

[40] "Laws: Cases and Codes : U.S. Code : Title 18 : Section 2701 (c)", 23 Jan 2000 URL: http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/i/chapters/121/sections/section_2701.html , (22 August 2002)

[41] "Laws: Cases and Codes : U.S. Code : Title 18 : Section 2702 (c)", 23 Jan 2000 URL: http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/i/chapters/121/sections/section_2702.html , (22 August 2002)

necessarily incident to the rendition of the service or to the protection of the rights or property of the provider of that service".[42]

The exception clearly gives system administrators the right to monitor traffic for specific purposes. First, the law gives the system administrator the right to monitor traffic when it is necessary to keep the system up and operating and keep the service available. The use of network sniffers, for example, is authorized for system administrators in the day-to-day maintenance activities of the network. Second, the law further expands the rights of the system administrators by allowing them to protect their network from unauthorized access and/or abuse by the words "protection of the rights or property of the provider of that service". This gives the system administrators the right to monitor traffic for unauthorized, unlawful, and malicious traffic that may be present. Law therefore authorizes the use of sniffers and Intrusion Detection Systems (IDS) as long as the reason for such traffic monitoring is to protect the network.

It should be noted that while the law give system administrators broad authorization to monitor their networks, it does not give them carte blanche to monitor and/or access anything and everything on their networks. The law specifies the purpose of the authorization and any action that takes place outside of that purpose would not be covered by the exemption to the law. An example of unauthorized access of data by a system administrator would be an administrator intercepting the direct emails of a particular customer for the purpose of determining account numbers, private information, or other information not directly connected to the providers network. Unless that system administrator could demonstrate that the reason for this intrusion was maintenance or protection, he/she would not be exempted from the law. It should be noted here that obtaining stored emails would probably not be a violation of the wiretap laws as ruled by a Federal Judge.[43] This would indicate that the law covers traffic traversing a network, but does not apply to data that has reached storage status on a device on the network.

Another issue raised in this law is the role of law enforcement and the system administrators pertaining to interception of communications. According to the law, system administrators may release contents of communications to law enforcement officials "(A) if the contents - (i) were inadvertently obtained by the service provider; and (ii) appear to pertain to the commission of a crime; or (B) if required by section 227 of the Crime Control Act of 1990."[44] According to this section of the law, system administrators can only release information to law enforcement officials when they inadvertently obtain information. In other words, law enforcement officials cannot use the exemptions from this law by getting

---

[42] "Laws: Cases and Codes : U.S. Code : Title 18 : Section 2702 (b)(5)", 23 Jan 2000 URL: http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/i/chapters/121/sections/section_2702.html , (22 August 2002)

[43] Mail Utilities, "Digging through Old Emails not Wiretap Violation, Federal Judge Rules", 29 Mar 2001, URL: http://www.mailutilities.com/news/archive/43/1102.html

[44] "Laws: Cases and Codes : U.S. Code : Title 18 : Section 2702 (b)(6)",23 Jan 2000 URL: http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/i/chapters/121/sections/section_2702.html , (22 August 2002)

system administrators to gather information for them. Another barrier to that action would be the sure argument that the system administrator was acting as an agent of law enforcement, an argument he/she would probably win. In addition, any information gathered must appear to pertain to the commission of a crime or it cannot be turned over to law enforcement.

In order for law enforcement officials to be able to directly gather communications for evidence, they must first obtain a court order as outlined in the Omnibus Crime Control and Safe Streets Act of 1968.[45] Nothing in other laws allows law enforcement, without a court order, to access communications in violation of the Wiretap Act. System administrators must be certain that all legal requirements are met before turning over information to law enforcement officials or they may find themselves liable in a criminal or civil suit.

In a situation where an intrusion of the network is detected, the system administrator is allowed under the law to investigate the intrusion by intercepting communications under the auspices of self protection as mentioned earlier. Another decision that the system administrator must face is the determination of when to call in law enforcement to the investigation. This decision should be influenced by the fact that the presence of law enforcement officials may hinder the ability of the system administrator to continue with his/her investigation as the system administrator may effectively lose the exclusions to the law. As stated earlier, law enforcement officials may not use system administrators to bypass the law. Once they become involved in an investigation, the system administrators' rights to monitor remain, but the proceeds of the monitoring may not be turned over to law enforcement. The reason is that the evidence obtained must be "inadvertently" obtained. If law enforcement were on the scene asking for particular types of evidence, the evidence obtained would not be by inadvertent means. As a result, system administrators should gather adequate evidence under the pretense of self-protection before law enforcement becomes involved.

Another law that may come into play is the Pen Register Statute, 18 USC 3121. It states "(a) In General. - Except as provided in this section, no person may install or use a pen register or a trap and trace device without first obtaining a court order under section 3123 of this title or under the Foreign Intelligence Surveillance Act of 1978 (50 U.S.C. 1801 et seq.). (b) Exception. - The prohibition of subsection (a) does not apply with respect to the use of a pen register or a trap and trace device by a provider of electronic or wire communication service - (1) relating to the operation, maintenance, and testing of a wire or electronic communication service or to the protection of the rights or property of such provider, or to the protection of users of that service from abuse of service or unlawful use of service; or (2) to record the fact that a wire or electronic communication was initiated or completed in order to protect such

[45] Delaney, Donald P., Denning, Dorothy E., Kaye, John, and McDonald, Alan R. "WIRETAP LAWS AND PROCEDURES WHAT HAPPENS WHEN THE U.S. GOVERNMENT TAPS A LINE", 23 September 1993, URL: http://www.cs.georgetown.edu/~denning/wiretap/Wiretap.txt, (22 August 2002)

provider, another provider furnishing service toward the completion of the wire communication, or a user of that service, from fraudulent, unlawful or abusive use of service; or (3) where the consent of the user of that service has been obtained. (c) Limitation. - A government agency authorized to install and use a pen register under this chapter or under State law shall use technology reasonably available to it that restricts the recording or decoding of electronic or other impulses to the dialing and signaling information utilized in call processing. (d) Penalty. - Whoever knowingly violates subsection (a) shall be fined under this title or imprisoned not more than one year, or both."[46]

The Pen and Register Statute is similar in purpose and function to the other two laws that have been reviewed in this paper. This includes the exemption for the provider of the service to allow for maintenance activities and for self-protection.

Logon Banners

Logon Banners refer to a text message displayed onscreen whenever someone logs onto a computer device, whether network based or independent. These banners can notify the user logging in of many things, but one of the most important is the authorized use of the system,

How important are logon banners? In recent years, they have received some attention as legal necessities. According to CERT, "Failure to include a logon banner regarding acceptable use of a computer system can make it difficult to prosecute violations when they occur. Indeed, legal cases exist in which defendants have been acquitted of charges for tampering with computer systems because no explicit notice was given prohibiting unauthorized use of the computer systems involved."[47]

"The system administrator is responsible for enabling the Logon Warning Banner for a system. The purpose of the banner is to ensure that all persons attempting to gain access to the system know that the system and its information are for official use only and that attempts to illegally logon to the system could lead to criminal penalties."[48] This is a clear statement of why logon banners are necessary. It also brings out the core of what the logon banner should say in order to be effective in a legal situation. Examples of legal banners may be found at http://www.itsc.state.md.us/info/InternetSecurity/BestPractices/WarnBanner.htm.

A system without a banner could be considered to be an open system for anyone to use. That is what the defense attorney will almost certainly argue, and according to earlier references stands a good chance of success. It is therefore incumbent upon the system administrator to identify his/her systems for authorized use only by logon banners. The banner should identify that the

---

[46] "Laws: Cases and Codes : U.S. Code : Title 18 : Section 3121", 23 Jan 2000 http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/ii/chapters/206/sections/section%5F3121.html (22 August 2002)

[47] Carnegie Mellon University, "Setting up a logon banner on Windows NT 4.0", 17 Mar 1999, URL: http://www.cert.org/security-improvement/implementations/i034.01.html (22 Aug 2002)

[48] ITSC, "Logon Warning Banners", URL: http://www.itsc.state.md.us/info/InternetSecurity/BestPractices/WarnBanner.htm (22 August 2002)

system is private and the information it contains is private. It should also note that any unauthorized access of the computer or data could lead to criminal sanctions. In this way, the argument of public versus private computer, legitimate or not, is nullified.

Another aspect (unfortunate) of the logon banner requirements is that many systems have default logon banners that welcome users onto the system. According to a document published by the DDN Security Coordination Center, a case was thrown out of court because the system in question had a logon banner that started with "Welcome to…"[49]

The absolute need for logon banners may have been reduced with recent legislation put in place as a result of the events of September 11, 2001. This, however, has not been tested in a court of law. Therefore, every system administrator should ensure that appropriate logon banners are in place on all of their network systems.

One limitation of banners is that they are normally only displayed at known entry points into the system. Therefore, if a hacker were to bypass the normal logon process, the logon banners may not be seen. It remains to be seen in court how this will be treated.

Conclusion

There are three laws, 18 USC 2511, 18 USC 2701, and 18 USC 3121 that apply to electronic communications that are in transit across system wires. The laws specifically prohibit the reception of these signals by anyone other than the recipient as designated by the sender of the information. The law has specific exemptions in place that allow system administrators to intercept signals for specific purposes. The two purposes that system administrators can intercept data is for network maintenance and operations and to protect the provider of the service from unauthorized activity. These exemptions are very broad making virtually everything open for the system administrator. However, it should be noted that a system administrator that is intercepting communications for purposes other than that proscribed by law is in violation of the law.

Logon banners are a vital necessity in the overall security planning and program for a network. Logon banners notify users, both authorized and unauthorized, that the system they are connected to is a private system to be used by authorized personnel only. The banner should even warn them that unauthorized access to the computer might result in the filing of criminal charges.

It is not reasonable to think that logon banners will actually prevent a would-be intruder from breaking into a computer. It does, however, give the system administrators and owners more options in prosecution of the offender than they may have without the logon banners.

[49] DDN Security Coordination Center, "Computer System Welcome Banners", 2 Mar 1990, URL: http://csrc.ncsl.nist.gov/secalert/ddn/1990/sec-9004.txt (22 August 2002)

Appendix A
Install Figures

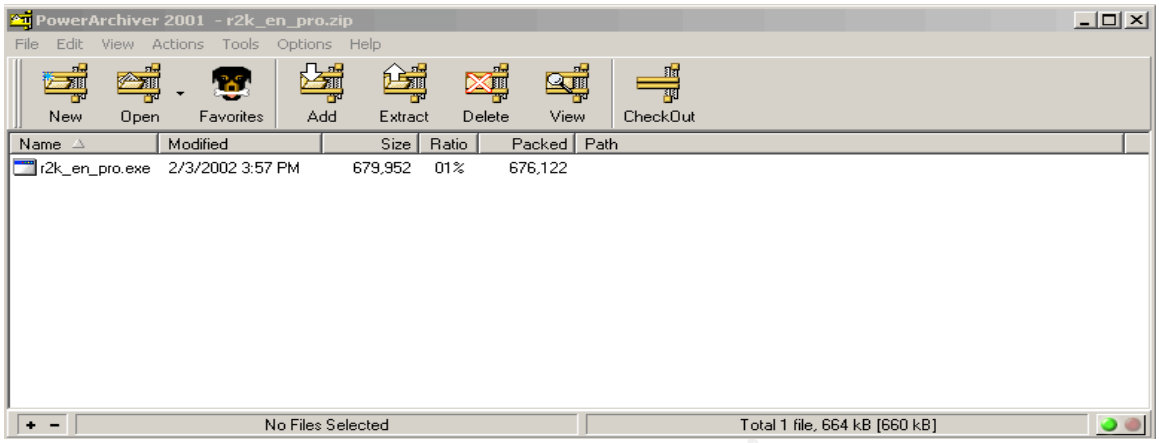


FIGURE A-1 (Contents of the downloaded installation zip file.)

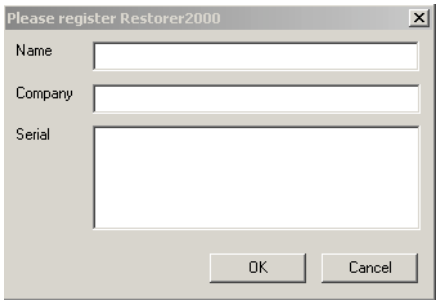


FIGURE A-2 (Registration Dialog Box.)

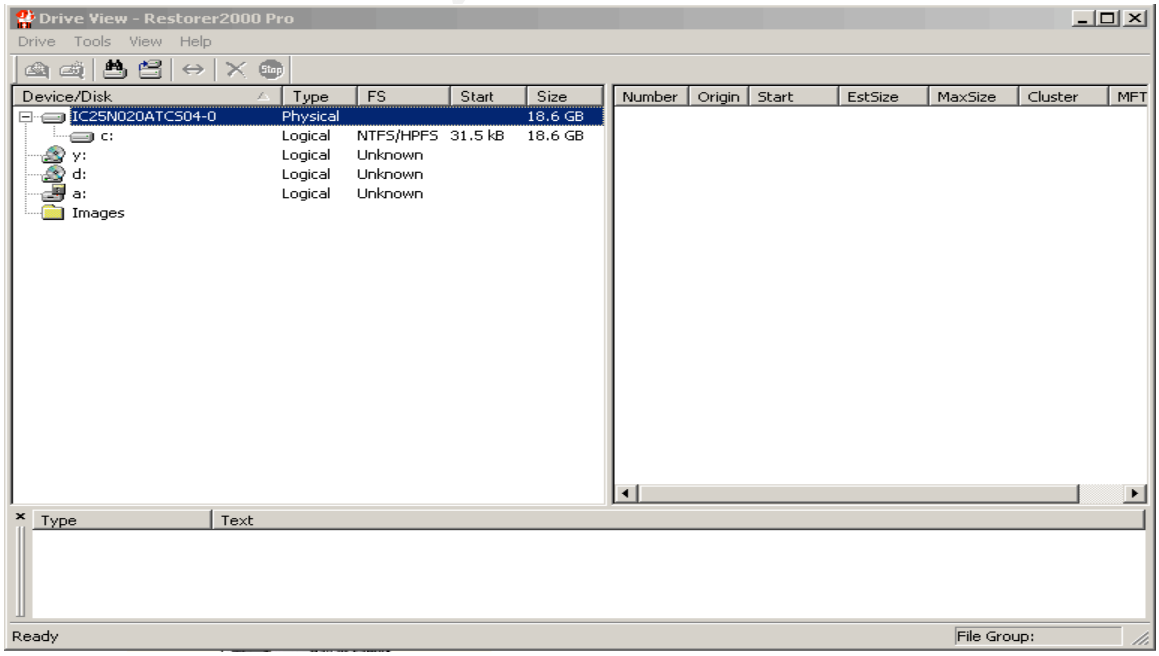


FIGURE A-3 (Main Console of Restorer 2000 Pro.)

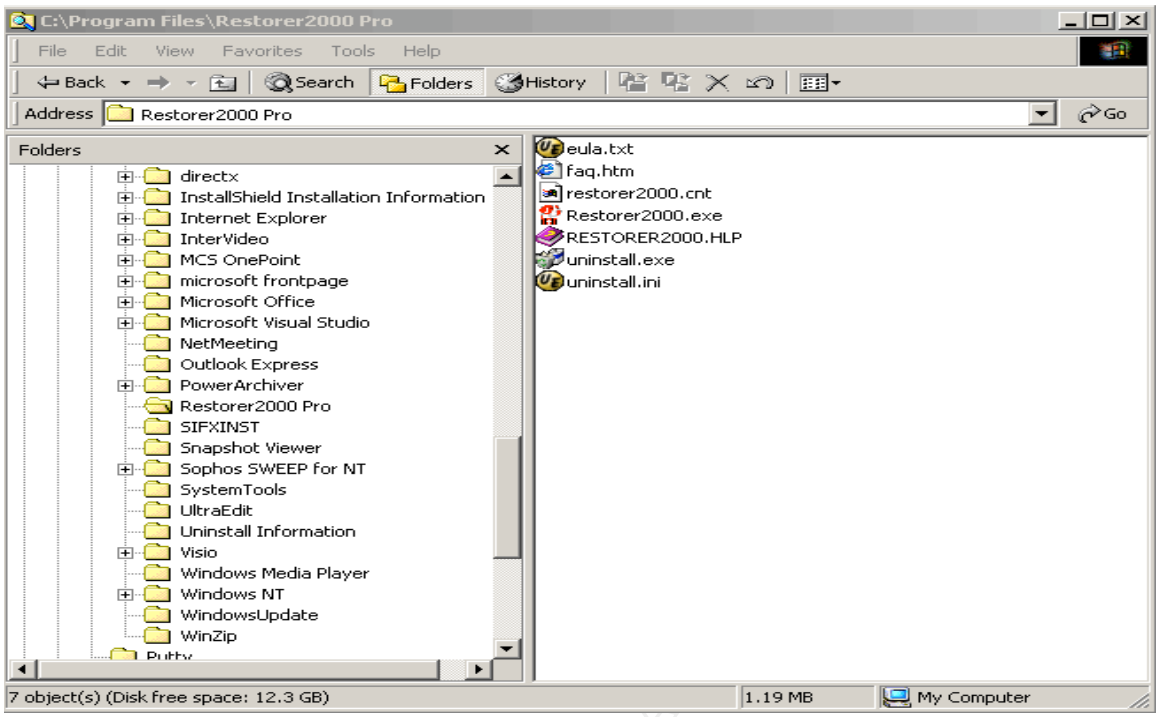


FIGURE A-4 (Contents of Restorer 2000 Pro Installation Directory.)

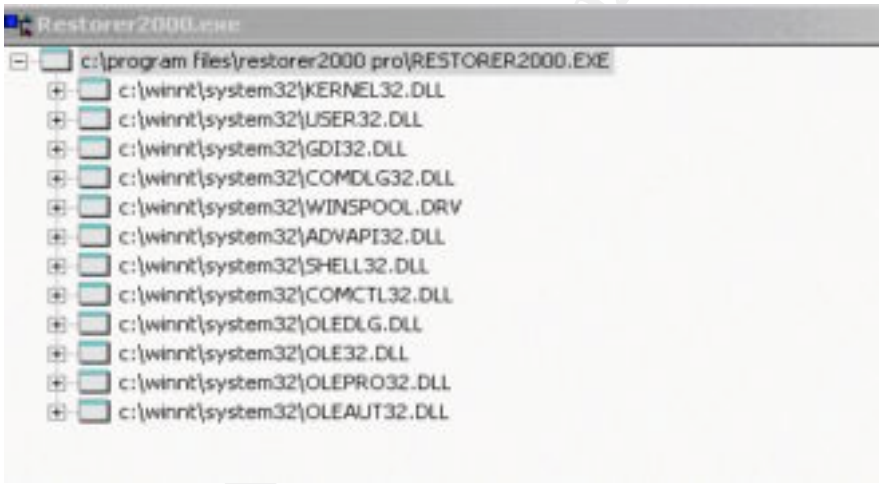


FIGURE A-5 (Restorer 2000 Pro File Dependencies.)

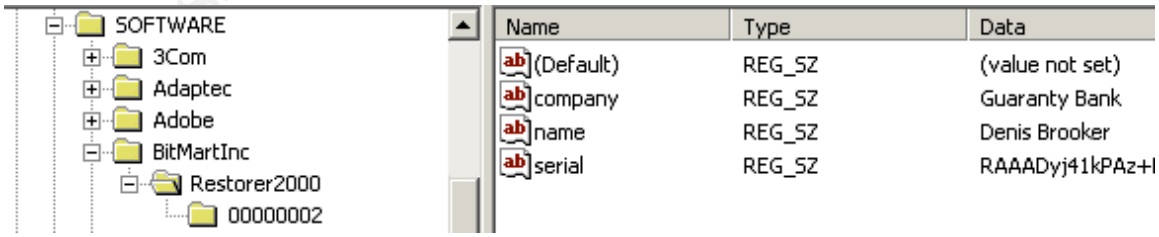


FIGURE A-6 (Registry Entry for Restorer 2000 Pro.)

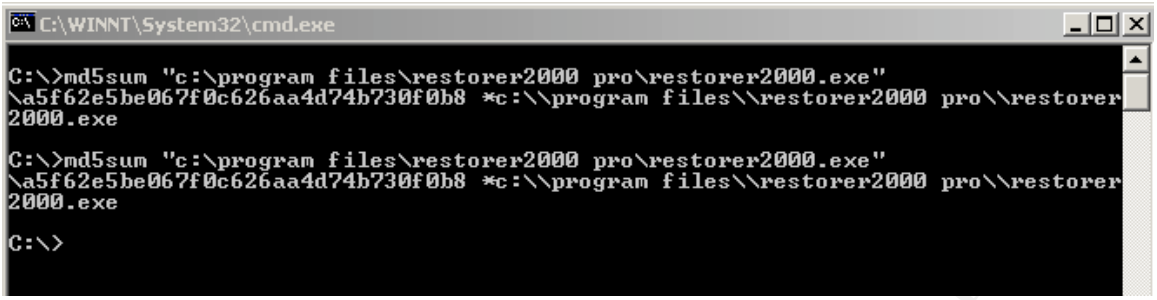


FIGURE A-7 (MD5 Hash Values of Restorer2000.exe before and after execution.)

Appendix B
Operations Figures

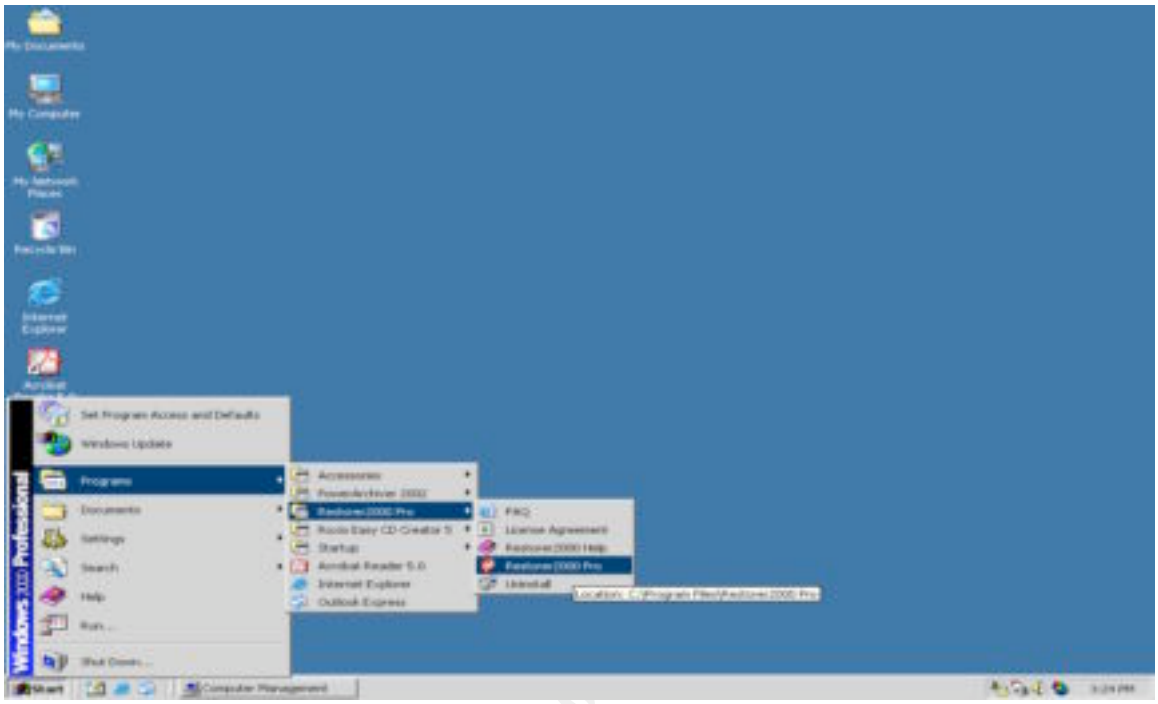


FIGURE B-1 (Starting the Program)

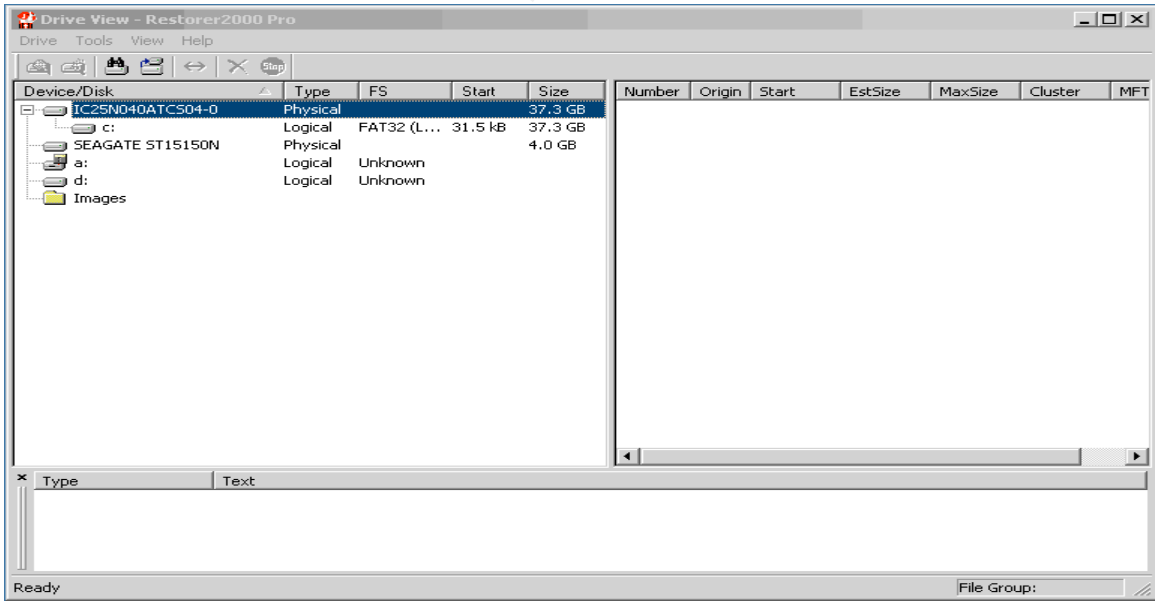


FIGURE B-2 (The Main Screen)

FIGURE B-3 (The Menu Section)

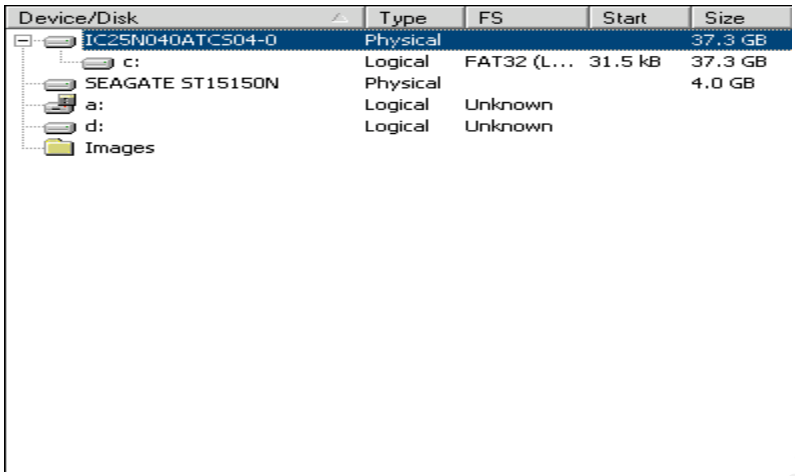


FIGURE B-4 (Device/Disk Section)

FIGURE B-5 (Files Section)

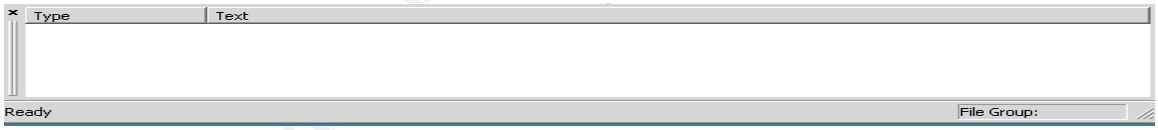


FIGURE B-6 (Status Section.)

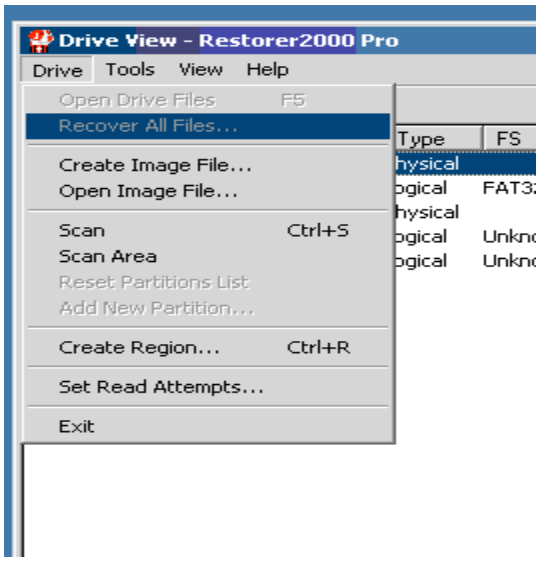


FIGURE B-7 (Drive Menu from a Physical Drive – The 'Recover All Files' is grayed out.)

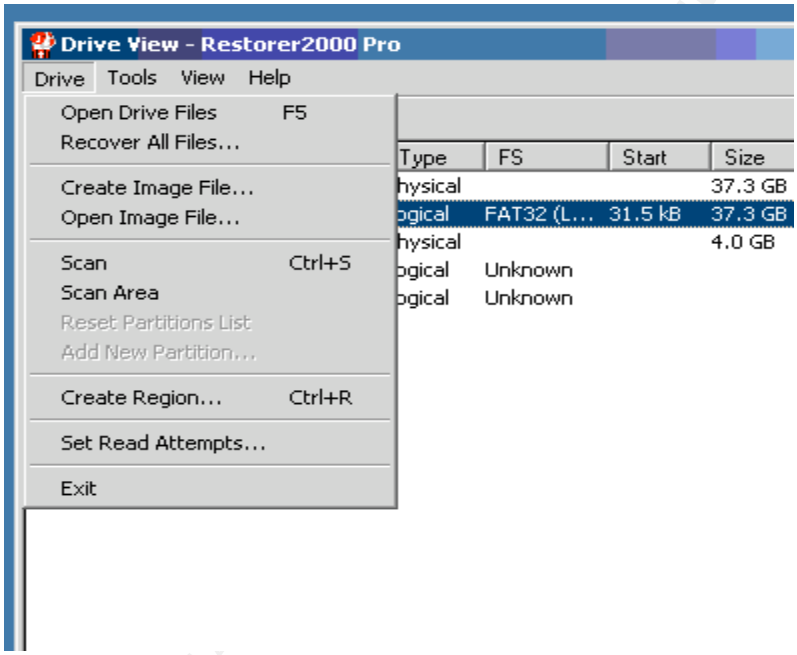


FIGURE B-8 (Drive Menu from a Logical Drive)

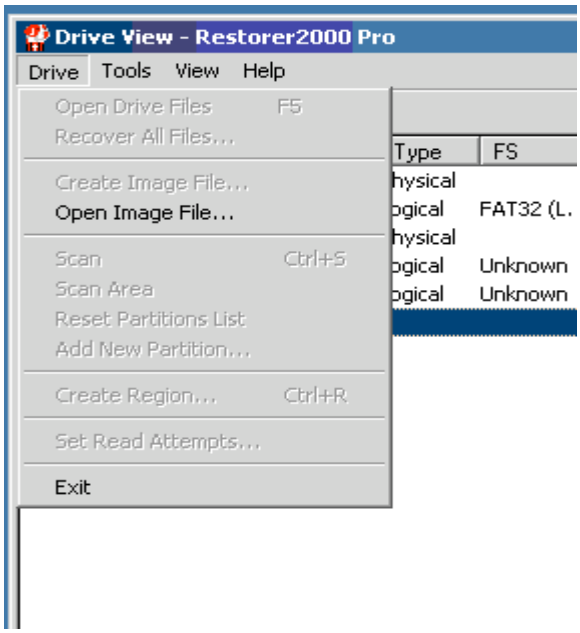


FIGURE B-9 (Drive Menu from the Image Folder)

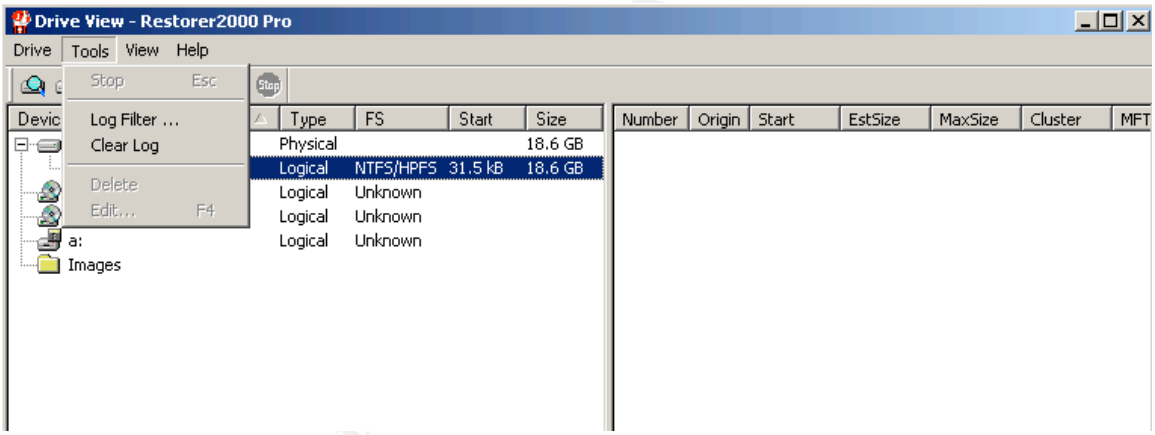


FIGURE B-10 (Tools Menu)

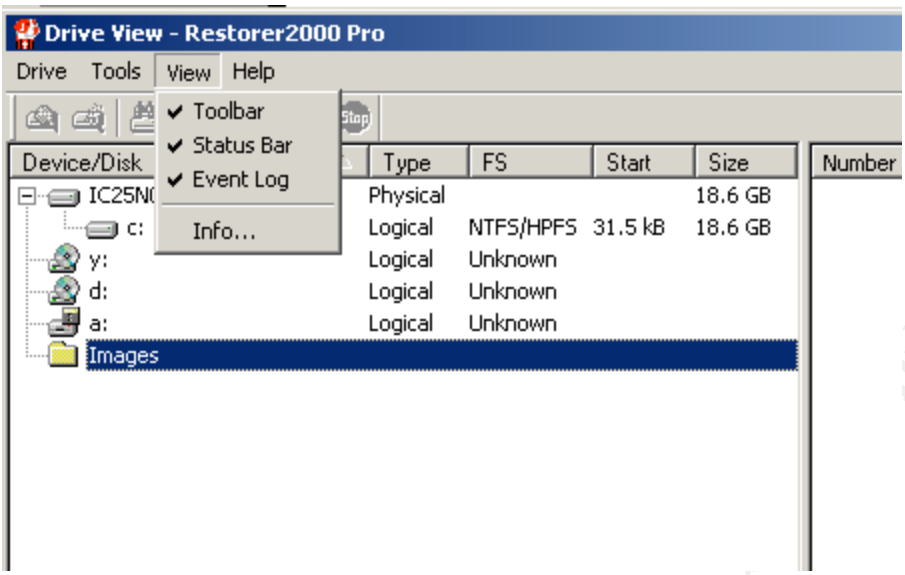


FIGURE B-11 (View Menu)

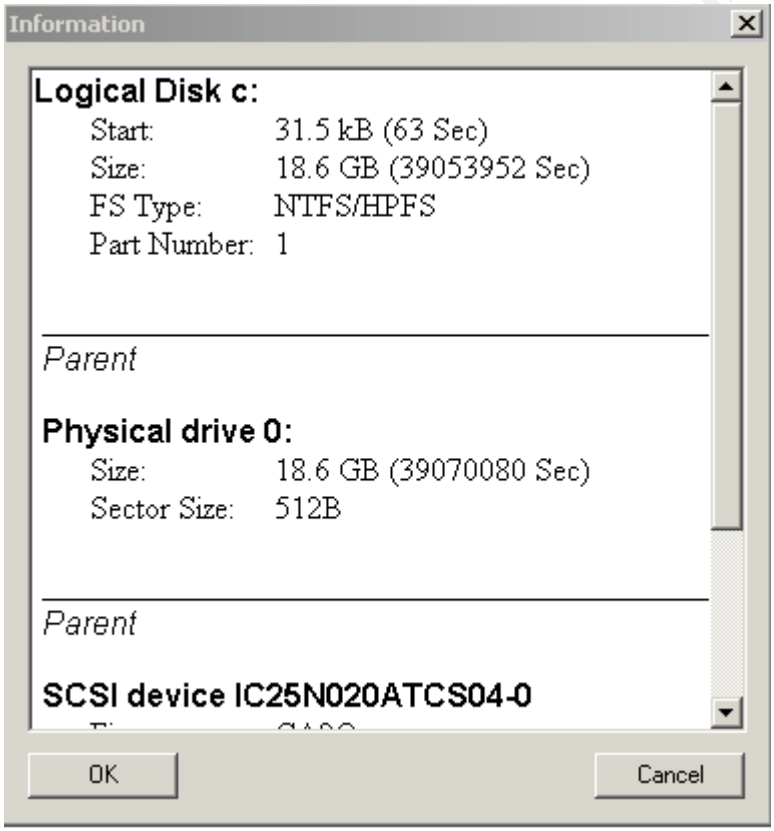


FIGURE B-12 (Information about a Logical or Physical Drive)

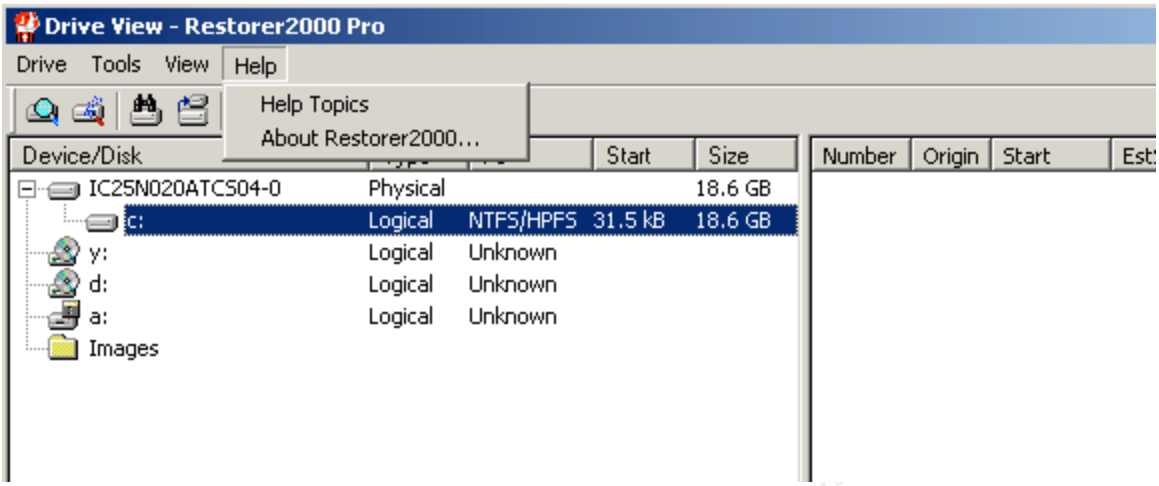


FIGURE B-13 (Help Topics)

FIGURE B-14 (About Restorer 2000 – Image has serial number removed.)

FIGURE B-15 (Icons)

Appendix c
Test 1 Figures

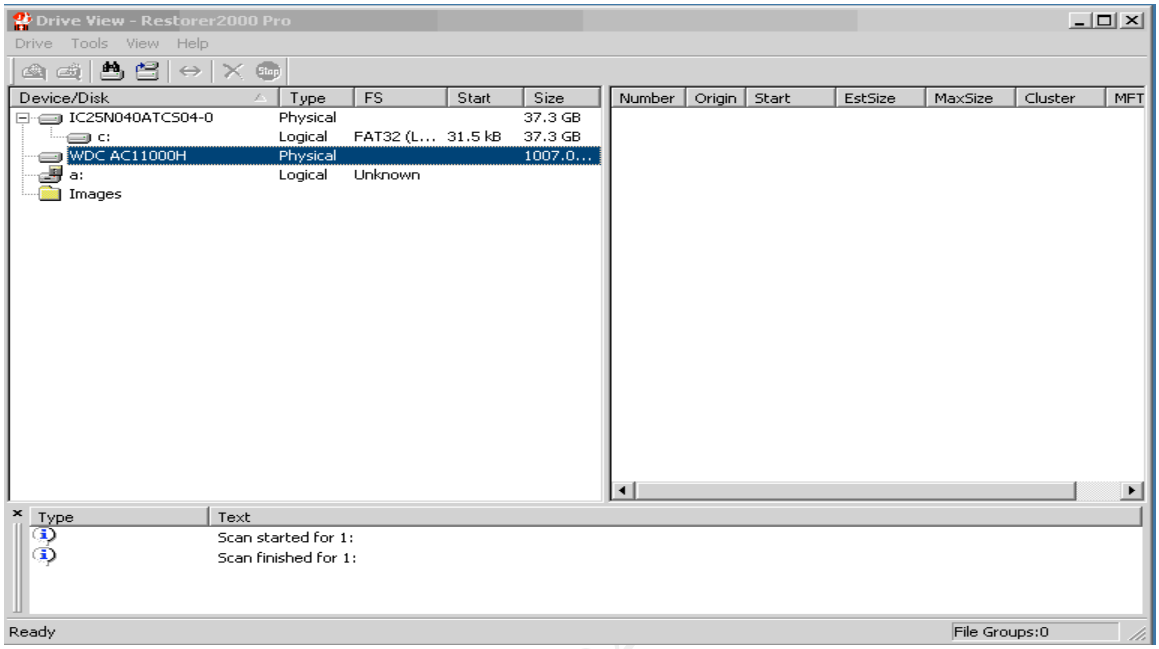


FIGURE C-1
(Shows there are no files on the drive. Note the IDE drive is selected and the scan shows started and completed in the bottom window. If files were found, they would appear in the right window.)

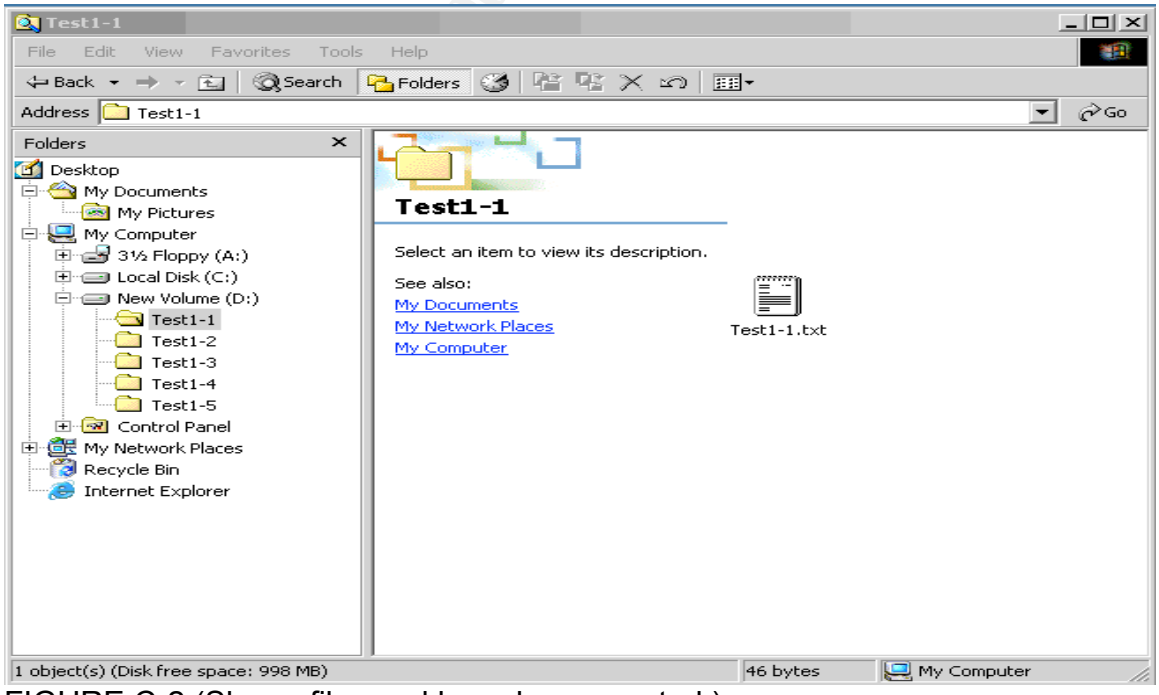


FIGURE C-2 (Shows files and have been created.)

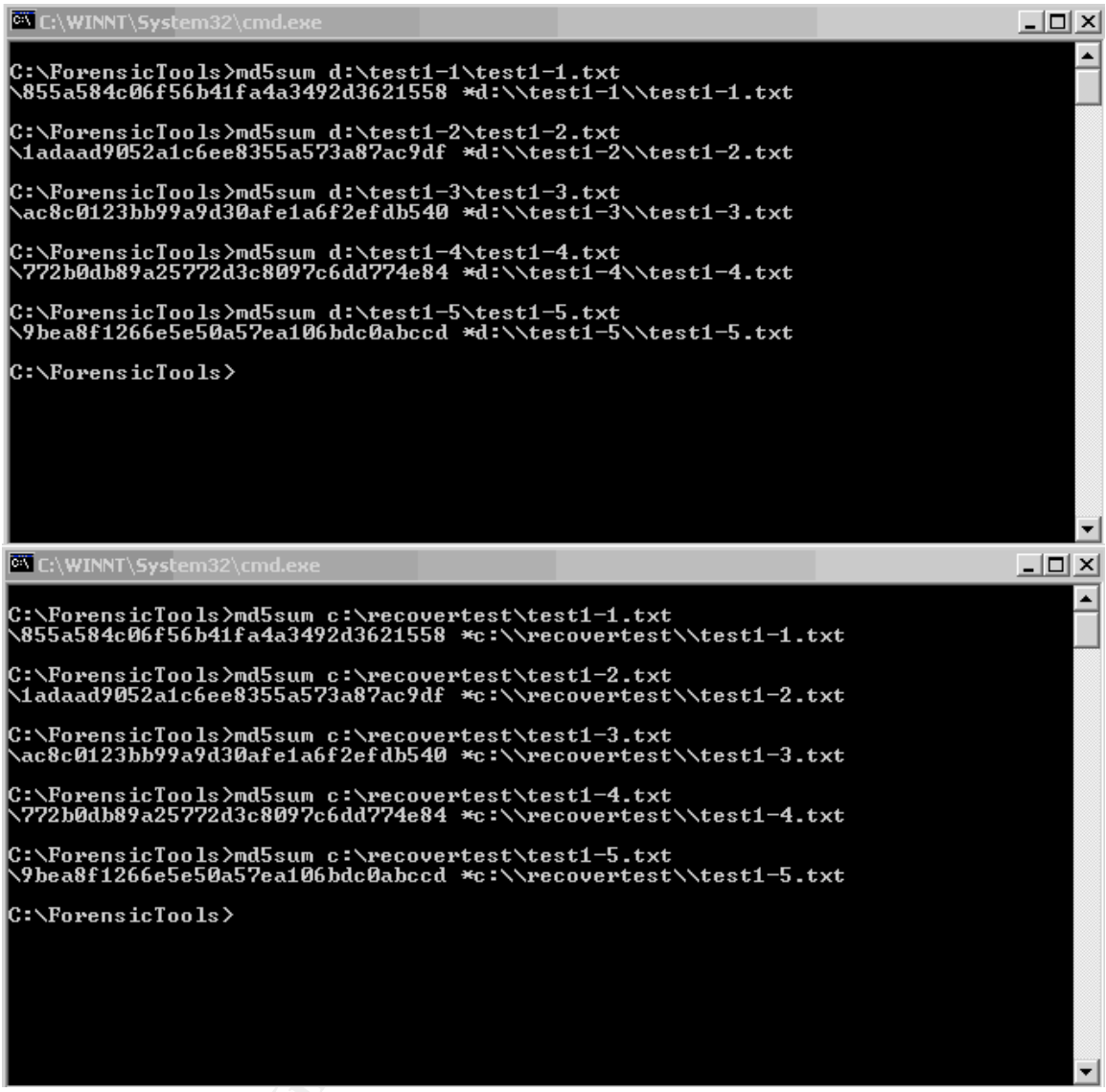


FIGURE C-3 (MD5 Hash Values of the Test Files [Top] and the Recovered Files [Bottom])

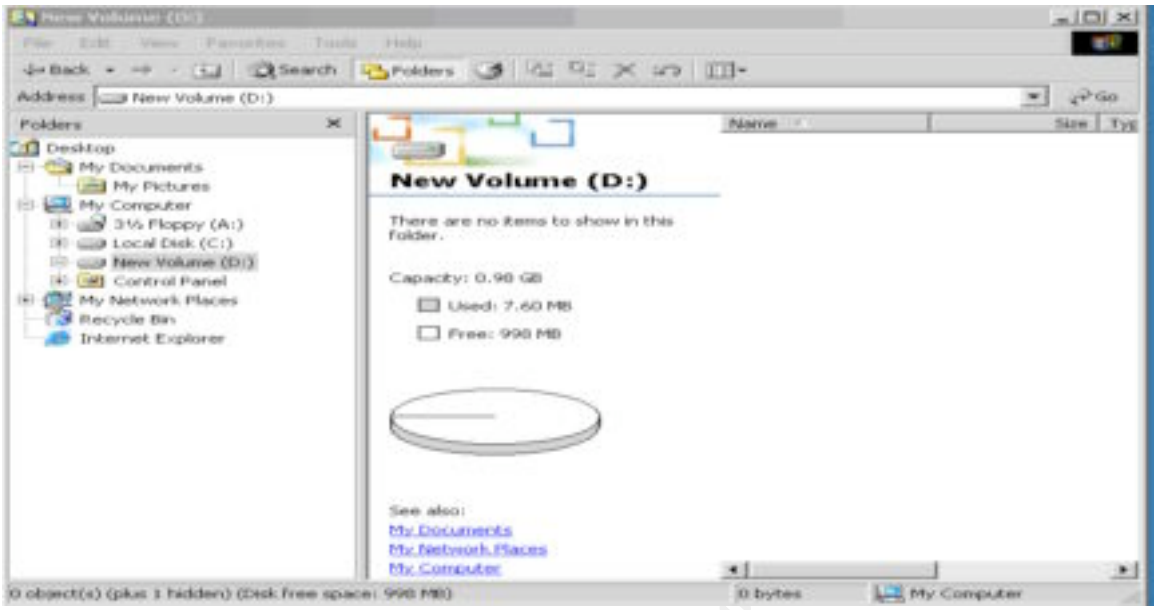


FIGURE C-4 (Files Have Been Deleted)

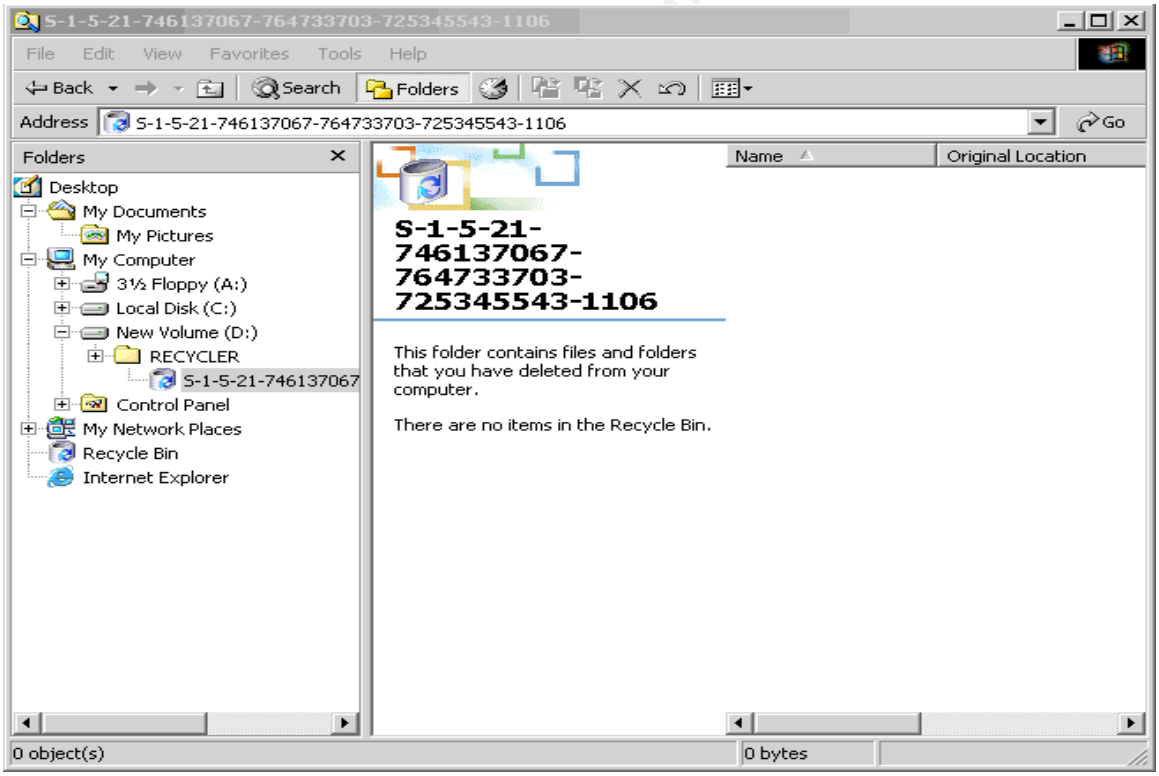


FIGURE C-5 (Recycle Bin on the IDE Drive has been emptied.)

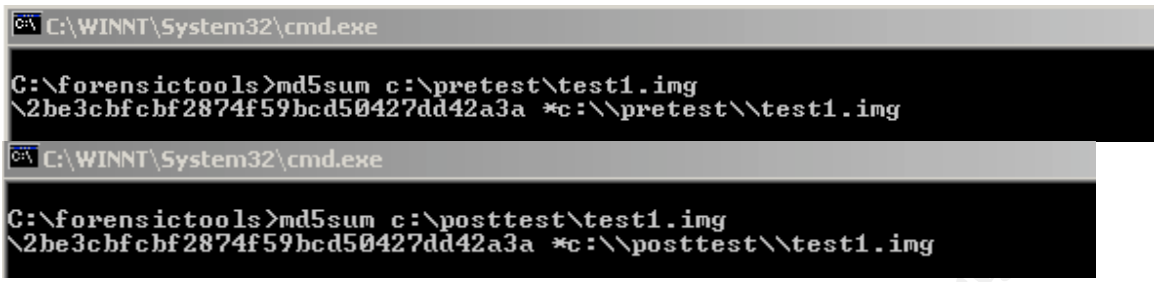


FIGURE C-6 (MD5 Hash Values of Drive Images before and after the testing.)

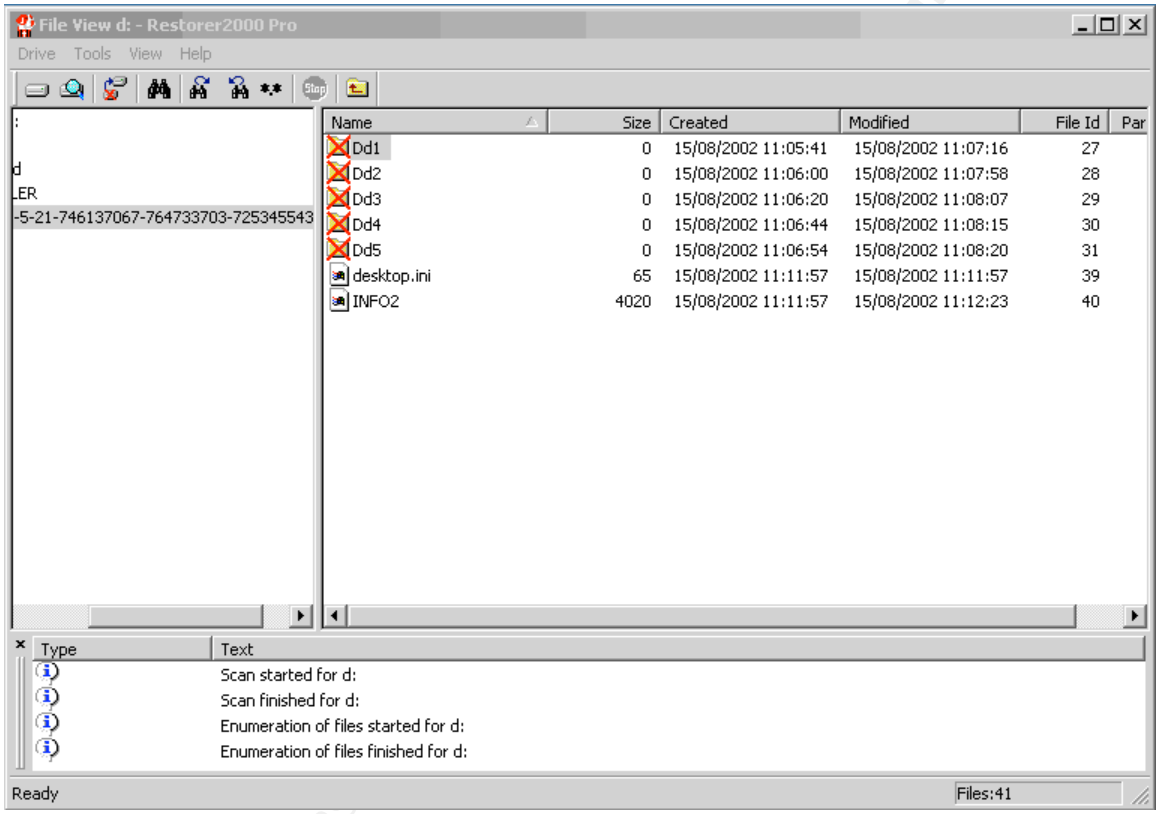


FIGURE C-7 (Restorer 2000 Pro found the deleted files.)

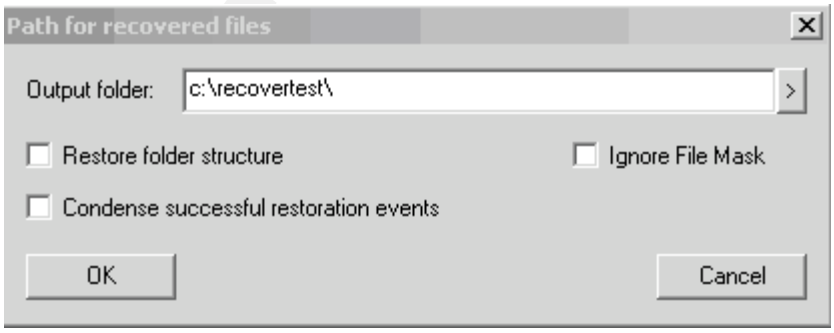


FIGURE C-8 (Restorer 2000 Pro Dialog Box)

Appendix D
Test 2 Figures

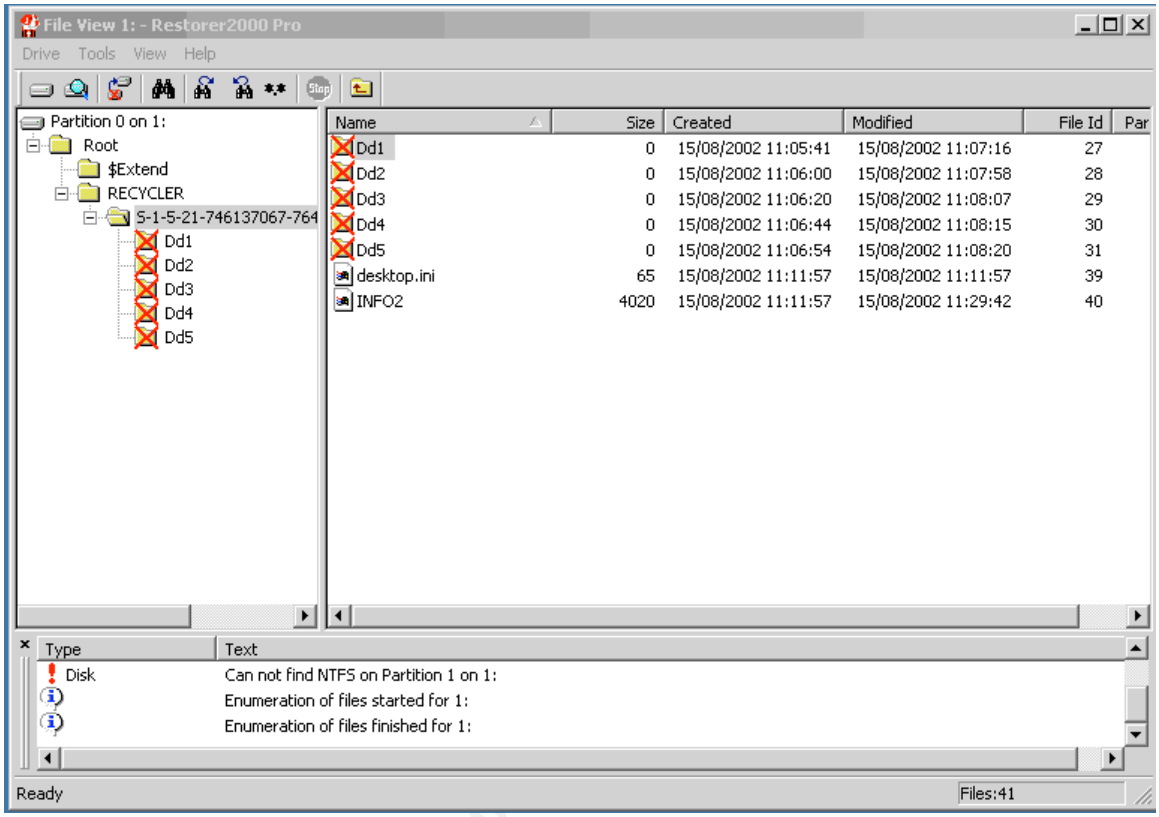


FIGURE D-1 (Files are seen from a drive with partitions removed.)

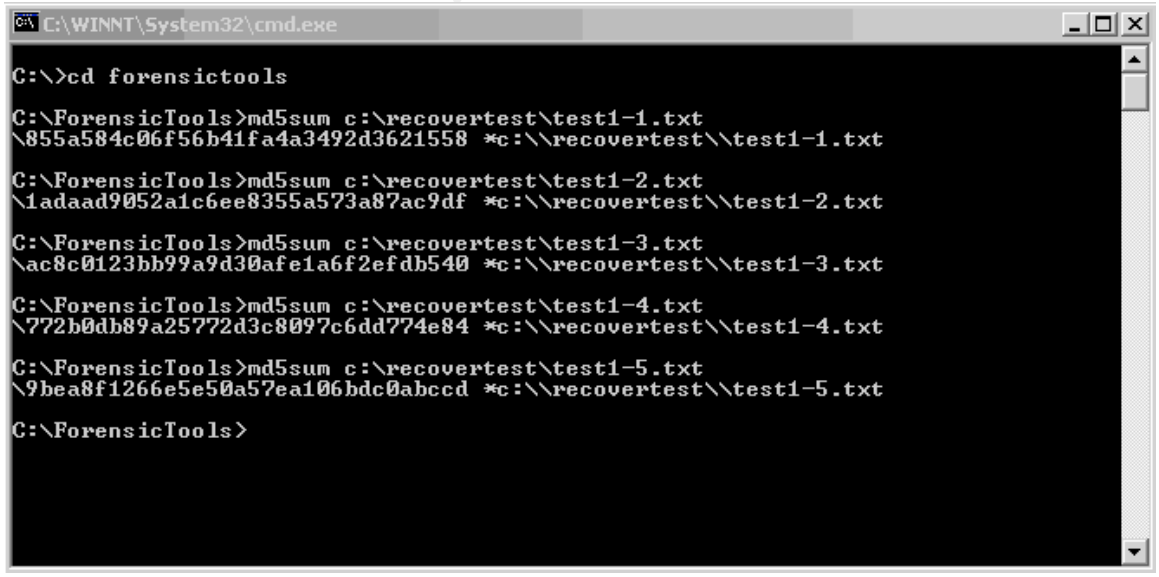


FIGURE D-2 (MD5 Hash Values from files recovered.)

Appendix E
Test 3 Figures

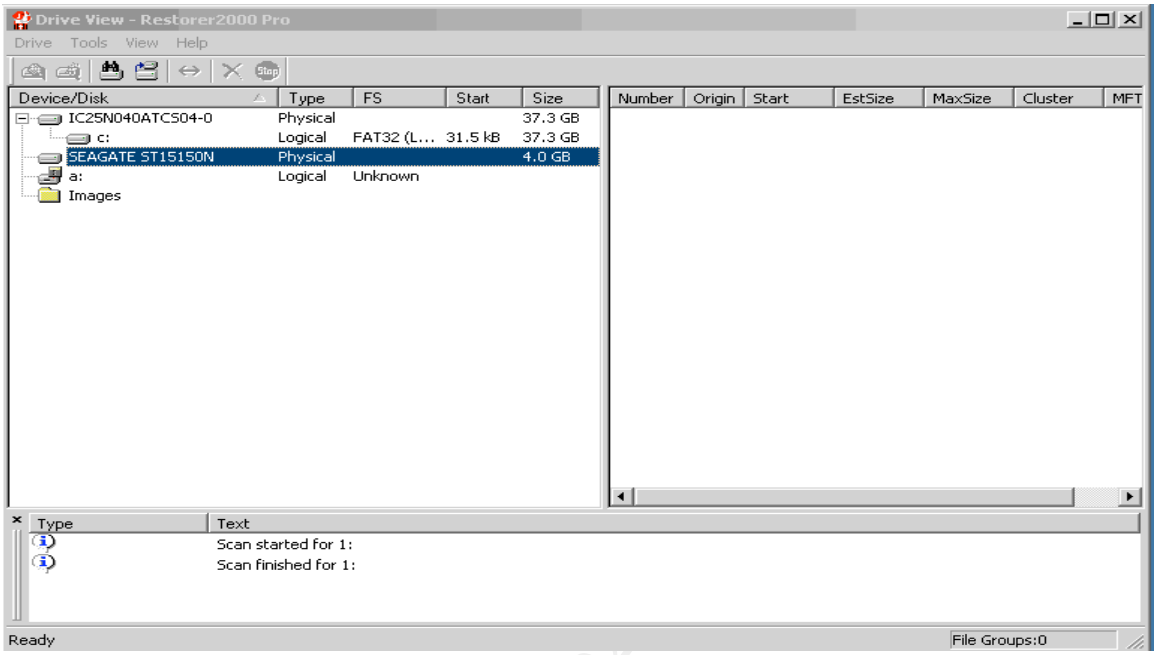


FIGURE E-1
(Shows there are no files on the drive. Note the SCSI drive is selected and the scan shows started and completed in the bottom window. If files were found, they would appear in the right window.)

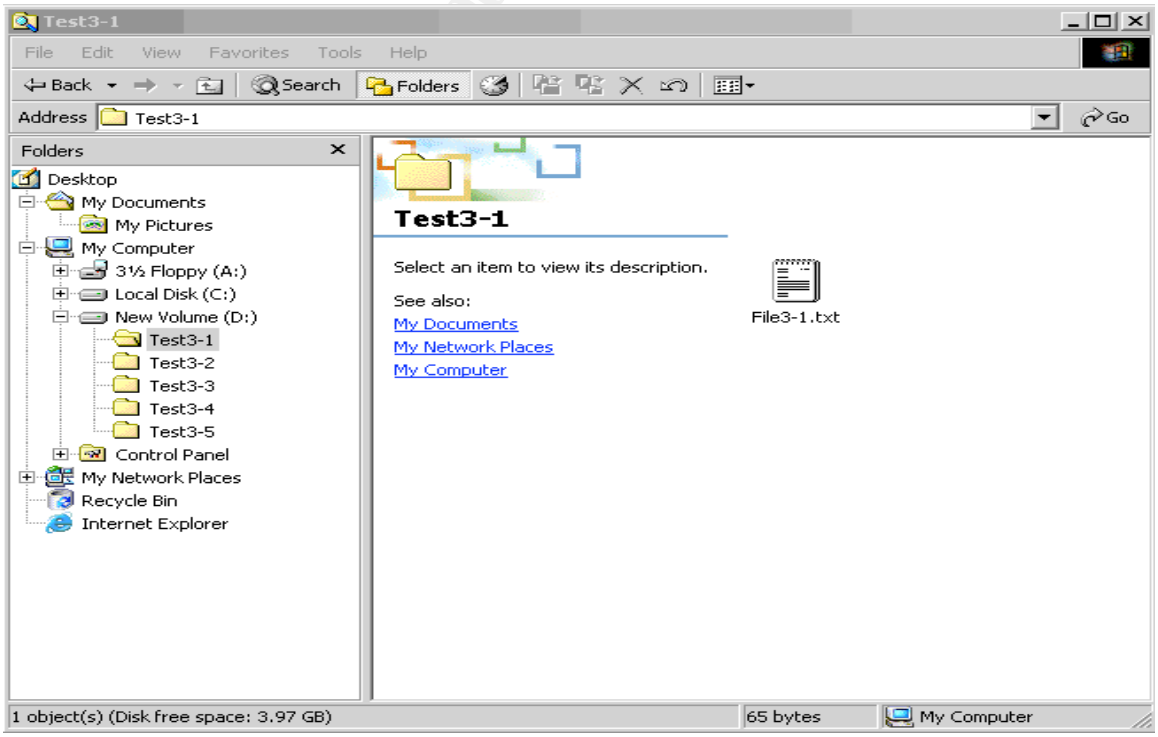


FIGURE E-2 (Test File in Place.)

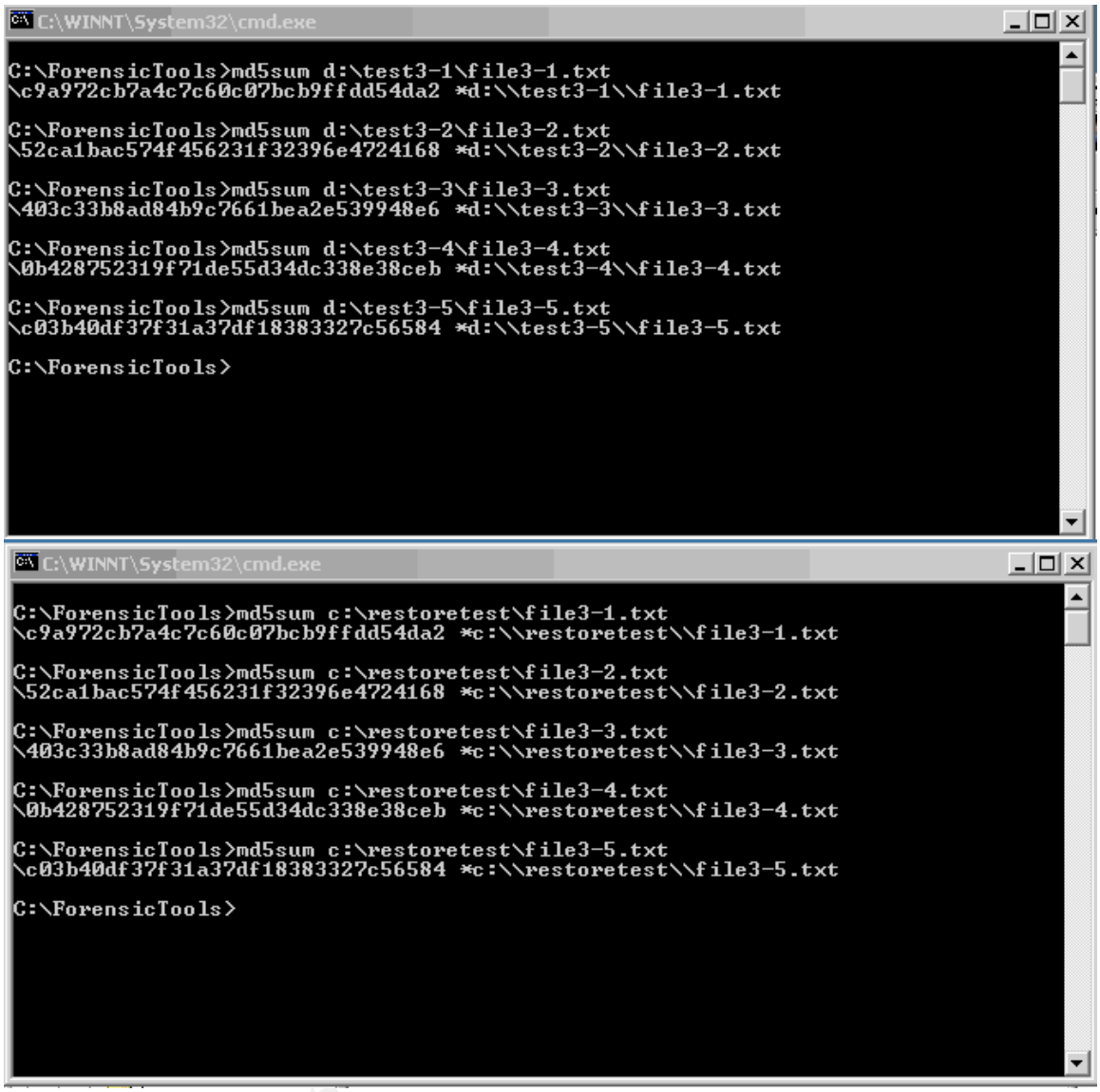


FIGURE E-3 (MD5 Hash Values on the Test Files [Top] and the recovered files [Bottom])

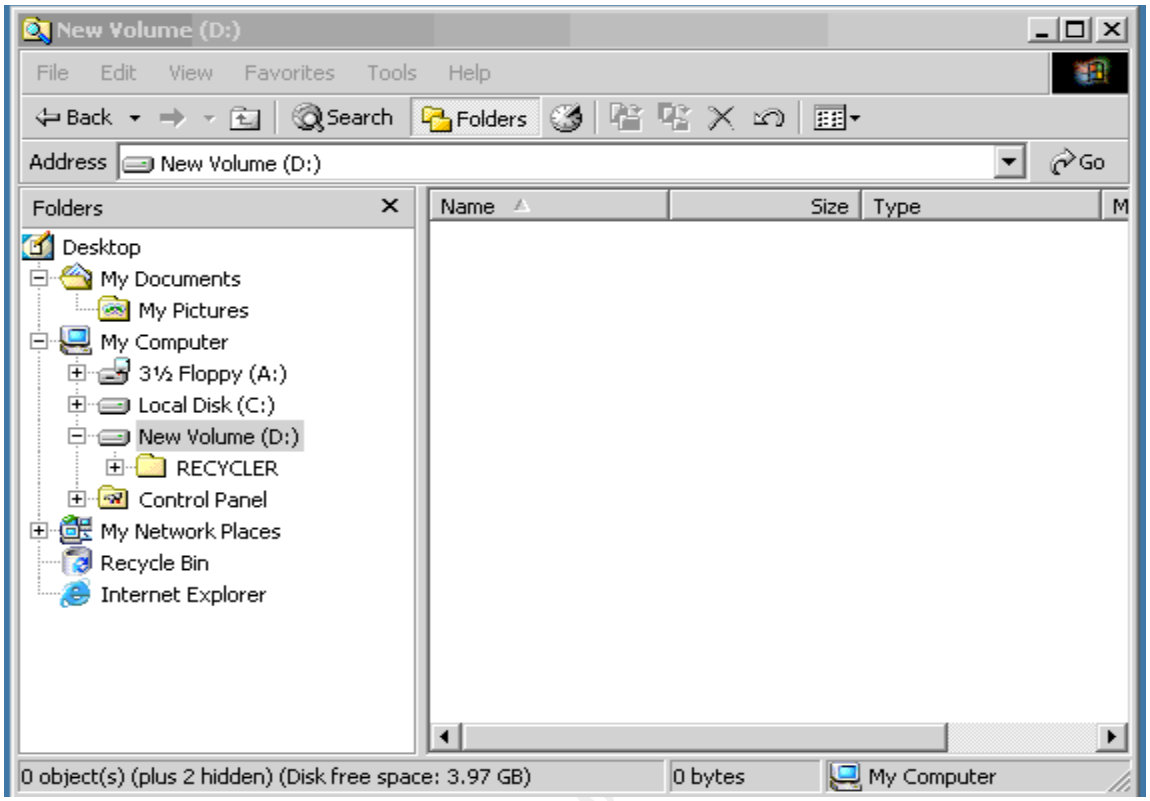


FIGURE E-4 (Shows no files accessible by Windows 2000 on the Drive.)

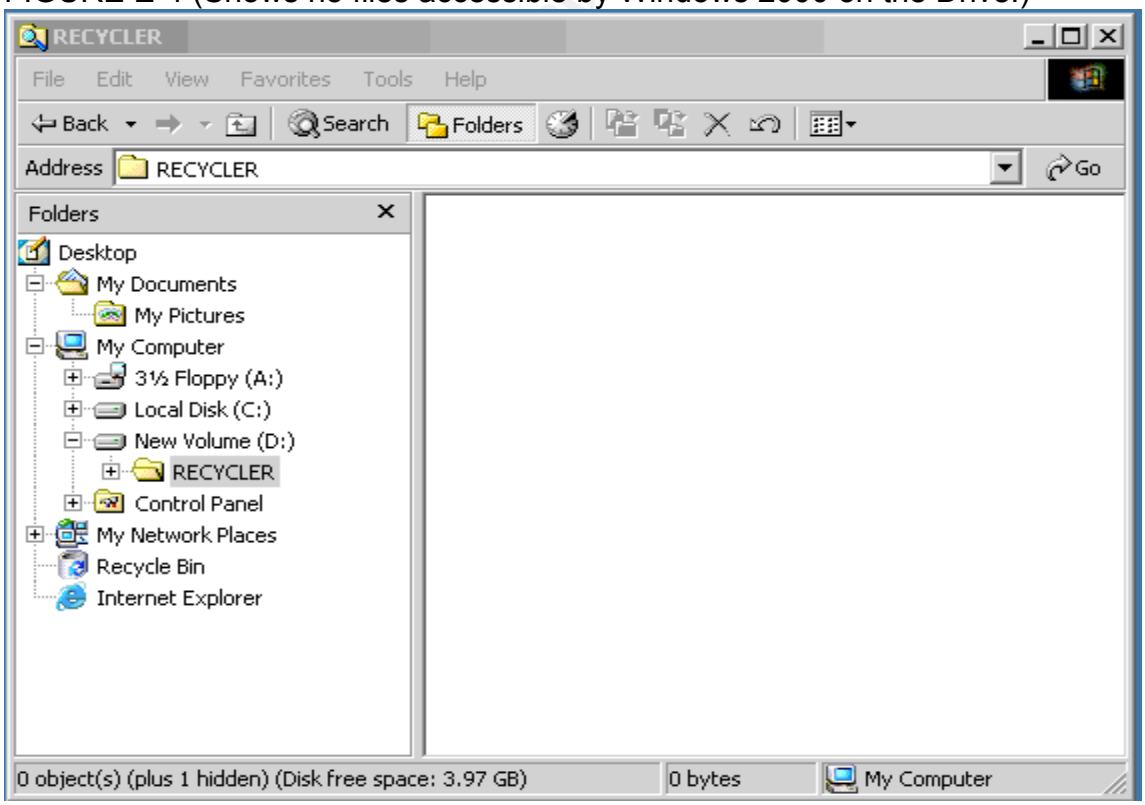


FIGURE E-5 (Shows all files deleted and removed from Recycle Bin.)

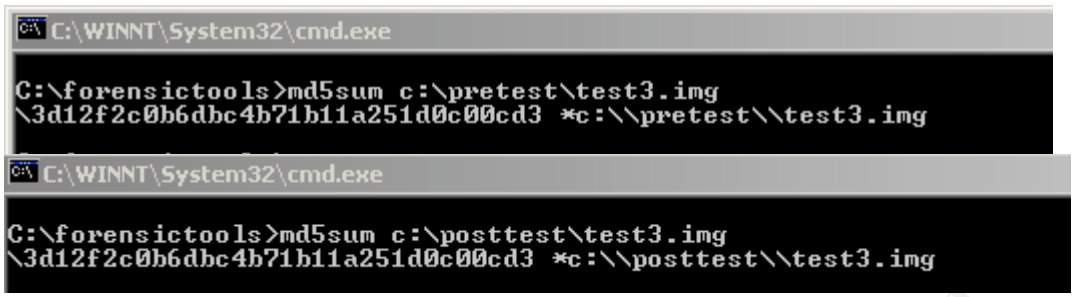


FIGURE E-6 (MD5 Hash Values from pretest and post-test drive images)

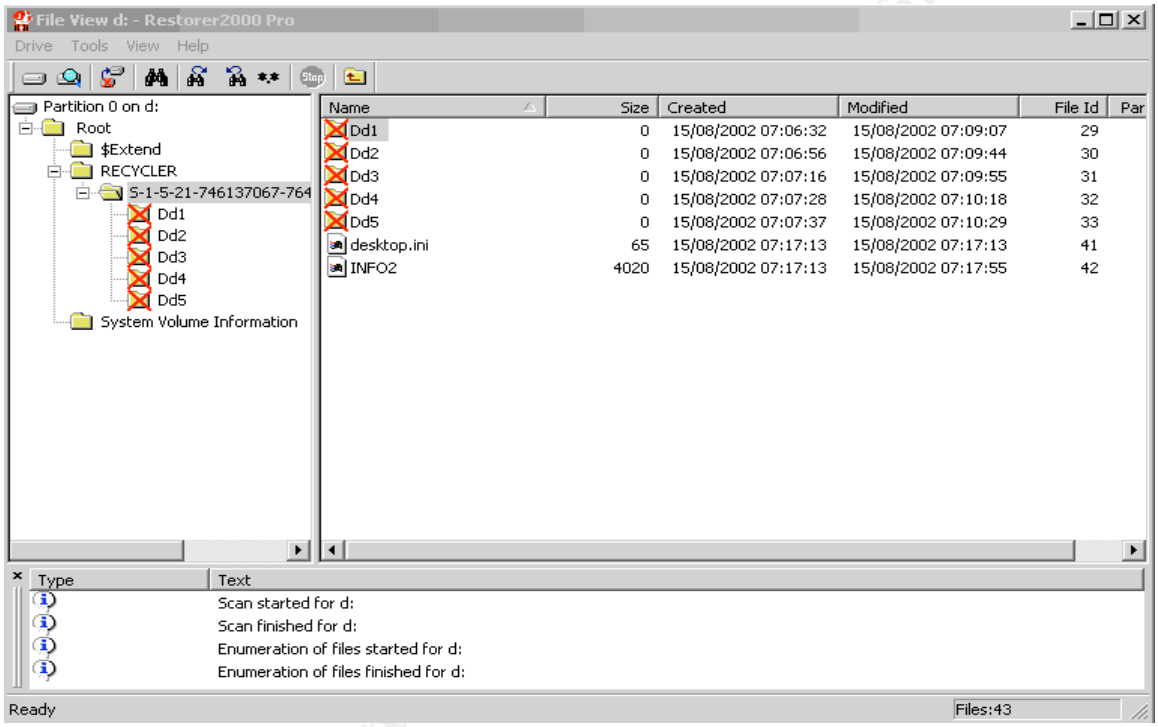


FIGURE E-7 (Shows Directories that have been deleted from the Recycle Bin.)

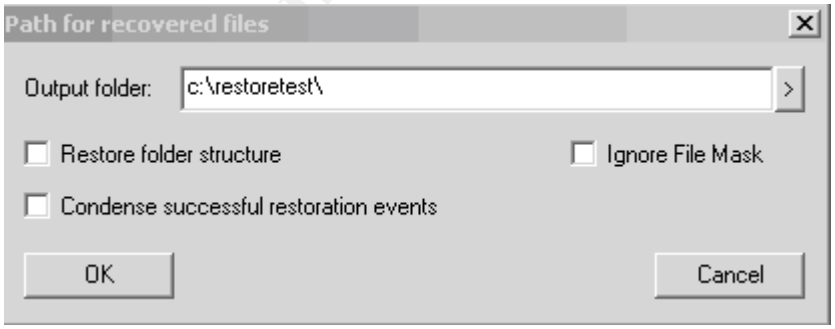


FIGURE E-8 (Restore Dialog Box)

Appendix F
Test 4 Figures

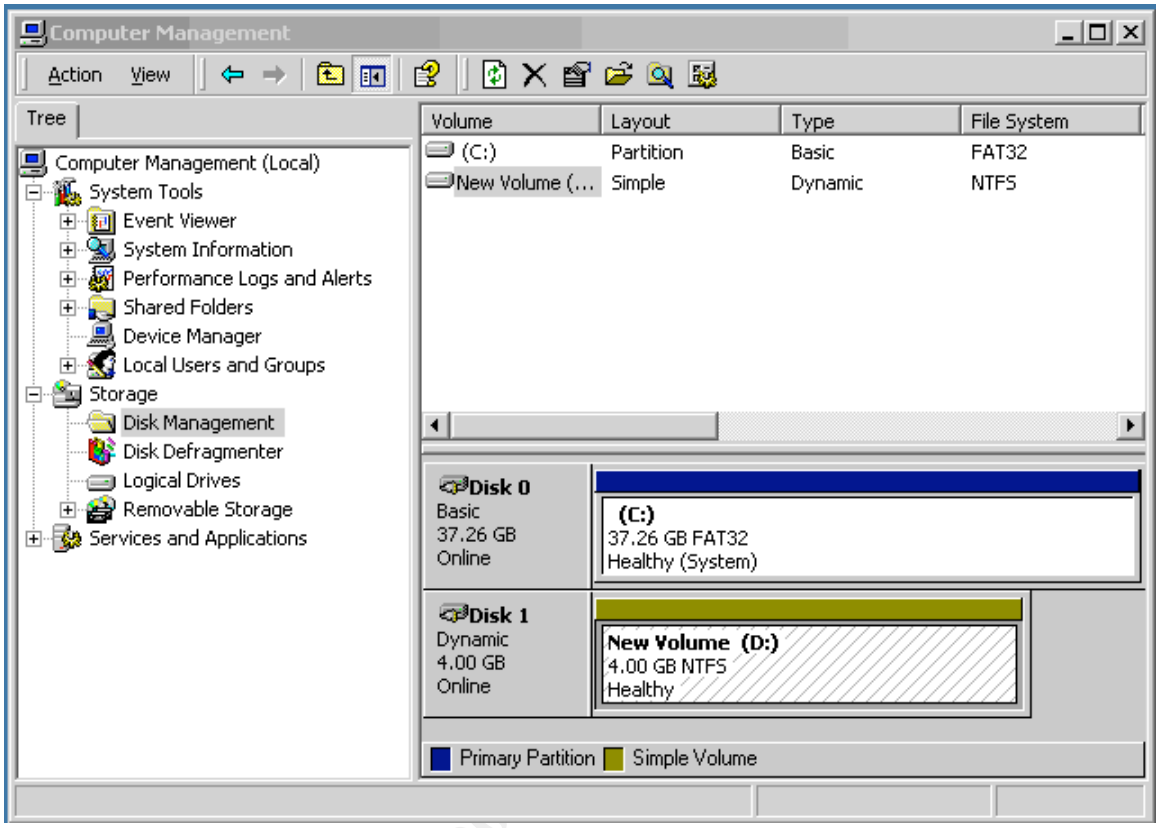


FIGURE F-1 (Original NTFS Partition.)

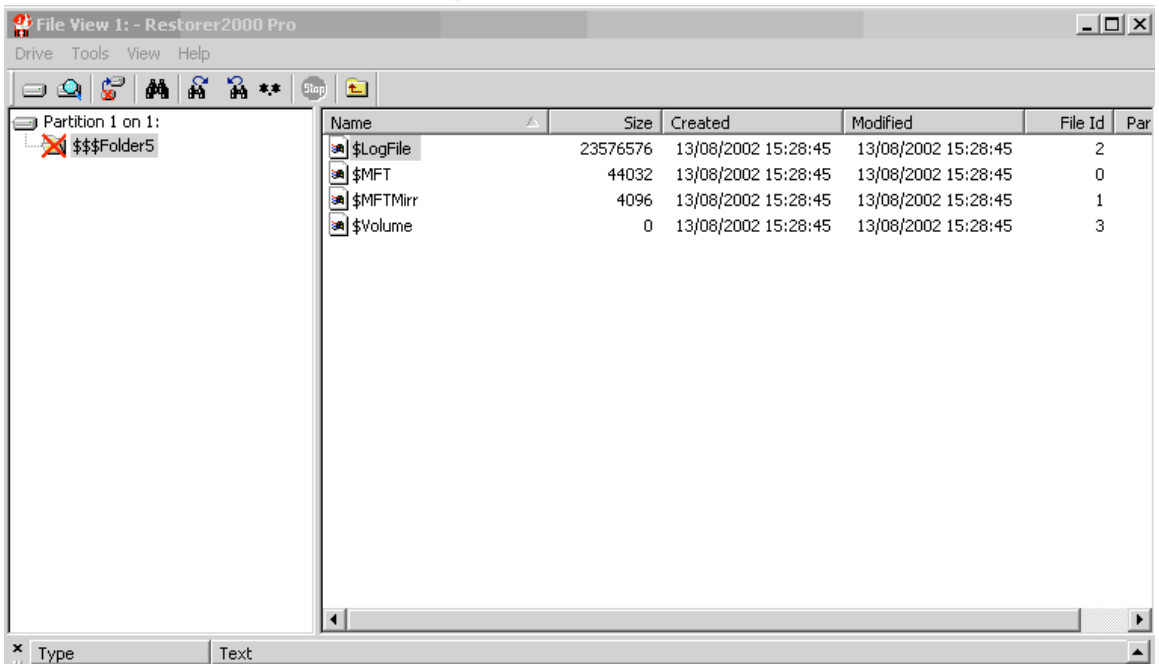


FIGURE F-2 (Files accessed by Restorer 2000 Pro)

Appendix G
Test 5 Figures

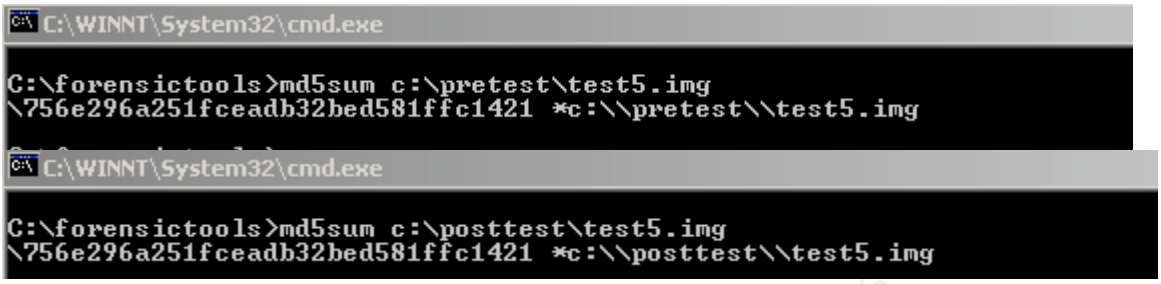


FIGURE G-1 (MD5 Hash Values of Drive Image Pretest and Post-test)

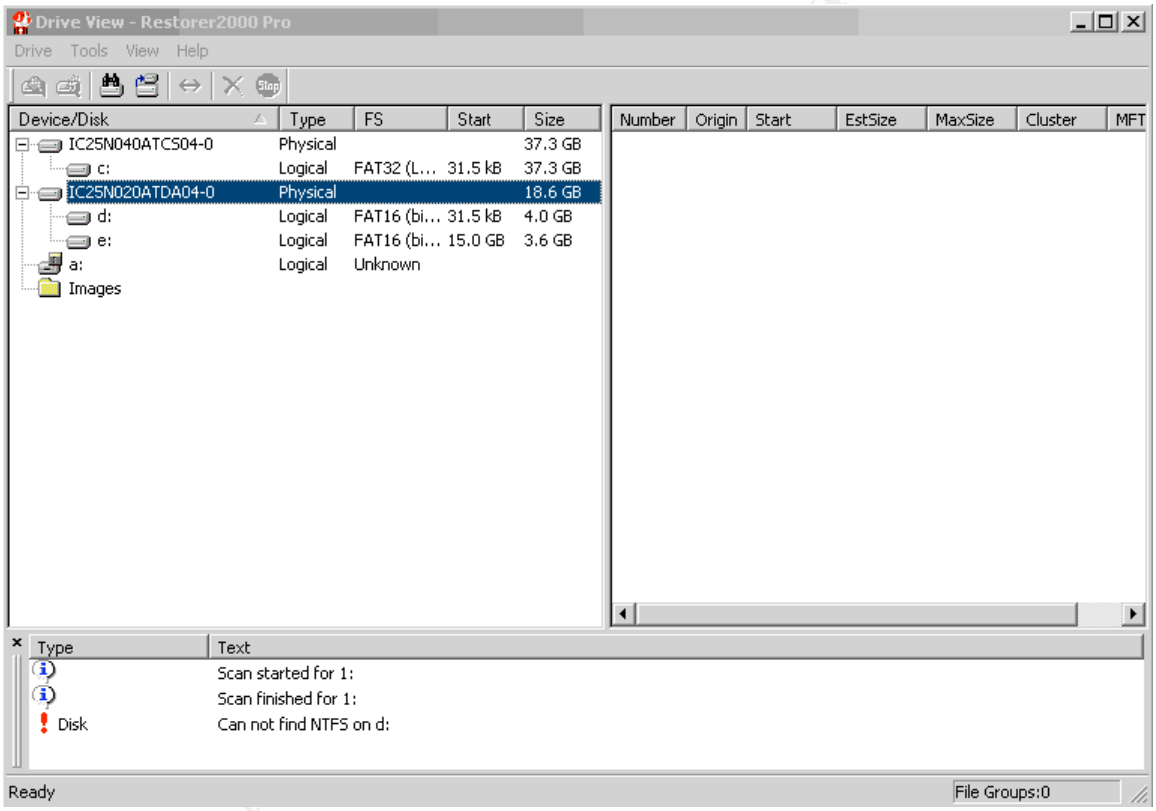


FIGURE G-2 (Non-NTFS Partitions do not show up.)

Appendix H
Test 6 Figures

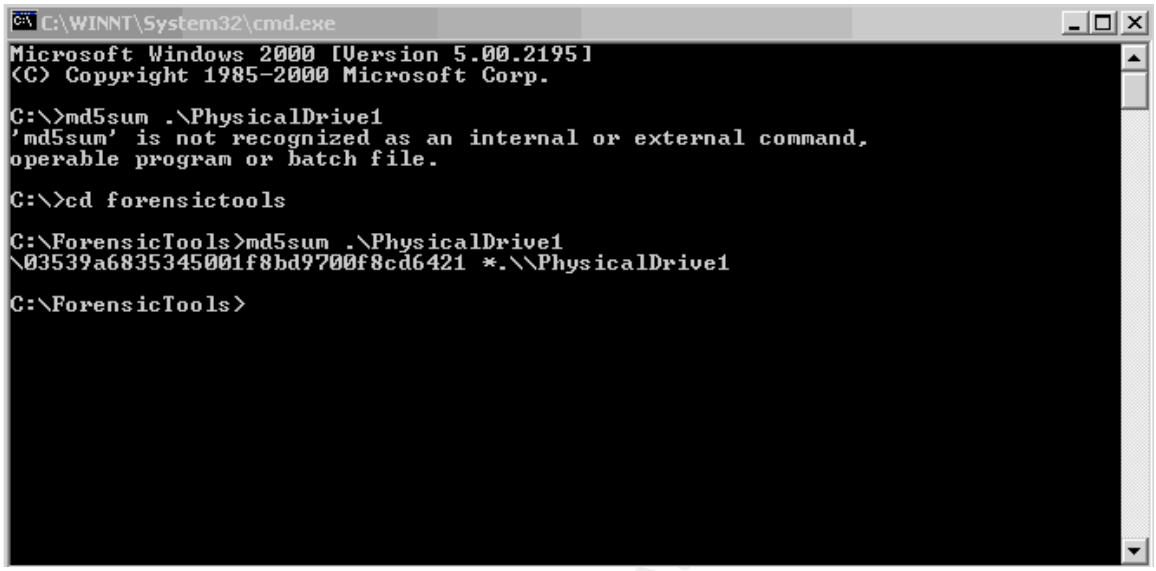


FIGURE H-1 (MD5 Hash Values for the Test Physical Drive)

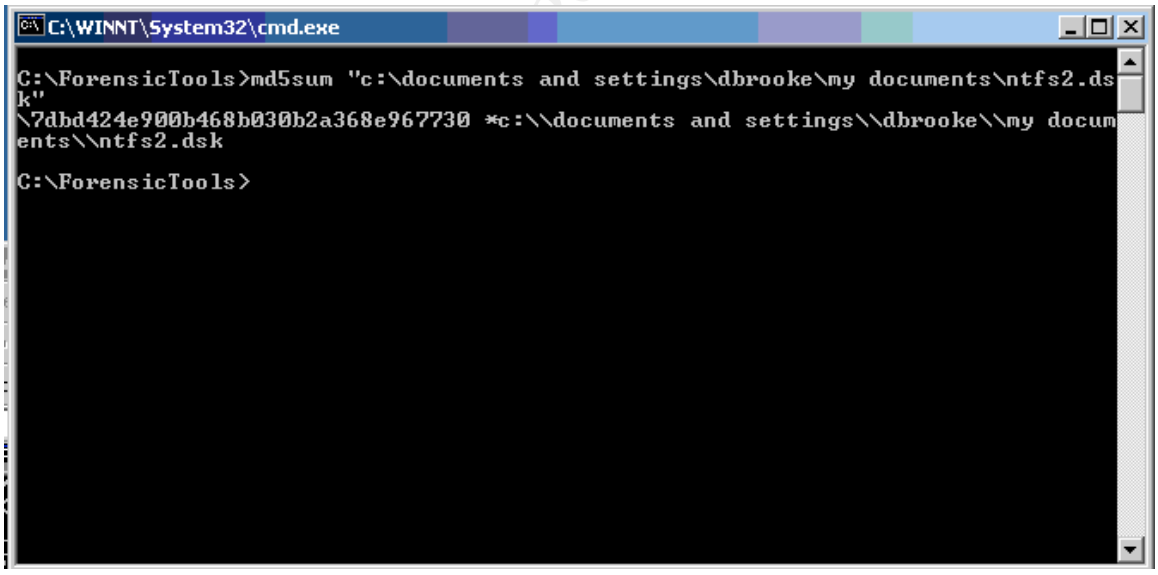


FIGURE H-2 (MD5 Hash Value for the Test Drive Physical Drive Image)

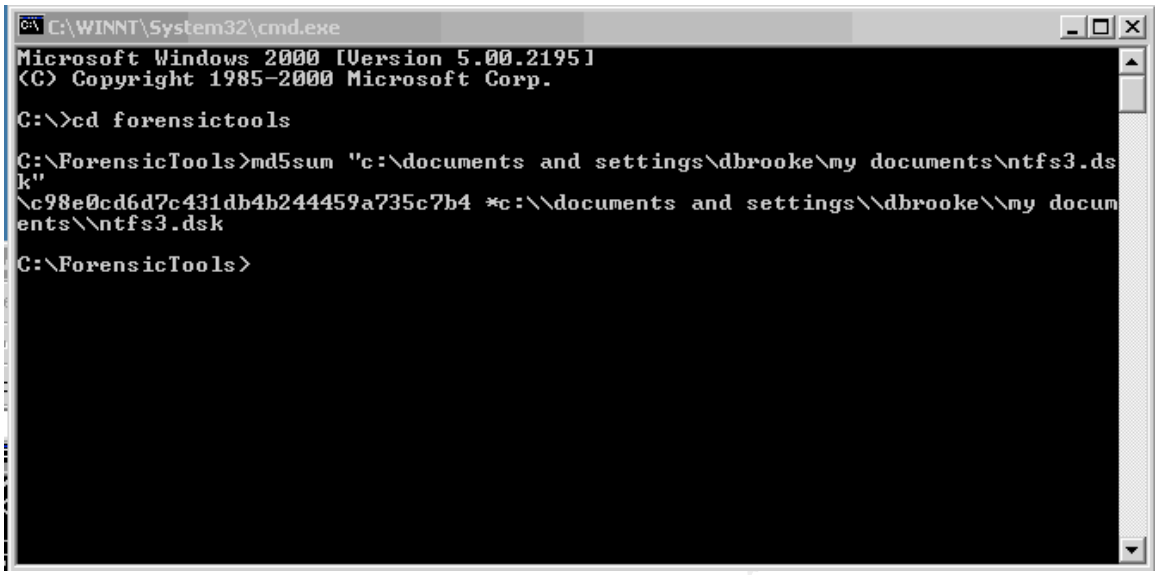


FIGURE H-3 (MD5 Hash Value of the Logical D Drive Image of the Test Drive)

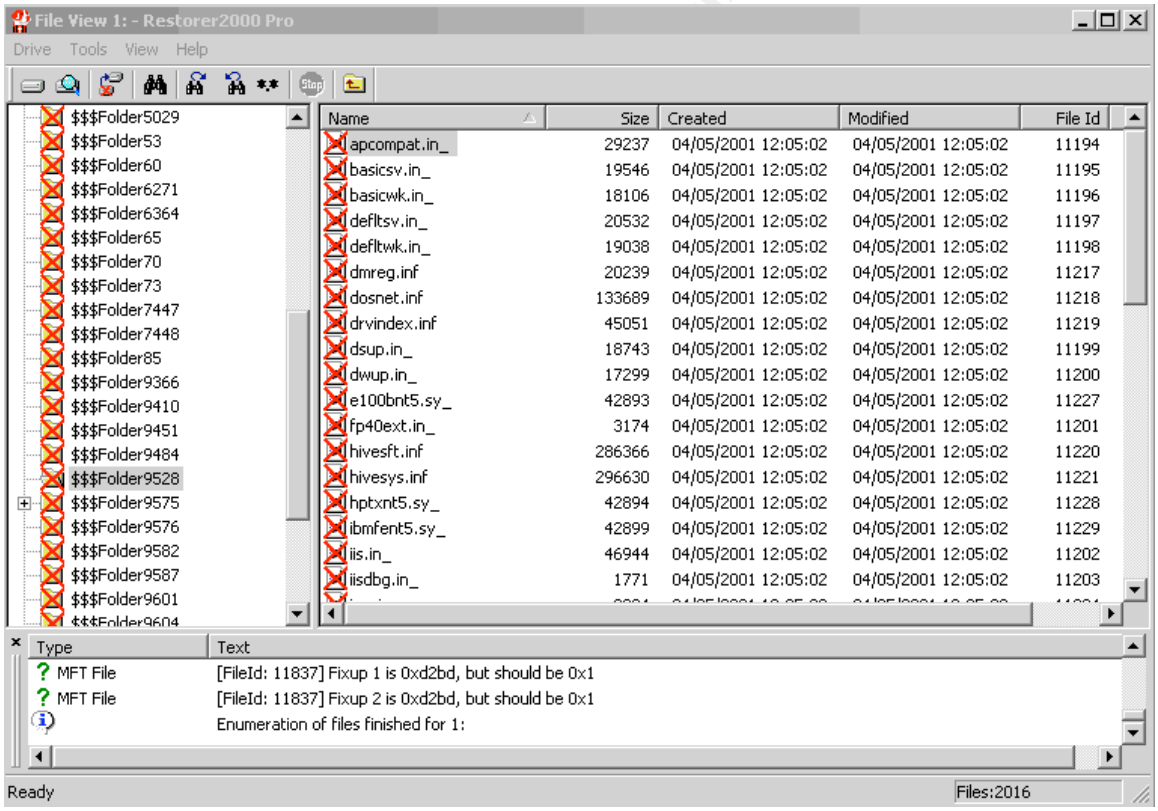


FIGURE H-4 (Shows files can be recovered from an image file.)

Appendix I
Test 7 Figures

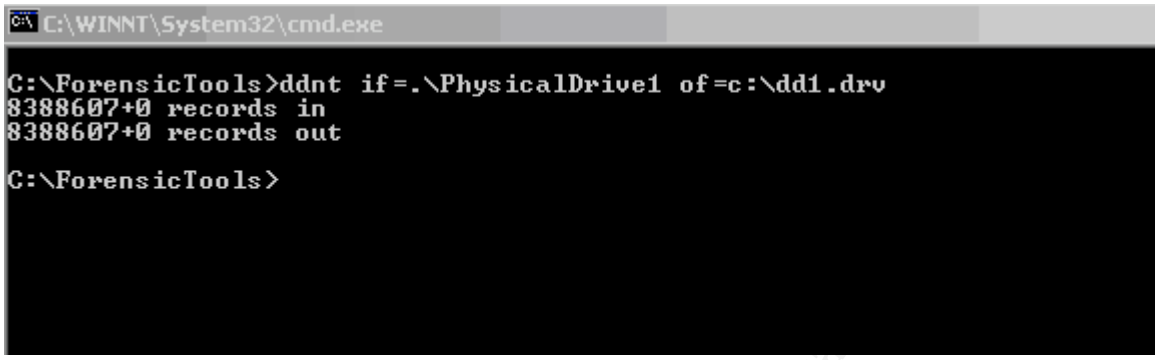


FIGURE I-1 (Using DD to get an image of the physical drive.)

FIGURE I-2 (Using DD to get an image of the volume.)

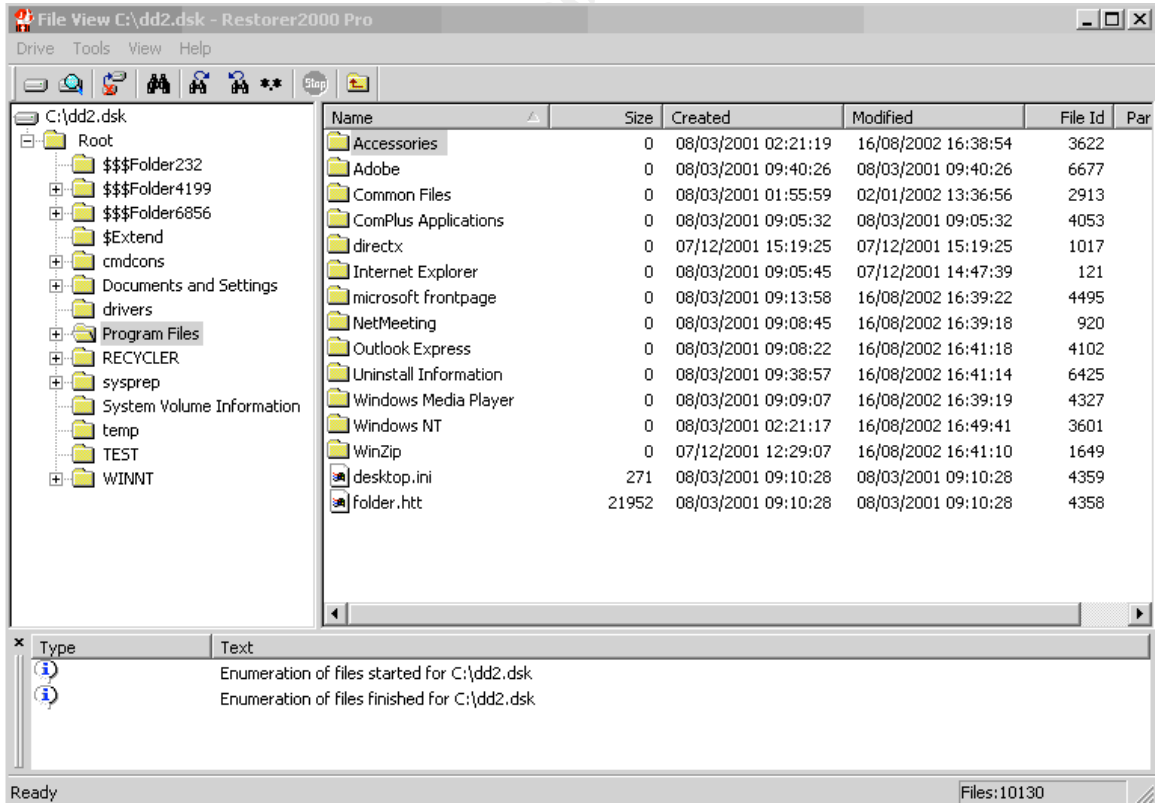


FIGURE I-3 (Restorer 2000 Pro Reads DD Image.)

Appendix J
Test 8 Figures

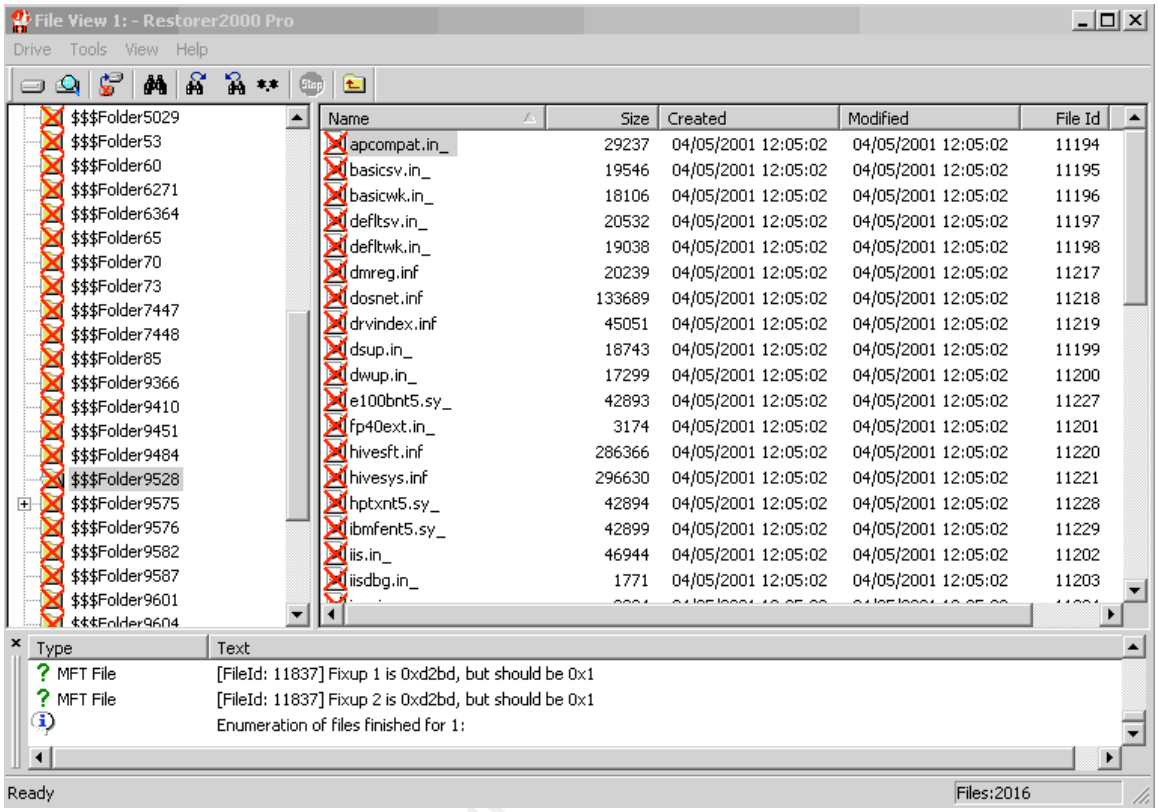


FIGURE J-1 (Files shown from NTFS partition through FAT32 partition.)

Appendix K
Test 9 Figures

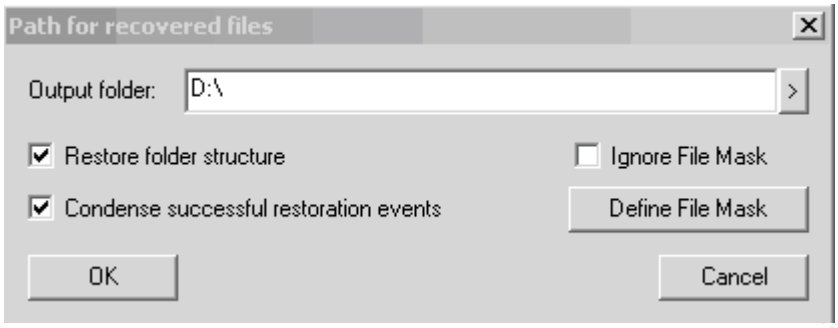


FIGURE K-1 (Recover Files – Note “Restore folder structure”)

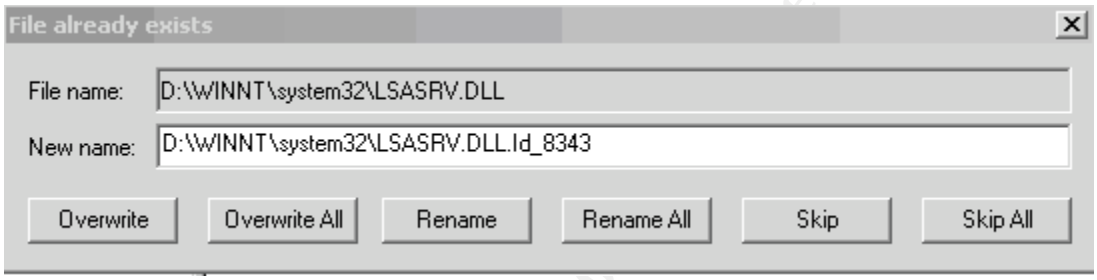


FIGURE K-2 (Files had to be overwritten.)

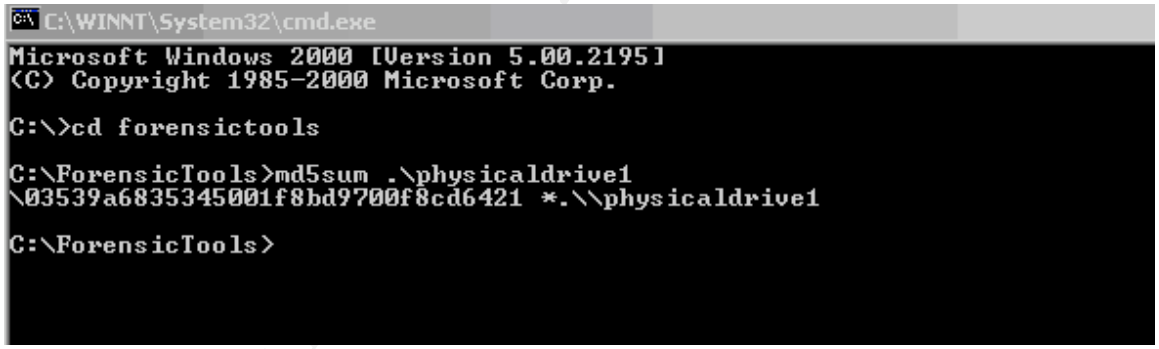


FIGURE K-3 (MD5 Hash Value from restored drive.)

Appendix L
Part 2 "Binary Analysis" Figures

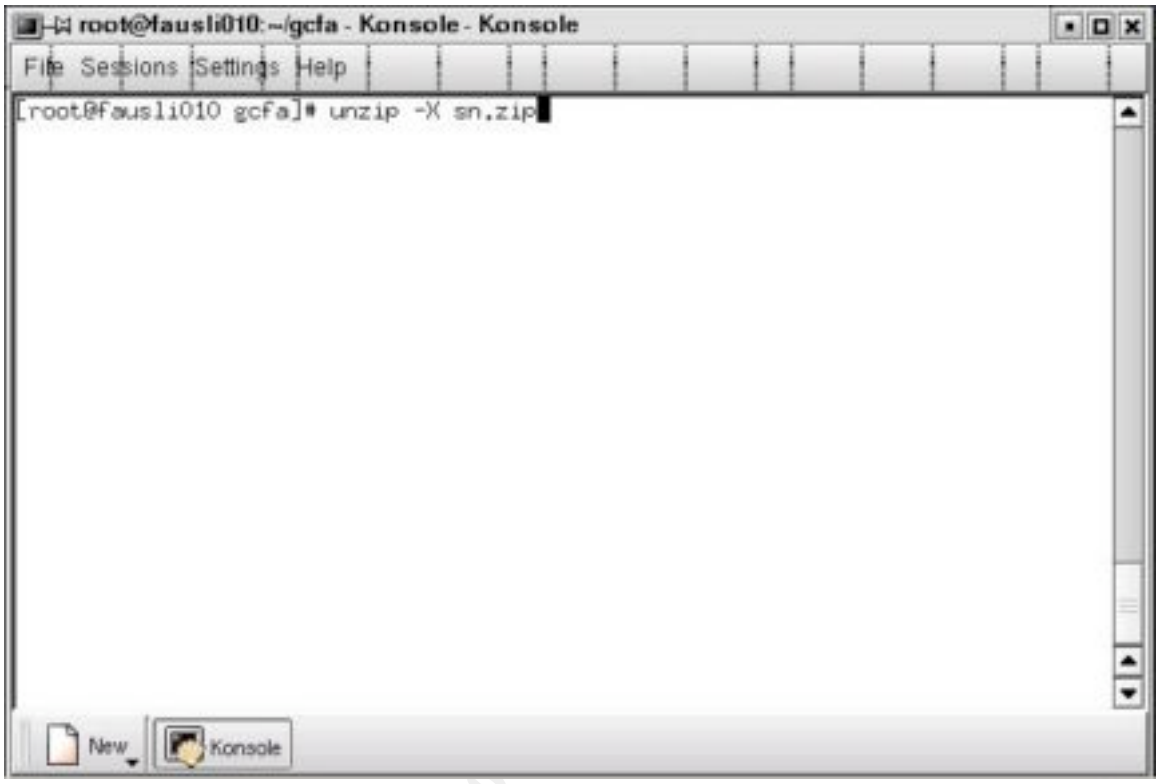


FIGURE L-1 (UNZIP Syntax)

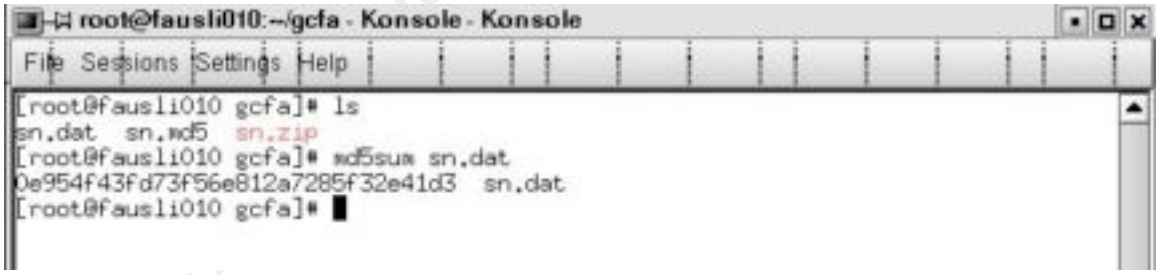


FIGURE L-2 (MD5 Hash Value of the Extracted File)

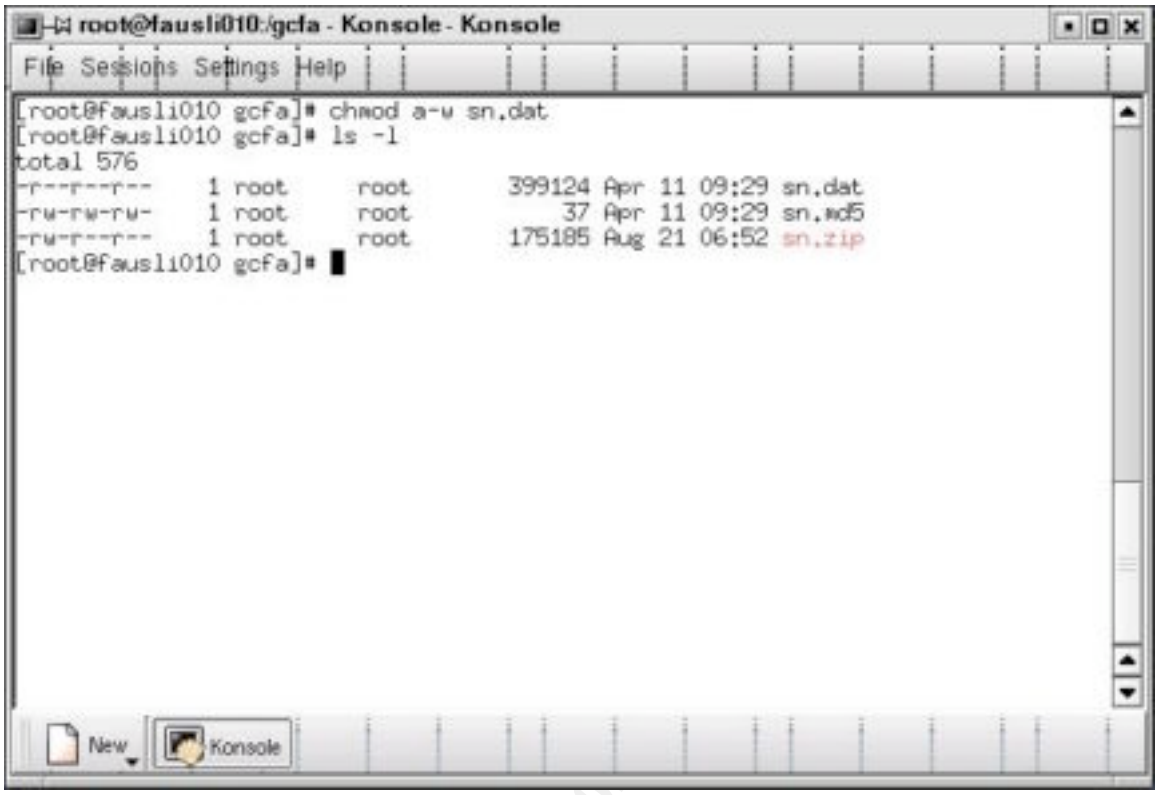


FIGURE L-3 (Making and confirming the file is Read Only)

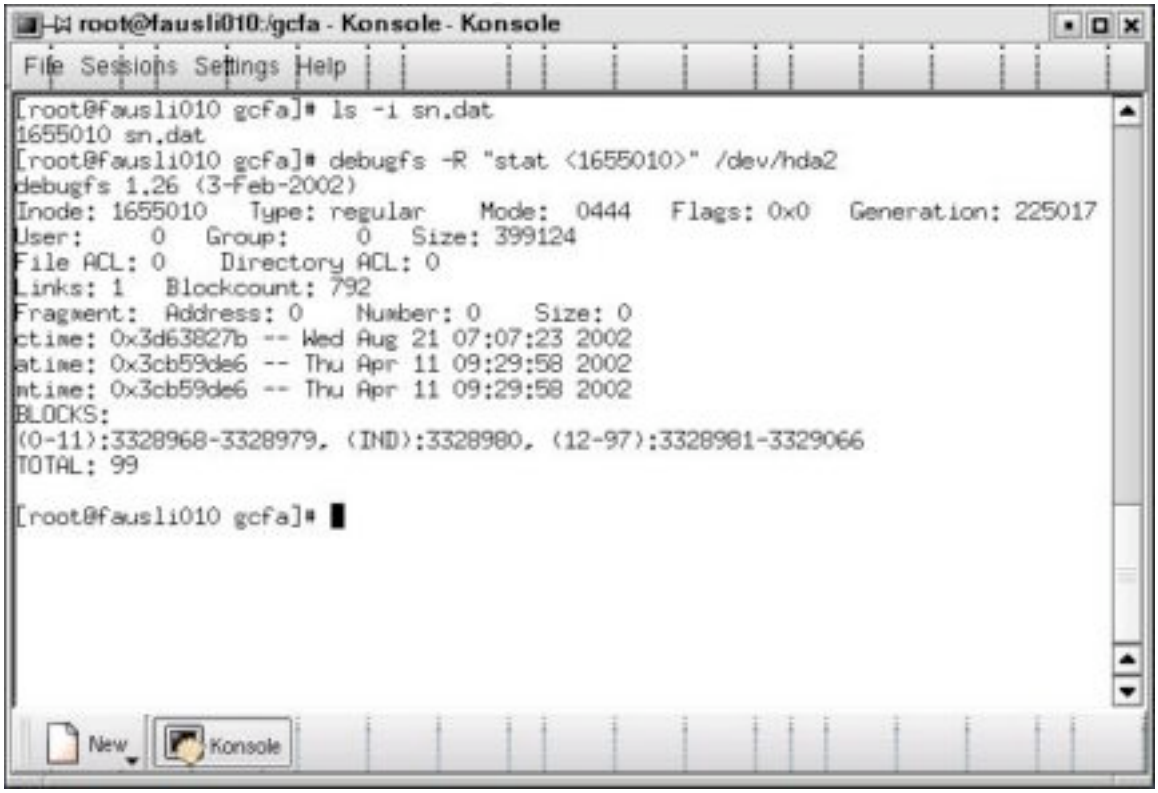


FIGURE L-4 (Determining MAC Information)

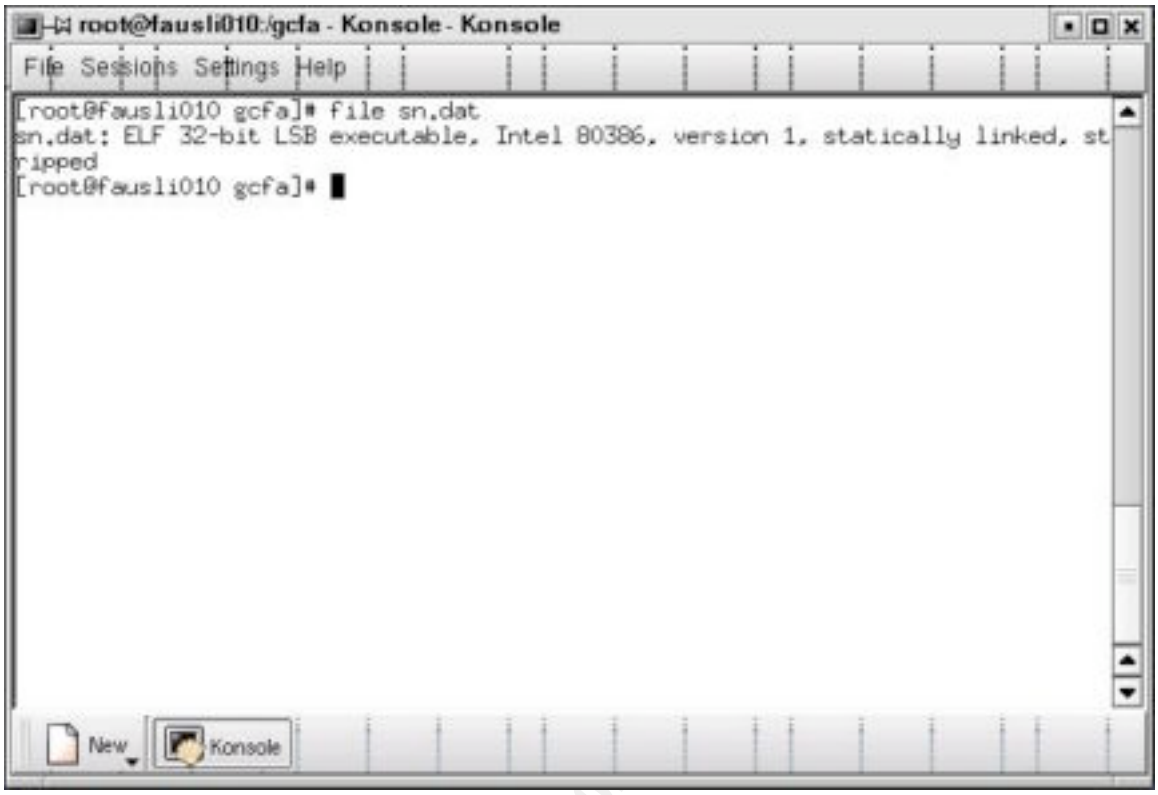


FIGURE L-5 (Results of File Command)

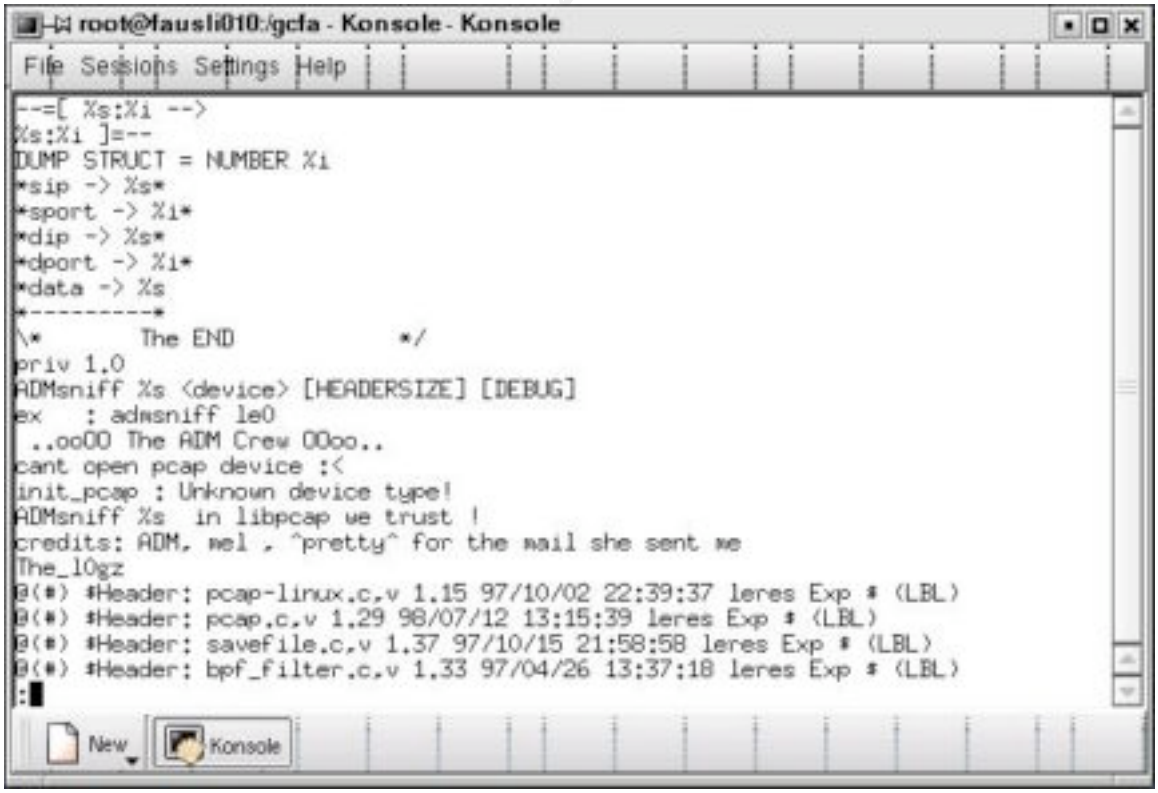


FIGURE L-6 (Partial Results from Strings Command)

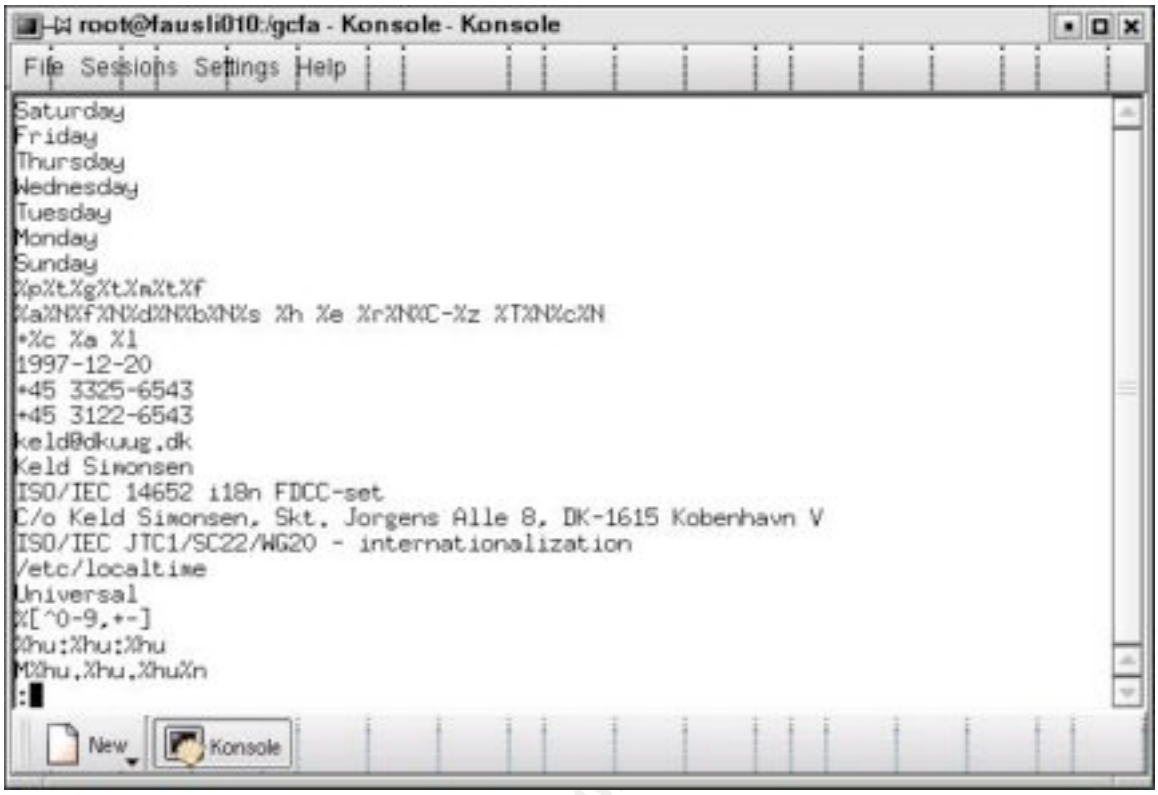


FIGURE L-7 (Partial Results from Strings Command)

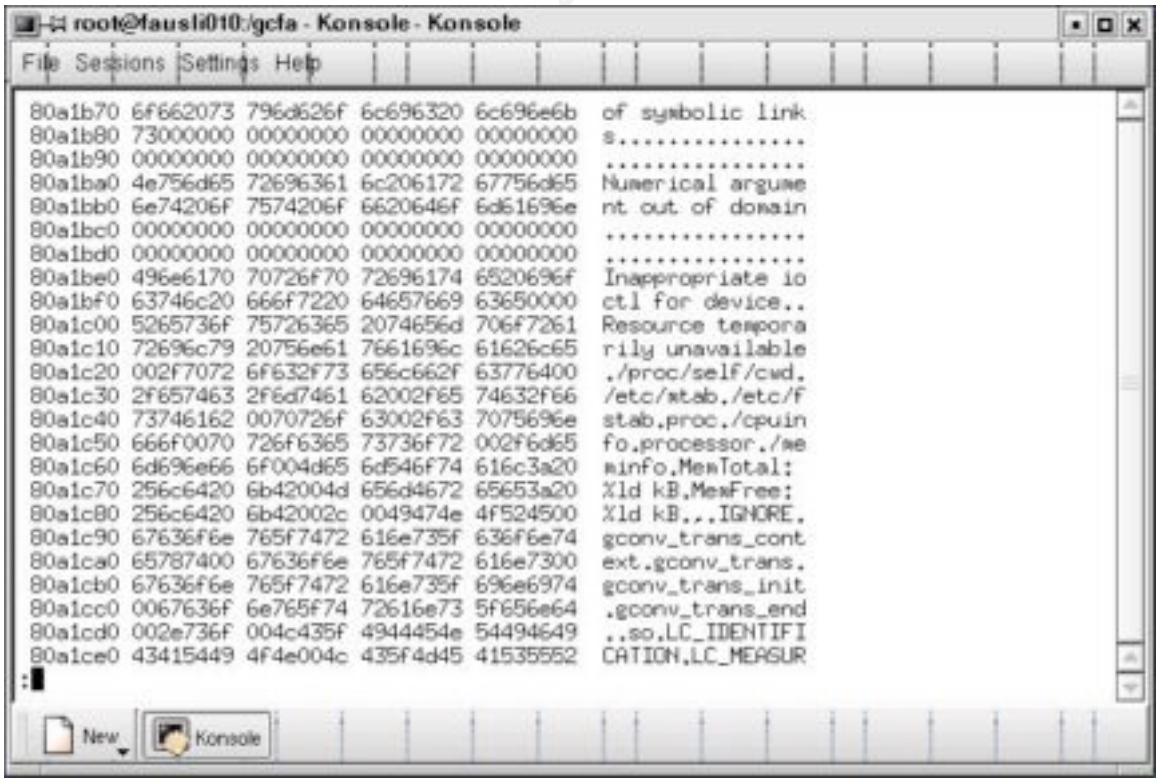


FIGURE L-8 (Partial objdump Results)

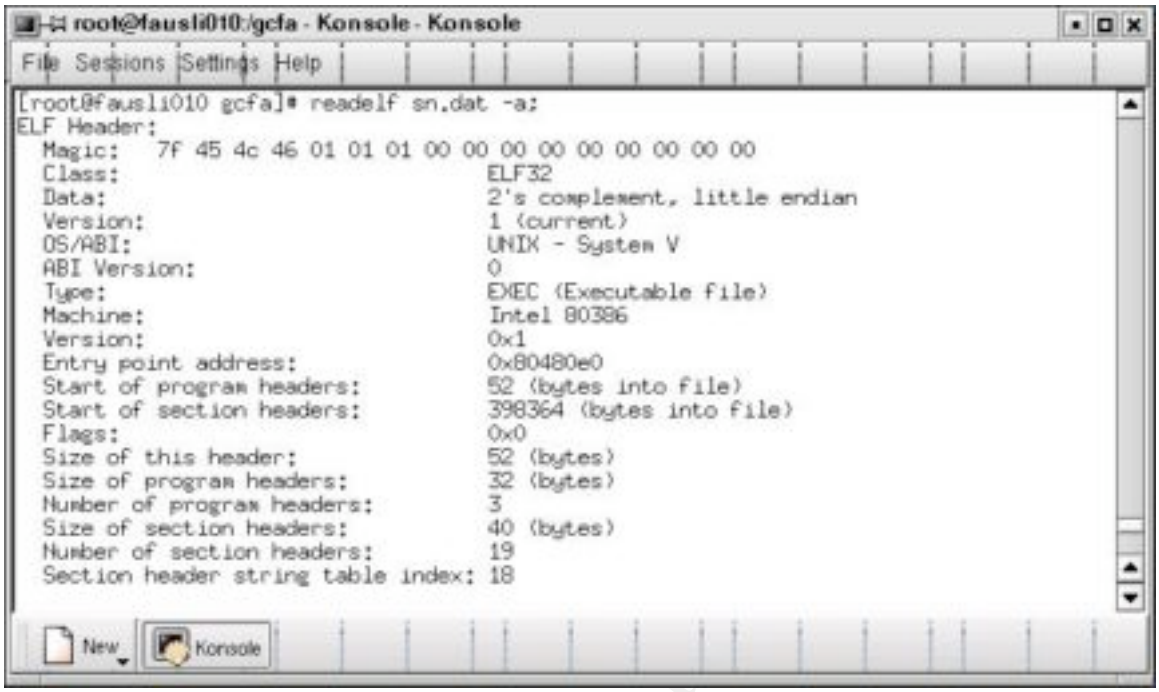


FIGURE L-9 (Partial ReadElf Results)

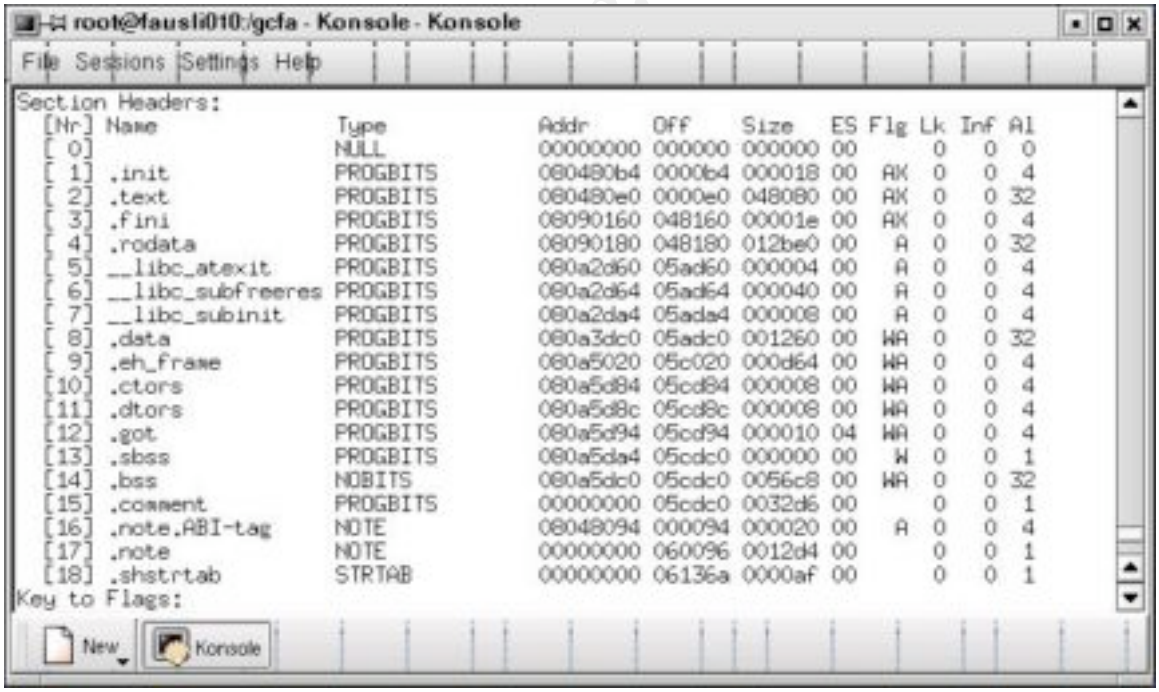


FIGURE L-10 (Partial ReadElf Results)

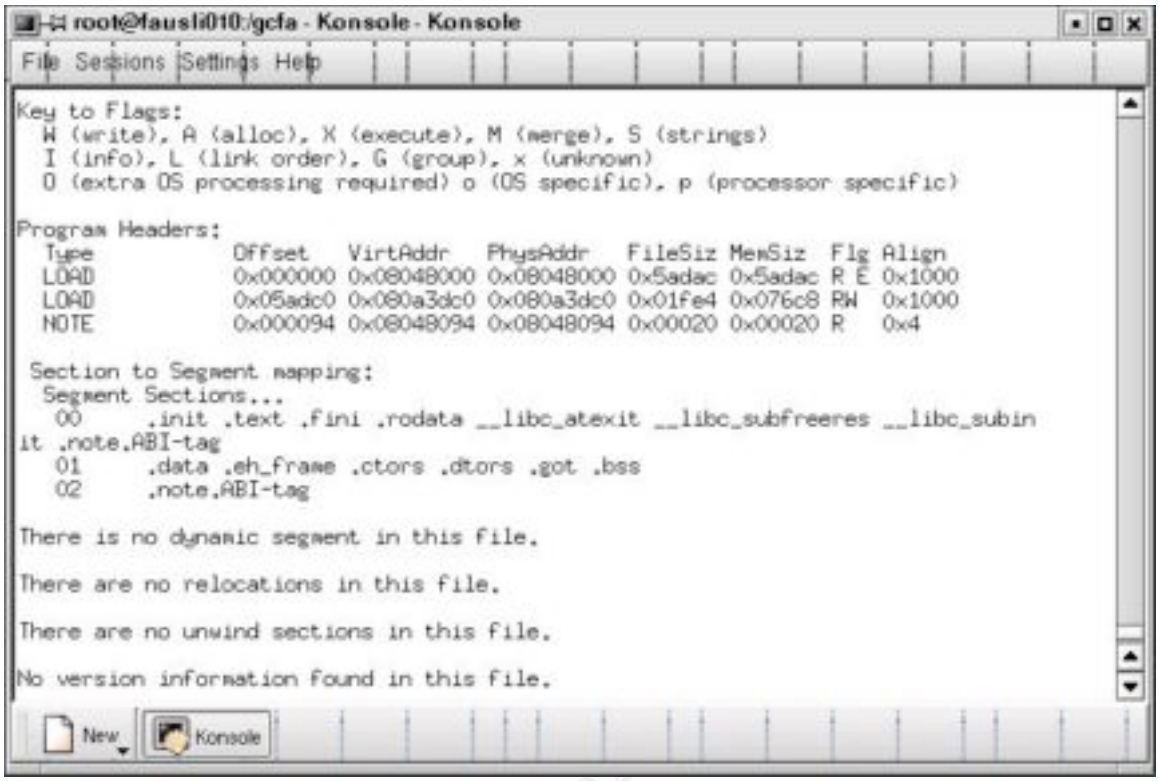


FIGURE L-11 (Partial ReadElf Results)

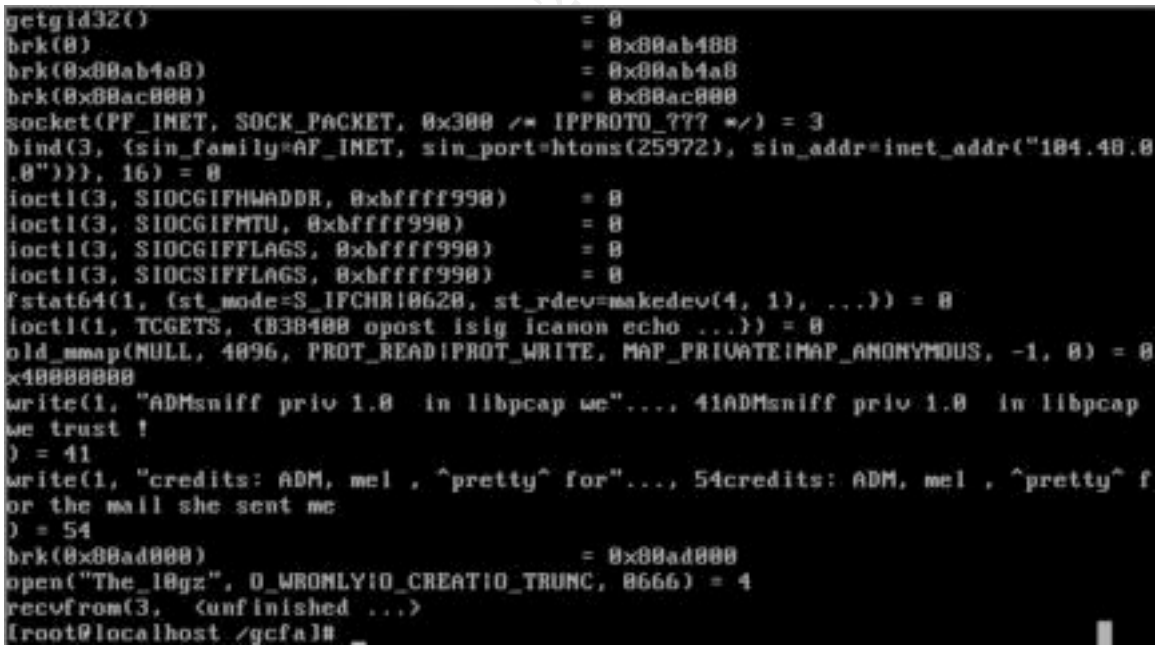


FIGURE L-12 (STRACE Results)

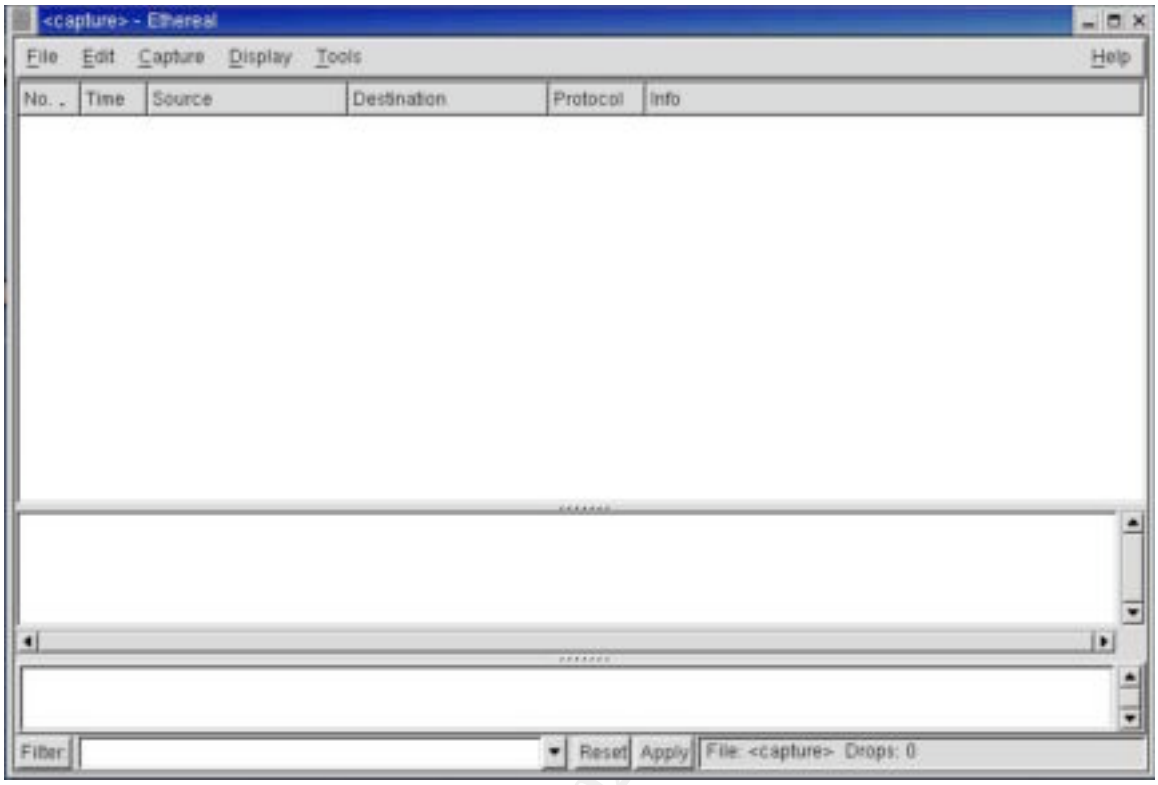


FIGURE L-13 (Ethereal Results showing No Traffic was being transmitted.)

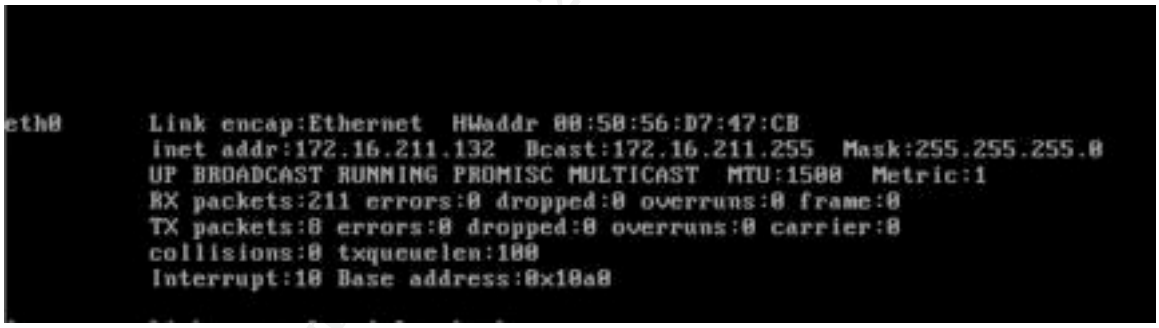


FIGURE L-14 (Shows the Ethernet Card has been placed in promiscuous mode.)

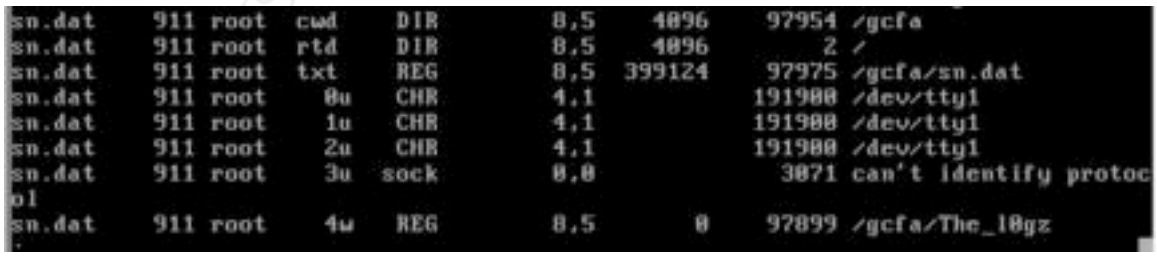


FIGURE L-15 (Open files by the "SN.DAT" program.)

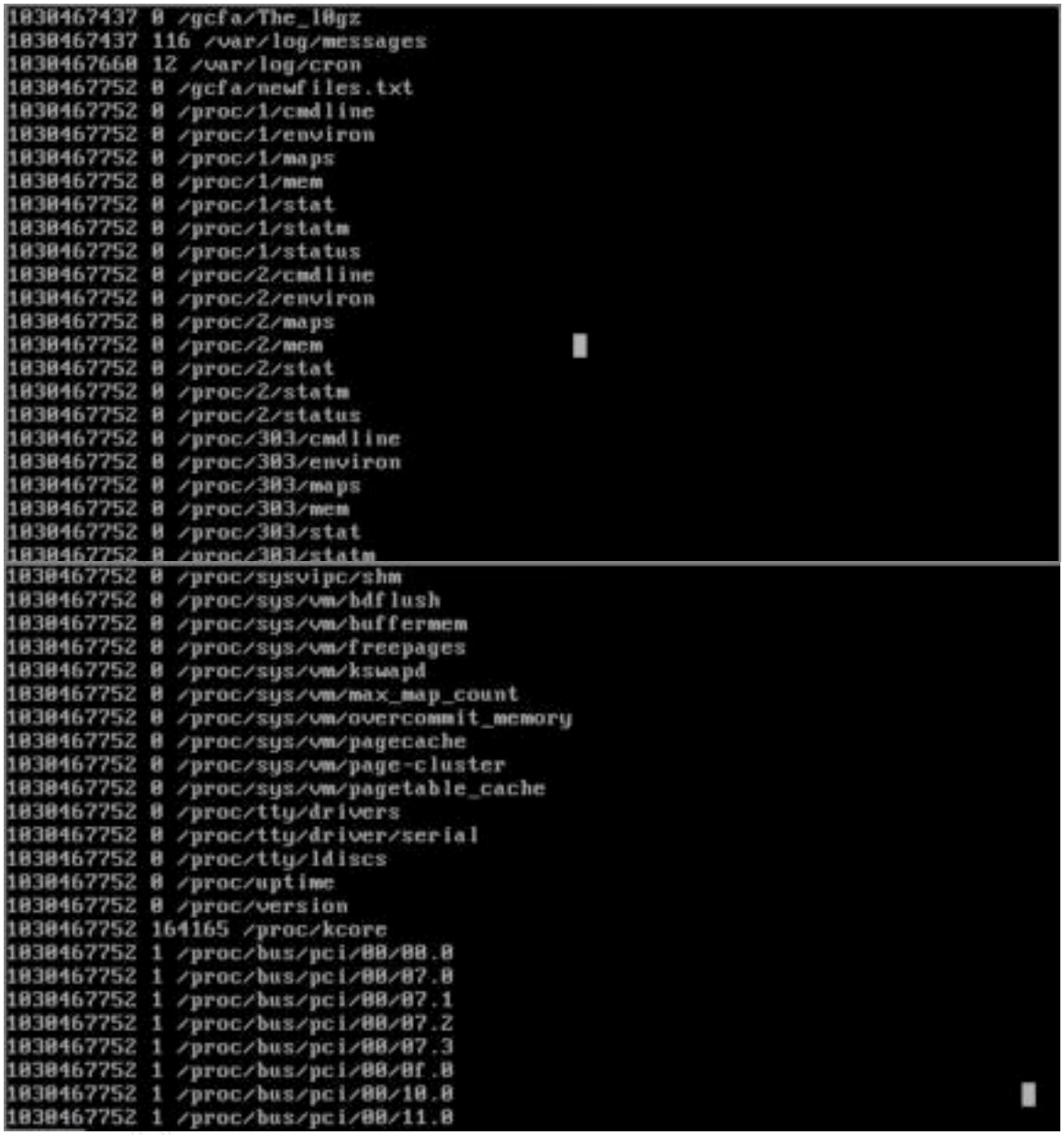
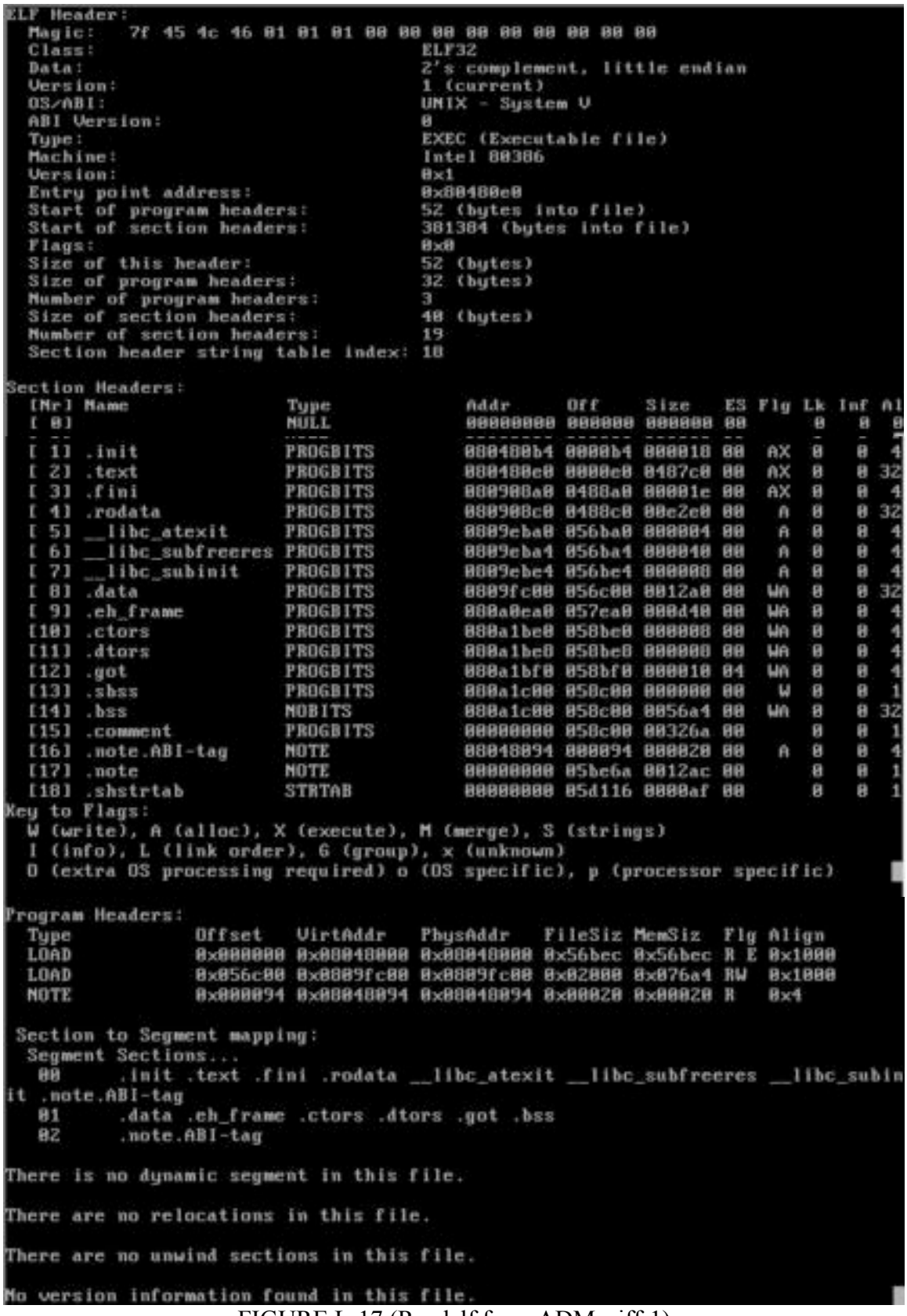


FIGURE L-16 (Partial list of files changed since “sn.dat”)



```
ELF Header:
  Magic:   7f 45 4c 46 01 01 01 00 00 00 00 00 00 00 00 00
  Class:                             ELF32
  Data:                              2's complement, little endian
  Version:                           1 (current)
  OS/ABI:                            UNIX - System V
  ABI Version:                       0
  Type:                              EXEC (Executable file)
  Machine:                           Intel 80386
  Version:                           0x1
  Entry point address:               0x80480e0
  Start of program headers:          52 (bytes into file)
  Start of section headers:          381384 (bytes into file)
  Flags:                             0x0
  Size of this header:               52 (bytes)
  Size of program headers:           32 (bytes)
  Number of program headers:         3
  Size of section headers:           40 (bytes)
  Number of section headers:         19
  Section header string table index: 18

Section Headers:
  [Nr] Name              Type            Addr     Off    Size   ES Flg Lk Inf Al
  [ 0]                   NULL            00000000 000000 000000 00      0   0  0
  [ 1] .init             PROGBITS        080480b4 0000b4 000018 00  AX  0   0  4
  [ 2] .text             PROGBITS        080480e0 0000e0 0487c0 00  AX  0   0 32
  [ 3] .fini             PROGBITS        080900a0 0488a0 00001e 00  AX  0   0  4
  [ 4] .rodata           PROGBITS        080908c0 0488c0 00e2e0 00   A  0   0 32
  [ 5] __libc_atexit     PROGBITS        0809eba0 056ba0 000004 00   A  0   0  4
  [ 6] __libc_subfreeres PROGBITS        0809eba4 056ba4 000040 00   A  0   0  4
  [ 7] __libc_subinit    PROGBITS        0809ebe4 056be4 000008 00   A  0   0  4
  [ 8] .data             PROGBITS        0809fc00 056c00 0012a0 00  WA  0   0 32
  [ 9] .eh_frame         PROGBITS        080a0ea0 057ea0 000d40 00  WA  0   0  4
  [10] .ctors            PROGBITS        080a1be0 058be0 000008 00  WA  0   0  4
  [11] .dtors            PROGBITS        080a1be8 058be8 000008 00  WA  0   0  4
  [12] .got              PROGBITS        080a1bf0 058bf0 000010 04  WA  0   0  4
  [13] .sbss             PROGBITS        080a1c00 058c00 000008 00   W  0   0  1
  [14] .bss              NOBITS          080a1c00 058c00 0056a4 00  WA  0   0 32
  [15] .comment          PROGBITS        00000000 058c00 00326a 00      0   0  1
  [16] .note.ABI-tag     NOTE            08048094 000094 000020 00   A  0   0  4
  [17] .note             NOTE            00000000 05be6a 0012ac 00      0   0  1
  [18] .shstrtab         STRTAB          00000000 05d116 0000af 00      0   0  1
Key to Flags:
  W (write), A (alloc), X (execute), M (merge), S (strings)
  I (info), L (link order), G (group), x (unknown)
  O (extra OS processing required) o (OS specific), p (processor specific)

Program Headers:
  Type           Offset   VirtAddr   PhysAddr   FileSiz MemSiz  Flg Align
  LOAD           0x000000 0x08048000 0x08048000 0x56bec 0x56bec R E 0x1000
  LOAD           0x056c00 0x0809fc00 0x0809fc00 0x02000 0x076a4 RW  0x1000
  NOTE           0x000094 0x08048094 0x08048094 0x00020 0x00020 R   0x4

 Section to Segment mapping:
  Segment Sections...
   00     .init .text .fini .rodata __libc_atexit __libc_subfreeres __libc_subin
it .note.ABI-tag
   01     .data .eh_frame .ctors .dtors .got .bss
   02     .note.ABI-tag

There is no dynamic segment in this file.

There are no relocations in this file.

There are no unwind sections in this file.

No version information found in this file.
```

FIGURE L-17 (Readelf from ADMsniff-1)

Part 1 References

John B, maruti sunil, “how to hack a web site?”
URL: http://www.faqts.com/knowledge_base/view.phtml/aid/11815/fid/118, (20 August 2002)

Bitmart.Net “NTFS Data Recovery & Undelete Software for Windows 2000. NT. XP. Unformat Utility. “ 8 Mar 2002” URL: http://www.bitmart.net/r2k.htm (20 August 2002)

“NTFS Data Recovery & Undelete Software for Windows 2000. NT. XP. Unformat Utility.
URL: http://www.bitmart.net/r2kfull.htm (8 Aug 2002)

“Webopedia. The #1 Online Encyclopedia Dedicated to Computer Technology”
URL: http://www.webopedia.com/TERM/M/malware.html (9 Aug 2002)

Spera, Christopher, “Restorer 2000 Pro”, User to User Reviews
URL: http://www.wugnet.com/csreviews/software/Restorer2000Pro/ (15 Aug 2002)

DLL,
URL: http://www.webopedia.com/TERM/D/DLL.html, (20 August 2002)

MD5Sum
URL: http://www.etree.org/md5com.html, (20 August 2002)

KillDisk, Lsoft Technologies, Version 1.1
URL: http://www.lsoft.net, (20 August 2002)

DD,
URL: http://www.redhat.com/swr/i386/fileutils-4.1-10.i386.html, (20 August 2002)

Restorer 2000 Pro Help File, “Using Regions”

SANS Institute, Track 8 – System Forensics, Investigation, and Response, 8.1 Investigative Incident Response and Basic Forensic Windows Principles – Hands On, Sans Institute, 2002

NTI, “Trial Illustration Posters”
URL: http://www.forensics-intl.com/trialill.html, (28 August 2002)

Part 2 References

Phrack Magazine, “Interface Promiscuity Obscurity”, 8 July 98
URL: http://www.phreak.org/archives/exploits/unix/network-sniffers/interface-promiscuity-obscurity.txt, (20 August 2002)

“Laws: Cases and Codes: U.S. Code: Title 18: Section 2701 (a)”, 23 Jan 2000
URL:
http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/i/chapters/121/sections/section_2701.html , (22 August 2002)

“Laws: Cases and Codes: U.S. Code: Title 18: Section 1030”, 23 Jan 2000
URLhttp://caselaw.lp.findlaw.com/scripts/ts_search.pl?title=18&sec=1030, (22 August 2002)

“Laws: Cases and Codes: U.S. Code: Title 18: Section 1030 (a)(2)(c)”, 23 Jan 2000
URLhttp://caselaw.lp.findlaw.com/scripts/ts_search.pl?title=18&sec=1030, (22 August 2002)

SANS Institute, Track 8 – System Forensics, Investigation, and Response, 8.4 – Forensic Frameworks and Best Practices – Managerial and Legal Issues,
Sans Institute, 2002,

KillDisk, Lsoft Technologies, Version 1.1
URL: http://www.lsoft.net, (20 August 2002)

Unzip Utility Reference
URL: http://www.info-zip.org/pub/infozip/UnZip.html, (20 August 2002)

MD5Sum
URL: http://www.etree.org/md5com.html, (20 August 2002)

Pfenning, Frank “The Elf Meta-Language”,
URL: http://www-2.cs.cmu.edu/~fp/elf.html, (20 August 2002)

SANS Institute, “Track 8 – System Forensics, Investigation, and Response, Book 8.3”
Pg 2-114

Simonsen, Keld, “News About Standardization”, 21 Nov 2000
URL: http://www.usenix.org/publications/login/standards/37.simonsen.html, (20 August 2002)

Symantec Anti-Virus Site, “Explanation of Trojan Horses” 29 Jul 2002
URL:http://service2.symantec.com/SUPPORT/nip.nsf/1b078893dcd782a985256771004dfaa5/4b119f1de20fb66188256862007b3a5e?OpenDocument

VMWARE Workstation 3.1, VMWARE, Inc.
URL: http://www.vmware.com/company/ (August 27, 2002)

VMWARE Help Program

ARIN Whois, “Search Results for 104.48.0.0”
URL: http://ws.arin.net/cgi-bin/whois.pl

Unix Manpages, “stat(2)”,
URL: http://www.mcsr.olemiss.edu/cgi-bin/man-cgi?fstat+2

UnixGuide.Net, “What is /proc/kcore?”
URL: http://www.unixguide.net/linux/faq/04.16.shtml , (August 27, 2002)

Google Search Engine
URL: http://www.google.com, (20 August 2002)

Part 3 References

"Laws: Cases and Codes : U.S. Code : Title 18 : Section 2511 (1)", 23 Jan 2000 URL: http://caselaw.lp.findlaw.com/scripts/ts_search.pl?title=18&sec=2511 , (22 August 2002)

"Laws: Cases and Codes : U.S. Code : Title 18 : Section 2511 (2)(a) (1)", 23 Jan 2000
URL: http://caselaw.lp.findlaw.com/scripts/ts_search.pl?title=18&sec=2511 , (22 August 2002)

"Laws: Cases and Codes : U.S. Code : Title 18 : Section 2701 (a)", 23 Jan 2000 URL:
http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/i/chapters/121/sections/section_2701.html , (22 August 2002)

"Laws: Cases and Codes : U.S. Code : Title 18 : Section 2702 (a)", 23 Jan 2000 URL:
http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/i/chapters/121/sections/section_2702.html , (22 August 2002)

"Laws: Cases and Codes : U.S. Code : Title 18 : Section 2701 (c)", 23 Jan 2000 URL:
http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/i/chapters/121/sections/section_2701.html , (22 August 2002)

"Laws: Cases and Codes : U.S. Code : Title 18 : Section 2702 (c)", 23 Jan 2000 URL:
http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/i/chapters/121/sections/section_2702.html , (22 August 2002)

"Laws: Cases and Codes : U.S. Code : Title 18 : Section 2702 (b)(5)", 23 Jan 2000
URL:
http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/i/chapters/121/sections/section_2702.html , (22 August 2002)

Mail Utilities, "Digging through Old Emails not Wiretap Violation, Federal Judge Rules", 29 Mar 2001, URL:
http://www.mailutilities.com/news/archive/43/1102.html

"Laws: Cases and Codes : U.S. Code : Title 18 : Section 2702 (b)(6)",23 Jan 2000
URL:
http://caselaw.lp.findlaw.com/casecode/uscodes/18/parts/i/chapters/121/sections/section_2702.html , (22 August 2002)

Delaney, Donald P., Denning, Dorothy E., Kaye, John, and McDonald, Alan R. "WIRETAP LAWS AND PROCEDURES WHAT HAPPENS WHEN THE U.S. GOVERNMENT TAPS A LINE", 23 September 1993, URL: http://www.cs.georgetown.edu/~denning/wiretap/Wiretap.txt, (22 August 2002)

Carnegie Mellon University, "Setting up a logon banner on Windows NT 4.0", 17 Mar 1999, URL: http://www.cert.org/security-improvement/implementations/i034.01.html (22 Aug 2002)

ITSC, "Logon Warning Banners", URL: http://www.itsc.state.md.us/info/InternetSecurity/BestPractices/WarnBanner.htm (22 August 2002)

DDN Security Coordination Center, "Computer System Welcome Banners", 2 Mar 1990, URL: http://csrc.ncsl.nist.gov/secalert/ddn/1990/sec-9004.txt (22 August 2002)