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

# Analysis of an Unknown Binary, Legal Issues, and "The Hack"

GCFA Practical Assignment
Version 1.4
Part 1, Part 2 - Option 1

Jeri L. Malone
October 4, 2004

Table of Contents

## Abstract

This paper will be written to meet the requirements of the current SANS™ Global Information Assurance Certification (GIAC) Certified Forensic Analyst (GCFA) certification. It includes the following:

- Part 1 - Analysis of an Unknown Image
  A zip file was retrieved from the SANS GIAC web site http://www.giac.org/gcfa/binary_v1_4.zip. The task requires a download of the image, performance of a full image analysis, and formal documentation of the forensic analysis steps; thus, demonstrating the concepts and technologies associated with the process.
- Part 2 - Option 1: Perform Forensic Analysis on a system
  Provides an explanation of an exposure on a partner network of an actual corporation. An investigation was triggered when a system administrator on a development host recognized root passwords had been changed. Subsequent investigations may prove the system or systems on the subnet also may have been compromised.
- Part 3 – Legal Issues of Incident Handling
  Addresses the distribution of copyrighted materials via systems available to the public.

The laptop configuration used for this assignment is a Toshiba © 8000 Pentium II with a dual partitioned host 20 GB Western Digital © hard drive. The user has the ability to select Red Hat ® Linux 7.2 (2.4.20) or Microsoft ® Windows 2000 © Professional (SP4). Another Toshiba 8100 Pentium III was also used as a backup. Its configuration is Windows 2000 SP4 with a 40 GB Western Digital hard drive purchased for SANS 2004 with VMware © 4.5.2 configured to host Red Hat Enterprise 3. (2.4.21-EL)

- The approach used to analyze follows:
  - Guarantee the integrity of the image data by using specific mount commands.
  - Use toolsets that include Sleuth Kit © commands such as ils, dcat, mactime and/or programs such as Autopsy, Lazarus, Foremost, text editors, hex editors, debuggers, dump and trace programs. The toolsets were provided by the SANS Institute in Forensics Track 8 and Helix compact disks.
  - Document all tasks performed.
  - A formal documented report is presented to the committee.

## Summary Statement:

The SANS GCFA practical assignment V1.4 presents each candidate with background information (copied portions from SANS website):

> An employee, John Price has been suspended from his place of employment when an audit discovered that he was using the organizations computing resources to illegally distribute copyrighted material. Unfortunately Mr. Price was able to wipe the hard disk of his office PC before investigators could be deployed. However, a single 3.5-inch floppy disk (the floppy disk image that you must use for this assignment can be downloaded here) was found in the drive of the PC. Although Mr. Price has subsequently denied that the floppy belonged to him, it was seized and entered into evidence:
>
> The floppy disk contains a number of files, including an unknown binary named 'prog'. Your primary task is to analyze this binary to establish its purpose, and how it might have been used by Mr. Price in the course of his alleged illegal activities. You should also examine the disk for any other evidence relating to this case. It is suspected that Mr. Price may have had access to other computers in the workplace.
>
> - Tag# fl-160703-jp1
> - 3.5 inch TDK floppy disk
> - MD5: 4b680767a2aed974cec5fbcbf84cc97a
> - fl-160703-jp1.dd.gz

The candidates then answer the following questions:

What type of program is it?

Name: bmap-1.0.20
File Mactime: Wed Jul 16 2003 02:05:33 487476 ..c -/-rwxr-xr-x 502 502 18 /prog
File Owner: The owner of the file is uid (502), groupid (502).
File Size: 174469
Md5sum: f38857e11e405e994dd3a1ea81c23893 bmap
Keywords associated with the file:
List found in the Appendix B

What is it used for?

To utilize the space that occurs at the end of blocks or in fragments. In the case of this image, the blocksize is 1024 bytes. Prog tool locates this slack space in files, can write to the slack space, and clear (rewrite) the slack space.

When was the last time it was used?

Within the binary, there is a notation containing the date 7/15/03. This date is placed there during the compile stage. This date does not correlate with the date shown in the mactime analysis noted above.

## Part 1: Analyze an Unknown Binary

### Acquisition

The laptop is isolated from any network and Vmware is not active on the system. The SANS Track 8 response_toolkit directory is used directly or installed later on the forensic system. The file retrieved from the GIAC website is named *binary_v1_4.zip*. The file was downloaded to a clean, formatted floppy drive and McAfee ® Virus Scan ™ was run on the file, resulting the message “no infected files”.
File details (Windows)

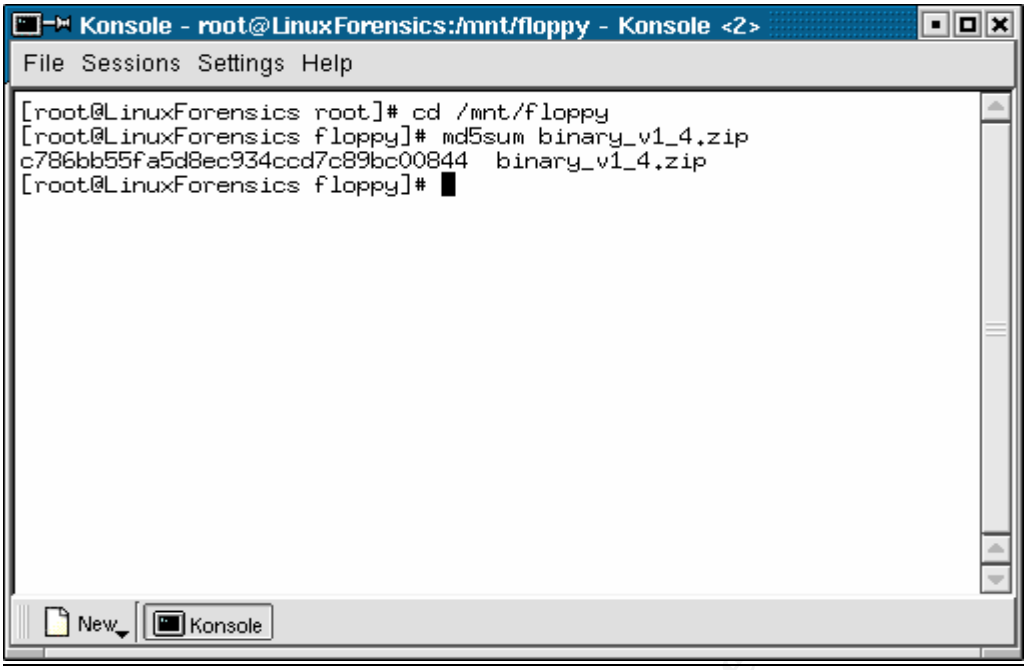
File size: 449 KB Compressed (zipped) Folder

### Integrity checking

Integrity checking is the most critical step of any analysis. Forensic analysts must ensure the files copied have not been changed in any way. It also means an investigation cannot be correct if the evidence is not an exact copy or can not be documented why there are discrepancies.

Before any tools are run there are certain commands at the operating system level that provide identification, cryptographic sums, and file system and content information. Commands such as *ls, md5sum and md5deep,* and *strings* are used to provide this information.

The *md5sum* performs a cryptographic hash value synonymous to a fingerprint that once calculated, uniquely identifies the file and determines if its bit-by-bit numeric pattern has not been changed. The md5sum for the zip file from the GIAC website for this file was not provided. An md5sum on the zip file provided the following value:




```
file binary_v1_4.zip
binary_v1_4.zip: Zip archive data, at least v2.0 to extract
```

G*unzip* was used to unzip the file. Gunzip is a Linux utility that will "uncompress" a file into its original state. Zipping a file is equivalent to compressing the file to a point where all unused space in the file is discarded and an index is built identifying where the dead space should be. To use an analogy, consider a double spaced typewritten document with the placement of two spaces behind every period at the end of a sentence. Removal of the extra spaces and blank lines between paragraphs essentially compresses the document or "zips" it. Unzip does the reverse by replacing the white space and creating the format that previously existed. Most importantly, zipping a file reduces the size of the file and *g-unzipping* the file returns it back to its original size. A *md5sum* was performed on the unzipped file. The following hash was provided:

```
d7641eb4da871d980adbe4d371eda2ad
```

Using the UNIX command "*man file*" produced a help document that describes the system's file [version 3.39]. The file command runs a series of checks with "*magic number*" validations. This means that normally a Unix file located in /usr/share/magic certain numerical values can associate these values to a particular file type; if the file type is known to the particular kernel of the Linux system being run.

The file command against the zip file produced the following:

```
file binary_v1_4.zip
binary_v1_4.zip: Zip archive data, at least v2.0 to extract
```

The command *ls –l* shows the current directory or "folder" list with the long list option showing the owner, date values, and directory permissions. Additionally, it showed the access time on the file to be that of the time that the file was g-unzipped. The *gunzip* command actually touches the file or creates read access, thereby updating the date and time stamp. This is unavoidable, because the image had to be accepted as it was presented through the chain of custody. Measures had to be taken to minimize any additional accesses.

Performing an *ls* presented three files: fl-160703-jp1.dd.gz, fl-160703-jp1.dd.gz.md5 and prog.md5. The *md5sum* was performed on fl-160703-jp1.dd.gz and the md5sum value for prog.md5 was displayed using the *cat* command.

```
4b680767a2aed974cec5fbcbf84cc97a  fl-160703-jp1.dd.gz
7b80d9aff486c6aa6aa3efa63cc56880  prog
```

**Analysis:**

To perform an analysis on the image, it was mounted using the Linux *mount* command. Mounting a file allows the system to point to an area specifically devoted to the image content. According to *Unix man pages:mount* found at URL http://www.rt.com/man/mount.8.html

> "All files accessible in a Unix system are arranged in one big tree, the file hierarchy, rooted at */*. These files can be spread out over several devices. The **mount** command serves to attach the file system found on some device to the big file tree. Conversely, the **umount**(8) command will detach it again". – Mount (8) Linux Programmer's Manual

This stops the system times or any associated file from being changed. The mount command used below shows the steps taken to ensure this mount point remains in an undisturbed state. As described further in the *man mount (8) options* from the same web page listed above, the command mount used alone shows the devices attached to the file system and its state: -t ext2 (Linux ext2 file system) (rw: read/write), ro (read only) noexec (no execution of binaries), nosuid (do not allow set user id or bits to take any effect on the mounted file), noatime (don't change inode access times on this file system) and the name of the mount point specifically specified (/mnt/Evidence)

```
[root@LinuxForensics Sans]# mount -t ext2 -o loop,ro,noatime,noexec,nosuid fl-160703-jp1.dd
/mnt/Evidence
[root@LinuxForensics Sans]# mount
/dev/hda3 on / type ext2 (rw)
none on /proc type proc (rw)
usbdevfs on /proc/bus/usb type usbdevfs (rw)
/dev/hda2 on /boot type ext2 (rw)
none on /dev/pts type devpts (rw,gid=5,mode=620)
/dev/hda7 on /member type vfat (rw)
/dev/hda8 on /home type ext2 (rw)
none on /dev/shm type tmpfs (rw)
/dev/hda9 on /tmp type ext2 (rw)
/dev/hda6 on /tools type vfat (rw)
```

```
/dev/hda5 on /vm type ext2 (rw)
/home/forensics/Sans/fl-160703-jp1.dd on /mnt/Evidence type ext2
(ro,noexec,nosuid,noatime,loop=/dev/loop0)
```

Within the /mnt/Evidence directory (cd /mnt/Evidence) there are 4 directories: John, Docs  lost+found, and May03.  There are prog, nc-1.10-16.i386.rpm..rpm, and ./~5456g.tmp  files.

The command *md5deep –r* supplies details of the /mnt/Evidence directory and provides a cryptographic hash value calculation of all files in the directory. Written by Special Agent Jesse Kornblum, md5deep is described as follows from the Introduction found at the web URL :  http://md5deep.sourceforge.net/:

> "md5deep is a cross-platform set of programs to compute MD5, SHA-1, or SHA-256 message digests on an arbitrary number of files. The programs run on Windows, Linux, *BSD, OS X, Solaris, and should run on most other platforms. md5deep is similar to the md5sum program found in the GNU Coreutils package…"

```
[root@LinuxForensics Sans]#  md5deep -r /mnt/Evidence
636be3f63d098684b23965390cea0705  /mnt/Evidence/John/sect-num.gif
1083d681b1e7a1581c70042a7e1417de  /mnt/Evidence/John/sectors.gif
7b80d9aff486c6aa6aa3efa63cc56880  /mnt/Evidence/prog
f5095084edd7d08e59c6067de84a347d  /mnt/Evidence/May03/ebay300.jpg
ba3f18b1fb92310057495475661d21f9  /mnt/Evidence/Docs/Letter.doc
82d58d80782a3c017738d00d3a33e2b9  /mnt/Evidence/Docs/Mikemsg.doc
e51fb91c9bed2829ca8bf5ae22c28a98  /mnt/Evidence/Docs/Kernel-HOWTO-html.tar.gz
df2ffba17a329e7f150d1df3af16c57b  /mnt/Evidence/Docs/MP3-HOWTO-html.tar.gz
4a8ca21db1f7fc3c37f203600e58cca7  /mnt/Evidence/Docs/Sound-HOWTO-html.tar.gz
d8a3bad9dcdbd81190510086033843ac  /mnt/Evidence/Docs/DVD-Playing-HOWTO-html.tar
535003964e861aad97ed28b56fe67720  /mnt/Evidence/nc-1.10-16.i386.rpm..rpm
f13ddc8775e4234f8d889a6e49bc69eb  /mnt/Evidence/.~5456g.tmp
[root@LinuxForensics Sans]#
```

Strings were examined using the *strings* command.

> "For each  *file*  given, GNU **strings** prints the printable character sequences that are at least  4 characters  long (or  the number given with the options below) and are followed by an unprintable character" .  – strings (1) GNU Development Tools  URL:  http://www.rt.com/man/strings.1.html

*Strings* found alphanumeric values and displayed those values with an offset value. The offset value assists in analyzing other forensic tools located where the character set appears in the image.  Using strings certain characters can be located, such as those that make letters, words, and sentences.  *Strings* is a powerful tool, and provides the analyzer with text based values found in a binary file rather than just machine code.

*Strings* run on the entire image provided a text file sized 62.8KB.  Interesting remnants are listed in Appendix A.

A combination of *ls* and *file* commands show:

```
[Forensics root]# cd /mnt/Evidence
```

```
[root@LinuxForensics Evidence]# ls
Docs  John  lost+found  May03  nc-1.10-16.i386.rpm..rpm  prog
[root@LinuxForensics Evidence]# file Docs
Docs: directory
[root@LinuxForensics Evidence]# cd Docs
[root@LinuxForensics Docs]# ls
DVD-Playing-HOWTO-html.tar  Letter.doc   MP3-HOWTO-html.tar.gz
Kernel-HOWTO-html.tar.gz    Mikemsg.doc  Sound-HOWTO-html.tar.gz
[root@LinuxForensics Docs]# file Letter.doc
Letter.doc: Microsoft Office Document
[root@LinuxForensics Docs]# file Mikemsg.doc
Mikemsg.doc: Microsoft Office Document
[root@LinuxForensics Docs]# file DVD-Playing-HOWTO-html.tar
DVD-Playing-HOWTO-html.tar: POSIX tar archive
[root@LinuxForensics Docs]# file Kernel-HOWTO-html.tar.gz
Kernel-HOWTO-html.tar.gz: gzip compressed data, deflated, original filename, `Kernel-HOWTO-
html.tar', last modified: Sun Jan 21 15:29:57 2001, os: Unix
[root@LinuxForensics Docs]# file MP3-HOWTO-html.tar.gz
MP3-HOWTO-html.tar.gz: gzip compressed data, deflated, original filename, `MP3-HOWTO-
html.tar', last modified: Wed Nov  1 13:38:32 2000, os: Unix
[root@LinuxForensics Docs]# file Sound-HOWTO-html.tar.gz
Sound-HOWTO-html.tar.gz: gzip compressed data, deflated, original filename, `Sound-HOWTO-
html.tar', last modified: Wed Mar 15 17:05:13 2000, os: Unix
[root@LinuxForensics Docs]# cd ../
[root@LinuxForensics Evidence]# ls
Docs  John  lost+found  May03  nc-1.10-16.i386.rpm..rpm  prog
[root@LinuxForensics Evidence]# cd John
[root@LinuxForensics John]# ls
sect-num.gif  sectors.gif
[root@LinuxForensics John]# file sect-num.gif
sect-num.gif: GIF image data, version 87a, 145 x 145,
[root@LinuxForensics John]# file sectors.gif
sectors.gif: GIF image data, version 87a, 282 x 131,
[root@LinuxForensics John]# cd ../lost+found/
[root@LinuxForensics lost+found]# ls
[root@LinuxForensics lost+found]# cd ../May03/
[root@LinuxForensics May03]# ls
ebay300.jpg
[root@LinuxForensics May03]# file ebay300.jpg
ebay300.jpg: JPEG image data, JFIF standard 1.01, resolution (DPI), 96 x 96
[root@LinuxForensics May03]# cd ../
[root@LinuxForensics Evidence]# ls
Docs  John  lost+found  May03  nc-1.10-16.i386.rpm..rpm  prog
[root@LinuxForensics Evidence]# file nc-1.10-16.i386.rpm..rpm
nc-1.10-16.i386.rpm..rpm: RPM v3 bin i386 nc-1.10-16
[root@LinuxForensics Evidence]# file prog
prog: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), for GNU/Linux 2.2.5, statically
linked, stripped
```

The activities above provide the analyst the following data:

- The image is mounted in a read-only mode
- Cryptographic sums of all files

- The prog binary md5sum matches that of the md5sum from the contents of the binary_v1_4.zip file
- Strings output that can be used to formulate a dirty word list
- Directory listings
- Known file types

**The Sleuth Kit**

The Sleuth Kit (TSK) is a tool kit written by Brian Carrier containing open source forensic analysis tools. TSK is based on another data analysis tool called The Coroner's Toolkit (TCT). It contains sixteen binary files that provide information about the file system, data, meta data, and filename structures. TSK provides tools that function at five layers:

1. File System Layer (information about the file system)
2. Content Layer Tools (blocks, blocksize)
3. Meta data (deals with data section of block owned by inode)
4. Human Interface Layer (File and directory entries)
5. Media Management (partitian)

The Sleuth Kit can be found at http://sleuthkit.sourceforge.net/. The Coroner's Toolkit was authored by Dan Farmer and Weitze Venema and can be found at http://www.fish.com/tct. The SANS Institute Forensic Analysis Cheat Sheet v1.0 was used as a reference to these activities.

**Timeline**

One of the first tasks to perform when investigating an incident is the placement of events in chronological order. A timeline allows the investigator to prepare his focus and establish a sequence of events to cross-reference against anomalies in the image copy. In forsenics, timelines can be created through a series of Sleuth Kit commands, *dls*, *fls*, *ils,* and *mactime*.

A Linux filesystem has a master table of pointers. This master table is called an Inode table. Red Hat glossary defines inode as "Data structure that contain information about files in a UNIX or UNIX-compatible *file system*." This definition and additional information, i.e. file systems can be found at http://www.redhat.com/docs/glossary/index.html#I. Inodes are the basis of a file system structure. Inodes contain information about the file including owner, file name, size, and mactimes.

Each disk segment has a master table of inode numbers. Each segment is the memory or storage address with a calculated blocksize. A *mactime* is the timestamp information of a file and is represented "m.a.c". The letters indicate the time last modified, accessed, and created The mactime command allows the investigator to time order the file accesses and begin analysis interpretation. Mactimes are explained this way in

"What are MACtimes? Dr. Dobb's Journal October 2000, " Powerful tools for digital databases" by Dan Farmer, URL: http://www.ddj.com/documents/s=880/ddj0010f/0010f.htm

> "…the term "Mactime" as a shorthand way to refer to the three (or in some versions of Linux, four) time attributes –mtime, atime, and ctime – that are attached to any file or directory in UNIX, NT, and other file systems. (Microsoft documents these three times as LastWriteTime, Last AccessTime, and CreationTime respectively)."

The *ils* command creates a file that provides inode listings of all the files of the image providing creation times. File system layer information is provided by the *fls* command, while *dls* command provides data layer system information, such as ELF file, data, and Microsoft ® file type.

An article of interest regarding fls, ils, and mactime is "Freeware Forensics Tools for UNIX" by Derek Cheng, CISSP, GCIH at http://www.securityfocus.com/printable/infocus/1503.

The following steps were taken to create the timelines:

- Using the *fls* command created a file with all filename data
- Using the *ils* command created a file with all of the deleted inode structures
- Using the *cat* command created an overall body file
- Using the *mactime* command created the timeline

The *fls -f linux-ext2 –r /home/forensics/Sans/fl-160703-jp1.dd* was used to capture filesystem information recursively for the directories in the image. The command provided a file or directory name, the size, and time in machine time format.
The *ils –f linux-ext2 –m /home/forensics/Sans/fl-160703-jp1.dd* was used to gather the inode information against the same image file, also in machine time format. The merging of the two files creates the mactime file.

```
[root@LinuxForensics Sans]# ils -f linux-ext2 -m /home/forensics/Sans/fl-160703-jp1.dd > fl-160703-jp1.ils
     0 class|host|start_time
     22 body|LinuxForensics|1087148382
     53 md5|file|st_dev|st_ino|st_mode|st_ls|st_nlink|st_uid|st_gid|st_rdev|st_size|st_atime|st_mtime|st_ctime|st_blksize|st_blocks
    177 0|<fl-160703-jp1.dd-alive-1>|0|1|0|----------|0|0|0|0|0|1058191689|1058191689|1058191689|1024|0
    273 0|<fl-160703-jp1.dd-dead-23>|0|23|33261|-rwxr-xr-x|0|0|0|0|100430|1058191935|1058191935|1058192353|1024|0
    379 0|<fl-160703-jp1.dd-dead-27>|0|27|33261|-rwxr-xr-x|0|502|502|0|546116|1058194030|1058191993|1058335380|1024|0
```

```
[root@LinuxForensics Sans]# cat /home/forensics/Sans/fl-160703-jp1.?ls > /home/forensics/Sans/fl-160703-jp1.mac
```

Activities can be viewed in the chronological order. The timeline showed events that ranged from January 2003 through July 2003 and can be viewed in Appendix C.

The command *fsstat -f linux-ext2 /home/forensics/Sans/fl-160703-jp1.dd* provides file system information against the entire image:

```
FILE SYSTEM INFORMATION
--------------------------------------------
File System Type: EXT2FS
Volume Name:
Last Mount: Wed Jul 16 02:12:33 2003
Last Write: Wed Jul 16 02:12:58 2003
Last Check: Mon Jul 14 10:08:08 2003
Unmounted properly
Last mounted on:
Operating System: Linux
Dynamic Structure
InCompat Features: Filetype,
Read Only Compat Features: Sparse Super,

METADATA INFORMATION
--------------------------------------------
Inode Range: 1 - 184
Root Directory: 2

CONTENT-DATA INFORMATION
--------------------------------------------
Fragment Range: 0 - 1439
Block Size: 1024
Fragment Size: 1024

BLOCK GROUP INFORMATION
--------------------------------------------
Number of Block Groups: 1
Inodes per group: 184
Blocks per group: 8192
Fragments per group: 8192

Group: 0:
  Inode Range: 1 - 184
  Block Range: 1 - 1439
    Super Block: 1 - 1
    Group Descriptor Table: 2 - 2
    Data bitmap: 3 - 3
    Inode bitmap: 4 - 4
    Inode Table: 5 - 27
    Data Blocks: 28 - 1439
```

The output revealed that the data was stored in 1024 byte fragments. If there is the possibility of data being stored in slack space, it will appear at the end of the fragments. The timeline shows the preparation for the attacks began January 28 through May 21, 2003. The increased activity in July shows the execution of the attack.

## FOREMOST

Foremost version 0.69, written by Kris Kendall and Jesse Kornblum helps categorize files by reading the header information and creating a text file of known file types. Foremost can be located at http://foremost.sourceforge.net/. Extraction of all unallocated data shows the likelihood of a deleted program and creates the output dls file, */home/forensics/Sans/fl-160703-jp1.dls.* Running foremost on the fl-160703-jp1.dls file produced one hit on 00000000.pgp. It had a length of 100000, concluding that the search criteria (pgp) were off.

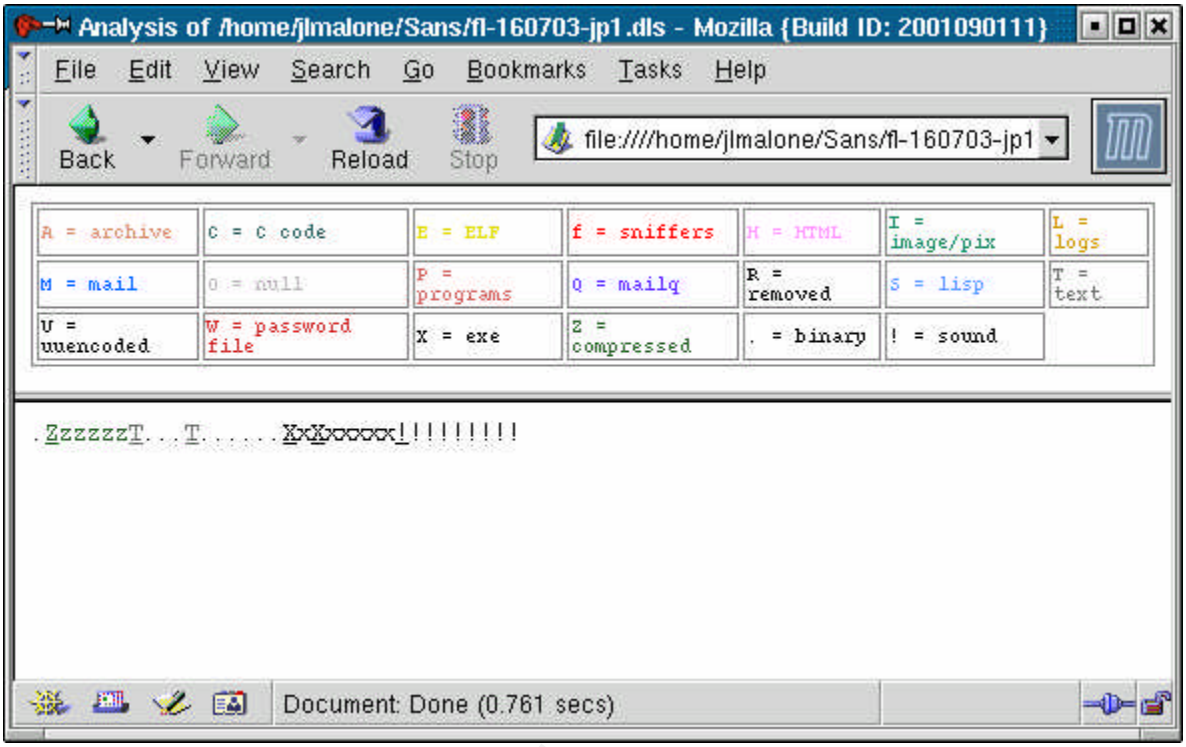
Using the hex editor did not find .pgp or 00000000.pgp (text) in the file. Rerunning foremost without the -q (quick) option to see if the results would remain the same created the file *audit.txt* in the /images/foremost/ directory. The results were different. This option produced 66 pgp files, one rpm file, and one tgz file. The question became, are these deleted files attempts at the hacker's program to compile and successfully run a program or shell? Reviewing the files with the hex editor revealed various references to compilers, cache data, and other items.

```
[root@LinuxForensics foremost-0.69]# ./foremost -o /images/foremost -c /usr/local/src/foremost-
0.69/foremost.conf /home/forensics/Sans/fl-160703-jp1.dls
foremost version 0.69
Written by Kris Kendall and Jesse Kornblum.

Opening /home/forensics/Sans/fl-160703-jp1.dls
/home/forensics/Sans/fl-160703-jp1.dls: 100.0%   630.0 KB    00:00 ETA
Foremost is done.

[root@LinuxForensics foremost]# cat /images/foremost/audit.txt

Foremost version 0.69 audit file
Started at Wed Jun  9 23:22:46 2004
Command line:
./foremost -o /images/foremost -c /usr/local/src/foremost-0.69/foremost.conf
/home/forensics/Sans/fl-160703-jp1.dls
Output directory: /images/foremost
Configuration file: /usr/local/src/foremost-0.69/foremost.conf

Opening /home/forensics/Practical/Sans/fl-160703-jp1.dls
File          Found at Byte  Int  Chop   Length     Extracted From
00000000.pgp          34244   X    X     100000     fl-160703-jp1.dls
00000001.pgp          35268   X    X     100000     fl-160703-jp1.dls
00000002.pgp          35780   X    X     100000     fl-160703-jp1.dls
  **edited for brevity **
00000065.pgp         125796   X    X     100000     fl-160703-jp1.dls
00000066.pgp         127140   X    X     100000     fl-160703-jp1.dls
00000067.rpm           6663   X          638457     fl-160703-jp1.dls
00000068.tgz           1024   X          150000     fl-160703-jp1.dls
```

## Lazarus

Lazarus, another tool commonly known by investigators was run. Lazarus reads an image and produces information from the raw data (e.g., slack space) into categories.

The Lazarus evaluation provided the following information and is 1K blocksize : Zz represents a compressed file spanning six blocks, and two executables totaling 8 blocks.




### Autopsy Forensic Tool

The tests performed using The Sleuth Kit, Foremost, and Lazarus provided useful information, but were rather time consuming and sometimes hard to read. The Autopsy tool (v.1.70) automates the tests performed using those command line tools, and provides a nice graphical interface. Autopsy is written by Dan Farmer and can be found at http://www.sleuthkit.org/autopsy/.

```
[root@LinuxForensics src]# ls
autopsy       kerncheck      md5deep-1.0      sleuthkit
autopsy-1.75  kregedit-0.1   memdump-1.0      sleuthkit-1.67
dcfldd-1.0    mac_daddy      ntfsprogs-1.8.2  tct
foremost-0.69 mac_daddy.tgz  NTFS-RPMS        tct-1.14
gpart-0.1h    md5deep-0.15   RPMS
[root@LinuxForensics src]#
```

Using Autopsy, the metadata information for the inodes listed above was reviewed. Examining inode 28 it was determined to be unallocated. Looking at the fragments beginning with 73 in hexadecimal, the ROOT ENTRY was found. Macro data pertaining to the word doc templates were also found.

A MP3 decoder for Linux file called *xmms-mpg123.1.2.7-13.i386.rpm* . provides help in web searches. It was shipped from Red Hat to fix problems users had playing mp3s. A question was posed to Experts Exchange http://www.experts-exchange.com/Operating_Systems/Linux/Q_20817247.html where a user asked how to listen to songs on Linux. Only a subscription to the service will provide an answer. Accessing *http://staff.xmms.org/priv/redhat8/* the source to my work production box was downloaded. By going to the Security Focus web site [1] and inputting *xmms* into the search dialog box, a Bugtraq id 7534 was found.

Info:
Input Validation Error
CVE-MAP-NOMATCH
Remote Yes
local No
Published" May 08,2003
Updated: May13, 2003
Vulnerable: X2 Studios XMMS Remote 0.1
not vulnerable: X2 Studios XMMS Remote 0.2

Discussion:
"A problem with the software package could make unauthorizede command execution possible.

It has been reported that a problem in the XMMS Remote software package could allow an attacker to pass arbitrary commands through its Perl Script. This could lead to attacks against system resources.

Exploit:
There is no exploit code required.

Solution:
Fixes available:
X2 Studios XMMS Remote 0.1:
X2 Studios Upgrade XMMS REmote 0.2
http://www/x2studios.com/download.php?id=9
Credit:
Discovery of this vulnerability credited to Chris Dolan.

URL: http://www.securityfocus.com/bid/7534/info

It is unknown how this information may apply to the project. This information was noted and the investigation continued with the focus on looking into the directories.

### Directories and files

The *John* directory contained two files. Through the Linux graphic program on the PC these file were examined and determined to be drawings. The first is a drawing of a disk drive with sector numbers. The picture sect-num displays the sectors numbered counter-clockwise. The sectors.gif shows a picture of a hard drive with a pie-like slice cutout of the sector.

[1] http://www.securityfocus.com/

```
[root@LinuxForensics John]# file sect-num.gif
        sect-num.gif: GIF image data, version 87a, 145 x 145,
[root@LinuxForensics John]# file sectors.gif
        sectors.gif: GIF image data, version 87a, 282 x 131,
```

The May03 directory contained one file, ebay300.jpg, using the file command. The image is a graphic of the eBay web site. It seemed out of the ordinary for a partition to include the ebay300.jpg file.

```
[root@LinuxForensics May03]# file ebay300.jpg
        ebay300.jpg: JPEG image data, JFIF standard 1.01, resolution (DPI), 96 x 96
```

The *Docs* directory contained Lettermsg.doc and Mikemsg.doc. The file command says these are Microsoft Office Documents. These files were viewed with Red Hat's editor. The Lettermsg.doc was opened with Microsoft Word. It appears to be a Contemporary letter template. Properties on this file show the file name is Contemporary Letter and the owner as John Price.

The Mikemsg.doc file is a four-line Microsoft Word document in Times New Roman 10 font. In the document properties, the title is Hey Mike, the author is John Price, and the company is CCNOU. This is the first indication that Mr. Price has misused company resources for personal use; because the company name is displayed (depending on the corporate policy).

```
[root@LinuxForensics Docs]# file Letter.doc
        Letter.doc: Microsoft Office Document
[root@LinuxForensics Docs]# file Mikemsg.doc
        Mikemsg.doc: Microsoft Office Document
```

Hey Mike,

I received the latest batch of files last night and I'm ready to rock-n-roll (ha-ha).

I have some advance orders for the next run. Call me soon.

JP

There were also three zipped tar files, and one unzipped tar file. Using the Mozilla browser to review, the tar file showed it could be viewed as one document. The pages continued in order, much like one would look though a book or continuous links on a web page. As mentioned earlier in the paper, the *.gz* files were unzipped using the *gunzip* command and tar files were extracted using the *tar –vxf* filename command into an isolated directory. The HOWTOs were browsed with Mozilla web browser, and no there were not any apparent abnormalities.

```
[root@LinuxForensics Docs]# file DVD-Playing-HOWTO-html.tar
        DVD-Playing-HOWTO-html.tar: POSIX tar archive
[root@LinuxForensics Docs]# file Kernel-HOWTO-html.tar.gz
```

```
        Kernel-HOWTO-html.tar.gz: gzip compressed data, deflated, original filename, `Kernel-
HOWTO-html.tar', last modified: Sun Jan 21 15:29:57 2001, os: Unix
[root@LinuxForensics Docs]# file MP3-HOWTO-html.tar.gz
        MP3-HOWTO-html.tar.gz: gzip compressed data, deflated, original filename, `MP3-
HOWTO-html.tar', last modified: Wed Nov  1 13:38:32 2000, os: Unix
[root@LinuxForensics Docs]# file Sound-HOWTO-html.tar.gz
        Sound-HOWTO-html.tar.gz: gzip compressed data, deflated, original filename, `Sound-
HOWTO-html.tar', last modified: Wed Mar 15 17:05:13 2000, os: Unix
```

The *lost+found* directory did not contain any information. An *ls* verified there were no entries in this directory.

```
[root@LinuxForensics John]# cd ../lost+found/
[root@LinuxForensics lost+found]# ls
[root@LinuxForensics lost+found]#
```

*Netcat* (nc-1.10-16.i386.rpm..rpm was underneath the root directory. Netcat, written by Hobbit is known as the Swiss Army knife and runs on Unix/Linux and Windows. Ethical hackers know that netcat (nc) is a favorite tool for a rootkit because it does many things, including: file transfer, perform port scanning, and retrieve web headers. The *–e* option can run as a background process until the server reboots, allowing unlimited connections from an attacker machine to the target machine when established correctly. Netcat runs on both Windows and Linux systems. More information about Netcat can be found at http://www.atstake.com/research/tools/network_utilities/.

```
[root@LinuxForensics Evidence]# file nc-1.10-16.i386.rpm..rpm
nc-1.10-16.i386.rpm..rpm: RPM v3 bin i386 nc-1.10-16
```

**Prog details**

Prog was moved into its own isolated directory /root/and started tcmpdump. It does not execute at first; prog returns a message saying no filename, try '--help' for help.

```
prog:1.0.20 (07/15/03) newt
Usage: prog [OPTION]... [<target-filename>]
use block-list knowledge to perform special operations on files

--doc VALUE
  where VALUE is one of:
  version  display version and exit
  help  display options and exit
  man  generate man page and exit
  sgml  generate SGML invocation info
--mode VALUE
  where VALUE is one of:
  m  list sector numbers
  c  extract a copy from the raw device
  s  display data
  p  place data
  w  wipe
  chk  test (returns 0 if exist)
  sb  print number of bytes available
```

```
  wipe  wipe the file from the raw device
  frag  display fragmentation information for the file
  checkfrag  test for fragmentation (returns 0 if file is fragmented)
--outfile <filename> write output to ...
--label useless bogus option
--name  useless bogus option
--verbose       be verbose
--log-thresh <none | fatal | error | info | branch | progress | entryexit> logging threshold ...
--target <filename> operate on ...
```

Using Google™ web search URL: http://www.google.com/ a string value presented by Prog usage statement was searched. Additionally, "use block-list knowledge to perform special operations on files" was used and a hit was provided by http://lwn.net/2000/0413/announce.php3 . It was for a program called bmap version 1.0.16; however, the version displayed on the usage statement on prog says 1.0.20. No references to the bmap program were easily found. Refining the Google search to *bmap 1.0* produced a linkage to Security Focus. This program was added Oct 22, 2001.

```
bmap 1.0.17 by Daniel Ridge, newt@scyld.com <http://online.securityfocus.com/tools/13559>
```

This still did not match the date on the prog –help, so continuous searches soon produced the correct bmap.1.0.20.tar.gz at http://garchive.movealong.org. This image was saved to the analyst laptop for further investigation.

Running prog as the root user against /etc/vmware/config/ without any options or *prog –m* provided no discernable difference in output. The file /etc/vmware/config was chosen because this string appeared in the strings output. The /etc/vmware/config of the forensic laptop was chosen, although VMware was not used in the analysis. *Prog –c /etc/vmware/config* wrote the contents to the terminal. This option is used to copy the contents back to stdout or it can create an output file.

```
 [root@LinuxForensics prog]# ./prog /etc/vmware/config
1409672
1409673
1409674
1409675
1409676
1409677
1409678
1409679
 [root@LinuxForensics prog]# ./prog --m /etc/vmware/config
1409672
1409673
1409674
1409675
1409676
1409677
1409678
1409679
[root@LinuxForensics prog]# ./prog --c /etc/vmware/config
vmnet1.hostonlyaddress = "192.168.2.1"
vmnet1.hostonlynetmask = "255.255.255.0"
```

```
control.fullpath = "/usr/bin/vmware-cmd"
loop.fullpath = "/usr/bin/vmware-loop"
dhcpd.fullpath = "/usr/bin/vmnet-dhcpd"
wizard.fullpath = "/usr/bin/vmware-wizard"
libdir = "/usr/lib/vmware"
vmware.fullpath = "/usr/bin/vmware"
[root@LinuxForensics prog]#
```

*Prog –s /etc/vmware/config* provided the block number, file size, slack size, and block size of this file. *Prog* –w */etc/vmware/config* should write to the slack area. It did select the block, but provided no update to this slack space because no parameters were passed to prog. This was fortunate for the analyzer, as this is a real VMWare application file on the forensics laptop. The file should not be overridden, especially without the aid of a sniffer or trace program to follow its actions. It also became apparent that the wrong file been used for prog (bmap). The file found in the strings output was *vmware-config.pl* and not */etc/vmware/config*. This is a costly mistake in a real investigation and proves the absolute need to document everything and verify what needs to be done before executing potential destructive and hazardous commands.

Through the activities mentioned above, no activity displayed using tcpdump.

```
[root@LinuxForensics prog]# ./prog --s /etc/vmware/config
getting from block 176209
file size was: 306
slack size: 3790
block size: 4096
[root@LinuxForensics prog]#
[root@LinuxForensics prog]# ./prog --w /etc/vmware/config
stuffing block 176209
file size was: 306
slack size: 3790
block size: 4096
write error
write error
write error
[root@LinuxForensics prog]#
```

Prog was tested using the author's user level id by using the *su myid* command. The program did not run and produced the error message *permission denied*. Once back at the root level id, the command ./prog –m /etc/vmware/config worked successfully.

Prog was also tested for copying and writing against file *progout*. Progout was created using the *prog –c prog > progout* . This created an extract copy of the prog binary and put the output into progout.

```
[root@LinuxForensics prog]# echo "here I am" | ./prog --mode p progout
stuffing block 189892
file size was: 488448
```

```
slack size: 3072
block size: 4096
[root@LinuxForensics prog]# ./prog --mode s  progout
getting from block 189892
file size was: 488448
slack size: 3072
block size: 4096
here I am
[root@LinuxForensics prog]# ./prog --mode wipe  progout
[root@LinuxForensics prog]# ./prog --mode s  progout
getting from block 189892
file size was: 488448
slack size: 3072
block size: 4096
[root@LinuxForensics prog]#
```

Another tool in the forensics arsenal is a program called *strace*. A Google web search on strace brings up a site http://www.die.net/doc/linux/man/man2/strace.1.html:

> "strace – trace system calls and signals. It intercepts and records the system calls which are called by a process and the signals which are received by a process. The name of each system call, its arguments and its return value are printed on the standard error or to the file specified with the –o option"

Attempts to *strace* the prog file were done against /mnt/Evidence/prog, but "permission denied" occurred.

```
[root@LinuxForensics prog]# strace /mnt/Evidence/prog
execve("/mnt/Evidence/prog", ["/mnt/Evidence/prog"], [/* 32 vars */]) = 0
strace: exec: Permission denied
```

The copy of prog isolated in the /home/forensics/Sans/Evidence/tmp/prog directory returned the following message:

```
[root@LinuxForensics prog]# strace /home/forensics/Sans/Evidence/tmp/prog/prog
execve("/home/forensics/Sans/Evidence/tmp/prog/prog",
["/home/forensics/Sans/Evidence/tmp/prog/prog"], [/* 32 vars */]) = 0
fcntl64(0, 0x1, 0, 0xbffffa04)          = 0
fcntl64(0x1, 0x1, 0, 0xbffffa04)        = 0
fcntl64(0x2, 0x1, 0, 0xbffffa04)        = 0
uname({sys="Linux", node="LinuxForensics", ...}) = 0
geteuid32()                             = 0
getuid32()                              = 0
getegid32()                             = 0
getgid32()                              = 0
brk(0)                                  = 0x80bedec
brk(0x80bee0c)                          = 0x80bee0c
```

```
brk(0x80bf000)                          = 0x80bf000
brk(0x80c0000)                          = 0x80c0000
write(2, "no filename. try \'--help\' for he"..., 36no filename. try '--help' for help.
) = 36
_exit(2)                                = ?
```

Assuming the command required additional options, the following were used: *strace –dfirv /mnt/Evidence/prog.*

| Options: | d-debug | f-fork child process via fork value |
|---|---|---|
| | r-print relative timestamp | v-print versions of the environment |

Again, the “permission denied” message occurred, but it provided debug information, instruction pointers, and showed there were no child processes spawned. View the messages relating to SIGSTOP and SIGTRAP.

```
strace -dfirv /mnt/Evidence/prog
     0.000000 [40058a01] execve("/mnt/Evidence/prog", ["/mnt/Evidence/prog"], [/* 32 vars */]) = 0
 [wait(0x137f) = 2434]
pid 2434 stopped, [SIGSTOP]
 [wait(0x57f) = 2434]
pid 2434 stopped, [SIGTRAP]
     0.074388 [400e523b] geteuid32( [wait(0x57f) = 2434]
pid 2434 stopped, [SIGTRAP]
)    = 0
 [wait(0x57f) = 2434]
pid 2434 stopped, [SIGTRAP]
     0.000262 [400e4b6a] execve("/mnt/Evidence/prog", ["/mnt/Evidence/prog"], ["PWD=/root",
"HOSTNAME=LinuxForensics", "LD_LIBRARY_PATH=/usr/local/lib", "QTDIR=/usr/lib/qt-2.3.1",
"LESSOPEN=|/usr/bin/lesspipe.sh %"..., "TPROC=2411", "KDEDIR=/usr", "USER=root",
"LS_COLORS=no=00:fi=00:di=01;34:l"..., "MACHTYPE=i386-redhat-linux-gnu",
"KDE_MULTIHEAD=false", "MAIL=/var/spool/mail/root", "INPUTRC=/etc/inputrc",
"BASH_ENV=/root/.bashrc", "GTK_RC_FILES=/etc/gtk/gtkrc:/roo"..., "XMODIFIERS=@im=none",
"LANG=en_US", "COLORTERM=", "DISPLAY=:0", "LOGNAME=root", "SHLVL=4",
"SESSION_MANAGER=local/LinuxForen"..., "SHELL=/bin/bash", "USERNAME=root",
"HOSTTYPE=i386", "QT_XFT=0", "OSTYPE=linux-gnu", "HISTSIZE=1000", "HOME=/root",
"TERM=xterm", "PATH=/usr/local/sbin:/sbin:/usr/"..., "_=/usr/bin/strace"] [wait(0x57f) = 2434]
pid 2434 stopped, [SIGTRAP]
) = -1 EACCES (Permission denied)
 [wait(0x57f) = 2434]
pid 2434 stopped, [SIGTRAP]
     0.001341 [4010aa3d] dup(2 [wait(0x57f) = 2434]
pid 2434 stopped, [SIGTRAP]
)        = 3
 [wait(0x57f) = 2434]
pid 2434 stopped, [SIGTRAP]
     0.000185 [4010a787] fcntl64(0x3, 0x3, 0x3, 0 [wait(0x57f) = 2434]
pid 2434 stopped, [SIGTRAP]
) = 1
 [wait(0x57f) = 2434]
pid 2434 stopped, [SIGTRAP]
     0.000162 [4010a4fd] close(3 [wait(0x57f) = 2434]
```

```
pid 2434 stopped, [SIGTRAP]
)       = 0
 [wait(0x57f) = 2434]
pid 2434 stopped, [SIGTRAP]
     0.000291 [4010a584] write(2, "strace: exec: Permission denied\n", 32strace: exec: Permission
denied
 [wait(0x57f) = 2434]
pid 2434 stopped, [SIGTRAP]
) = 32
 [wait(0x57f) = 2434]
pid 2434 stopped, [SIGTRAP]
     0.000201 [400e4afd] _exit(1)       = ?
 [wait(0x100) = 2434]
pid 2434 exited
```

The *Objdump* tool is a forensic aid to retrieve information from the binary. This tool is most valuable to analysts possessing a programming background; otherwise, the output is not easily understood. The interesting items this tool provided were the headers (*objdump -h prog*) which appeared in the strings output and the file output (*objdump -f prog*).

```
[root@LinuxForensics prog]# objdump
 [root@LinuxForensics prog]# objdump prog
Usage: objdump OPTION... FILE...
Display information from object FILE.

 At least one of the following switches must be given:
  -a, --archive-headers    Display archive header information
  -f, --file-headers       Display the contents of the overall file header
  -p, --private-headers    Display object format specific file header contents
  -h, --[section-]headers  Display the contents of the section headers
  -x, --all-headers        Display the contents of all headers
  -d, --disassemble        Display assembler contents of executable sections
  -D, --disassemble-all    Display assembler contents of all sections
  -S, --source             Intermix source code with disassembly
  -s, --full-contents      Display the full contents of all sections requested
  -g, --debugging          Display debug information in object file
  -G, --stabs              Display (in raw form) any STABS info in the file
  -t, --syms               Display the contents of the symbol table(s)
  -T, --dynamic-syms       Display the contents of the dynamic symbol table
  -r, --reloc              Display the relocation entries in the file
  -R, --dynamic-reloc      Display the dynamic relocation entries in the file
  -V, --version            Display this program's version number
  -i, --info               List object formats and architectures supported
  -H, --help               Display this information

 [root@LinuxForensics prog]# objdump -f prog

prog:     file format elf32-i386
architecture: i386, flags 0x00000102:
EXEC_P, D_PAGED
start address 0x080480e0

[root@LinuxForensics prog]# objdump -h prog
```

```
prog:     file format elf32-i386

Sections:
Idx Name          Size      VMA       LMA       File off  Algn
  0 .init         00000018  080480b4  080480b4  000000b4  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  1 .text         0004bc20  080480e0  080480e0  000000e0  2**5
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  2 .fini         0000001e  08093d00  08093d00  0004bd00  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
  3 .rodata       0001cce0  08093d20  08093d20  0004bd20  2**5
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  4 __libc_atexit 00000004  080b0a00  080b0a00  00068a00  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  5 __libc_subfreeres 00000040  080b0a04  080b0a04  00068a04  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  6 .data         0000b0e0  080b1000  080b1000  00069000  2**5
                  CONTENTS, ALLOC, LOAD, DATA
  7 .eh_frame     00001530  080bc0e0  080bc0e0  000740e0  2**2
                  CONTENTS, ALLOC, LOAD, DATA
  8 .ctors        00000008  080bd610  080bd610  00075610  2**2
                  CONTENTS, ALLOC, LOAD, DATA
  9 .dtors        00000008  080bd618  080bd618  00075618  2**2
                  CONTENTS, ALLOC, LOAD, DATA
 10 .got          00000010  080bd620  080bd620  00075620  2**2
                  CONTENTS, ALLOC, LOAD, DATA
 11 .bss          000017ac  080bd640  080bd640  00075640  2**5
                  ALLOC
 12 .comment      00000339  00000000  00000000  00075640  2**0
                  CONTENTS, READONLY
 13 .note.ABI-tag 00000020  08048094  08048094  00000094  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
 14 .note         00001388  00000000  00000000  00075979  2**0
                  CONTENTS, READONLY
[root@LinuxForensics prog]# objdump -i prog
BFD header file version 2.11.90.0.8
elf32-i386
 (header little endian, data little endian)
  i386
a.out-i386-linux
 (header little endian, data little endian)
  i386
efi-app-ia32
 (header little endian, data little endian)
  i386
elf32-little
 (header little endian, data little endian)
  i386
elf32-big
 (header big endian, data big endian)
  i386
srec
 (header endianness unknown, data endianness unknown)
  i386
symbolsrec
```

```
 (header endianness unknown, data endianness unknown)
  i386
tekhex
 (header endianness unknown, data endianness unknown)
  i386
binary
 (header endianness unknown, data endianness unknown)
  i386
ihex
 (header endianness unknown, data endianness unknown)
  i386
trad-core
 (header endianness unknown, data endianness unknown)

            rea  elf32-i386 a.out-i386-linux
 efi-app-ia32 elf32-little elf32-big
         i386 elf32-i386 a.out-i386-linux efi-app-ia32 elf32-little elf32-big

              srec symbolsrec tekhex binary ihex trad-core
         i386 srec symbolsrec tekhex binary ihex ---------

[root@LinuxForensics prog]# objdump -p prog | more

prog:     file format elf32-i386

Program Header:
    LOAD off    0x00000000 vaddr 0x08048000 paddr 0x08048000 align 2**12
         filesz 0x00068a44 memsz 0x00068a44 flags r-x
    LOAD off    0x00069000 vaddr 0x080b1000 paddr 0x080b1000 align 2**12
         filesz 0x0000c630 memsz 0x0000ddec flags rw-
    NOTE off    0x00000094 vaddr 0x08048094 paddr 0x08048094 align 2**2
         filesz 0x00000020 memsz 0x00000020 flags r--
```

The *readelf –a prog* command was entered below to provide all (-a) information about an ELF (Executable and Linking Format). The man page for ELF can be found at http://www.mcsr.olemiss.edu/cgi-bin/man-cgi?elf+3; it provides a synopsis, description and additional information for ELF. All information shown is valuable, especially the operating system, architecture, and magic number (7f 45 4c 46 01 01 01 00) which identifies it specifically as a Linux binary.

```
[root@LinuxForensics prog]# readelf -a prog
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:          486796 (bytes into fil
e)
  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:         17
  Section header string table index: 16

Section Headers:
  [Nr] Name              Type            Addr     Off    Si
ze   ES Flg Lk Inf Al
  [ 0]                   NULL            00000000 000000 00
0000 00      0   0  0
  [ 1] .init             PROGBITS        080480b4 0000b4 00
0018 00  AX  0   0  4
  [ 2] .text             PROGBITS        080480e0 0000e0 04
bc20 00  AX  0   0 32
  [ 3] .fini             PROGBITS        08093d00 04bd00 00
001e 00  AX  0   0  4
  [ 4] .rodata           PROGBITS        08093d20 04bd20 01
cce0 00   A  0   0 32
  [ 5] __libc_atexit     PROGBITS        080b0a00 068a00 00
0004 00   A  0   0  4
  [ 6] __libc_subfreeres PROGBITS        080b0a04 068a04 00
0040 00   A  0   0  4
  [ 7] .data             PROGBITS        080b1000 069000 00
b0e0 00  WA  0   0 32
  [ 8] .eh_frame         PROGBITS        080bc0e0 0740e0 00
1530 00  WA  0   0  4
  [ 9] .ctors            PROGBITS        080bd610 075610 00
0008 00  WA  0   0  4
  [10] .dtors            PROGBITS        080bd618 075618 00
0008 00  WA  0   0
 4
  [11] .got              PROGBITS        080bd620 075620 00
0010 04  WA  0   0  4
  [12] .bss              NOBITS          080bd640 075640 00
17ac 00  WA  0   0 32
  [13] .comment          PROGBITS        00000000 075640 00
0339 00      0   0  1
  [14] .note.ABI-tag     NOTE            08048094 000094 00
0020 00   A  0   0  4
  [15] .note             NOTE            00000000 075979 00
1388 00      0   0  1
  [16] .shstrtab         STRTAB          00000000 076d01 00
008a 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 (proc
essor specific)

Program Headers:
```

```
  Type           Offset   VirtAddr   PhysAddr   FileSiz Mem
Siz  Flg Align
  LOAD           0x000000 0x08048000 0x08048000 0x68a44 0x6
8a44 R E 0x1000
  LOAD           0x069000 0x080b1000 0x080b1000 0x0c630 0x0
ddec RW  0x1000
  NOTE           0x000094 0x08048094 0x08048094 0x00020 0x0
0020 R   0x4

 Section to Segment mapping:
  Segment Sections...
   00     .init .text .fini .rodata __libc_atexit __libc_su
bfreeres .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.
```

When using Autopsy, a Keyword Search on the word "newt" was performed. Block (Fragment) 644 (1024 bytes) in images/fl-160703-jp1.dd was presented and occurs at position 111. In addition to display, a notation of "(7/15/03) newt" and several lines down displayed "7/15/03…. How did we get here?" This is obviously a message that was generated during the increased activity timeline.

The string search through Autopsy was increased to include additional keywords such as /etc, fstab, /dev, syn, packet, nc, packet, wipe, root, UID, ./, hop, install, %s and %d, and ELF. The objective was to determine patterns from the prog binary where successful execution of these commands would be written back into some slack space and provided back to the attacker. These keywords provided many fragment numbers on the search words ELF, /dev, fstab, nc and ./. Since there are no active processes to review, the values of the %s and %d could not be provided. However, they are related to the file name and slack size of the fragment it was executed against respectively.

**Questionable strings:**

- There were 3,334 string entries with /dev/. It is unknown why all these entries are in the prog file.

- The string *root* would provide an indication that somehow the userid was compromised; although the string *passwd* did not appear in the image. *Root* appeared thirty-eight times.

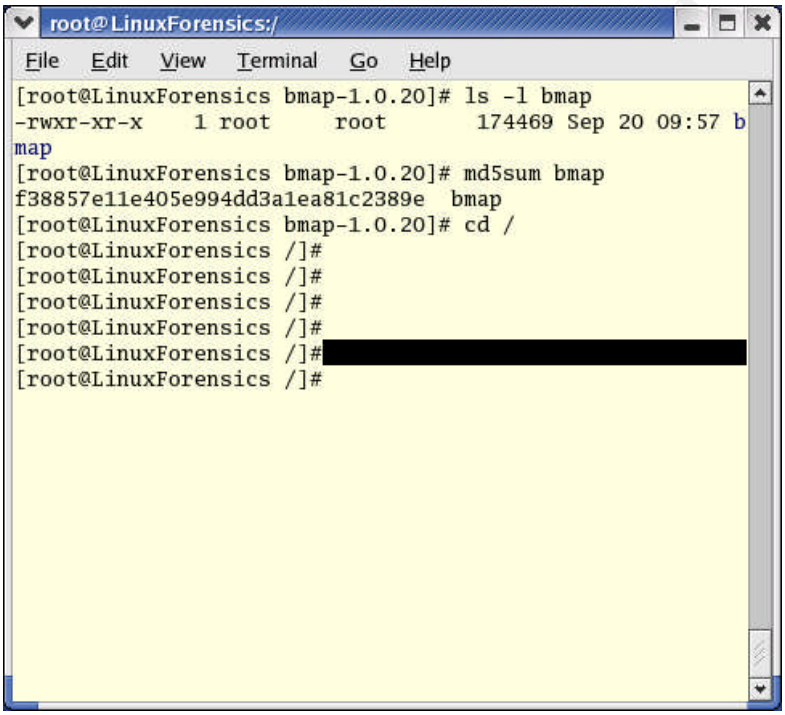
The keyword */bin/sh* only appears once at block 194 offset 212 and identifies the shell script was gained.

- Symbols *%s* and *%d* were two variables that occurred within prog 238 and 42 times respectively. This indicated the use of a script for automation. *00gferg* appeared nine times, and appears in the HOWTO documents. A Google web search did not provide any hits to help determine an explanation.

**Retrieve source and compile.**

As mentioned earlier, the bmap-1.0.20 binary was found at http://www.garchive.movealong.org/bmap-1.0.20. The official web site ftp://ftp.scyld.com/pub/forensic_computing/bmap is no longer available. The author is, Daniel Ridge. The bmap binary was unable to compile on the Red Hat 7.2 forensic laptop probably due to the libraries included. The make command compiled successfully using the backup laptop configured with Red Hat Enterprise 3.

It is unlikely there will be the same cryptographic hash value between the two binaries. The prog binary is statically linked, that is, it has been compiled with all libraries necessary to run. It is also stripped, which removes all comments. The bmap binary uses local libraries.




**Conclusion**

Mr. Price is the owner of the program. Files contained in the Docs directory identify John Price as the owner of these documents.

The use of the Ebay picture could not be determined, but Ebay may be used in future activities, such as selling illegally reproduced DVDs.

Prog is a probable part of a hacker's rootkit. One rootkit definition was provided by a Google web search:

> "A hacker security tool that captures passwords and message traffic to and from a computer. A collection of tools that allows a hacker to provide a backdoor into a system, collect information on other systems on the network, mask the fact that the system is compromised, and much more. Rootkit is a classic example of Trojan Horse software. Rootkit is available for a wide range of operating systems." -- http://www.tsl.state.tx.us/ld/pubs/compsecurity/glossary.html"

The program, which must run with a root level id, will utilize unused space of existing programs or files. It does not change the mactime of the file it has stuffed with data, nor will it be detectable. An excellent description of bmap is "Linux Data Hiding and Recovery" by Anton Chuvakin, Phd, 3/10/2002, 11:28 and can be found at URL: http://www.linuxsecurity.com/feature_stories/data-hiding-forensics.html. Bmap does not change mactimes. There are other methods to alter the access times of a file or directory including the *touch* command that allows backdating of a mactime,.but the touch command was not found in the image copy. Prog wrote to the slack space of the file and was wiped from the same slack area. This is the same behavior of bmap.

Automation Projects such as the "Metasploit Project" provide an interface for launching exploits. According to its web content, it is an "advanced open-source platform for developing, testing, and using exploit code". There is no clear indication that the Metasploit Project was used in this project. The intent of this conversation is to make the reader aware there is the development of open-source, commercial, and underground tools that automate the hacking of machine thereby aiding in the *uploading of rootkits to one or many machines*. In this case, the identification of the /bin/sh command is easily found, however the identification of the exploit launched to get root access is not easily found. An educated assumption is that an exploit was launched with some automation. More information on the Metasploit Project can be found at http://www.metasploit.com/.

Now that the binary is identified, a meeting with need to know personnel should be held. At this point, explain to the system or network administrator what the program is and the behavior of the program. Even though this program on this floppy did not work, there is potentially the fully operation version that may have existed on wiped hard drive. That version contained the netcat program which is a program that can do file transfers to other machines, so there is a probability this code has been copied to other systems on the network. You should provide a verbal overview and follow up with a written report containing your forensic activity logs and notes. You should then advise the administrator to open an security incident investigation.

It is time to involve the company Incident Response Team. An IRT has documented procedures on investigating attacks and personnel are trained to review intrusion detection logs and network traffic analysis. They should be aware of key contact

personnel including Human Resources and Legal professionals inside the corporation. This team may be able to stop an attack in progress.

**Interview questions**

Forensic work dictates a series of analysis, creating hypotheses, and validations of these ideas. This keeps the investigator from making conclusions without knowing all the facts. A thorough investigation would include an interview of the computer owner subject. The questions posed to the suspect should have intention to obtain truthful information about the state of the machine and the binaries on the floppy disk.

Questioning a suspect can make him become defensive. Attemp to assure the owner that these questions are intended for an incident investigation and there is no presumption of guilt. The approach should be one that makes the subject feel at ease and provide the answers to the alleged accusations.

The following questions should be asked while interviewing Mr. Price.

*Has there been any system administrator activity on your computer and did you obtain a ticket number associated with a trouble or maintenance call?*

A trouble call to the help desk on the hardware may explain why his computer was recently wiped. If a hard drive is wiped, an operating system should have been installed before returning the computer to the work area. Exceptions to this rule may be company used computer sale for replacement of obsolete equipment. If this is not the case, then Mr. Price appears to have intentionally eliminated potentially incriminating evidence.

*Do any such as coworkers or family members have access to your computer?*

If Mr. Price has a shared corporate computer that does not properly authenticate individual use by unique userid and passwords, it is possible than an unauthorized user could have performed activities posing as Mr. Price. The possibility of family members accessing this machine is not part of the case facts. If Mr. Price were the only user, an investigator would have to check all known users of the system.

*There was a floppy drive that contained several unknown files. Can you tell me about a floppy disk found in your computer and/or what these files may be?*

We want to ask if he has any knowledge of a floppy that contains an unknown binary.

*Have you ever heard of, or used, programs that copy mp3 files?*

Explain to Mr. Price there were documents on the floppy that provided step-by-step information to create mp3s and DVDs. If Mr. Price is aware of such programs, then he may

suggest programs that he has heard of or used. He may inadvertently mention the program name used in this binary.

*Why has your hard drive been cleared?*

Unless there is a new hard drive to be loaded with company approved hardware and software in a standardized load process, there is no reason a hard drive should be wiped. This is highly suspicious, and indicates the need to “destroy evidence”.

## Part 2: Perform Forensic Analysis on a System

A server farm exists that collects statistical information for an old ISP. These servers have been in production mode for approximately three years. The system administrator suddenly was unable to login; a password change had taken place. This event prompted the administrator to contact the corporate and network investigation groups where network based intrusion detection logs were reviewed. Tier 2 support technicians were coincidently studying increased port scanning activity. The goal of this analysis was to identify if a server had been compromised and the extent of the invasion by the intruder. *This report has been sanitized to protect proprietary information.*

Collector 1 was imaged to a 40 GB hard drive. The image was acquired following the corporate chain of custody rules.

Analysis Computer Hardware Description

| | | |
|---|---|---|
| Forensic-Computers.com Computer | Air-Lite IV | Intel Pentium © 4 |
| Hardware: | Maxtor 4G160JB | Type: Disk drives |
| Floppy disk drive | | Type: Floppy disk drives |

Source Drive Image MASSter™ Drive Lock Solo S/N 22551

Analysis Computer System Description

Operating System Information:
Microsoft Windows 2000
Microsoft ® Windows
Version 5.0 (Build 2195: Service Pack 4)
Copyright © 1981-1999 Microsoft Corp.

Physical Memory:
Physical memory available to Windows: 1,015,280KB

Local Disk Properties:
Type: LocalDisk
File system: FAT32

| | | |
|---|---|---|
| Used space: | 9,394,134,016 bytes | 18.0 GB |
| Free space: | 62,550,704,128 bytes | 58.2 GB |
| Capacity: | 81,944,838144 bytes | 76.3 GB |

Software used:

EnCase Version 4.19a
www.EnCase.com
EnCase ® is a registered trademark of Guidance Software. All rights reserved.

Another Screen Capture Tool (ASCT) 1.0.18 By Tony Belcastro/TCB Software
http://www.bhwhost.com//tcb_software
Copyright © 2001 Tony Belcastro

Microsoft ® Word 2000 (9.0.3821 SR- 1)


WinZip 8.1 SR-1 (5266)

www.winzip.com
WinZip License
Evaluation Version

WinHex 10.65 SR-8

For Evaluation Purposes only

Tcpdump
www.tcpdump.org

Red Hat Linux 7.2
Machine Name: LinuxForensics
Same configuration from Part 1.

## Acquire image

The image was acquired as one in a series of sixteen images taken by the local technical forensics expert of the company. Details are provided below. The analysis on the images were divided between forensic team members in three states and the Collector 1 image became my assigned responsibility. Consequently, the image has not been pre-analyzed and is part of the corporate investigation.

Airborne Express delivered the package to the lab. The hard drive was removed from the packaging. After all the accompanying paperwork was reviewed, the hard drive was connected to the Image MASSter Drive lock hardware using the IDE cable with an IDE output cable connector to the Forensic-Computer.com computer as the source. The following information was recorded from the hard drive label: Western Digital WD Caviar ™ Enhanced IDE Hard Drive, Serial Number WMAD11743178, 40.0 GB. The pre-attached tag is labeled Collector 1

The following information is recorded from the “Forensics Investigation Imaging Log” for XYZ Telecom

| | |
|---|---|
| Date: 6/23/04 Time: 11:05 | System Time Zone: |
| System Name: host.xyztel.net | IP Address: nnn.nnn.n.17 |

Serial Number: n/a IP Address:

Function: n/a IP Address

Hardware Description Operating System:

Photograph taken (Y/N)

Administrator/Owner:

System State (On, Off, network active, etc…)
Off – Hard Drive removed and labeled “Collector 1”.

Hard Drive Serial Number: WK139175
Jumpers : : : : - original settings
MD5Sum **0a60b0d3384436a45b384177a87445e5**

Hard Drive Geometry:
IBM Model BTTA-371446E182115S
CHS: 1683/16/63
CBA: 28,229,040 Sectors
14.4 GB

Notes:
BIOS time: +1 second GTC time.

Used original jumper setting for MD5sum on Toshiba Laptop.

Used Image MASSter Solo 2 Pro Serial #22551 with drive lock.
100%,
Full Version,
Ext dir,
Direct:norm,
Identical drive – no,
Autorun - no,
Safe - no,
Bad sector = cont.

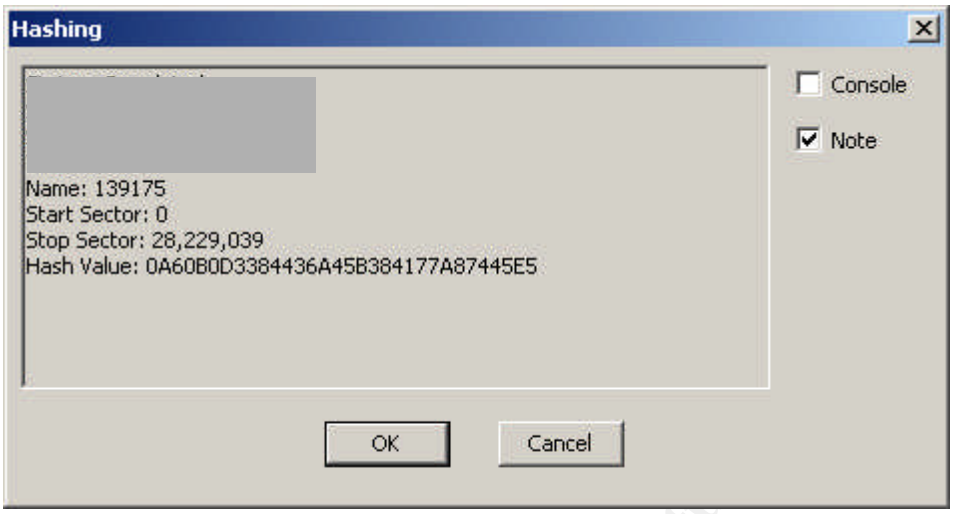
Master IBM Sn#wkowk139175 13784 MB 28229040 sectors to target WD400 SN # WMAD11743178 –38163 MB. Copied 13784 MB in 22 minutes 43 seconds. Success.

Investigator: Greg L. Date: 6/23/04 Time: 12:00

Received “Collector 1 Drive S/N: WK139175 from Greg L.

Name: John B. Date: 6/24/04 Time: 9:00 AM

Confidential Document (when completed) of XYZ Telecom




**Interviews**

A road trip started the investigative process in June 2004 where images were taken of servers on the subnet. Another road trip occurred in July 2004. One team member acquired additional images, while my responsibility was to secure as much information as possible regarding the network. A meeting was arranged with the network and firewall administrators to determine the scope of the network. This network represents the merge of a legacy Internet facing network into a larger network via a frame switched network migrated into the backbone network of another ISP provider. All of the servers on this subnet are in a production mode. This Internet Data Center manages a combination of Class A and Class C IP address ranges that provide managed services to pre-existing customers. The purpose of the infrastructure is to provide remote access services to employees supporting the Telework initiative.

The routers provide scalability as they can also perform firewall capability from broadcast traffic based on layer 3 addressing. The routers will further divide networks and subnets. This depends on the addressing scheme used. The firewall is used to only filter statistical data to the Collector, while the public Internet is used to share data common to all supported customers. Virtual Local area networks (VLANS) provide broadcast containment, communication, and security between VLANS. Additionally, AIX based servers exist on the network. The firewall administrator believes all collector servers run at the same software level.

Other servers on one of the subnets built are known as local interface gateways. They are servers that may use System Network Architecture (SNA) a IBM Corporation proprietary networking protocol,[2] or a Nortel tunnel termination device. They also use a combination of SSL, IPSEC, and PPTP protocols. Some of server boxes terminate using L2TP. There is a mixture of SNA and TCP/IP connectivity that alternate because the servers support clients that may have one of the other protocols in use.

[2] http://www.yale.edu/pclt/COMM/SNA.HTM

There are three types of applications run on these servers: database, collectors, and web servers. The Apache Web servers allow the customers to access self-generating reports. DB2 is the SQL language interface to the database server.

The firewall/network administrator provided hand-delivered network diagrams that were logged and issued a sequenced date and time stamp. The following information was observed about the server.

| | |
|---|---|
| Name: | Collector 1 |
| Desc. | PC |
| Mfg. | AMD |
| Sfw | Red Hat (Linux) 7.1 |
| Public IP: | 192.168.201.81 |

The associated network diagram is represented as Appendix I. This reflects a partial representation of the network.

There are production gateways in multiple geographic locations. Every gateway server at these locations will either be used for production, quality assurance (QA), development, or support.

The collectors reside on the Internet, also called the “dirty net”. They receive statistical information on status data for customer usage. After the collectors get data, they dump the information into a DB2 database, which is managed by the same people that manage the servers. Information regarding the DB2 product can be found at http://www-306.ibm.com/software/data/db2/.

The Web server authenticates via a web URL.

Finally, there is a private Class A network used for backup purposes. According to the administrator, no access exists into the Corporate Intranet. Secure shell (SSH) access is permitted. SSH is described by Rory Krause as “secure Telnet program”, http://www.linuxjournal.com/article.php?sid=5462.

This interview with the administrator provided information on the target server in question to concentrate on for a forensic analysis. Additional meetings with development, production, and QA personnel responsible for applications and server administrators were held. The information provided may prove useful later in the investigation.

**Setup**

The hard drive was powered on using the Solo Image MASSter drive lock, and using the same pin formation as provided by the Forensic Investigator’s log. That is, pins 1 and 5 were capped and pins 2 through 4 were open. EnCase V4.19 was launched from the forensic computer. The following was selected on the EnCase graphical user interface: Selected New: Case Name: Practical Collector 1 Case 2004-Coll-HD, my

examiner's name, and setup location of temporary files to Program Files\EnCase4 directory.Launch EnCase

Selected Specify a new path for device 1 New Case for Device 1\Device\Devices Chose Devices\Local\"1" Local Drive and allowed the program to select the location for the Evidence Files as c:\Program Files\EnCase 4\Choose Devices\Tab

Here is the representative screen of EnCase. The image selected Name 1, label WDC WD400 ASPI. It also verifies that the image is in read only mode.

| Name | Label | Access | Sectors | Size | Write Blocked | Read file system |
|---|---|---|---|---|---|---|
| A | | | 0 | Not ready | | * |
| C | | Windows | 160,087,662 | 76.3 GB | | * |
| 0 | Maxtor 4G16 ASPI | | 268,435,455 | 128 GB | | * |
| 1 | WDC WD400 ASPI | | 78,156,288 | 37.3 GB | | * |

The EnCase main screen opened the case in a left navigation window with a larger window to display the content based on the icon or toolbar request to the right. The left navigation bar expands into a tree structure. Expand the box on device 1 and the right display window exhibits, in tabular format, the directories on this hard drive. The following are the directories on device 1 (in alphabetical order):

| | |
|---|---|
| / | hda8 |
| /boot | hda1 |
| /home | hda6 |
| /usr/ | hda5 |
| /var/ | hda7 |
| swap1/ | |

## Acquisition

There was no need to acquire the file from the hard drive; this disk was the original image created during the actual investigation. Since the image is accessible only in read-only mode, no timestamps will corrupt the image.

EnCase was used to generate cryptographic hash values for each of these directories by right clicking on the directory name and selecting hash.

| Directory name | Hash | Start Sector | Stop Sector |
|---|---|---|---|
| / | AEBD5F4B43CFAEA221535A85ED16F7F5 | 22,928,283 | 22,828,364 |
| /boot | 31A52A78B87180BD2FF45B99F1850799 | 63 | 112,454 |
| /home | 07F9BF53D6237193D4DAA27B7C08E33 | 10,940,328 | 21,768,074 |
| /usr/ | A85C64374DE4D9144174319676A973C2 | 112,518 | 10,940,264 |
| /var/ | 37BB7115869F50DF4B88DABE09630AD | 21,768,138 | 22,298,219 |
| swap1 | DD5671F56349C890241CCBAC4A7E5290 | 22,828,428 | 22,888,654 |
| Unused Disk Space | EnCase would not calculate a hash value | | |

This information was recorded on the 2004-Coll1-HD Fact Sheet. This sheet can be found as Appendix F. The fact sheet collects information about file systems and data structures.

In order to create or display a report showing this data, you must selectively check each box on the left navigator bar. These boxes include the name of the case, the device data information, and all other directories and subdirectories found. Click report on the lower right of the right panel. A report will generate on the bottom panel. Using the right mouse button, click on each directory and export into a newly created and segregated directory c:\My Docments\ Practical Collector 1\EnCase\.

| | | |
|---|---|---|
| Boot directory.rtf | Rich Text Format | hda1 |
| Home Directory.rtf | Rich Text Format | hda6 |
| Physical Disk description.rtf | Rich Text Format | |
| Swap1 directory.rtf | Rich Text Format | |
| Usr directory.rtf | Rich Text Format | |
| Var directory.rtf | Rich Text Format | |

These were copied into the 2004-Coll1-HD Fact sheet.

**Timeline**

This Red Hat 7.1 System has been an active server since 2001. Timelines generated for this system are astronomical in size. The approach towards the timeline will concentrate on the six-month time frame prior to the actual late June 2004 investigation, ranging from January 1 through June 24, 2004.

Beginning on a system of this size requires one to think like a hacker. The operating methods of a hacker will vary depending on the experience of the script kiddie or expert programmer:

> ("…The term is also often used as a derogatory moniker for individuals who do not contribute to the development of new security-related programs, especially exploits, but rather benefit from the work of others".)–script kiddie definition, http://en.wikipedia.org/wiki/Script_kiddie )

The hacks begin with reconnaisance, enumerating users, and exploitation. The next step in the investgation turned towards locating evidence of these activities. The timeline can be viewed in Appendix J.

**Hash Analysis**

A copy of the file NSRL was downloaded to a temporary file on the backup Windows XP computer and burned to a CD/RW. The NSRL 2.0 version can be obtained from ftp://ftp.nist.gov/pub/itl/div897/nsrl/ver_2_0/nsrl_2_0.iso.

To create the equivalent of an MD5deep command, EnCase includes the ability to create hash sets for each binary found; this will not include the lost files. Because of

this EnCase offered to hash 55,992 of 70,270 files. The NSRL will aid in this search by having the known values of these binaries in a text file that reduces the load onto the EnCase program and Windows operating system. Doing large searches and calculations hogs the CPU cycle resources. The hash set was named 139175. View\Hashset|Search allowed the search to compute hashes for all selected files that qualified The captured hash value source (139175) and the category "Known". The NSRL was imported into the program and a search was performed to create all known hash values. The hash set was then sorted.

### Strings

EnCase performs string searches. By selecting the search icon, the analyst can create a new folder and put in any number of strings. Grep, Active Code pages, and UTF –8. . The following keywords were used on this case, but the output does not include strings that are sensitive, such as company names, and internal application names.

| | | | | |
|---|---|---|---|---|
| IP | hack | collector | root | Passwd |
| Sniff | Collect | Bot | Chat | ftp |
| telnet | Named | Echo | Netstat | Linuxconf |
| Chargen | Systat | Finger | Tftp | Rexec |
| Rusers | rlogin | Uucp | Pop | Whoami |
| Suid | Login | Cpio | Netcat | Cron |
| Time | Suid | /etc/passwd | Etc/shadow | chkrootkit |
| Shell | Ssh | Cron | daytime | Tcpdump |
| Ethereal | ADMSniff | exploit | | |

### Web Accesses

EnCase contains a directory of scripts to use. The August 2004 latest scripts were acquired from the Guidance Software EnCase download site. This tool produced over five thousand html pages carved from the image. Through techniques such as sorting by creation, modification times and file size, the analyst may determine if any web pages carved may have been used in the exploitation of this machine. The list of pages was sorted in descending order.

Unfortunately, several pages were accessed by one of the many system administrators. Evidence linked to Hotmail ©, Amazon.com, ClassMates.com that included personal registration numbers.

References to customer reports were found. Due to the sensitivity of this data, they cannot be reproduced in this report.

- Performance
- Latency
- Usage Reports
- Status reports

There are also examples of web pages: Note these are the documents stored on the local image only, because there are no networks connected to the forensic hardware.

- "Pine® - a **P**rogram for **I**nternet **N**ews & **E**mail - is a tool for reading, sending, and managing electronic messages." Information regarding Pine is located at http://www.washington.edu/pine/. Since this organization uses Microsoft Outlook for mail, it doesn't seem likely that Pine should be used.
- Reference documents
    - Ghostscript and device drivers
    - Linux-Pam
    - IBM DB2 Universal Database 7.1 for Linux
    - mod_ssl
    - Linux threads
    - SGML text

### Zip Files

Zip files were found on this image. Two hundred and forty-two zip files appeared when the EnCase query was run that checked for zip files in unallocated space. The thought process here is to determine if any files possessing a *.gz* extension may have been used in the exploit. Browsing several of these files provided unfamiliar content; however Google web searches provided validity of these files as part of any rootkit or exploit. Many of these zip files are members of software install directories.

### Using the WinHex Hexeditor

Using the hexeditor WinHex proved fruitful and ran faster than EnCase. The read only Collector 1 image was opened using File/Open/Hard Disk 2, which is the Collector 1 Hard Drive. WinHex can be found at http://www.sf-soft.de/winhex/index-m.html and is marketed by X-Ways Software Technology AG.

The Search is used on the taskbar to find entries in the dirty word list. On the WinHex select dialog box, the option for entries was 100 and the radio box archive was selected. Once the search is complete on the selected number, these hits are produced in a dialog box that allows the user to go directly to the hit by selecting the offset of the hit. A dirty word list can be found in Appendix E.

- Rootkit – This entry was found at least 77 times. Browsing through these entries produced messages from a commercial host-based policy check manager. However at offset
- /bin/sh (select 200 entries) – This search stopped when it had encountered at least 200 hits. The presence of /bin/sh means someone has use of the system and can run code. Further analysis will determine user and group entries.
- /etc/passwd (select 300 entries) – Hits stopped at 300 entries. These were checked for the probability of someone attempting to see if there were either clear text passwords or to download them for use in a password cracker program
- chkroot (select 100 entries) – This was run to find if someone wanted to determine if their uid was the equivalent of root
- IRC (select 300 entries) – Hits stopped at 300 entries. There were clear instances of IRC, and the libraries were the executables were stored under Linuxconf

- Nmap (select 300 entries) – Had 100 hits before even 1 percent complete. Nmap can be used to do port scans finding open ports. Fortunately under a random sampling of nmap, the majority of occurrences related to the term *unmap*.
- Netcat (select 400 entries) – Found 14 occurrences. To determine if port scanning or file transfers were performed once a hacker was on the network. One occurrence showed the command nc host 515. Port 515 is the line printer daemon (LPD). Memory jogging: There is an LPD vulnerability and can be found at http://www.cert.org/advisories/CA-2001-30.html. This was noted on the log for further investigation.
- Lpd (select 400 entries) – Found 400 hits before five percent complete. Browsing only two of these entries showed they were embedded in scripts.
- The appearance of a password crack dictionary.

Observations of this image have provided a wealth of data. An intruder must obtain password files, cracked passwords, obtain shell permissions, and install rootkits to be successful at his task. Pine mail has been installed, IRC channels have been opened, and nmap may have been used to map the network and obtain IP addresses, and, the machine has been compromised. This might have occurred from the lpd vulnerability. From an internal view, there has been a misuse of company resources and a violation of company policy by a system administrator who used a server to access Internet mail, do online shopping, and browse for non-corporate data. If these observations can provide more truth, then the hypothesis should be correct.

Pine mail was installed. Pine mail is an open source program developed and managed by the University of Washington. It was created in 1989 after their existing mail program was replaced due to business requirements. Unable to find the absolute desired mail replacement, the coders modified and enhanced a program called “Ben”, a user friendly mail program. Further information on Pine can be found at http://www.washington.edu/pine/overview/project-history.html and downloadable versions include and run on Linux, Unix, and pc. Pine is not corporate foundation approved software.

Internet Relay Chat (IRC) is described in RFC 1459. It consists of a server and a client and functions on the TCP Protocol. One definition of IRC found at http://www.free-definition.com/Internet-Relay-Chat.html states IRC

> “is a form of instant communication over the Internet that allows both one-to-one communication and group communication (through rooms of discussion called "Channel"). --

The corporate policy disallows the use of any Internet messaging or chat service. Chat and messaging services are provided by approved software accessible to members of the internal company.

### The search for the login id

The analyst must determine if system directories or files have been changed by knowing the owner of the system directories. Issue the command *cat /etc/passwd* to list userids. Look at the first ids on the system, and determine the userid number and group id number associated with it. This command will also quicky tell the analyst at a

glance if the userids are stored in /etc/shadow. Userids with an x in the second column can only be accessed by the root account when they are in /etc/shadow[3].

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/var/spool/news:
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
gopher:x:13:30:gopher:/var/gopher:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
…
forensics:x:502:502:forensics user:/home/forensics:/bin/bash
```

### The trail: ./bash_history

The /root/.bash_history provides command listings for anyone using the root account. The 9K file contained a mixture of commands that appear to be utilized by the system administrator and the hacker. Active employees were added to the system in a consistent format.

```
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/male6 -s /bin/ksh -c " male full name 6" -m
male6
chmod 640 /opt/male6/.profile
passwd male6
exit
uname
unalias passwd
```

There were legitimate users added on the fly:

```
useradd -c "female3" -m female3
passwd female3
```

Some of the persons are no longer employed with the company, or work from virtual office locations. Yet other individuals added work in the central or northeast region of the United States.

Listed below are notable commands:

[3] http://www.tldp.org/LDP/lame/LAME/linux-admin-made-easy/shadow-file-formats.html

```
netstat-ln
cat (maps, mem, environ, stat, mem,cpuinfo,processes, version
vi (/ec/fstab, /etc/hosts, /etc/messages, /var/log/messages, cron, xinetd.conf
groupadd
useradd
passwd
chmod
echo processor
startx
su
ping
service status xinetd
uptime
ssh
ftp
tar -vxf
make (test, install
gunzip
find / -name syslogd
view syslog.conf
id
telnet nnn.nn.n.nn
ftp nn.nn.n.nnadduser
finger
```

The hacker collected system files, hierarchy, processes, and user information. They started and stopped services. They found paths to other systems and connected using ftp ,ssh, or telnet. Files were copied and cron entries were added. Log files were viewed, and at least four files were copied. Two passwords were reset in this timeframe and the root password was changed.. The /root/.bash_history file can be found in Appendix H.

**The /bin directory**

Browsing the directories in the gui format of EnCase produces a tree presentation of the directories of the captured or acquired image under investigation. The bin directory is the location of binary files used by all users, such as *ls, cat, file*, and *login*. The bin directory owner account is root. This can be verified by referring to the previous discussion of enumerating users. A short list of commands that are found in /bin can be found at http://www.linuxforum.com/linux-filesystem/bin.html.

This is the output of ls –ld /bin from the LinuxForensics laptop.

```
drwxr-xr-x    2 root     root        4096 May  5  2003 /bin
```

In order to determine if directories or files have been changed, again, the analyst should know the owner of the system directories. Root owns the bin directory. The /bin/tmp directory contained the file nn.nnn.12.nn. Once bookmarked, the binary file was extracted and exported to My Documents\Collector\nn-nnn-12-nn.

This binary details:

```
File size:          23.4KB
State:              Original
Exported this file to a floppy disk.
```

At this point, the Linux operating system will be used. The userid under which the following commands were performed is *root*. The binary was then copied to a floppy drive, then to Linux, and placed in an isolated directory under /home/forensics/Sans/Evidence/tmp/part2/binary/nn.nnn.12.nn. Using Linux, an *md5sum* command was issued, and the *file* command produced the following:

```
0158e3703278b6e6bb037a85f9679b9a  nn-nnn-12-nn
nn-nnn-12-nn: ELF 32-bit LSB executable, Intel 80386, version 1, dynamically linked (uses
shared libs), not stripped
```

A Google search on the hash from the md5sum did not provide a match.

*Foremost –c /usr/local/src/foremost-0.69/foremost.conf /home/forensics/tmp/nn-nnn-12-nn* was used to carve out files by headings. There were four files carved out along with the file audit.txt. Within the first three pgp files were the words *team teso*. Teso is a UNIX exploit.

```
[root@LinuxForensics part2]# ./nn-nnn-12-nn
7350logout - sparc|x86/solaris login remote root (version 0.7.0) -sc.
team teso.

usage: ./nn-nnn-12-nn [-h] [-v] [-D] [-p] [-t num] [-a addr] [-d dst]

-h      display this usage
-v      increase verbosity
-D      DEBUG mode
-T      TTYPROMPT mode (try when normal mode fails)
-p      spawn ttyloop directly (use when problem arise)
-t num  select target type (zero for list)
-a a    acp option: set &args[0]. format: "[sx]:0x123"
        (manual offset, try 0x26500-0x28500, in 0x600 steps)
-d dst  destination ip or fqhn (default: 127.0.0.1)
```

A Google search on "*7350logout - sparc|x86/solaris login remote root (version 0.7.0) - sc. team teso"* provided a link to the site http://examples.oreilly.com/networksa/tools/. A preliminary view of the file showed many of the strings to be identical to the ones in nn-nnn-12-nn. The text file was downloaded to a floppy desk for further examination.

Making certain tcpdump was running first, the binary was run from the current working directory. The program executed, quit, and did not connect to any host.

```
[root@LinuxForensics part2]# ./nn-nnn-12-nn -d nn.nnn.12.nn
7350logout - sparc|x86/solaris login remote root (version 0.7.0) -sc.
team teso.
WARNING: target out of list. list:
```

```
num . description
----+-----------------------------------------------------
  1 | Solaris 2.6|2.7|2.8 sparc
  2 | Solaris 2.6|2.7|2.8 x86
    '
[root@LinuxForensics part2]# ./nn-nnn-12-nn -t2 -d nn.nnn.12.nn
7350logout - sparc|x86/solaris login remote root (version 0.7.0) -sc.
team teso.

# using target: Solaris 2.6|2.7|2.8 x86
failed to connect
```

The tcpdump output shows the arp requests to find the gateway out of this network to find the host nn.nnn.12.nn. The executing binary was unable to get a route out provided a "failed to connect" message.

```
[root@LinuxForensics cdrom]# tcpdump
tcpdump: listening on eth0
21:54:51.343283 192.168.1.109.netbios-dgm > 192.168.1.255.netbios-dgm: NBT UDP
PACKET(138) (DF) 21:54:51.343794 192.168.1.109.netbios-dgm >
192.168.1.255.netbios-dgm: NBT UDP PACKET(138) (DF) 21:54:51.820459 arp who-
has 192.168.1.1 tell 192.168.1.109
21:54:52.812998 arp who-has 192.168.1.1 tell 192.168.1.109
… snip
22:12:16.122996 arp who-has 192.168.1.1 tell 192.168.1.109
```

The final analysis on *7350-logout by team teso* did not run because the Collector 1 server is not a Solaris 2.6 through 2.8 x86 architecture. However, the presence of this binary, indicates the toolkit or arsenal of the hacker to use, once inside the network.

The command *strace ./nn-nnn-12-nn* provided different information from that in Part 1 of this document. The first time this binary was run, nothing happened. The investigator reviewed the usage statement again and determined an IP address must follow as a command line parameter.

```
[root@LinuxForensics part2]#[root@LinuxForensics part2]# strace ./nn-
nnn-12-nn
execve("./nn-nnn-12-nn", ["./nn-nnn-12-nn"], [/* 32 vars */]) = 0
uname({sys="Linux", node="LinuxForensics", ...}) = 0 brk(0)
= 0x804cb1c
open("/etc/ld.so.preload", O_RDONLY)    = -1 ENOENT (No such file or
directory)
open("/usr/local/lib/i686/mmx/libc.so.6", O_RDONLY) = -1 ENOENT (No such
file or directory)
stat64("/usr/local/lib/i686/mmx", 0xbfffebbc) = -1 ENOENT (No such file
or directory)
open("/usr/local/lib/i686/libc.so.6", O_RDONLY) = -1 ENOENT (No such
file or directory)
stat64("/usr/local/lib/i686", 0xbfffebbc) = -1 ENOENT (No such file or
directory)
```

```
open("/usr/local/lib/mmx/libc.so.6", O_RDONLY) = -1 ENOENT (No such file
or directory)
stat64("/usr/local/lib/mmx", 0xbfffebbc) = -1 ENOENT (No such file or
directory)

…snip

munmap(0x40017000, 70056)               = 0
write(2, "7350logout - sparc|x86/solaris l"..., 827350logout -
sparc|x86/solaris login remote root (version 0.7.0) -sc.
team teso.

) = 82
write(2, "usage: ./nn-nnn-12-nn [-h] [-v] "..., 71usage: ./nn-nnn-12-nn
[-h] [-v] [-D] [-p] [-t num] [-a addr] [-d dst]

) = 71
write(2, "-h\tdisplay this usage\n-v\tincreas"..., 356-h     display
this usage
-v      increase verbosity
-D      DEBUG mode
-T      TTYPROMPT mode (try when normal mode fails)
-p      spawn ttyloop directly (use when problem arise)
-t num  select target type (zero for list)
-a a    acp option: set &args[0]. format: "[sx]:0x123"
        (manual offset, try 0x26500-0x28500, in 0x600 steps)
-d dst  destination ip or fqhn (default: 127.0.0.1)

) = 356
_exit(1)                                = ?
```

The investigator used the readelf –a command to find out more about the binary.

```
[root@LinuxForensics part2]# readelf -a nn-nnn-12-nn 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:               0x8048a30
  Start of program headers:          52 (bytes into file)
  Start of section headers:          22632 (bytes into file)
  Flags:                             0x0
  Size of this header:               52 (bytes)
  Size of program headers:           32 (bytes)
  Number of program headers:         6
  Size of section headers:           40 (bytes)
  Number of section headers:         30
  Section header string table index: 27
readelf: Error: Unable to read in 236 bytes of (null)
Program Headers:
```

```
  Type           Offset   VirtAddr   PhysAddr   FileSiz MemSiz  Flg
Align
  PHDR           0x000034 0x08048034 0x08048034 0x000c0 0x000c0 R E 0x4
  INTERP         0x0000f4 0x080480f4 0x080480f4 0x00013 0x00013 R   0x1
      [Requesting program interpreter: /lib/ld-linux.so.2]
  LOAD           0x000000 0x08048000 0x08048000 0x03871 0x03871 R E
0x1000
  LOAD           0x003880 0x0804c880 0x0804c880 0x00274 0x0029c RW
0x1000
  DYNAMIC        0x003a54 0x0804ca54 0x0804ca54 0x000a0 0x000a0 RW  0x4
  NOTE           0x000108 0x08048108 0x08048108 0x00020 0x00020 R   0x4

 Section to Segment mapping:
  Segment Sections...
readelf: readelf.c:2571: process_program_headers: Assertion
`string_table != ((void *)0)' failed.
Aborted
[root@LinuxForensics part2]#
```

### Unusual files

```
      06/17/04 05:54:47PM     06/20/04 01:28:23PM          6,457,837   •
      65891
0           hostid@nn.n.1.nn  •          06/21/04 12:45:01PM
```

This file was located in /root/ directory and appeared suspicious. The file was exported from EnCase to a clean floppy. An *md5sum* was taken and the *file* command was issued. After copying the file to the same temporary directores as before, this exploit gave a root shell immediately. Google was searched prior to the execution and no match was found. The command used was *Strings –radix=o nn.n.1.nn > /home/forensics/Sans/Evidence/tmp/part2/nn.n.1.nn-out*. The output of the strings can be found in Appendix G. This local exploit is the one that gives the privileged root shell displayed with # at the prompt. There may be others, but this one definitely worked.

### Conclusion

It has been determined that the Collector 1 image has been compromised. The best solution is to remove this and all associated collectors from the network. These machines should be remediated immediately and undergo thorough reviews before placing them back online.

The data showed there have been exploits and the initial hypothesis was only partially correct. The problems stemmed from faulty userid administration, cracked passwords and local exploits elevated them to root level. Once on the server, the hackers telnet, ftp, or ssh to other hosts on the same subnetand the same activities and toolkits were installed. The status of the other servers is undetermined.

The local systems administrators will have to ensure the system backups also are not contaminated. The operating system may have to be rebuilt.

## Part 3: National and Regional Legal Issues

This addresses the distribution of copyrighted materials via systems available to the public.

Grolier Encyclopedia of Knowledge defines copyright as

> "…the exclusive right to publish and sell the expression embodied in a literary, musical, or artistic work, and of other works that involve original creative effort. In recent years copyright protection has been extended to computer programs and data bases, and copyrightlike protection to semiconductor chips". [4]

According to Grolier, there have been several versions of the copyright law.

- In 1790 the first copyright law passed in United States, built from the British law commonly referred to Statute of Anne. Several changes were made in the 19th century and in the 20th century expanded to include motion pictures, sound recording, and computer programs.
- The U. S. Copyright Statute of 1976 then included statements to include tangible works of authorship.
- The U. S. Copyright Statute of 1988 revised the laws to include protection of the Berne Convention laws and to adhere to them. The Berne Convention dealt with international issues.

Additional changes to the copyright law were found at the web site for the U. S. Copyright Office located at http://www.copyright.gov/title17. The latest version of the copyright is Title 17, United States Code Chapter 5. It now includes semiconductors, original designs, sound recordings, and music videos.

Infringement of copyrights is defined by U. S Code (§) 501 Infringement of copyright, and criminal offenses are covered by U. S Code (§) 506. There is pending legislation introduced 6/22/2004 known as the Inducing Infringement of Copyrights Act of 2004'[5].

Companies have to deal with copyright laws daily. Issues exist as employees load unauthorized or unlicensed software on the corporate computers. Large corporations use open source code sparingly and rely on legally purchased software that inter-operates with hardware and other software components of the business. Introduction of viruses and worms into the intranet cost companies millions of dollars annually. Now any corporation has to create a common office, system, and network operational environment for its workers and clients. The introduction of pirated software also creates a legal and financial loss to the corporation. There are further complications if the creation of pirated software occurs while on corporate facilities.

When proprietary documents are stolen from a corporation the results can damage a corporation name and cause loss of its competive advantage. It also damages the corporate image and can impact its value to stockholders.

[4] Grolier Encyclopedia of Knowledge, p.250-251
[5] http://thomas.loc.gov/cgi-bin/query/z?c108:S.2560:

Kazaa and Napster are two companies that were distributing copyrighted media through the Internet. These file-sharing services are accused of causing millions of dollars to the copyright holders. Software companies claim their losses are into the several billion dollar range.

The is a warning message on the video tapes of the 1970 –1980 era. This message appeared on video motion pictures prior to the start of the movie.

> FBI Warning
>
> Federal law provides severe civil and criminal penalties for the unauthorized reproduction, distribution, or exhibition of copyrighted motion pictures and video tapes (Title 17, United States Code. Sections 501 and 506). The Federal Bureau of Investigation investigates allegations of criminal copyright infringement. (Title 17, United States Code, Section 506).

Now there is a new FBI warning for copyright violations of certain types of media. The warning is an anti-piracy seal that explain new conditions for unauthorized copying and distribution of digital content. Fines can reach $250,000.

The Computer Crime and Intellectual Property Section (CCIPS), of the U.S. Department of Justice provides the method of reporting Internet Crime. Copyright piracy can be reported to the local FBI office, U. S. Customs and Border Patrol Protection local Office, and the Internet Fraud Complaint Center.[6]

Regional Case: http://www.cybercrime.gov/villaIndict.htm

This case began when people complained that a man was selling software claimed to be “new and in the Box”. Postal inspectors corresponded with him and determined that the subject was reproducing and distributing software over the Internet.

Mr. Price: Accused of distributing copyrighted material.

Mr. Price is guilty of copyright infringement if any media contained any licensed software, music (as evidenced by the DVD and mp3 HOWTO documents. If he distributed these documents over the Internet then he has violated the current copyright law identified in Title 17 of the United States Code, Section 506.

The Incident Response policy for the company may suggest the steps to take once an incident has occurred. Here are some suggestions:

- Advise personnel not to take action for perform their own investigations. Consider the incident a crime scene
- Refer all incident related questions to Legal or Security personnel
- Control access to problem management reports and do not include all the details

[6] CCIPS, http://www/cybercrime.gov/reporting.htm

- Provide point of contact information and originator of the incident (this may be the same person)
- Provide as much system specific information, such as hostname, IP address, physical location, network name, date and time

The IRT will verify if an incident has indeed occurred and pre-evaluate the business impact. They will also report a validated incident to Law Enforcement if a crime has been committed.

A forensic analysis is normally performed and has been in the case of Mr. Price. Incident handlers and forensic personnel normally follow chain of custody rules if the incident goes to court. A court case can be lost if preservation of evidence can not be maintained.

If Mr. Price was distributing child pornography, this case would have to go to Law Enforcement immediately. This violates Title 18, United States Code Chapter 110 § 2256.

# Appendix

## Appendix A: Strings found in the fl-160703-jp1.dd image file

(not all inclusive)

lost+found
John
progt
May03
Docs
nc-1.10-16.i386.rpm..rpm
prog
.~5456g.tmp
sect-num.gif
sectors.gif
vmware
vmware-config.pl
vmware
LOGNAME=root
xmms-mpg123-1.2.7-13.i386.rpm..rpmUU
vmware
cd ..
vmware-config.pl
vmware
LOGNAME=root
ebay300.jpg
Letter.doc
Mikemsg.doc
DVD-Playing-HOWTO-html.tar.gzª
Kernel-HOWTO-html.tar.gz
MP3-HOWTO-html.tar.gzµ
Sound-HOWTO-html.tar.gz
DVD-Playing-HOWTO-html.tar
July 14, 2003
14 July, 2003
_PID_GUID
Hey Mike,
I received the latest batch of files last night and I
m ready to rock-n-roll (ha-ha).
I have some advance orders for the next run. Call me soon.
DVD-Playing-HOWTO-html.tar
Sound-HOWTO-html.tar
root
nc-1.10-16.src.rpm
nc
/bin/sh
astest 1027442875
1.10-16
nc-1.10
Changelog
README
scripts
README
alta

dist.sh
iscan
probe
webproxy
webrelay
websearch
nc.1.gz
/usr/bin/
/usr/share/doc/
/usr/share/doc/nc-1.10/
/usr/share/doc/nc-1.10/scripts/
/usr/share/man/man1/
-O2 -march=i386 -mcpu=i686
cpio
gzip
DVD-Playing-HOWTO-1.html
00gferg
mft_getopt
no index
invalid index %d
argv[%d] is NULL
argv[%d] (%s) is not an option
examining a filename or url!
%s is a well-formed argument
checking against %s
flag-
flagized option invokation
examining an enum!
matched against an enum val
examining a venum!
matched against an venum val
arg matches against %s
process_match
true
matches against %s
invalid value for enum
mft_log_init
nbd-server
MFT_LOG_THRESH
none
fatal
error
info
branch
progress
entryexit
mft_log_shutdown
unspecified
enter
exit
operate on ...
target
entryexit
progress
branch
info

error
fatal
none
logging threshold ...
log-thresh
be verbose
verbose
name
useless bogus option
label
write output to ...
outfile
test for fragmentation (returns 0 if file is fragmented)
checkfrag
display fragmentation information for the file
frag
wipe the file from the raw device
print number of bytes available
test (returns 0 if exist)
wipe
place data
display data
extract a copy from the raw device
list sector numbers
operation to perform on files
mode
generate SGML invocation info
sgml
generate man page and exit
display options and exit
help
display version and exit
version
autogenerate document ...
1.0.20 (07/15/03)
newt
use block-list knowledge to perform special operations on files
prog
main
off_t too small!
07/15/03
invalid option: %s
try '--help' for help.
how did we get here?
no filename. try '--help' for help.
target filename: %s
Unable to stat file: %s
%s is not a regular file.
%s has multiple links.
Unable to open file: %s
Unable to determine blocksize
target file block size: %d
unable to raw open %s
Unable to determine count
Unable to allocate buffer
%s has holes in excess of %ld bytes...

error mapping block %d (%s)
nul block while mapping block %d.
seek failure
read error
write error
%s fragmented between %d and %d
%d %s
getting from block %d
file size was: %ld
slack size: %d
block size: %d
seek error
# File: %s  Location: %Ld  size: %d
stuffing block %d
%s has slack
%s does not have slack
%s has fragmentation
%s does not have fragmentation
bmap_get_slack_block
NULL value for slack_block
Unable to stat fd
Unable to determine blocksize
error getting block count
fd has no blocks
mapping block %lu
error mapping block %d. ioctl failed with %s
error mapping block %d. block returned 0
bmap_get_block_count
unable to stat fd
unable to determine filesystem blocksize
filesystem reports 0 blocksize
computed block count: %d
stat reports %d blocks: %d
bmap_get_block_size
bmap_map_block
nul block while mapping block %d.
bmap_raw_open
NULL filename supplied
Unable to stat file: %s
%s is not a regular file.
unable to determine raw device of %s
unable to stat raw device %s
device mismatch 0x%x != 0x%x
unable to open raw device %s
raw fd is %d
bmap_raw_close
/.../image
bogowipe
write error
Wrong medium type
No medium found
Disk quota exceeded
Remote I/O error
Is a named type file
No XENIX semaphores available
Not a XENIX named type file

Structure needs cleaning
Stale NFS file handle
Operation now in progress
Operation already in progress
No route to host
Host is down
Connection refused
Connection timed out
No buffer space available
Connection reset by peer
Network is unreachable
Network is down
Address already in use
Protocol family not supported
Operation not supported
Socket type not supported
Protocol not supported
Protocol not available
Message too long
Destination address required
Too many users
Streams pipe error
Remote address changed
File descriptor in bad state
Name not unique on network
Bad message
RFS specific error
Multihop attempted
Protocol error
Communication error on send
Srmount error
Advertise error
Link has been severed
Object is remote
Package not installed
Machine is not on the network
Out of streams resources
Timer expired
No data available
Device not a stream
Bad font file format
Invalid slot
Invalid request code
No anode
Exchange full
Invalid request descriptor
Invalid exchange
Level 2 halted
No CSI structure available
Protocol driver not attached
Link number out of range
Level 3 reset
Level 3 halted
Level 2 not synchronized
Channel number out of range
Identifier removed

No message of desired type
Directory not empty
Function not implemented
No locks available
File name too long
Resource deadlock avoided
Numerical result out of range
Broken pipe
Too many links
Read-only file system
Illegal seek
No space left on device
File too large
Text file busy
Too many open files
Too many open files in system
Invalid argument
Is a directory
Not a directory
No such device
Invalid cross-device link
File exists
Device or resource busy
Block device required
Bad address
Permission denied
Cannot allocate memory
No child processes
Bad file descriptor
Exec format error
Argument list too long
No such device or address
Input/output error
Interrupted system call
No such process
No such file or directory
Operation not permitted
Success
Too many references: cannot splice
Cannot send after transport endpoint shutdown
Transport endpoint is not connected
Transport endpoint is already connected
Software caused connection abort
Network dropped connection on reset
Cannot assign requested address
Address family not supported by protocol
Protocol wrong type for socket
Socket operation on non-socket
Interrupted system call should be restarted
Invalid or incomplete multibyte or wide character
Cannot exec a shared library directly
Attempting to link in too many shared libraries
.lib section in a.out corrupted
Accessing a corrupted shared library
Can not access a needed shared library
Value too large for defined data type

Too many levels of symbolic links
Numerical argument out of domain
Inappropriate ioctl for device
Resource temporarily unavailable
,ccs=
TOP_PAD_
MMAP_MAX_
TRIM_THRESHOLD_
MMAP_THRESHOLD_
Arena %d:
system bytes     = %10u
in use bytes     = %10u
Total (incl. mmap):
max mmap regions = %10u
max mmap bytes   = %10lu
malloc: top chunk is corrupt
free(): invalid pointer %p!
malloc: using debugging hooks
realloc(): invalid pointer %p!
Unknown error
/etc/suid-debug
MALLOC_CHECK_
/proc/sys/kernel/osrelease
FATAL: kernel too old
FATAL: cannot determine library version
/usr/lib/gconv
/etc/localtime
Universal
/proc/self/cwd
/proc
/etc/mtab
/etc/fstab
proc
/cpuinfo
processor
/meminfo
MemTotal: %ld kB
MemFree: %ld kB
/lib/
/usr/lib/
symbol=%s;  lookup in file=%s
file=%s;  needed by %s (relocation dependency)
binding file %s to %s: %s symbol `%s'
downloads
Reads and writes data across network connections using TCP or UDP.
The nc package contains Netcat (the program is actually nc), a simple
utility for reading and writing data across network connections, using
the TCP or UDP protocols. Netcat is intended to be a reliable back-end
tool which can be used directly or easily driven by other programs and
scripts.  Netcat is also a feature-rich network debugging and
exploration tool, since it can create many different connections and
has many built-in capabilities.
You may want to install the netcat package if you are administering a
network and you'd like to use its debugging and network exploration
capabilities.
"astest

## Appendix B: Strings associated with compiled binary

```
/lib/ld-linux.so.2
libc.so.6
strcpy
ioctl
stdout
strerror
sys_errlist
getenv
malloc
vsnprintf
fflush
calloc
write
fprintf
sys_nerr
dprintf
read
openlog
strncmp
strcasecmp
strdup
memset
open64
syslog
strcmp
stderr
__errno_location
exit
atoi
_IO_stdin_used
__libc_start_main
strlen
lseek64
toupper
close
free
__fxstat64
__lxstat64
__gmon_start__
GLIBC_2.2
GLIBC_2.1
GLIBC_2.3
GLIBC_2.0
PTRh
QVh0
8-ts
RPSQ
RPSQ
I,RPSQ
p8hx
wF;U
wF;U
mft_getopt
no index
```

```
invalid index %d
argv[%d] is NULL
argv[%d] (%s) is not an option
examining a filename or url!
%s is a well-formed argument
checking against %s
flag-
flagized option invokation
examining an enum!
matched against an enum val
examining a venum!
matched against an enum val
arg matches against %s
process_match
true
matches against %s
invalid value for enum
mft_log_init
nbd-server
MFT_LOG_THRESH
none
fatal
error
info
branch
progress
entryexit
mft_log_shutdown
unspecified
enter
exit
%s: %s
violet
blue
green
yellow
orange
white
%s: %s
<table bgcolor=%s><tr><td>%s: %s</td></tr></table><br>
<table bgcolor=%s><tr><td>%s</td></tr></table><br>
<table bgcolor=%s><tr><td></td></tr></table><br>
Brazil
.TH %s "%d" "%s" "%s" "%s"
.SH NAME
%s \- %s
.SH SYNOPSIS
.B %s
[\fIOPTION\fR]...
.SH DESCRIPTION
\fB\-\-%s\fR %s
\fB\-\-%s\fR \fIARG\fR %s
\fB\-\-%s\fR \fIINT\fR %s
\fB\-\-%s\fR \fIFILENAME\fR %s
\fB\-\-%s\fR \fIVALUE\fR %s
\fIVALUE\fR can be one of:
      \fB%s\fR
```

```
 | \fB%s\fR
      \fBSHORTHAND INVOKATION:\fR
Any of the valid values for \fB--%s\fR can be supplied directly as options.
For instance, \fB--%s\fR can be used in place of \fB--%s=%s\fR.
      \fB%s\fR %s
--%s %s
.SH REPORTING BUGS
Report bugs to %s.
Usage: %s [OPTION]...
 [<%s-filename>]
--%s  %s
--%s <arg> %s
--%s <int> %s
--%s <filename> %s
--%s <
 | %s
> %s
--%s VALUE
  where VALUE is one of:
  %s  %s
<tt>%s</tt> invocation
<tt>%s [&lt;OPTIONS&gt;]
 [&lt;%s-filename&gt;]
</tt>
Where <bf>OPTIONS</bf> may include any of:
<descrip>
<tag>--%s</tag>   %s
<tag>--%s &lt;arg&gt;</tag> %s
<tag>--%s &lt;int&gt;</tag> %s
<tag>--%s &lt;filename&gt;</tag> %s
<tag>--%s &lt;
&gt;</tag> %s
<tag>--%s VALUE</tag>
<tag>%s</tag>  %s
</descrip>
<tag>--%s</tag> %s
%s:%s %s
autogenerate document ...
mode
operation to perform on files
outfile
write output to ...
label
useless bogus option
name
verbose
be verbose
log-thresh
logging threshold ...
target
operate on ...
bmap
use block-list knowledge to perform special operations on files
newt@scyld.com
1.0.20 (09/20/04)
main
09/20/04
```

```
invalid option: %s
try '--help' for help.
how did we get here?
no filename. try '--help' for help.
target filename: %s
Unable to stat file: %s
%s is not a regular file.
%s has multiple links.
Unable to open file: %s
Unable to determine blocksize
target file block size: %d
unable to raw open %s
Unable to determine count
Unable to allocate buffer
%s has holes in excess of %ld bytes...
error mapping block %d (%s)
nul block while mapping block %d.
seek failure
read error
write error
%s fragmented between %d and %d
%d %s
getting from block %d
file size was: %ld
slack size: %d
block size: %d
seek error
# File: %s  Location: %Ld  size: %d
stuffing block %d
%s has slack
%s does not have slack
%s has fragmentation
%s does not have fragmentation
version
display version and exit
help
display options and exit
generate man page and exit
sgml
generate SGML invocation info
list sector numbers
carve
extract a copy from the raw device
slack
display data in slack space
putslack
place data into slack
wipeslack
wipe slack
checkslack
test for slack (returns 0 if file has slack)
slackbytes
print number of slack bytes available
wipe
wipe the file from the raw device
frag
display fragmentation information for the file
```

```
checkfrag
test for fragmentation (returns 0 if file is fragmented)
none
fatal
error
info
branch
progress
entryexit
bmap_get_slack_block
NULL value for slack_block
Unable to stat fd
Unable to determine blocksize
error getting block count
fd has no blocks
mapping block %lu
error mapping block %d. ioctl failed with %s
error mapping block %d. block returned 0
bmap_get_block_count
unable to stat fd
unable to determine filesystem blocksize
filesystem reports 0 blocksize
computed block count: %d
stat reports %d blocks: %d
bmap_get_block_size
bmap_map_block
nul block while mapping block %d.
bmap_raw_open
NULL filename supplied
Unable to stat file: %s
%s is not a regular file.
unable to determine raw device of %s
unable to stat raw device %s
device mismatch 0x%x != 0x%x
unable to open raw device %s
raw fd is %d
bmap_raw_close
/.../image
bogowipe
write error
/dev/md10
/dev/md0
/dev/md1
/dev/md2
** edited for brevity **
/dev/sdaq2
/dev/sdaq3


/dev/xdb9
A. K.
A. K.
```

## Appendix C: Timeline:fl-160703-jp1.dd

| Timeline of fl-160703-jp1 | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Date | Time | Size | M | A | C | Permissions | UID | GID | Inode | |
| Tue Jan 28 2003 | 10:56:00 | 20680 | m | a | . | -/-rwxr-xr-x | 502 | 502 | 25 | /John/sectors.gif |
| | | 19088 | m | a | . | -/-rwxr-xr-x | 502 | 502 | 24 | /John/sect-num.gif |
| Mon Feb 03 2003 | 06:08:00 | 1024 | m | . | . | d/drwxr-xr-x | 502 | 502 | 12 | /John |
| Sat May 03 2003 | 06:10:00 | 1024 | m | . | . | d/drwxr-xr-x | 502 | 502 | 14 | /May03 |
| Wed May 21 2003 | 06:09:00 | 27430 | m | a | . | -/-rwxr-xr-x | 502 | 502 | 19 | /Docs/Kernel-HOWTO-html.tar.gz |
| | | 29184 | m | a | . | -/-rwxr-xr-x | 502 | 502 | 13 | /Docs/DVD-Playing-HOWTO-html.tar |
| Wed May 21 2003 | 06:12:00 | 32661 | m | a | . | -/-rwxr-xr-x | 502 | 502 | 20 | /Docs/MP3-HOWTO-html.tar.gz |
| Wed Jun 11 2003 | 09:09:00 | 29696 | m | a | . | -/-rw------- | 502 | 502 | 16 | /Docs/Letter.doc |
| Mon Jul 14 2003 | 10:08:09 | 12288 | m | . | c | d/drwx------ | 0 | 0 | 11 | /lost+found |
| | | 0 | m | a | c | ---------- | 0 | 0 | 1 | <fl-160703-jp1.dd-alive-1> |
| Mon Jul 14 2003 | 10:11:50 | 26843 | m | a | . | -/-rwxr-xr-x | 502 | 502 | 21 | /Docs/Sound-HOWTO-html.tar.gz |
| Mon Jul 14 2003 | 10:12:02 | 56950 | m | a | . | -/-rwxr-xr-x | 502 | 502 | 22 | /nc-1.10-16.i386.rpm..rpm |
| Mon Jul 14 2003 | 10:12:15 | 100430 | m | a | . | -rwxr-xr-x | 0 | 0 | 23 | <fl-160703-jp1.dd-dead-23> |
| Mon Jul 14 2003 | 10:12:48 | 13487 | m | a | . | -/-rwxr-xr-x | 502 | 502 | 26 | /May03/ebay300.jpg |
| Mon Jul 14 2003 | 10:13:13 | 546116 | m | . | . | -rwxr-xr-x | 502 | 502 | 27 | <fl-160703-jp1.dd-dead-27> |
| Mon Jul 14 2003 | 10:13:52 | 2592 | m | . | c | -/-rw-r--r-- | 0 | 0 | 28 | /.~5456g.tmp |
| Mon Jul 14 2003 | 10:19:13 | 100430 | . | . | c | -rwxr-xr-x | 0 | 0 | 23 | <fl-160703-jp1.dd-dead-23> |
| Mon Jul 14 2003 | 10:22:36 | 1024 | m | . | . | d/drwxr-xr-x | 502 | 502 | 15 | /Docs |
| Mon Jul 14 2003 | 10:24:00 | 487476 | m | . | . | -/-rwxr-xr-x | 502 | 502 | 18 | /prog |
| Mon Jul 14 2003 | 10:43:44 | 26843 | . | . | c | -/-rwxr-xr-x | 502 | 502 | 21 | /Docs/Sound-HOWTO-html.tar.gz |
| | | 1024 | . | . | c | d/drwxr-xr-x | 502 | 502 | 15 | /Docs |
| Mon Jul 14 2003 | 10:43:53 | 13487 | . | . | c | -/-rwxr-xr-x | 502 | 502 | 26 | /May03/ebay300.jpg |
| Mon Jul 14 2003 | 10:43:57 | 56950 | . | . | c | -/-rwxr-xr-x | 502 | 502 | 22 | /nc-1.10-16.i386.rpm..rpm |
| Mon Jul 14 2003 | 10:45:48 | 29184 | . | . | c | -/-rwxr-xr-x | 502 | 502 | 13 | /Docs/DVD-Playing-HOWTO-html.tar |
| Mon Jul 14 2003 | 10:46:00 | 27430 | . | . | c | -/-rwxr-xr-x | 502 | 502 | 19 | /Docs/Kernel-HOWTO-html.tar.gz |
| Mon Jul 14 2003 | 10:46:07 | 32661 | . | . | c | -/-rwxr-xr-x | 502 | 502 | 20 | /Docs/MP3-HOWTO-html.tar.gz |
| Mon Jul 14 2003 | 10:47:10 | 546116 | . | a | . | -rwxr-xr-x | 502 | 502 | 27 | <fl-160703-jp1.dd-dead-27> |
| Mon Jul 14 2003 | 10:47:57 | 29696 | . | . | c | -/-rw------- | 502 | 502 | 16 | /Docs/Letter.doc |
| Mon Jul 14 2003 | 10:48:15 | 19456 | m | a | c | -/-rw------- | 502 | 502 | 17 | /Docs/Mikemsg.doc |
| Mon Jul 14 2003 | 10:48:53 | 19088 | . | . | c | -/-rwxr-xr-x | 502 | 502 | 24 | /John/sect-num.gif |
| | | 20680 | . | . | c | -/-rwxr-xr-x | 502 | 502 | 25 | /John/sectors.gif |
| Mon Jul 14 2003 | 10:49:25 | 1024 | . | . | c | d/drwxr-xr-x | 502 | 502 | 12 | /John |
| Mon Jul 14 2003 | 10:50:15 | 1024 | . | . | c | d/drwxr-xr-x | 502 | 502 | 14 | /May03 |
| Wed Jul 16 2003 | 02:03:00 | 546116 | . | . | c | -rwxr-xr-x | 502 | 502 | 27 | <fl-160703-jp1.dd-dead-27> |
| Wed Jul 16 2003 | 02:03:13 | 1024 | m | . | c | -/drwxr-xr-x | 0 | 0 | 2 | /John/ (deleted-realloc) |
| Wed Jul 16 2003 | 02:05:33 | 487476 | . | . | c | -/-rwxr-xr-x | 502 | 502 | 18 | /prog |
| Wed Jul 16 2003 | 02:06:15 | 12288 | . | a | . | d/drwx------ | 0 | 0 | 11 | /lost+found |

| Date | Time | Size | M | A | C | Permissions | UID | GID | Inode | |
|---|---|---|---|---|---|---|---|---|---|---|
| Wed Jul 16 2003 | 02:09:35 | 1024 | . | a | . | d/drwxr-xr-x | 502 | 502 | 12 | /John |
| Wed Jul 16 2003 | 02:09:49 | 1024 | . | a | . | d/drwxr-xr-x | 502 | 502 | 14 | /May03 |
| Wed Jul 16 2003 | 02:10:01 | 1024 | . | a | . | d/drwxr-xr-x | 502 | 502 | 15 | /Docs |
| Wed Jul 16 2003 | 02:11:36 | 2592 | . | a | . | -/-rw-r--r-- | 0 | 0 | 28 | /.~5456g.tmp |
| Wed Jul 16 2003 | 02:12:39 | 1024 | . | a | . | -/drwxr-xr-x | 0 | 0 | 2 | /John/ (deleted-realloc) |
| Wed Jul 16 2003 | 02:12:45 | 487476 | . | a | . | -/-rwxr-xr-x | 502 | 502 | 18 | /prog |

## Appendix D: Dirty Word List – fl-160703-jp1.dd

| String | Hits |
|---|---|
| %d | 42 |
| %s | 238 |
| ./ | 19 |
| 00gferg | 9 |
| a.out | 1 |
| address | 15 |
| bin | 7 |
| bin/sh | 1 |
| bmap | 6 |
| cat | 33 |
| chmod | |
| connect | 8 |
| debug | 4 |
| dev | 3334 |
| dump | |
| echo | 1 |
| elf | |
| etc | 15 |
| gid | |
| group | 0 |
| gz | 11 |
| hack | |
| history | 0 |
| hop | 2 |
| install | 11 |
| log | 10 |
| login | 1 |
| lost | 2 |
| mkdir | |

| String | Hits |
|---|---|
| syn | 1 |
| telnet | |
| tmp | 4 |
| touch | |
| uid | 1 |
| var/log | 0 |
| vmware | 6 |
| wipe | 3 |
| xmms | 1 |
| mpeg2dec | 9 |
| nc | 74 |
| nc.1.gz | 1 |
| newt | 1 |
| null | 3 |
| prog | 18 |
| rexec | |
| rhosts | |
| rlogin | |
| root | 38 |
| rshell | |
| slack | 5 |
| slack space | 0 |
| sniff | 0 |
| ssh | |

**Appendix E: Dirty Word List – Collector 1**

| String | Hits | String | Hits |
|---|---|---|---|
| IP | | hack | |
| Sniff | | Collect | |
| telnet | | Named | |
| Chargen | | Systat | |
| Rusers | | rlogin | |
| Suid | | Login | |
| Time | | Suid | |
| Shell | | Ssh | |
| root | | Passwd | |
| Chat | | ftp | |
| Netstat | | Linuxconf | |
| Tftp | | Rexec | |
| Pop | | Whoami | |
| Netcat | 14 | Cron | |
| Etc/shadow | | Rootkit | 77 |
| daytime | | Bin/sh | > 200 |
| collector | | chkroot | 6 |
| Bot | | IRC | > 300 |
| Echo | | Nmap | > 300 |
| Finger | | | |
| Uucp | | | |
| Cpio | | | |
| /etc/passwd | > 300 | | |
| Cron | | | |

## Appendix F: 2004-Coll1-HD Fact sheet

XYZ Telecom:
This information was recorded on File Structure

| Directory name | Hash | Start Sector | Stop Sector |
|---|---|---|---|
| / | AEBD5F4B43CFAEA221535A85ED16F7F5 | 22,928,283 | 22,828,364 |
| /boot | 31A52A78B87180BD2FF45B99F1850799 | 63 | 112454 |
| /home | 07F9BF53D6237193D4DAA27B7C08E33 | 10,940,328 | 21,768,074 |
| /usr/ | A85C64374DE4D9144174319676A973C2 | 112,518 | 10,940,264 |
| /var/ | 37BB7115869F50DF4B88DABE09630AD | 21,768,138 | 22,298,219 |
| swap1 | DD5671F56349C890241CCBAC4A7E5290 | 22,828,428 | 22,888,654 |
| Unused Disk Space | EnCase would not calculate a hash value | | |

| Description | EnCase Exported Report to Directory |
|---|---|
| Physical Disk description.rtf | |
| / | Root directory.rtf |
| /boot | Boot directory.rtf |
| /home | Home Directory.rtf |
| /usr/ | Usr directory.rtf |
| /var/ | Var directory.rtf |
| swap1 | Swap1 directory.rtf |
| Unused Disk Space | |

This is the description of the ***/* directory**

| | |
|---|---|
| Name: | / |
| Description: | Volume, Sector 22298283-22828364, 258.8MB |
| Last Accessed: | 06/22/04 12:02:03AM |
| Last Written: | 05/23/03 11:23:57AM |
| Entry Modified: | 05/23/03 11:23:57AM |
| Logical Size: | 1,024 |
| Physical Size: | 1,024 |
| Starting Extent: | 0/-C257 |
| File Extents: | 1 |
| Physical Location: | 11,416,984,064 |
| Evidence File: | 1 |
| File Identifier: | 2 |
| Full Path: | Practical Collector 1 - Case 2004-Coll-HD\1\/ |
| Short Name: | hda8 |
| Original Path: | Practical Collector 1 - Case 2004-Coll-HD\1\/ |
| File Extents | |

| Start Sector | Sectors | Start Cluster | Clusters |
|---|---|---|---|
| 22,298,797 | 2 | 257 | 1 |

**Volume**

| | | | |
|---|---|---|---|
| File System: | EXT2 | Drive Type: | Fixed |
| Sectors per cluster: | 2 | Bytes per sector: | 512 |
| Total Sectors: | 530,082 (258.8MB) | Total Capacity: | 271,401,984 bytes |
| Total Clusters: | 265,041 (196.7MB) | Unallocated: | 206,292,992 bytes |
| Free Clusters: | 201,458 (62.1MB) | Allocated: | 65,108,992 bytes |
| Volume Name: | | Volume Offset: | 22,298,283 |

This is the description of the **/boot directory**

| | |
|---|---|
| Name: | /boot |
| Description: | Volume, Sector 63-112454, 54.9MB |
| Last Accessed: | 06/22/04 12:02:03AM |
| Last Written: | 06/21/04 12:27:52PM |
| Entry Modified: | 06/21/04 12:27:52PM |
| Logical Size: | 1,024 |
| Physical Size: | 1,024 |
| Starting Extent: | 0/boot-C256 |
| File Extents: | 1 |
| Physical Location: | 294,400 |
| Evidence File: | 1 |
| File Identifier: | 2 |
| Full Path: | Practical Collector 1 - Case 2004-Coll-HD\1\/boot |
| Short Name: | hda1 |
| Original Path: | Practical Collector 1 - Case 2004-Coll-HD\1\/boot |

File Extents

| Start Sector | | Sectors | Start Cluster | Clusters |
|---|---|---|---|---|
| 575 | 2 | 256 | 1 | |

**Volume**

| | | | |
|---|---|---|---|
| File System: | EXT2 | Drive Type: | Fixed |
| Sectors per cluster: | 2 | Bytes per sector: | 512 |
| Total Sectors: | 112,392 (54.9MB) | Total Capacity: | 57,544,704 bytes |
| Total Clusters: | 56,196 | Unallocated: | 52,153,344 bytes (49.7MB) |
| Free Clusters: | 50,931 (5.1MB) | Allocated: | 5,391,360 bytes |
| Volume Name: | | Volume Offset: | 63 |

This is the description of the **/home directory**:

Name: /home

| | |
|---|---|
| Description: | Volume, Sector 10940328-21768074, 5.2GB |
| Last Accessed: | 06/22/04 12:02:03AM |
| Last Written: | 01/23/04 12:49:12AM |
| Entry Modified: | 01/23/04 12:49:12AM |
| Logical Size: | 4,096 |
| Physical Size: | 4,096 |
| Starting Extent: | 0/home-C508 |
| File Extents: | 1 |
| Physical Location: | 5,603,528,704 |
| Evidence File: | 1 |
| File Identifier: | 2 |
| Full Path: | Practical Collector 1 - Case 2004-Coll-HD\1\/home |
| Short Name: | hda6 |
| Original Path: | Practical Collector 1 - Case 2004-Coll-HD\1\/home |

File Extents

| Start Sector | Sectors | Start Cluster | Clusters |
|---|---|---|---|
| 10,944,392 | 8 | 508 | 1 |

**Volume**

| | | | |
|---|---|---|---|
| File System: | EXT2 | Drive Type: | Fixed |
| Sectors per cluster: | 8 | Bytes per sector: | 512 |
| Total Sectors: | 10,827,747 | Total Capacity: | 5,543,804,928 bytes (5.2GB) |
| Total Clusters: | 1,353,468 (3.6GB) | Unallocated: | 3,918,491,648 bytes |
| Free Clusters: | 956,663 (1.5GB) | Allocated: | 1,625,313,280 bytes |
| Volume Name: | | Volume Offset: | 10,940,328 |

This is the description of the **/usr directory**:

| | |
|---|---|
| Name: | /usr |
| Description: | Volume, Sector 112518-10940264, 5.2GB |
| Last Accessed: | 06/22/04 12:02:06AM |
| Last Written: | 12/15/03 02:47:54PM |
| Entry Modified: | 12/15/03 02:47:54PM |
| Logical Size: | 4,096 |
| Physical Size: | 4,096 |
| Starting Extent: | 0/usr-C508 |
| File Extents: | 1 |
| Physical Location: | 59,689,984 |
| Evidence File: | 1 |
| File Identifier: | 2 |
| Full Path: | Practical Collector 1 - Case 2004-Coll-HD\1\/usr |
| Short Name: | hda5 |
| Original Path: | Practical Collector 1 - Case 2004-Coll-HD\1\/usr |

File Extents

| Start Sector | Sectors | Start Cluster | Clusters |
|---|---|---|---|
| 116,582 8 | 508 | 1 | |

**Volume**

| | | | |
|---|---|---|---|
| File System: | EXT2 | Drive Type: | Fixed |
| Sectors per cluster: | 8 | Bytes per sector: | 512 |
| Total Sectors: | 10,827,747 (5.2GB) | Total Capacity: | 5,543,804,928 bytes |
| Total Clusters: | 1,353,468 (1.3GB) | Unallocated: | 1,428,090,880 bytes |
| Free Clusters: | 348,655 (3.8GB) | Allocated: | 4,115,714,048 bytes |
| Volume Name: | | Volume Offset: | 112,518 |

This is the description of the **/var directory**:

| | |
|---|---|
| Name: | /var |
| Description: | Volume, Sector 21768138-22298219, 258.8MB |
| Last Accessed: | 06/22/04 12:02:34AM |
| Last Written: | 10/25/02 09:20:34AM |
| Entry Modified: | 10/25/02 09:20:34AM |
| Logical Size: | 1,024 |
| Physical Size: | 1,024 |
| Starting Extent: | 0/var-C257 |
| File Extents: | 1 |
| Physical Location: | 11,145,549,824 |
| Evidence File: | 1 |
| File Identifier: | 2 |
| Full Path: | Practical Collector 1 - Case 2004-Coll-HD\1\/var |
| Short Name: | hda7 |
| Original Path: | Practical Collector 1 - Case 2004-Coll-HD\1\/var |

File Extents

| Start Sector | Sectors | Start Cluster | Clusters |
|---|---|---|---|
| 21,768,652 | 2 | 257 | 1 |

**Volume**

| | | | |
|---|---|---|---|
| File System: | EXT2 | Drive Type: | Fixed |
| Sectors per cluster: | 2 | Bytes per sector: | 512 |
| Total Sectors: | 530,082 (258.8MB) | Total Capacity: | 271,401,984 bytes |
| Total Clusters: | 265,041 (219.1MB) | Unallocated: | 229,775,360 bytes |
| Free Clusters: | 224,390 (39.7MB) | Allocated: | 41,626,624 bytes |
| Volume Name: | | Volume Offset: | 21,768,138 |

This is the description of the **unused Disk area**:

| | |
|---|---|
| Name: | Unused Disk Area |
| Logical Size: | 27,785,221,120 |
| Physical Size: | 27,785,221,120 |
| Starting Extent: | 0S1 |
| File Extents: | 7 |
| Physical Location: | 512 |
| Evidence File: | 1 |
| Full Path: | Practical Collector 1 - Case 2004-Coll-HD\1\Unused Disk Area |

File Extents

| Start Sector | Sectors | Start Cluster | Clusters |
|---|---|---|---|
| 1 | 62 | | |
| 112,455 | 63 | | |
| 10,940,265 | 63 | | |
| 21,768,075 | 63 | | |
| 22,298,220 | 63 | | |
| 22,828,365 | 63 | | |
| 23,888,655 | 54,267,633 | | |

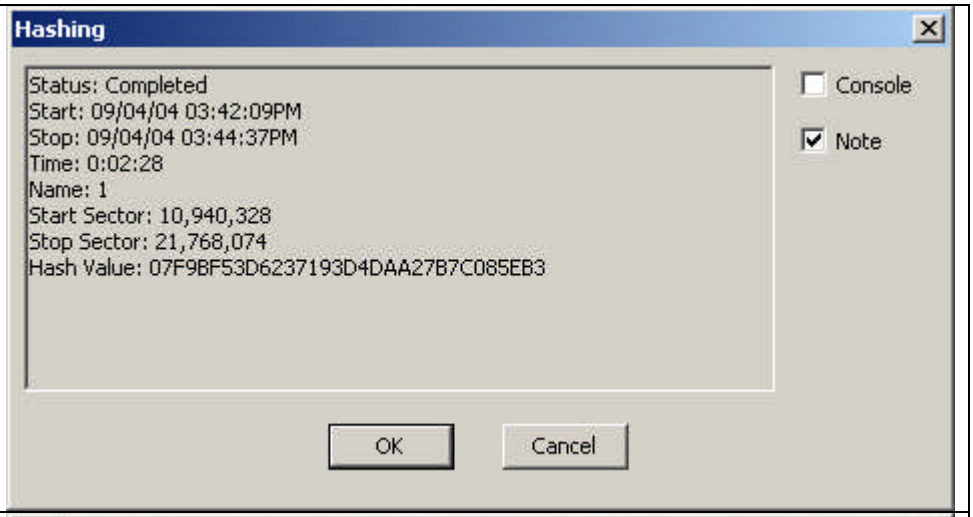
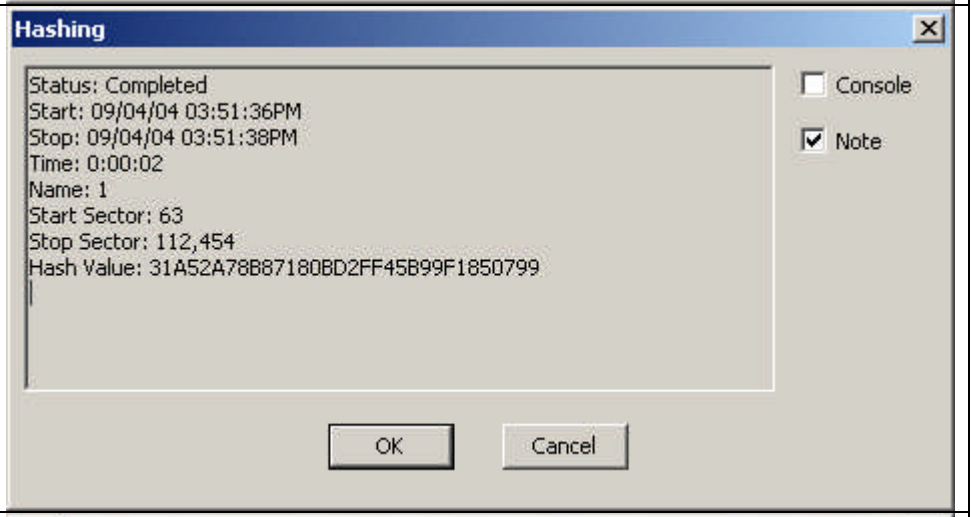
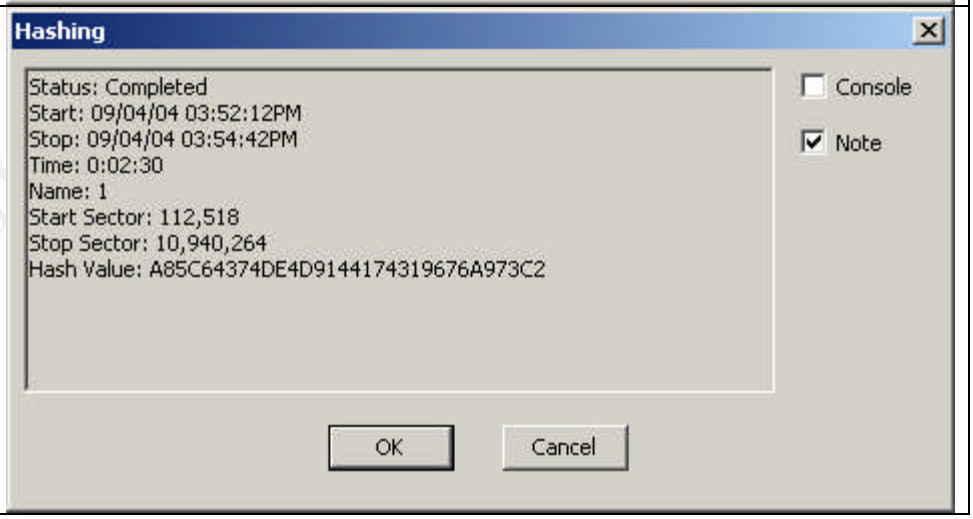
This is the description of the **swap1 Drectory**

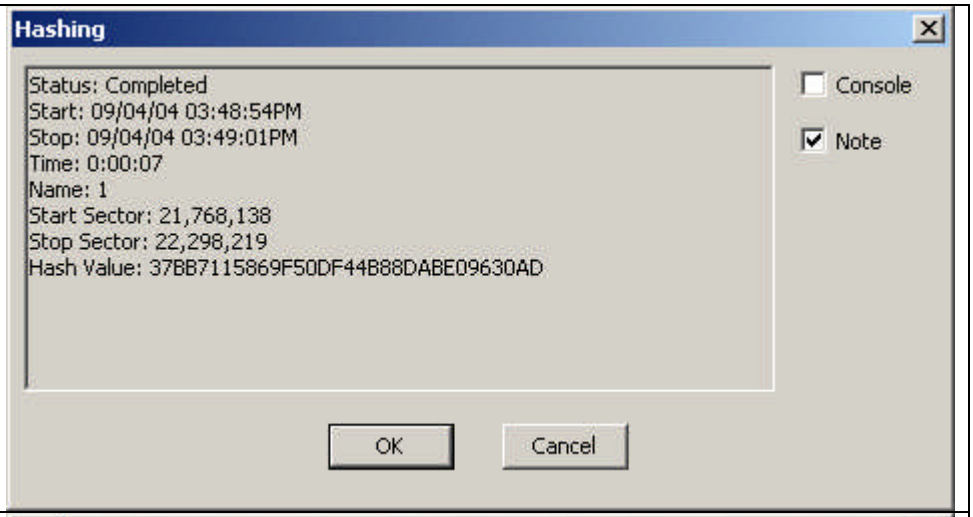
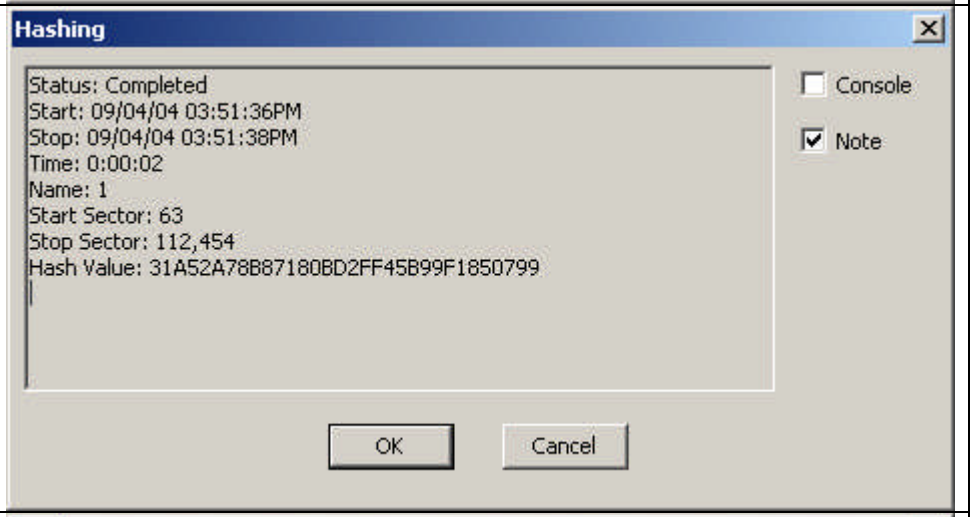
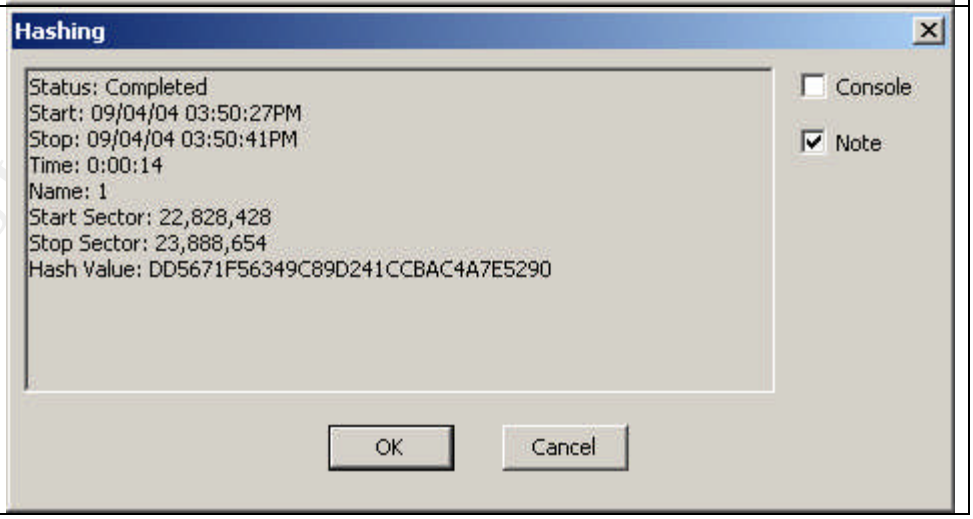
| | |
|---|---|
| Name: | swap1 |
| Description: | Volume, Sector 22828428-23888654, 517.7MB |
| Logical Size: | |
| Physical Size: | 0 |
| Evidence File: | 1 |
| Full Path: | Practical Collector 1 - Case 2004-Coll-HD\1\swap1 |

**Volume**

| | | | |
|---|---|---|---|
| File System: | Unknown | Drive Type: | Fixed |
| Sectors per cluster: | 1 | Bytes per sector: | 512 |
| Total Sectors: | 1,060,227 (517.7MB) | Total Capacity: | 542,836,224 bytes |
| Total Clusters: | 1,060,227 (517.7MB) | Unallocated: | 542,836,224 bytes |
| Free Clusters: | 1,060,227 | Allocated: | 0 bytes (0 bytes) |
| Volume Name: | | Volume Offset: | 22,828,428 |

HASH VALUES

| | |
|---|---|
| Home |  |
| /boot |  |
| /usr |  |

| | |
|---|---|
| /var |  |
| / |  |
| /swap1 |  |

## Appendix G: Strings from nn-nnn-12-nn binary

```
/lib/ld-linux.so.2
__gmon_start__
libc.so.6
printf
connect
memmove
usleep
memcpy
perror
malloc
sleep
optarg
socket
select
fflush
send
calloc
write
fprintf
inet_addr
__deregister_frame_info
read
memcmp
sscanf
getopt
memset
getchar
gethostbyname
stderr
getsockopt
htons
__errno_location
exit
_IO_stdin_used
__libc_start_main
strlen
fcntl
__register_frame_info
close
free
GLIBC_2.0
Solaris 2.6|2.7|2.8 x86
Solaris 2.6|2.7|2.8 sparc
Manual target sparc
Manual target x86
127.0.0.1
usage: %s [-h] [-v] [-D] [-p] [-t num] [-a addr] [-d dst]
-h    display this usage
-v    increase verbosity
-D    DEBUG mode
-T    TTYPROMPT mode (try when normal mode fails)
-p    spawn ttyloop directly (use when problem arise)
-t num      select target type (zero for list)
-a a  acp option: set &args[0]. format: "[sx]:0x123"
```

```
      (manual offset, try 0x26500-0x28500, in 0x600 steps)
-d dst      destination ip or fqhn (default: 127.0.0.1)
7350logout - sparc|x86/solaris login remote root (version 0.7.0) -sc.
team teso.
ht:vDTpa:d:
%c:0x%lx
give args address in [sx]:0x123 format, dumb pentester!
invalid [sx] manual target
WARNING: target out of list. list:
# using target: %s
failed to connect
# setting TTYPROMPT
gera
TTYPROMPT
login:
# detected first login prompt
foo 7350
pass
# detected second login prompt
### attach and press enter!
# send long login bait, waiting for password prompt
# press enter at the prompt
Password:
# received password prompt, success?
7350
# waiting for shell (more than 15s hanging = failure)
# detected shell prompt, successful exploitation
################################################################################
unset HISTFILE;id;uname -a;uptime;
# returning into 0x%08lx
envcount = %d (0x%x)
padding with %ld (0x%lx) chars
7350
WIRE-BUFFER
xp_setenv:send
no room to store shellcode (%lu bytes given, %u needed)
CODE-BUFFER
failed telnet_prompt.
failed exploitation. possible causes:
# 1. login patched
# 2. wrong target type (sparc|x86)
# 3. weird/no solaris version <= 2.4
# 4. TTYPROMPT weirdness, try again with -T option
# 5. try with -p -v options
good luck.
rbuf:
from wire
after processing
# telnetd either died or invalid response
num . description
----+-----------------------------------------------------
%3d | %s
    :    0x%08lx
    '
read user
read remote
```

```
................................ !"#$%&'()*+,-
./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~.
................................................................................
........................................
/* %s, %u bytes */
%02x
to wire
first,second: %02x %02x   2last,last: %02x %02x
 ¿ÿÿ ¿ÿÿ
/bin/ksh
RWP«
«°;èàÿÿÿ/bin/ksh
ÿÿÿÿ
ÿÿÿÿ
þÿÿo
ÿÿÿo
ðÿÿo:
init.c
/usr/src/bs/BUILD/glibc-2.1.3/csu/
gcc2_compiled.
int:t(0,1)=r(0,1);0020000000000;0017777777777;
char:t(0,2)=r(0,2);0;127;
long int:t(0,3)=r(0,1);0020000000000;0017777777777;
unsigned int:t(0,4)=r(0,1);0000000000000;0037777777777;
long unsigned int:t(0,5)=r(0,1);0000000000000;0037777777777;
long long int:t(0,6)=r(0,1);01000000000000000000000;07777777777777777777;
long long unsigned int:t(0,7)=r(0,1);0000000000000;01777777777777777777777;
short int:t(0,8)=r(0,8);-32768;32767;
short unsigned int:t(0,9)=r(0,9);0;65535;
signed char:t(0,10)=r(0,10);-128;127;
unsigned char:t(0,11)=r(0,11);0;255;
float:t(0,12)=r(0,1);4;0;
double:t(0,13)=r(0,1);8;0;
long double:t(0,14)=r(0,1);12;0;
complex int:t(0,15)=s8real:(0,1),0,32;imag:(0,1),32,32;;
complex float:t(0,16)=r(0,16);4;0;
complex double:t(0,17)=r(0,17);8;0;
complex long double:t(0,18)=r(0,18);12;0;
void:t(0,19)=(0,19)
../include/libc-symbols.h
/usr/src/bs/BUILD/glibc-2.1.3/build-i386-linux/config.h
../include/libintl.h
../intl/libintl.h
../include/features.h
../include/sys/cdefs.h
../misc/sys/cdefs.h
/usr/lib/gcc-lib/i386-redhat-linux/egcs-2.91.66/include/stddef.h
../include/locale.h
../locale/locale.h
lconv:T(10,1)=s48decimal_point:(10,2)=*(0,2),0,32;thousands_sep:(10,2),32,32;\
grouping:(10,2),64,32;int_curr_symbol:(10,2),96,32;\
currency_symbol:(10,2),128,32;mon_decimal_point:(10,2),160,32;\
mon_thousands_sep:(10,2),192,32;mon_grouping:(10,2),224,32;\
positive_sign:(10,2),256,32;negative_sign:(10,2),288,32;\
int_frac_digits:(0,2),320,8;frac_digits:(0,2),328,8;\
p_cs_precedes:(0,2),336,8;p_sep_by_space:(0,2),344,8;\
n_cs_precedes:(0,2),352,8;n_sep_by_space:(0,2),360,8;\
```

```
p_sign_posn:(0,2),368,8;n_sign_posn:(0,2),376,8;;
../include/xlocale.h
../locale/xlocale.h
__locale_struct:T(13,1)=s36__locales:(13,2)=ar(0,1);0;5;(13,3)=*(13,4)=xslocale_da
ta:,0,192;\
__ctype_b:(13,5)=*(0,9),192,32;__ctype_tolower:(13,6)=*(0,1),224,32;\
__ctype_toupper:(13,6),256,32;;
__locale_t:t(13,7)=(13,8)=*(13,1)
../sysdeps/unix/sysv/linux/_G_config.h
../sysdeps/unix/sysv/linux/bits/types.h
size_t:t(16,1)=(0,4)
__u_char:t(15,1)=(0,11)
__u_short:t(15,2)=(0,9)
__u_int:t(15,3)=(0,4)
__u_long:t(15,4)=(0,5)
__u_quad_t:t(15,5)=(0,7)
__quad_t:t(15,6)=(0,6)
__int8_t:t(15,7)=(0,10)
__uint8_t:t(15,8)=(0,11)
__int16_t:t(15,9)=(0,8)
__uint16_t:t(15,10)=(0,9)
__int32_t:t(15,11)=(0,1)
__uint32_t:t(15,12)=(0,4)
__int64_t:t(15,13)=(0,6)
__uint64_t:t(15,14)=(0,7)
__qaddr_t:t(15,15)=(15,16)=*(15,6)
__dev_t:t(15,17)=(15,5)
__uid_t:t(15,18)=(15,3)
__gid_t:t(15,19)=(15,3)
__ino_t:t(15,20)=(15,4)
__mode_t:t(15,21)=(15,3)
__nlink_t:t(15,22)=(15,3)
__off_t:t(15,23)=(0,3)
__loff_t:t(15,24)=(15,6)
__pid_t:t(15,25)=(0,1)
__ssize_t:t(15,26)=(0,1)
__rlim_t:t(15,27)=(0,3)
__rlim64_t:t(15,28)=(15,6)
__id_t:t(15,29)=(15,3)
__fsid_t:t(15,30)=(15,31)=s8__val:(15,32)=ar(0,1);0;1;(0,1),0,64;;
__daddr_t:t(15,33)=(0,1)
__caddr_t:t(15,34)=(10,2)
__time_t:t(15,35)=(0,3)
__swblk_t:t(15,36)=(0,3)
__clock_t:t(15,37)=(0,3)
__fd_mask:t(15,38)=(0,5)
__fd_set:t(15,39)=(15,40)=s128fds_bits:(15,41)=ar(0,1);0;31;(15,38),0,1024;;
__key_t:t(15,42)=(0,1)
__ipc_pid_t:t(15,43)=(0,9)
__blkcnt_t:t(15,44)=(0,3)
__blkcnt64_t:t(15,45)=(15,6)
__fsblkcnt_t:t(15,46)=(15,4)
__fsblkcnt64_t:t(15,47)=(15,5)
__fsfilcnt_t:t(15,48)=(15,4)
__fsfilcnt64_t:t(15,49)=(15,5)
__ino64_t:t(15,50)=(15,4)
__off64_t:t(15,51)=(15,24)
```

```
__t_scalar_t:t(15,52)=(0,3)
__t_uscalar_t:t(15,53)=(0,5)
__intptr_t:t(15,54)=(0,1)
../linuxthreads/sysdeps/pthread/bits/pthreadtypes.h
../sysdeps/unix/sysv/linux/bits/sched.h
__sched_param:T(18,1)=s4sched_priority:(0,1),0,32;;
_pthread_fastlock:T(17,1)=s8__status:(0,3),0,32;__spinlock:(0,1),32,32;;
_pthread_descr:t(17,2)=(17,3)=*(17,4)=xs_pthread_descr_struct:
pthread_attr_t:t(17,5)=(17,6)=s36__detachstate:(0,1),0,32;\
__schedpolicy:(0,1),32,32;__schedparam:(18,1),64,32;\
__inheritsched:(0,1),96,32;__scope:(0,1),128,32;\
__guardsize:(16,1),160,32;__stackaddr_set:(0,1),192,32;\
__stackaddr:(17,7)=*(0,19),224,32;__stacksize:(16,1),256,32;;
pthread_cond_t:t(17,8)=(17,9)=s12__c_lock:(17,1),0,64;\
__c_waiting:(17,2),64,32;;
pthread_condattr_t:t(17,10)=(17,11)=s4__dummy:(0,1),0,32;;
pthread_key_t:t(17,12)=(0,4)
pthread_mutex_t:t(17,13)=(17,14)=s24__m_reserved:(0,1),0,32;\
__m_count:(0,1),32,32;__m_owner:(17,2),64,32;\
__m_kind:(0,1),96,32;__m_lock:(17,1),128,64;;
pthread_mutexattr_t:t(17,15)=(17,16)=s4__mutexkind:(0,1),0,32;;
pthread_once_t:t(17,17)=(0,1)
_pthread_rwlock_t:T(17,18)=s32__rw_lock:(17,1),0,64;__rw_readers:(0,1),64,32;\
__rw_writer:(17,2),96,32;__rw_read_waiting:(17,2),128,32;\
__rw_write_waiting:(17,2),160,32;__rw_kind:(0,1),192,32;\
__rw_pshared:(0,1),224,32;;
pthread_rwlock_t:t(17,19)=(17,18)
pthread_rwlockattr_t:t(17,20)=(17,21)=s8__lockkind:(0,1),0,32;\
__pshared:(0,1),32,32;;
pthread_t:t(17,22)=(0,5)
wchar_t:t(19,1)=(0,3)
wint_t:t(19,2)=(0,4)
_G_int16_t:t(14,1)=(0,8)
_G_int32_t:t(14,2)=(0,1)
_G_uint16_t:t(14,3)=(0,9)
_G_uint32_t:t(14,4)=(0,4)
_IO_stdin_used:G(0,1)
GCC: (GNU) egcs-2.91.66 19990314/Linux (egcs-1.1.2 release)
GCC: (GNU) egcs-2.91.66 19990314/Linux (egcs-1.1.2 release)
GCC: (GNU) egcs-2.91.66 19990314/Linux (egcs-1.1.2 release)
GCC: (GNU) egcs-2.91.66 19990314/Linux (egcs-1.1.2 release)
GCC: (GNU) egcs-2.91.66 19990314/Linux (egcs-1.1.2 release)
GCC: (GNU) egcs-2.91.66 19990314/Linux (egcs-1.1.2 release)
01.01
01.01
01.01
01.01
01.01
01.01
.symtab
.strtab
.shstrtab
.interp
.note.ABI-tag
.hash
.dynsym
.dynstr
```

```
.gnu.version
.gnu.version_r
.rel.got
.rel.bss
.rel.plt
.init
.plt
.text
.fini
.rodata
.data
.eh_frame
.ctors
.dtors
.got
.dynamic
.bss
.stab
.stabstr
.comment
.note
ÿÿÿo
þÿÿo
```

## Appendix H: ./bash_history

| | |
|---|---|
| Name: | .bash_history |
| File Ext: | bash_history |
| Description: | File |
| Last Accessed: | 06/21/04 01:02:23PM |
| Last Written: | 06/21/04 09:56:43AM |
| Entry Modified: | 06/21/04 09:56:43AM |
| Logical Size: | 8,588 |
| Physical Size: | 9,216 |
| Starting Extent: | 0/-C22907 |
| File Extents: | 1 |
| Permissions: | • |
| Bookmarks: | 6 |
| Physical Location: | 11,440,177,664 |
| Evidence File: | 139175 |
| File Identifier: | 60311 |
| Hash Value: | 1a5f41c00b3c297a1d946ae4974160c2 |
| Hash Set: | 139175 |
| Full Path: | Collector 1\139175\/\root\.bash_history |

File Extents

| Start Sector | Sectors | Start Cluster | Clusters |
|---|---|---|---|
| 22,344,097 | 18 | 22,907 | 9 |

Permissions

| | |
|---|---|
| Id: | 0 |
| Property: | Owner |

| | |
|---|---|
| Id: | 0 |
| Property: | Group |

| | |
|---|---|
| Property: | Owner |
| Permissions: | [G-R] [G-W] |

.bash_history.txt

```
slocate xinetd
/etc/rc.d/init.d/xinetd restart
netstat -ln
ps -ef
cd /etc
ls -tlr xinetd.d
ls -l xinet*
vi xinetd.conf
vi /etc/services
cd xinetd.d
/etc/rc.d/init.d/xinetd restart
netstat -ln
ta
```

```
exit
cat ide/ide0/hda/settings
cat ide/ide0/hda/identify
cat ide/ide0/hda/model
df
exit
cd /etc
cd xinetd.d
ls -ltr
slocate xinetd.d
slocate xinetd
/etc/rc.d/init.d/xinetd restart
netstat -ln
exit
cd /proc/21173
ls -l
ls -l fd
ls -l maps
cat maps
cat mem
cat cmdline
cat environ
cat stat
cat statm
ps -ef
cd /proc/23757
ls -l
cat cmdline
cat fd
ls -l fd
ps -ef
cd /proc/25863
ls -l
ls -l fd
ls -l maps
cat maps
cat statm
cat status
cat statm
cat stat
ls -l
ps -f
ps -ef
cd ../24862
ls -l
cat fd
ls -l fd
cat maps
cat statm
cat stat
cat status
exit
ifconfig
vi /etc/fstab
vi /etc/hosts
umount xxx:/xxx
mount mount   /nfs/xxxxxxxx/hostuid
```

```
mount   /nfs/xxxxxxxx/hostuid
mkdir /nfs/xxxxxxxx
mkdir /nfs/xxxxxxxx/hostuid
mount   /nfs/xxxxxxxx/hostuid
exit
uname
unalias passwd
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/male1 -s /bin/ksh -c "male full name 1" -m male1
chmod 640 /home/male1/.profile
passwd male1
exit
uname
unalias passwd
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/male2 -s /bin/ksh -c " male full name 2" -m male2
chmod 640 /opt/male2/.profile
passwd male2
exit
uname
unalias passwd
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/male3 -s /bin/ksh -c " male full name 3" -m male3
chmod 640 /opt/male3/.profile
passwd male3
exit
uname
unalias passwd
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/male4 -s /bin/ksh -c " male full name 4" -m male4
chmod 640 /opt/male4/.profile
passwd male4
exit
uname
unalias passwd
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/male5 -s /bin/ksh -c " male full name 5" -m male5
chmod 640 /opt/male5/.profile
passwd male5
exit
uname
unalias passwd
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/female1 -s /bin/ksh -c " female full name 1" -m
female1
chmod 640 /opt/female1/.profile
passwd female1
exit
uname
unalias passwd
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/male6 -s /bin/ksh -c " male full name 6" -m male6
chmod 640 /opt/male6/.profile
passwd male6
exit
uname
unalias passwd
```

```
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/female2 -s /bin/ksh -c " female full name 2" -m
female2
chmod 640 /opt/female2/.profile
passwd female2
exit
uname
unalias passwd
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/male7 -s /bin/ksh -c " male full name 7" -m male7
chmod 640 /opt/male7/.profile
passwd male7
exit
uname
unalias passwd
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/male8 -s /bin/ksh -c " male full name 8" -m male8
chmod 640 /opt/male8/.profile
passwd male8
exit
uname
unalias passwd
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/male9 -s /bin/ksh -c " male full name 9" -m male9
chmod 640 /opt/male9/.profile
passwd male9
exit
uname
unalias passwd
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/male10 -s /bin/ksh -c " male full name 10" -m male10
chmod 640 /opt/male10/.profile
passwd male10
exit
uname
unalias passwd
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/male11 -s /bin/ksh -c " male full name 11" -m male11
chmod 640 /opt/male11/.profile
passwd male11
exit
uname
unalias passwd
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/male12 -s /bin/ksh -c " male full name 12" -m male12
chmod 640 /opt/male12/.profile
passwd male12
exit
uname
unalias passwd
groupadd xxxxxxxx
useradd -g xxxxxxxx -d /opt/male13 -s /bin/ksh -c " male full name 13" -m male13
chmod 640 /opt/male13/.profile
passwd male13
exit
uname
hostname=`uname -a | awk '{print $2}'`
```

```
filename=/tmp/systeminfo.$hostname
cat /dev/null > $filename
echo "Processor:" >> $filename
cat /proc/cpuinfo >> $filename
### get Operating system Level
echo "Linux OS Level:" >> $filename
uname -r >> $filename
echo >> $filename
### get list of runing processes
echo "Processes:" >> $filename
ps -ef >> $filename
echo >> $filename
for i in `cd /var/spool/cron/ ; ls -1 `; do echo
"******************************** \r" >> $filename; echo "CRON for $i:" >>
$filename; cat /var/spool/cron/$i >> $filename; echo >> $filename; done
cd /tmp
ls -al
exit
cd /tmp
ls -al system*
scp systeminfo.host.xyztel.net root@nnn.nnn.12.nnn:/tmp
scp systeminfo.host.xyztel.net male7@nnn.nnn.19.nn:/tmp
scp systeminfo.host.xyztel.net root@10.nn.1.nn:/tmp
exit
pwd
cd bin
ls
cd ln*
ls
cat version.dat
exit
ls
cd /
ls
cd /etc
ls
cd /usr/bin
ls
cd bin
ls
cd lnx*
ls
cat version.dat
exit
exit
vi /var/log/messages
ps -ef | grep member  | wc
exit
startx
ksh
startx
ls
cd /
ls
startx
exit
vi /etc/messages
```

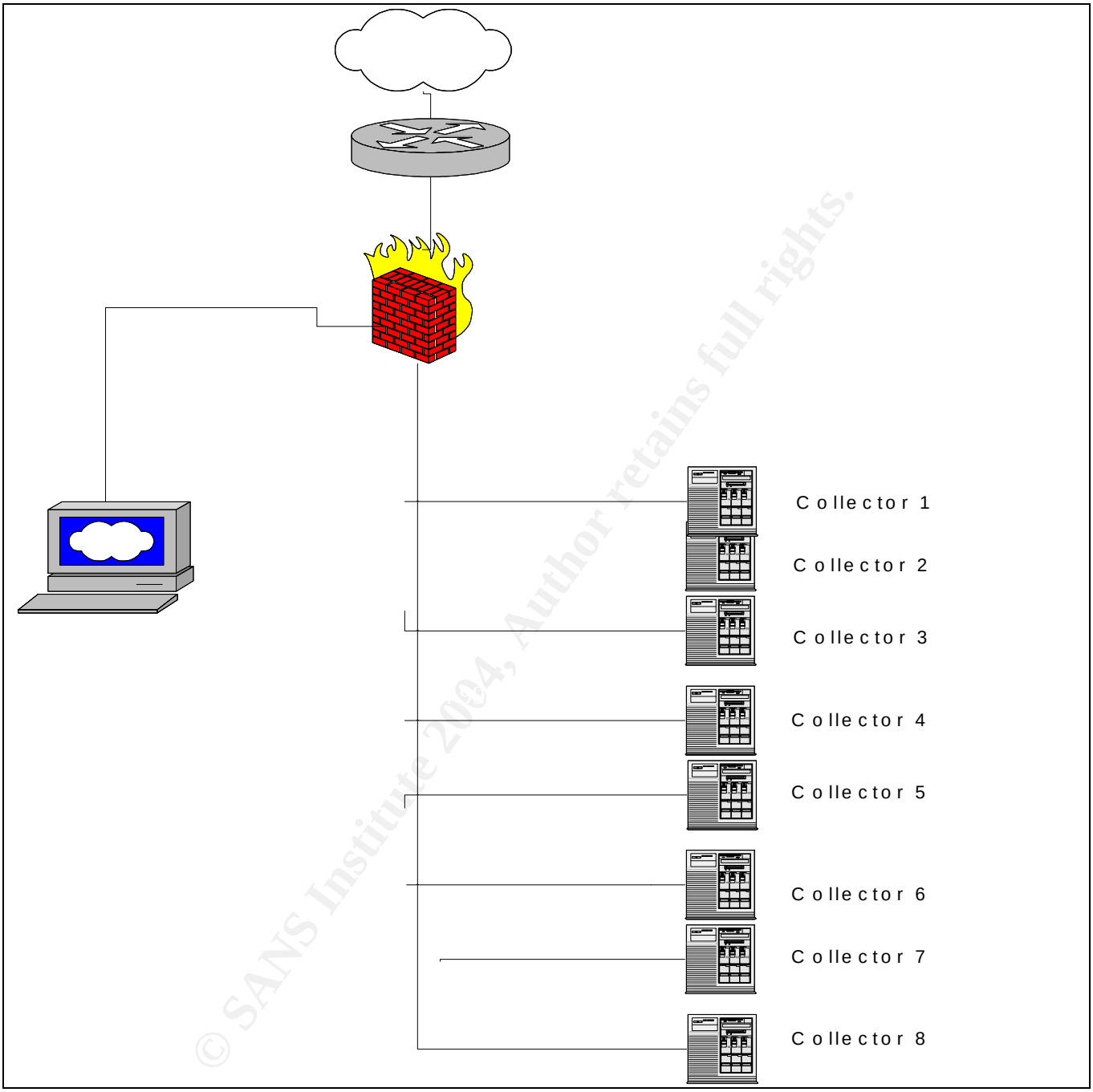
```
vi /var/log/messages
ps -ef
su - hostuid
su - hostuid
exit
ping nn.nn.59.nn
ping nn.nnn.9.nn
ping nnn.nnn.12.nn
ping nnn.nnn.9.nn
ping nnn.nn.27.nnn
ping nnn.nn.0.nnn
ping nn.nn.59.nn
ping nnn.nnn.107.nn
ping nnn.nnn.197.nnn
ping nnn.nnn.100.nn
ping nnn.nnn.196.nn
ping nnn.nnn.12.nnn
ping nn.nn.118.nnn
cd /var/log
ls -tlr
cd cron
vi cron
exit
cd /etc
ls -tlr
vi xinetd.conf
cd xinetd.d
ls -tlr
cat specialfilename
services
service
service status xinetd
service restart xinetd
service  xinetd status
service  xinetd restart
exit
uname -a
useradd -c "male14" -s /bin/ksh -m male14
passwd male14
exit
uptime
passwd male7
exit
/etc/rc.d/init.d/xinetd restart
exit
cd /var/log
ls -tlr
vi messages
netstat -ln
netstat -ln
ps -ef
ls -l
ls -tlr
tail messages
cd /hostuiddata/log
ls -tlr
ssh -l hostuid nn.n.1.n
```

```
exit
ssh nnn.nn.0.nnn
exit
cd /var/log
vi cron
date
ts
exit
set -o vi
pwd
set -o vi
ls -ltr
ls -ltr
ifconfig -a
exit
passwd male14
pwd
ls -ail
cd /tmp
exit
uname -a
useradd -c "female3" -m female3
passwd female3
s_TF8+VD
exit
cd /etc/xinetd
cd /etc/xinetd.d
ls -tlr
vi specialfilename
cd ..
ls -tlr xin*
vi xinetd.conf
slocate xinetd
/etc/rc.d/init.d/xinetd restart
exit
kill -9 26595
ps -ef --cols 300 | grep "pts/6"
kill 26534
ps -ef --cols 300 | grep "pts/6"
kill -9 26534
ps -ef --cols 300 | grep "pts/6"
ps -ef --cols 300 | grep "pts/6"
ps -ef --cols 300 | grep "df"
/hostuid/hostuid/bin/killmember
/hostuid/hostuid/bin/killhttpd
ps -ef
ts
exit
cd /home/install
ls -tlr
cd perl-src
ls -l
ftp host2
ftp nn.n.1.nn
tar -xvf file-1.08.tar
cd *108
cd *1.08
```

```
perl Makefile.PL
make
make test
make install
cd ..
tar -xvf file-0.12.tar
cd file-0.12
perl Makefile.POL
perl Makefile.PL
make
make test
ls -l /usr/local/lib/perl5/site_perl/5.8.0/file/0.12/ParserDetails.ini
perl -v
ls -l /usr/local/lib/perl5/site_perl/5.6.0/file/0.12/ParserDetails.ini
ls -l
slocate Temp.pm
ls -l /usr/local/lib/perl5/site_perl
ls -l /usr/local/lib/perl5
ls -l /usr/local/lib/
ls -l /usr/local/lib/perl
make test
cd ..
ftp nn.n.1.nn
tar -xvf File-0.14.tar.gz
gunzip File-0.14.tar.gz
tar -xvf File-0.14.tar.gz
tar -xvf File-0.14.tar
cd File-0.14
perl Makefile.PL
ftp nn.n.1.nn
gunzip Test-0.47.tar.gz
mv *.tar ..
cd ..
tar -xvf Test-0.47.tar
cd *47
perl Makefile.PL
make
make test
make install
cd../ File-0.14
cd ..
ls -tlr
cd File-0.14
make test
make install
cd ..
ls -l
cd FILE-0.12
make test
make install
cd ..
ls -tlr
tar -xvf File-2.09.tar
cd *2.09
perl Makefile.PL
make
make test
```

```
make install
exit
ps -ef --cols 500 | grep inet
vi /etc/xinetd.d/specialfilename
cat /etc/xinetd.d/specialfilename
cdc
exit
ps -ef --cols 500 | grep inet
/etc/rc.d/init.d/xinetd restart
ps -ef --cols 500 | grep inet
exit
vi /etc/xinetd.d/specialfilename
cat /etc/xinetd.d/specialfilename
ps -ef --cols 500 | grep inet
/etc/rc.d/init.d/xinetd restart
ps -ef --cols 500 | grep inet
exit
su - xxx
exit
crontab -l
exit
exit
find / -name syslogd
cd /var/adm
cd /var
ls
dir
cd log
ls
ls -al
more messages
view messages
ls -al
cd /etc
ls -al sys*
more syslog
view syslog.conf
exit
w
rebut
id
id
hostname
telnet nnn.nn.0.nn
telnet nnn.nn.1.nn
ftp nnn.nn.1.nn
telnet nnn.nn.1.nn
adduser
cd ..
uname -a
ls
finger
finger hostuid
logout
exit
```

## Appendix I: Collector1 Network Diagram



## Appendix J: Timeline Collector1

…snip

| File Offset | Length | Name | In Report | Is Deleted | Last Accessed | File Created | Last Written |
|---|---|---|---|---|---|---|---|
| Entry | Modified | File Deleted | Logical Size | | | | |
| 05:30:16PM | 05/23/03 11:24:17AM | | 270,536 | • | 419341 | | |
| 0 | | .bash_history | • | 06/19/04 12:07:35PM | | 06/03/03 10:24:24PM | |
| 06/03/03 10:24:24PM | | | 282 | • | 290324 | | |
| 170 | 14 | killhttpd | | 06/22/04 12:55:00AM | | 06/04/03 09:37:46AM | 06/04/03 |
| 09:37:46AM | | 2,372 | • | 435487 | | | |
| 0 | | .bash_logout | • | 03/05/04 11:07:32AM | | 01/23/04 12:49:12AM | |
| 01/23/04 12:49:12AM | | | 24 | • | 209679 | | |
| 0 | | /home | • | 06/22/04 12:02:03AM | | 01/23/04 12:49:12AM | 01/23/04 |
| 12:49:12AM | | 4,096 | | 2 | hda6 | | |
| 0 | | .screenrc | • | 01/23/04 12:49:12AM | | 01/23/04 12:49:12AM | |
| 01/23/04 12:49:12AM | | | 3,728 | • | 209683 | | |
| 0 | | .bash_history | • | 06/19/04 12:09:22PM | | 03/05/04 11:07:33AM | |
| 03/05/04 11:07:33AM | | | 12 | • | 209692 | | |
| 8754449 | 7 | Lost File | | 05/10/04 12:24:34AM | | 05/10/04 12:24:03AM | |
| 05/10/04 12:26:40AM | | | 20,981,681 | • | 614047 | | |
| 205108 | 6 | Lost File | | 05/14/04 11:16:49PM | | 05/14/04 11:16:17PM | |
| 05/14/04 11:17:28PM | | | 537,260 | • | 613959 | | |
| 6441048 | 8 | Lost File | | 05/14/04 11:16:49PM | | 05/14/04 11:16:19PM | |
| 05/14/04 11:18:54PM | | | 20,440,650 | • | 613960 | | |
| 32766 | 19 | File-108.tar | | 05/16/04 12:37:06AM | | 05/16/04 12:36:17AM | |
| 05/16/04 12:36:17AM | | | 102,400 | • | 161282 | | |
| 76931 | 19 | File-108.tar | | 05/16/04 12:37:06AM | | 05/16/04 12:36:17AM | |
| 05/16/04 12:36:17AM | | | 102,400 | • | 161282 | | |
| 56764 | 19 | File-108.tar | | 05/16/04 12:37:06AM | | 05/16/04 12:36:17AM | |
| 05/16/04 12:36:17AM | | | 102,400 | • | 161282 | | |
| 11348 | 19 | File-108.tar | | 05/16/04 12:37:06AM | | 05/16/04 12:36:17AM | |
| 05/16/04 12:36:17AM | | | 102,400 | • | 161282 | | |
| 36948 | 19 | File-108.tar | | 05/16/04 12:37:06AM | | 05/16/04 12:36:17AM | |
| 05/16/04 12:36:17AM | | | 102,400 | • | | | |
| 36948 | 19 | File-108.tar | | 05/16/04 12:37:06AM | | 05/16/04 12:36:17AM | |
| 05/16/04 12:36:17AM | | | 102,400 | • | 161282 | | |
| 36948 | 19 | File-108.tar | | 05/16/04 12:37:06AM | | 05/16/04 12:36:17AM | |
| 05/16/04 12:36:17AM | | | 102,400 | • | 161282 | | |
| 31164 | 19 | File-108.tar | | 05/16/04 12:37:06AM | | 05/16/04 12:36:17AM | |
| 05/16/04 12:36:17AM | | | 102,400 | • | 161282 | | |
| 32768 | 17 | File-108.tar | | 05/16/04 12:37:06AM | | 05/16/04 12:36:17AM | |
| 05/16/04 12:36:17AM | | | 102,400 | • | 161282 | | |
| 31164 | 19 | File-108.tar | | 05/16/04 12:37:06AM | | 05/16/04 12:36:17AM | |
| 05/16/04 12:36:17AM | | | 102,400 | • | 161282 | | |
| 208835 | 17 | FILE-0.12.tar | | 05/16/04 12:38:06AM | | 05/16/04 12:36:32AM | |
| 05/16/04 12:36:32AM | | | 358,400 | • | 161283 | | |
| 307664 | 19 | FILE-0.12.tar | | 05/16/04 12:38:06AM | | 05/16/04 12:36:32AM | |
| 05/16/04 12:36:32AM | | | 358,400 | • | 161283 | | |
| 208835 | 17 | FILE-0.12.tar | | 05/16/04 12:38:06AM | | 05/16/04 12:36:32AM | |
| 05/16/04 12:36:32AM | | | 358,400 | • | 161283 | | |
| 305841 | 25 | FILE-0.12.tar | | 05/16/04 12:38:06AM | | 05/16/04 12:36:32AM | |
| 05/16/04 12:36:32AM | | | 358,400 | • | 161283 | | |
| 305945 | 20 | FILE-0.12.tar | | 05/16/04 12:38:06AM | | 05/16/04 12:36:32AM | |
| 05/16/04 12:36:32AM | | | 358,400 | • | 161283 | | |

…snip

0 hostuid@nn.n.1.nn • 06/21/04 12:45:01PM 06/20/04 12:21:59PM
06/20/04 12:21:59PM 34,957 • 60315
0 hostuid@nn.n.1.nn • 06/21/04 12:45:01PM 06/20/04 12:21:59PM
06/20/04 12:21:59PM 34,957 • 60315
0 hostuid@nn.n.1.nn • 06/21/04 12:45:01PM 06/20/04 12:21:59PM
06/20/04 12:21:59PM 34,957 • 60315
0 hostuid@nn.n.1.nn • 06/21/04 12:45:01PM 06/20/04 12:21:59PM
06/20/04 12:21:59PM 34,957 • 60315
06/17/04 05:54:47PM 06/20/04 01:28:23PM 6,457,837 • 65891
0 rar@nn.n.1.nn • 06/21/04 12:45:01PM 06/20/04 06:42:39PM
06/20/04 06:42:39PM 9,028 • 60317
0 rar@nn.n.1.nn • 06/21/04 12:45:01PM 06/20/04 06:42:39PM
06/20/04 06:42:39PM 9,028 • 60317
0 rar@nn.n.1.nn • 06/21/04 12:45:01PM 06/20/04 06:42:39PM
06/20/04 06:42:39PM 9,028 • 60317
0 rar@nn.n.1.nn • 06/21/04 12:45:01PM 06/20/04 06:42:39PM
06/20/04 06:42:39PM 9,028 • 60317

## Works Cited


108th Congress 2d Session S.2560, "Inducing Infringement of Copyrights Act of 2004 (Introduced in Senate)", 22 June 2004, URL: http://thomas.loc/gov/cgi-bin/query/z?c108:S.2560:, 20 Sep 2004

7350logout, "Index of /networksa/tools", URL: http://examples.oreilly.com/networksa/tools/ , 28 Sep 2004

Carrier, Brian, "The Sleuth Kit and Autopsy ", 24 May 2004, URL: http://www.sleuthkit.org/ and http://sleuthkit.sourceforge.net/, 12 August 2004

Carrier, Brian, "Open Source Digital Forensics Tools: The Legal Argument October 2002", @ Stake Research Project, 2003 @Stake,Inc. URL: http://www.atstake.com/research/reports/acrobat/atstake_opensource_forensics.pdf, 12 Aug, 2004

Cheng, Derek, "Freeware Forensics Tools for UNIX, 1 November 2001, Security Focus Infocus, 1999 – 2004 Security Focus, URL: http://www.securityfocus.com/infocus/1503, 23 Aug 2004

Christias, Panagiotis,"elf - object file access library", @1994 Man-cgi 1.11, URL: http://www.mcsr.olemiss.edu/cgi-bin/man-cgi?elf+3, 16 Sep, 2004

Chuvakin, Anton, "Linux Data Hiding and Recovery, Guardian Digital Inc, 10 March 2002, , URL: http://www.linuxsecurity.com/feature_stories/data-hiding-forensics.html, 11 Sep, 2004

/etc/shadow, " Linux Administration Made Easy, Chapter 6: General System Administration Issues", URL: http://www.tldp.org/LDP/lame/LAME/linux-admin-made-easy/shadow-file-formats.html, 27 Sep 2004

Farmer, Dan, "Oct00: What are MACtimes?, Powerful tools for digital databases", Dr. Dobb's Journal October 2000, URL: http://www.ddj.com/documents/s=880/ddj0010f/0010f.htm , 7 Sep 2004

"Mount (8) Linux Programmer's Manual "mount - mount a file system", URL: http://www.rt.com/man/mount.8.html, 12 Aug 2004

Kornblum, Jesse, "md5deep – the latest version", URL: http://md5deep.sourceforge.net/, 9 September 2004,

Kornblum, Jesse, "Foremost – the latest version", URL: http://foremost.sourceforge.net/, 9 September, 2004,

Krause, Rory, "Issue 94: Using ssh Port Forwarding to Print at Remote Locations",

01 February 2002, URL: http://www.linuxjournal.com/article.php?sid=5462, 12 Aug 2004

Cygnus support,"Strings (1) GNU Development Tools" "strings - print the strings of printable characters in files", 1993 Free Software Foundation, Inc., 25 June 1993, URL: http://www.rt.com/man/strings.1.html,11 Sep 2004

Gilbert, H, "Introduction to SNA" , 2 Feb 1995, URL: http://www.yale.edu/pclt/COMM/SNA.HTM, 6 Sept 2004

Google ™ , © 2004 Google – Searching 4,285,199,774 Web pages, URL: http://www.google.com

Grollier Encyclopedia of Knowledge, Grolier Incorporated, Danbury, 1991, p. 250-251

Hotmail, Microsoft Corporation, URL: http://www.msn.com/, 19 Sep 2004

Hobbit, Netcat 1.10, Released 1996, Copyright © 2004 @Stake, Inc., URL: http://www.atstake.com/research/tools/network_utilities/ , 20 Sep 2004.

Index of /bmap-1.0.20, Aug 2004, URL: http://garchive.movealong.org/bmap-1.0.20/

Nortel tunnel termination device, Nortel Networks Limited, 1994-2004, URL: http://www.nortelnetworks.com/index.html

Pine Information Center, The University of Washington Pine Information Center, Pine and Pico are trademarks of the University of Washington. Copyright © 1989 - 2004 by the University of Washington., URL: http://www.washington.edu/pine/, 19 Sep 2004

Red Hat Glossary: "inode", Red Hat Documentation, URL: http://www.redhat.com/docs/glossary/index.html#I, 9/17/2004.

"Scp – Secure Copy (remote file copy program)", 8 Nov 1995, URL: http://www-hep2.fzu.cz/computing/adm/scp.html, 7 Sept 2004

"Script kiddie", Last updated 20:02 14 Sep 2004
This article is licensed under the <a href="http://www.gnu.org/copyleft/fdl.html">GNU Free Documentation License</a>. It uses material from the <a href="http://www.wikipedia.org/wiki/Script_kiddie">Wikipedia article "Script kiddie "</a>. , 19 Sep 2004

Strace: URL: http://www.die.net/doc/linux/man/man1.strace.1.html

Sorenson, Holt, "Incident Response Tools for Unix, Part Two: File-System Tools", , SecurityFocusInfocus, 1999 – 2004, URL: http://www.securityfocus.com/infocus/1247 , 17 Sep 2003

tcpdump - dump traffic on a network, URL: http://www.tcpdump.org/tcpdump_man.html, 28 Sep 2004

The Linux Forum, "1.3 /bin", 01 June 2004, URL: http://www.linuxforum.com/linux-filesystem/bin.html, 27 September 2004

"The Metasploit Project", 13 August 2004, Metasploit.com, URL: http://www.metasploit.com/, 31 Aug 2004

The SANS Institute Forensic Analysis Cheat Sheet v1.0 ,Sans Institute, URL: http://www.sans.org/

United States Department of Justice, "Computer Crime and Intellectual Property Section (CCIPS): How to Report Internet-Related Crime", 20 May 2004, URL: http://www.cybercrime.gov/reporting.htm , 16 Sep 2004

United States Department of Justice, "Hialeah Man Arrested for Criminal Copyright Infringement", 20 Sep 2004, URL: http://www.cybercrime.gov/villaIndict.htm , 16 Sep 2004

WinHex: Data Recovery & Computer Forensics Software, Hex Editor & Disk Editor, X-Ways Software Technology AG, URL: http://www.sf-soft.de/winhex/index-m.html, 20 Sept. 2004

Wright, Timothy E., "An Introduction to the Field Guide for Investigating Computer Crime Part 1, 17 April 2000", SecurityFocus™ Infocus, © 1999 – 2004, URL: http://www.securityfocus.com/infocus/1244, July 2004

Wright, Timothy E., "The Field Guide for Investigating Computer Crime: Search and Seizure Basics Part Three", 28 July 2000, SecurityFocus™ Infocus, © 1999 – 2004, URL: http://www.securityfocus.com/infocus/1246 , July 2004

Wright, Timothy E., "The Field Guide for Investigating Computer Crime: Search and Seizure Planning Part Four, 1 Sep 2000", SecurityFocus™ Infocus, © 1999 – 2004, URL: http://www.securityfocus.com/infocus/1247 , July 2004

Wright, Timothy E., "The Field Guide for Investigating Computer Crime: Search and Seizure Approach, Documentation, and Location", 10 Nov 2000, SecurityFocus™ Infocus, © 1999 – 2004, URL: http://www.securityfocus.com/infocus/1248, July 2004

Wright, Timothy E., "The Field Guide for Investigating Computer Crime, Part 6 Search and Seizure – Evidence Retrieval and Processing", 1 Sep 2000, SecurityFocus™ Infocus, © 1999 – 2004, URL: http://www.securityfocus.com/infocus/1247 , July 2004

Last Updated: July 17th, 2018

# Upcoming SANS Forensics Training


| | | | |
|---|---|---|---|
| Community SANS Columbia FOR500 | Columbia, MD | Jul 23, 2018 - Jul 28, 2018 | Community SANS |
| SANS Riyadh July 2018 | Riyadh, Kingdom Of Saudi Arabia | Jul 28, 2018 - Aug 02, 2018 | Live Event |
| Security Operations Summit & Training 2018 | New Orleans, LA | Jul 30, 2018 - Aug 06, 2018 | Live Event |
| SANS Boston Summer 2018 | Boston, MA | Aug 06, 2018 - Aug 11, 2018 | Live Event |
| SANS San Antonio 2018 | San Antonio, TX | Aug 06, 2018 - Aug 11, 2018 | Live Event |
| San Antonio 2018 - FOR508: Advanced Digital Forensics, Incident Response, and Threat Hunting | San Antonio, TX | Aug 06, 2018 - Aug 11, 2018 | vLive |
| SANS August Sydney 2018 | Sydney, Australia | Aug 06, 2018 - Aug 25, 2018 | Live Event |
| SANS New York City Summer 2018 | New York City, NY | Aug 13, 2018 - Aug 18, 2018 | Live Event |
| Mentor Session - AW FOR508 | Phoenix, AZ | Aug 14, 2018 - Sep 13, 2018 | Mentor |
| Community SANS Columbia FOR610 | Columbia, MD | Aug 20, 2018 - Aug 25, 2018 | Community SANS |
| SANS Virginia Beach 2018 | Virginia Beach, VA | Aug 20, 2018 - Aug 31, 2018 | Live Event |
| Data Breach Summit & Training 2018 | New York City, NY | Aug 20, 2018 - Aug 27, 2018 | Live Event |
| SANS Chicago 2018 | Chicago, IL | Aug 20, 2018 - Aug 25, 2018 | Live Event |
| Mentor Session - FOR508 | Copenhagen, Denmark | Aug 22, 2018 - Oct 06, 2018 | Mentor |
| SANS San Francisco Summer 2018 | San Francisco, CA | Aug 26, 2018 - Aug 31, 2018 | Live Event |
| SANS Copenhagen August 2018 | Copenhagen, Denmark | Aug 27, 2018 - Sep 01, 2018 | Live Event |
| SANS Tokyo Autumn 2018 | Tokyo, Japan | Sep 03, 2018 - Sep 15, 2018 | Live Event |
| SANS Amsterdam September 2018 | Amsterdam, Netherlands | Sep 03, 2018 - Sep 08, 2018 | Live Event |
| SANS vLive - FOR585: Advanced Smartphone Forensics | FOR585 - 201809, | Sep 04, 2018 - Oct 11, 2018 | vLive |
| SANS Tampa-Clearwater 2018 | Tampa, FL | Sep 04, 2018 - Sep 09, 2018 | Live Event |
| Threat Hunting & Incident Response Summit & Training 2018 | New Orleans, LA | Sep 06, 2018 - Sep 13, 2018 | Live Event |
| Threat Hunting & IR Summit - FOR526: Memory Forensics In-Depth | New Orleans, LA | Sep 08, 2018 - Sep 13, 2018 | vLive |
| Threat Hunting & IR Summit - FOR508: Advanced Digital Forensics, Incident Response, and Threat Hunting | New Orleans, LA | Sep 08, 2018 - Sep 13, 2018 | vLive |
| SANS Baltimore Fall 2018 | Baltimore, MD | Sep 08, 2018 - Sep 15, 2018 | Live Event |
| Threat Hunting & IR Summit - FOR572: Advanced Network Forensics and Analysis | New Orleans, LA | Sep 08, 2018 - Sep 13, 2018 | vLive |
| SANS Munich September 2018 | Munich, Germany | Sep 16, 2018 - Sep 22, 2018 | Live Event |
| Community SANS Madrid FOR508 (in Spanish) | Madrid, Spain | Sep 17, 2018 - Sep 22, 2018 | Community SANS |
| SANS London September 2018 | London, United Kingdom | Sep 17, 2018 - Sep 22, 2018 | Live Event |
| Community SANS Toronto FOR508 | Toronto, ON | Sep 17, 2018 - Sep 22, 2018 | Community SANS |
| Community SANS Columbia FOR508 | Columbia, MD | Sep 17, 2018 - Sep 22, 2018 | Community SANS |
| Network Security 2018 - FOR585: Advanced Smartphone Forensics | Las Vegas, NV | Sep 23, 2018 - Sep 28, 2018 | vLive |