DIGITAL FORENSICS LAB SERIES

Lab 6: Introduction to Single Purpose Forensic Tools

Objective: Digital Forensics Fundamentals

Document Version: 2015-09-28

Development was funded by the Department of Labor (DOL) Trade Adjustment Assistance Community College and Career Training (TAACCCT) Grant No. TC-22525-11-60-A-48; The National Information Security, Geospatial Technologies Consortium (NISGTC) is an entity of Collin College of Texas, Bellevue College of Washington, Bunker Hill Community College of Massachusetts, Del Mar College of Texas, Moraine Valley Community College of Illinois, Rio Salado College of Arizona, and Salt Lake Community College of Utah.

This workforce solution was funded by a grant awarded by the U.S. Department of Labor's Employment and Training Administration. The solution was created by the grantee and does not necessarily reflect the official position of the U.S. Department of Labor. The Department of Labor makes no guarantees, warranties or assurances of any kind, express or implied, with respect to such information, including any information on linked sites, and including, but not limited to accuracy of the information or its completeness, timeliness, usefulness, adequacy, continued availability or ownership.

Contents

Introduction

This lab includes the following tasks:

1. Using File Hashing Tools to Verify Integrity
2. Mounting a Partition with Deleted Files and Folders
3. Using Foremost to Carve Files
4. Using a HEX Editor

Objective: Digital Forensics Fundamentals

Performing this lab will provide the student with a hands-on lab experience meeting the Digital Forensics Fundamentals Objective:

The candidate will demonstrate an understanding of forensic methodology, key forensics concepts, and identifying types of evidence on current Windows operating systems.

Foremost – Foremost is a file carving utility that allows you to carve files that were "deleted" out of a disk image or a mounted partition. Foremost was created by Jesse Kornblum and is available for download from this link: http://foremost.sourceforge.net/

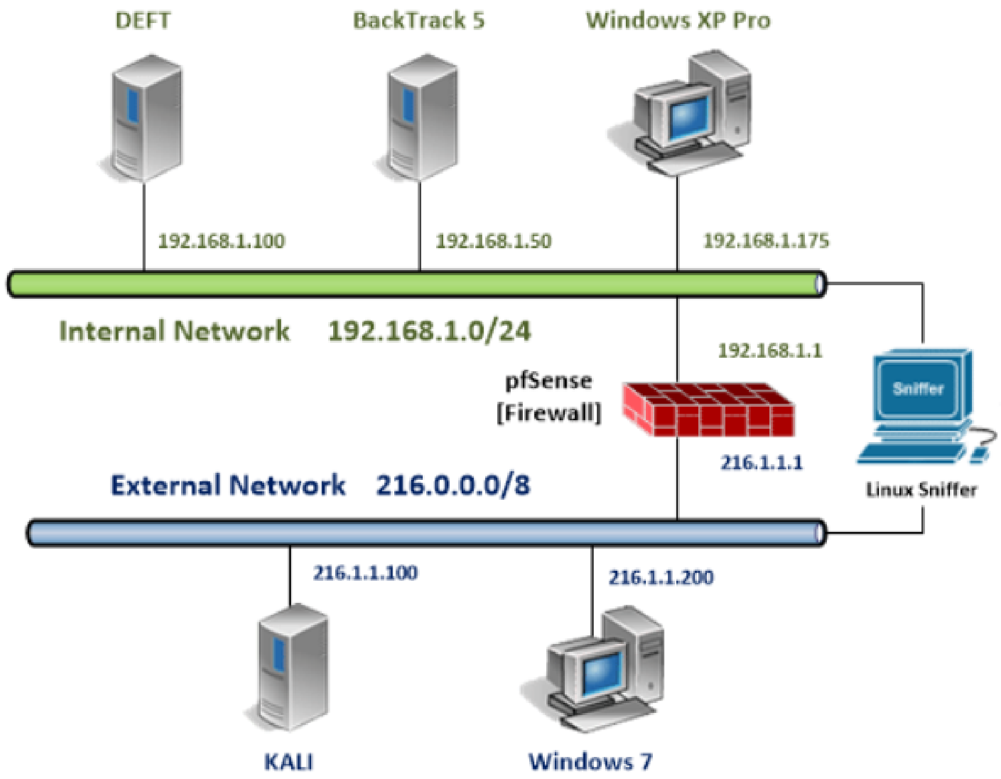
Hexadecimal - A numbering system where numbers 0-9 and letters A-F are used. Also known as base 16, hexadecimal is commonly used in computer forensics and networking.

HEX Editor– A Graphical User Interface (GUI) or command line tool that can be utilized to analyze the hexadecimal code of files. File headers have hexadecimal signatures that are unique to a particular type of file. For example, a JPEG file has a file signature of JFIF.

md5sum – A command that is used from the terminal to verify a MD5 hash. Message Digest 5 is a 128-bit hashing algorithm that aids forensic examiners by “proving” that the copy of the media they are working on is "equivalent" to the original.

sha1sum – A command that is used from the terminal to verify a sha1 hash. Secure Hash Algorithm is a 160-bit hashing algorithm that aids forensic examiners by “proving” that the copy of the media they are working on is "equivalent" to the original.

Lab Topology



Lab Settings

The information in the table below will be needed in order to complete the lab. The task sections below provide details on the use of this information.

Virtual Machine	IP Address	Account (if needed)	Password (if needed)
Kali Linux External Machine	216.1.1.100	root	toor

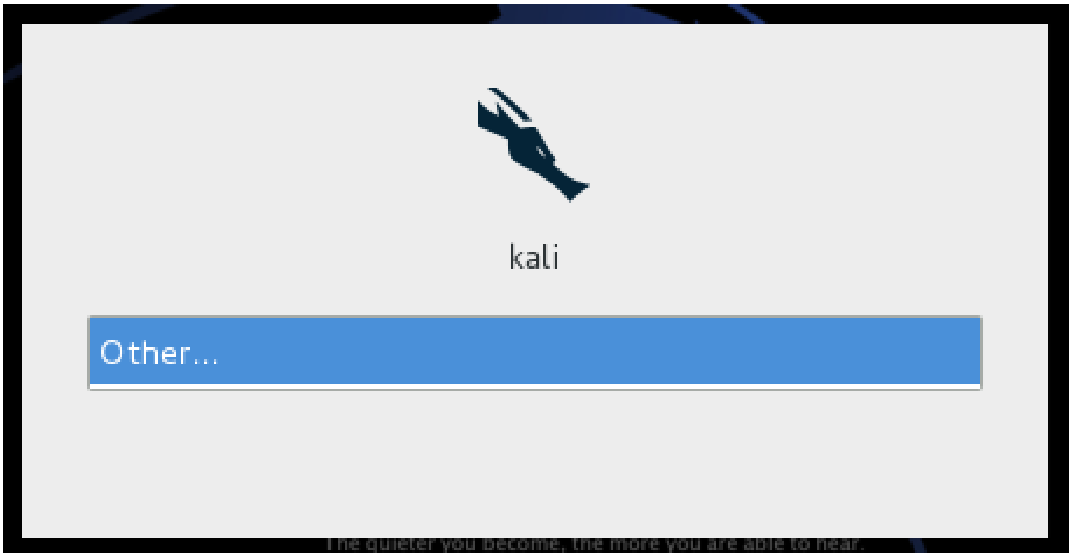
1 Using File Hashing Tools to Verify Integrity

Hashing algorithms, such as SHA1 and MD5, can be used to verify the integrity of data. MD5 stands for Message Digest 5, and is a 128-bit algorithm. SHA1, stands for Secure Hash Algorithm, and is a 160-bit algorithm. A SHA1 hash is more reliable than MD5.

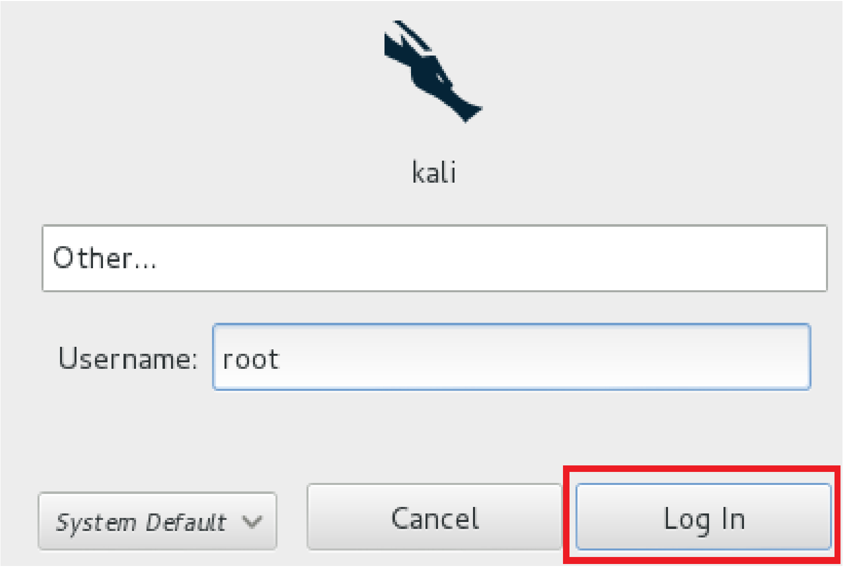
Keep in mind that **Linux commands are case sensitive**. The commands below must be entered exactly as shown.

1.1 Using Hashing Tools

1. Click on the **KALI Machine on the External Network** on the topology. Click the **Other** link.

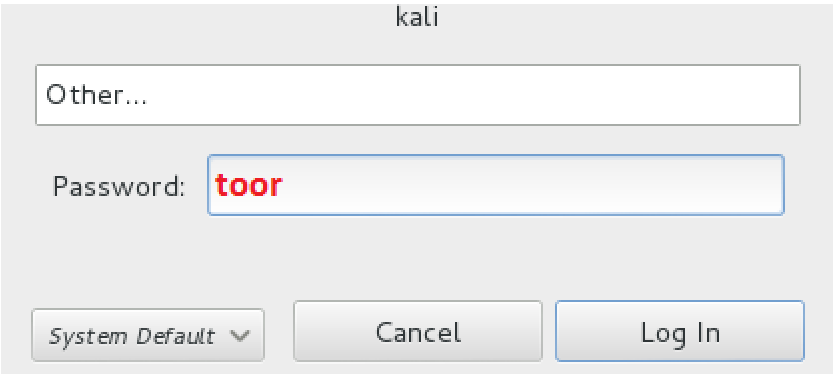


2. For the username for the Kali system, type **root**, then click the **Log in** button.



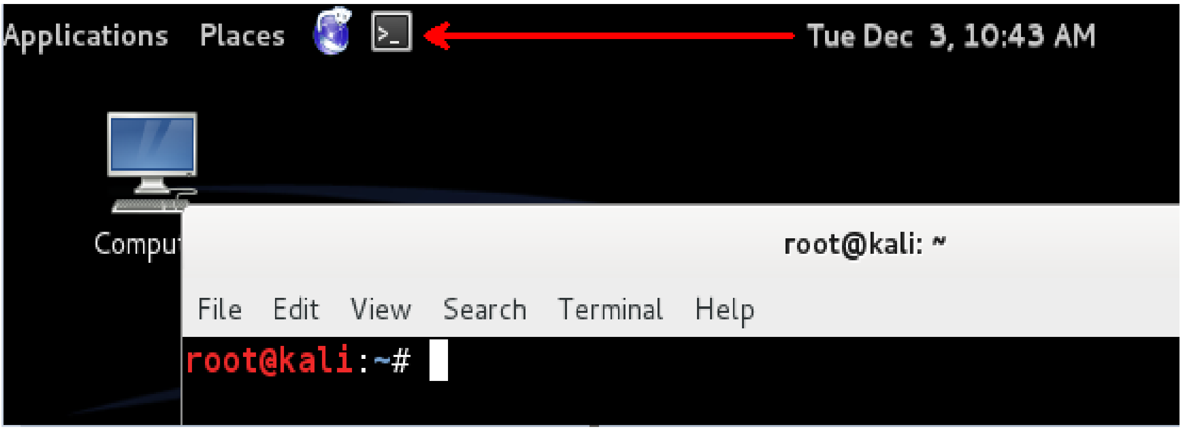
Username: root

3. For the password, type **toor**, then click the **Log In** button:



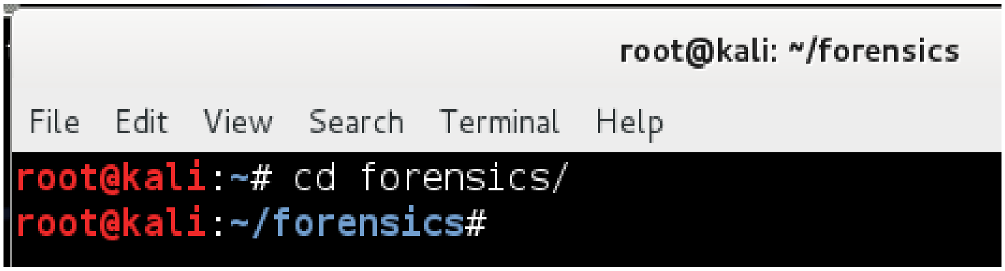
Password: toor

4. Open a terminal by clicking on the black icon to the right of the world icon.



```
root@kali:~#
```

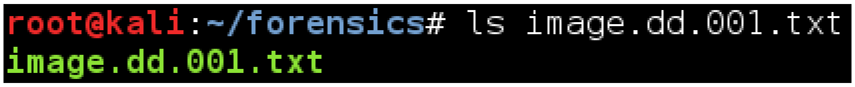
5. Switch to the forensics directory by typing the following command:
root@kali:~# **cd forensics**



```
root@kali:~# cd forensics/
root@kali:~/forensics#
```

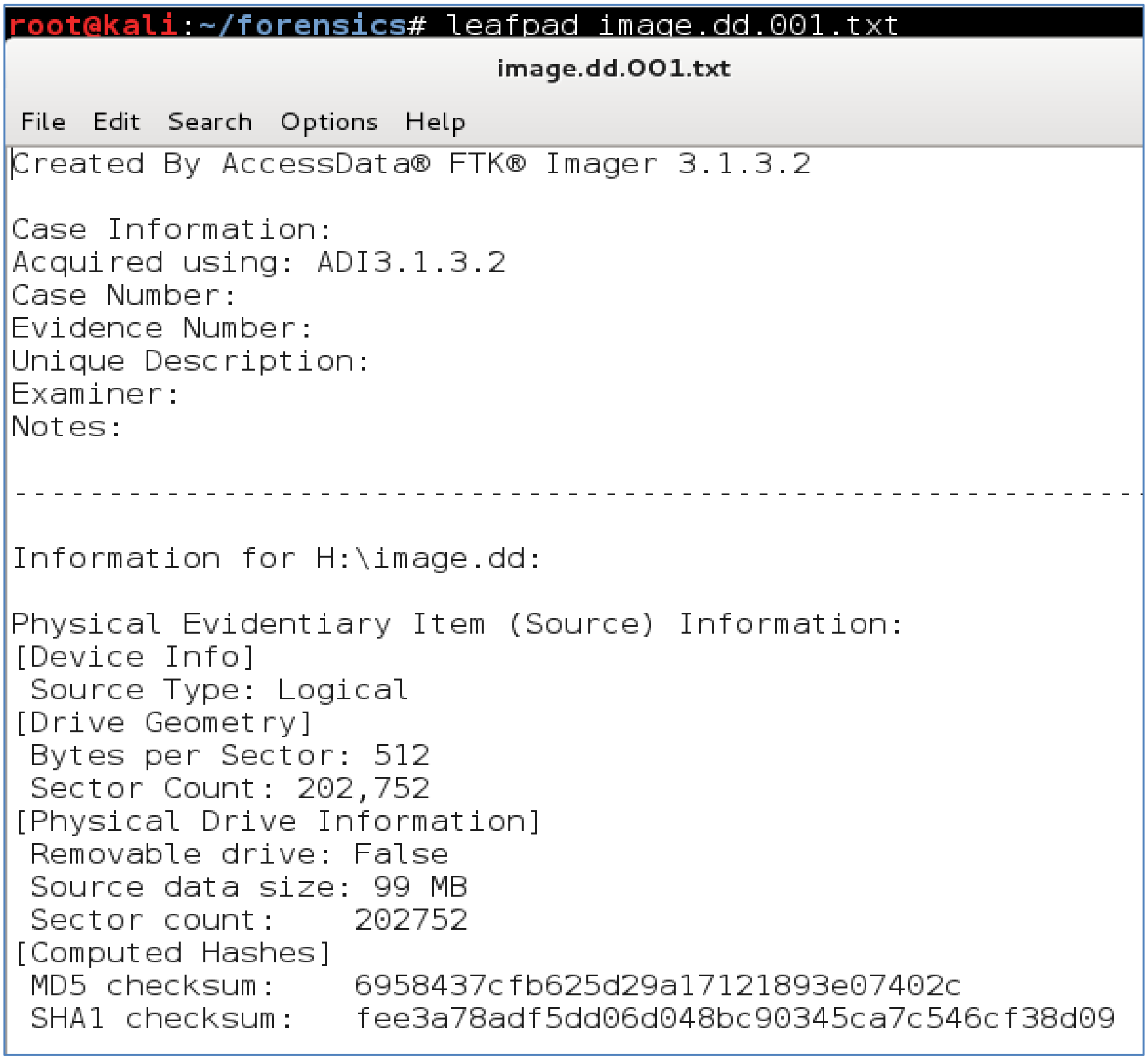
When an investigator takes an image using FTK Imager, they receive a corresponding txt file with the SHA1 and MD5 hashes. The text file will have a name similar to the image file.

6. Type the following command to view the file with the hashing information:
   root@kali:~/forensics# **ls image.dd.001.txt**



```
root@kali:~/forensics# ls image.dd.001.txt
image.dd.001.txt
```

7. Type the following command to view the file from the GUI:
   root@kali:~/forensics# **leafpad image.dd.001.txt**

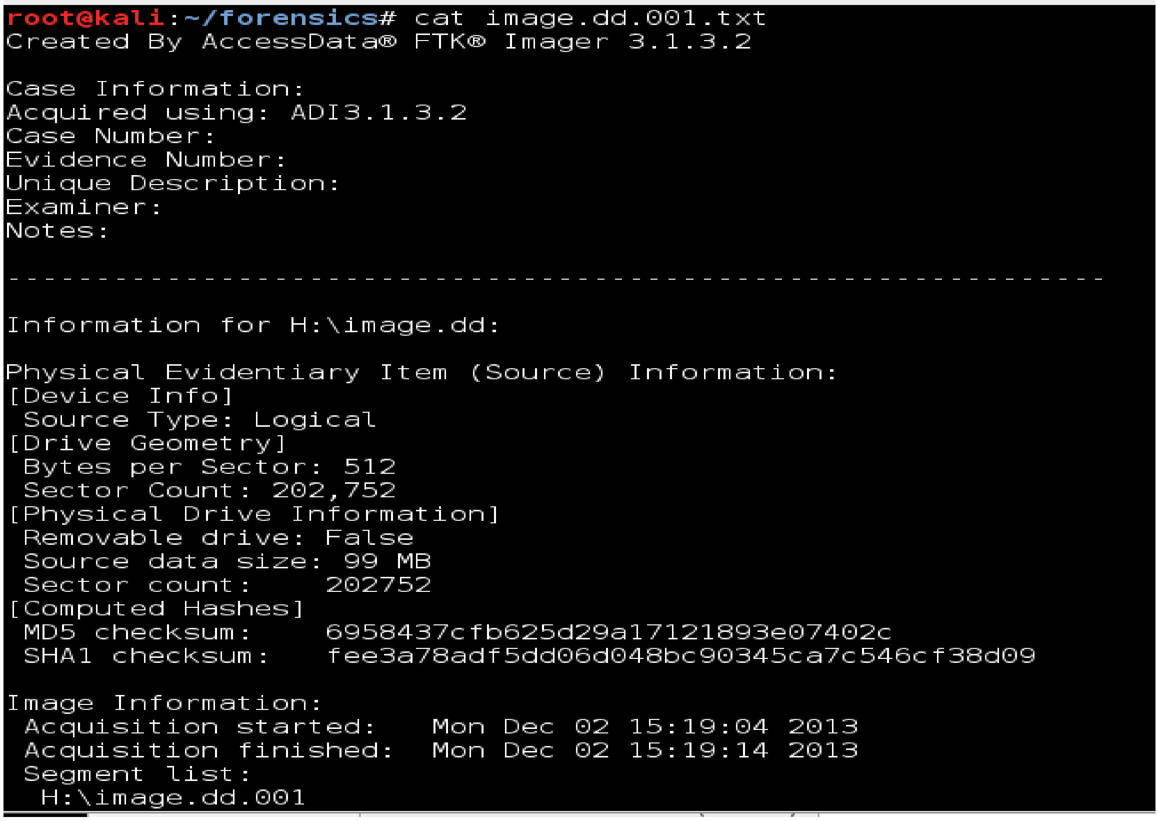


```
root@kali:~/forensics# leafpad image.dd.001.txt
```

```
Created By AccessData® FTK® Imager 3.1.3.2

Case Information:
Acquired using: ADI3.1.3.2
Case Number:
Evidence Number:
Unique Description:
Examiner:
Notes:

------------------------------------------------------------

Information for H:\image.dd:

Physical Evidentiary Item (Source) Information:
[Device Info]
 Source Type: Logical
[Drive Geometry]
 Bytes per Sector: 512
 Sector Count: 202,752
[Physical Drive Information]
 Removable drive: False
 Source data size: 99 MB
 Sector count:    202752
[Computed Hashes]
 MD5 checksum:    6958437cfb625d29a17121893e07402c
 SHA1 checksum:   fee3a78adf5dd06d048bc90345ca7c546cf38d09
```

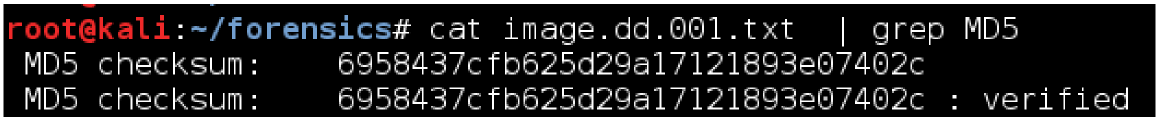
8. Close the file when you are finished viewing it with the leafpad application.

9. Type the following command to view the file contents from the terminal:
   root@kali:~/forensics# **cat image.dd.001.txt**



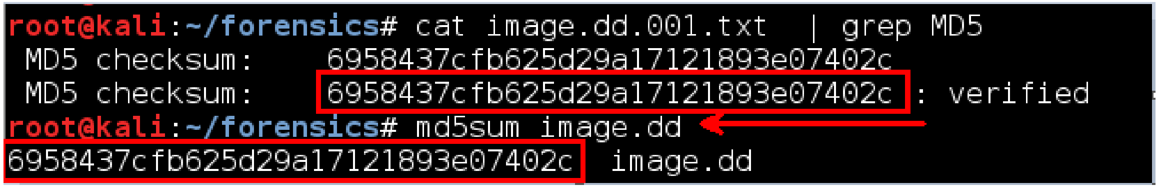
```
root@kali:~/forensics# cat image.dd.001.txt
Created By AccessData® FTK® Imager 3.1.3.2

Case Information:
Acquired using: ADI3.1.3.2
Case Number:
Evidence Number:
Unique Description:
Examiner:
Notes:

--------------------------------------------------------------

Information for H:\image.dd:

Physical Evidentiary Item (Source) Information:
[Device Info]
 Source Type: Logical
[Drive Geometry]
 Bytes per Sector: 512
 Sector Count: 202,752
[Physical Drive Information]
 Removable drive: False
 Source data size: 99 MB
 Sector count:    202752
[Computed Hashes]
 MD5 checksum:    6958437cfb625d29a17121893e07402c
 SHA1 checksum:   fee3a78adf5dd06d048bc90345ca7c546cf38d09

Image Information:
 Acquisition started:   Mon Dec 02 15:19:04 2013
 Acquisition finished:  Mon Dec 02 15:19:14 2013
 Segment list:
  H:\image.dd.001
```

8. Type the following command to view the MD5 hash:
   root@kali:~/forensics# **cat image.dd.001.txt | grep MD5**



```
root@kali:~/forensics# cat image.dd.001.txt  | grep MD5
 MD5 checksum:    6958437cfb625d29a17121893e07402c
 MD5 checksum:    6958437cfb625d29a17121893e07402c : verified
```

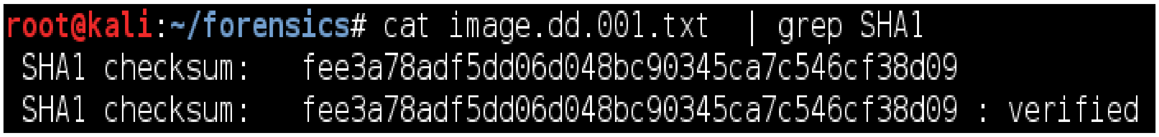
9. Type the following command to view the file with the hashing information:
   root@kali:~/forensics# **md5sum image.dd**



```
root@kali:~/forensics# cat image.dd.001.txt  | grep MD5
 MD5 checksum:    6958437cfb625d29a17121893e07402c
 MD5 checksum:    6958437cfb625d29a17121893e07402c : verified
root@kali:~/forensics# md5sum image.dd
6958437cfb625d29a17121893e07402c  image.dd
```

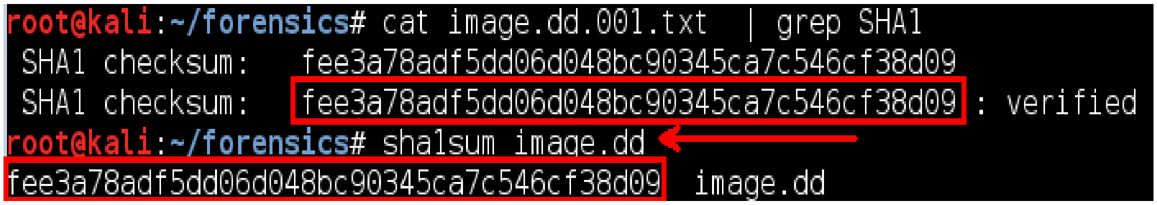
10. Notice that the MD5 sum matches the sum from the FTK acquisition text file.

11. Type the following command to view the SHA1 hash:
    root@kali:~/forensics# **cat image.dd.001.txt | grep SHA1**



```
root@kali:~/forensics# cat image.dd.001.txt  | grep SHA1
 SHA1 checksum:   fee3a78adf5dd06d048bc90345ca7c546cf38d09
 SHA1 checksum:   fee3a78adf5dd06d048bc90345ca7c546cf38d09 : verified
```

12. Type the following command to view the file with the hashing information:
    root@kali:~/forensics# **sha1sum image.dd**



```
root@kali:~/forensics# cat image.dd.001.txt  | grep SHA1
 SHA1 checksum:   fee3a78adf5dd06d048bc90345ca7c546cf38d09
 SHA1 checksum:   fee3a78adf5dd06d048bc90345ca7c546cf38d09 : verified
root@kali:~/forensics# sha1sum image.dd
fee3a78adf5dd06d048bc90345ca7c546cf38d09  image.dd
```

13. Notice that the SHA1 sum matches the sum from the FTK acquisition text file.

1.2 Conclusion

When an image is collected with FTK Imager, the incident responder gets a corresponding text file with the image MD5 and SHA1 hash values, as well as the acquisition date and time information. The md5sum and sha1sum utilities can be utilized from the terminal to hash a data set to verify the integrity of the data.

1.3 Discussion Questions

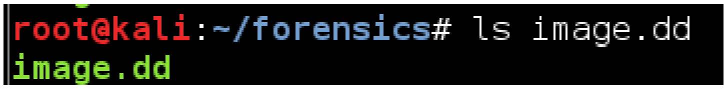
1. What is the default username and password for Kali Linux?
2. How many bits is the MD5 hashing algorithm?
3. How many bits is the SHA1 hashing algorithm?
4. Which hashing algorithm is more accurate, MD5 or SHA1?

2 Mounting a Partition with Deleted Files and Folders

It is imperative that all forensics investigators who utilize Linux know how to use the mount command. The mount command is used so the user can access files and folders on a partition. The mount command can be used with the read-only option, which will help to prevent the investigator from contaminating the disk.

2.1 Mounting a Partition

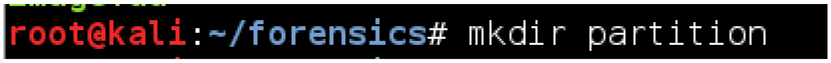
1. Type the following command to view the image.dd file located in the folder: root@kali:~/forensics# **ls image.dd**



```
root@kali:~/forensics# ls image.dd
image.dd
```

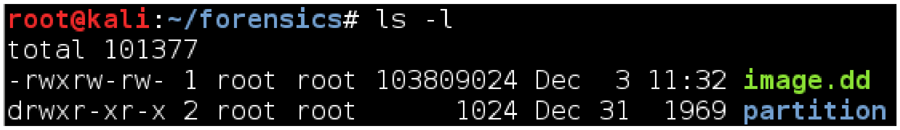
Next, we will create a folder to mount the partition to, or *mount point*. Typically, directories designated as mount points are created in the /mnt or /media directory on Linux systems. However, partitions can be mounted to any folder on the disk. It is always a best practice to avoid mounting to a directory in use, like /etc or /usr.

2. Type the following command to make a directory called partition: root@kali:~/forensics# **mkdir partition**



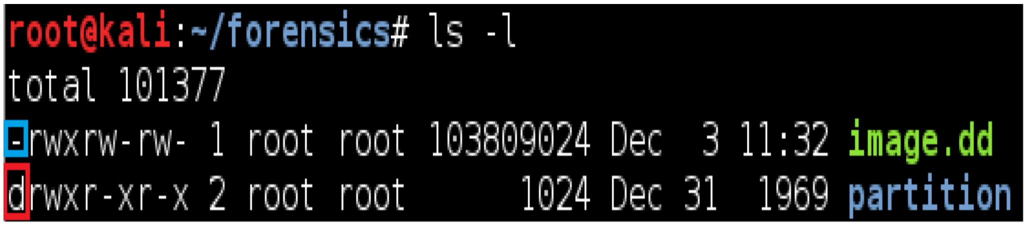
```
root@kali:~/forensics# mkdir partition
```

3. Type the following command to view your newly created directory: root@kali:~/forensics# **ls -l**



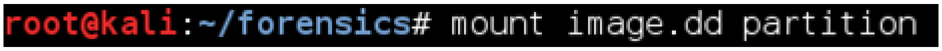
```
root@kali:~/forensics# ls -l
total 101377
-rwxrw-rw- 1 root root 103809024 Dec  3 11:32 image.dd
drwxr-xr-x 2 root root      1024 Dec 31  1969 partition
```

When Linux permissions are listed using ls –l, the first column designates whether the item is a file or a folder. A dash indicates a file. A d indicates that it is a directory.



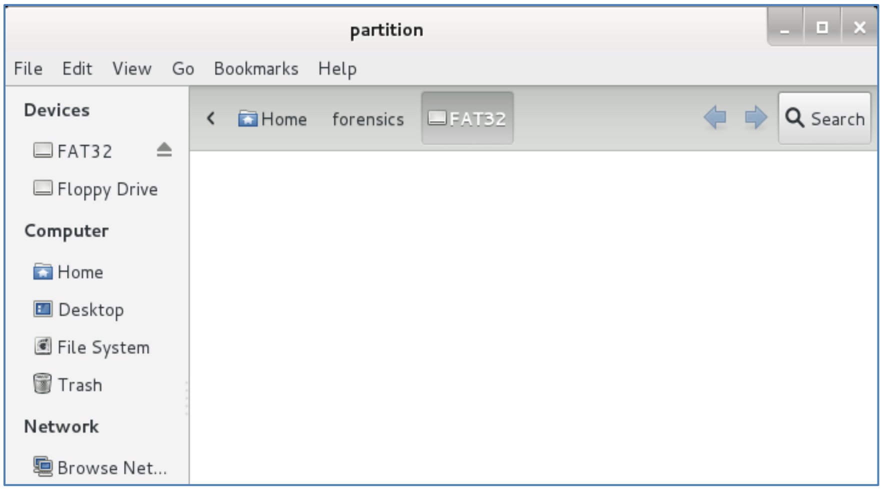
```
root@kali:~/forensics# ls -l
total 101377
-rwxrw-rw- 1 root root 103809024 Dec  3 11:32 image.dd
drwxr-xr-x 2 root root      1024 Dec 31  1969 partition
```

4. Mount the image to the partition directory by typing the following command:
   root@kali:~/forensics# **mount image.dd partition**



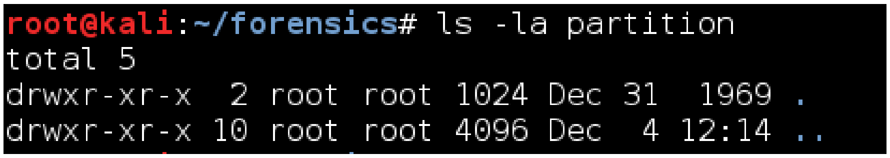
```
root@kali:~/forensics# mount image.dd partition
```

A Window will appear with a volume name of FAT32. Notice that no files are present.



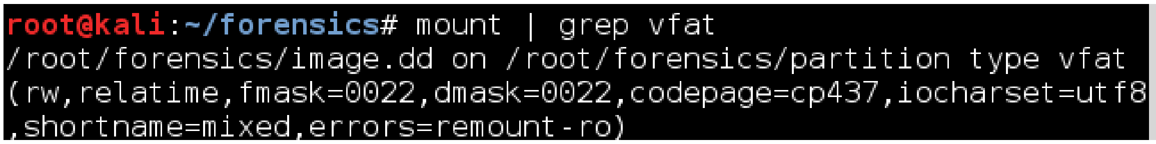
When we mounted the disk to try to recover the files on the partition, no files were present. In Task 2, we will use Foremost to recover deleted files from the disk.

5. Close the partition window.
6. To verify that no hidden files are present within the mounted image, type:
   root@kali:~/forensics# **ls -la partition**



```
root@kali:~/forensics# ls -la partition
total 5
drwxr-xr-x  2 root root 1024 Dec 31  1969 .
drwxr-xr-x 10 root root 4096 Dec  4 12:14 ..
```

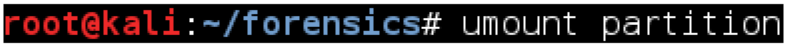
7. To verify that the image is mounted from the terminal, type:
   root@kali:~/forensics# **mount | grep vfat**



```
root@kali:~/forensics# mount | grep vfat
/root/forensics/image.dd on /root/forensics/partition type vfat
(rw,relatime,fmask=0022,dmask=0022,codepage=cp437,iocharset=utf8
,shortname=mixed,errors=remount-ro)
```

There are no files to recover. We will use the umount command to unmount the device.

8. Type the following command to umount the image.dd partition image.
   root@kali:~/forensics# **umount partition**



```
root@kali:~/forensics# umount partition
```

2.2 Conclusion

Mounting a partition can involve creating a mount point, then designating that folder so the partition can be mounted. Once the disk partition is mounted, the files and folders can be accessed by the end users. Typically, folders in /mnt or /media are used as mount points on Linux systems, but any directory can serve as a mount point.

2.3 Discussion Questions

1. What is the command to unmount a partition?
2. In what directories are mount points typically created on Linux?
3. When the ls –l command is used, what is designated in the first column to indicate a file?
4. When the ls –l command is used, what is designated in the first column to indicate a folder?

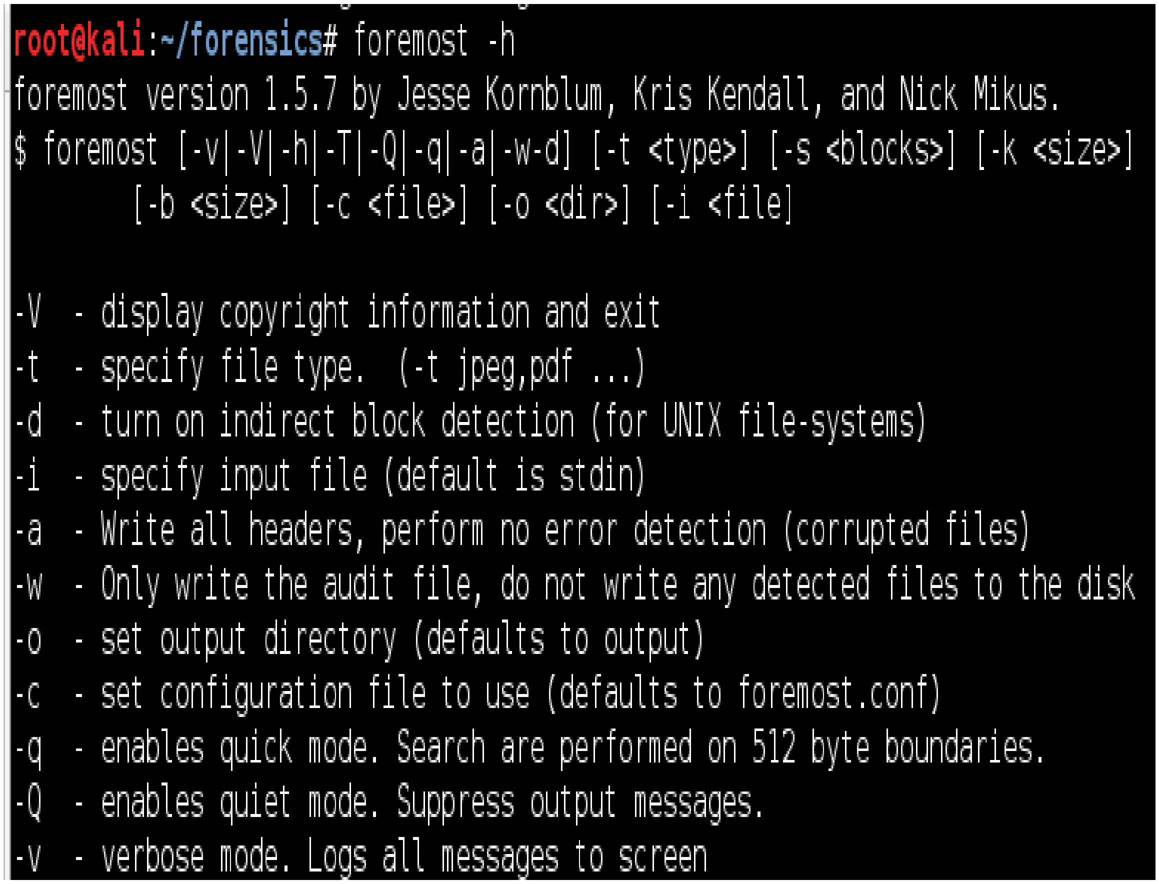
3 Using Foremost to Carve Files

One of the tasks that forensics investigators typically perform is recovering files. There are many single purpose recovery tools, such as Foremost, that allow users to recover files.

One of my colleagues told me that the first place he looked on a hard drive image was the Recycle Bin to see what the person deleted, since you can often learn a lot about a person from knowing what they were trying to delete.

3.1 Using Foremost

1. Type the following command to view the available options for Foremost:
   root@kali:~/forensics# **foremost -h**

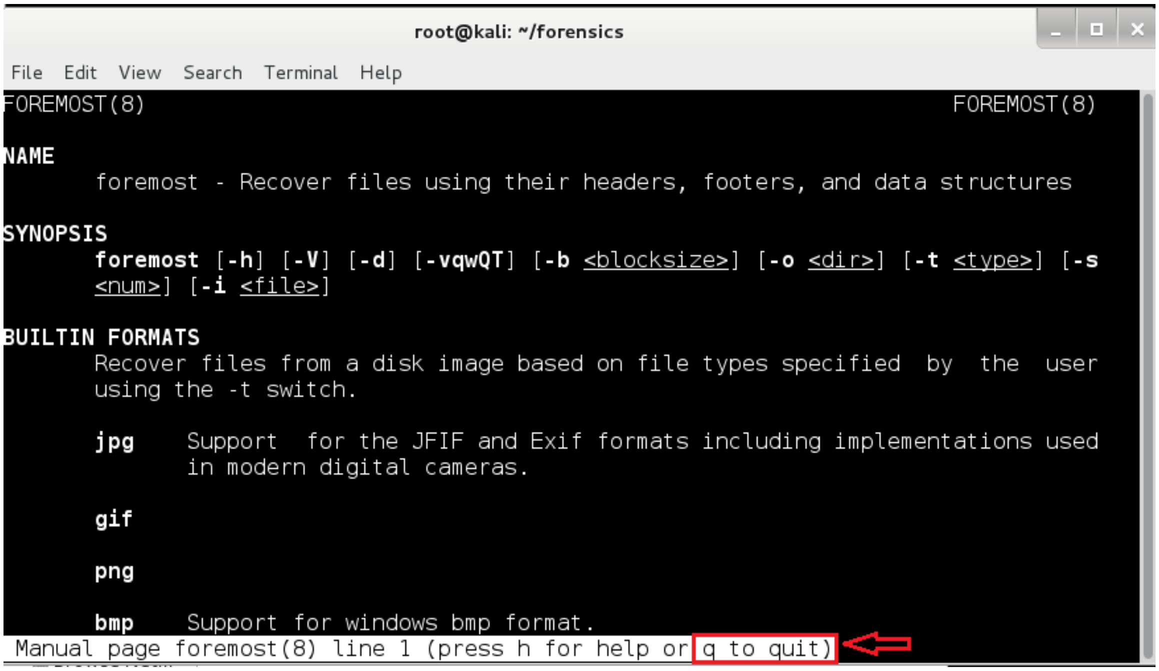


```
root@kali:~/forensics# foremost -h
foremost version 1.5.7 by Jesse Kornblum, Kris Kendall, and Nick Mikus.
$ foremost [-v|-V|-h|-T|-Q|-q|-a|-w-d] [-t <type>] [-s <blocks>] [-k <size>]
        [-b <size>] [-c <file>] [-o <dir>] [-i <file]

-V  - display copyright information and exit
-t  - specify file type.  (-t jpeg,pdf ...)
-d  - turn on indirect block detection (for UNIX file-systems)
-i  - specify input file (default is stdin)
-a  - Write all headers, perform no error detection (corrupted files)
-w  - Only write the audit file, do not write any detected files to the disk
-o  - set output directory (defaults to output)
-c  - set configuration file to use (defaults to foremost.conf)
-q  - enables quick mode. Search are performed on 512 byte boundaries.
-Q  - enables quiet mode. Suppress output messages.
-v  - verbose mode. Logs all messages to screen
```

2. To view detailed information about the foremost command, type the following:
   root@kali:~/forensics# **man foremost**

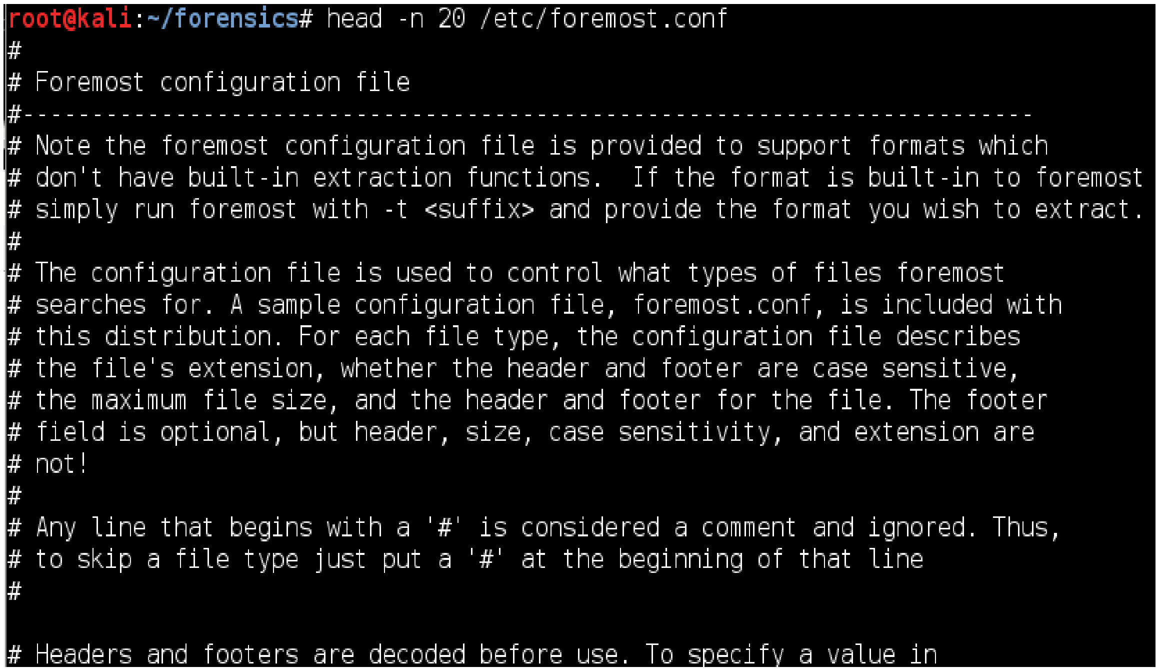
```
root@kali:~/forensics# man foremost
```

3. Click the **q** (quit) button to exit from the manual page.



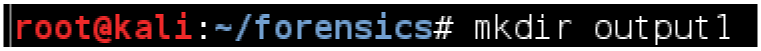
```
FOREMOST(8)                                                    FOREMOST(8)

NAME
       foremost - Recover files using their headers, footers, and data structures

SYNOPSIS
       foremost [-h] [-V] [-d] [-vqwQT] [-b <blocksize>] [-o <dir>] [-t <type>] [-s
       <num>] [-i <file>]

BUILTIN FORMATS
       Recover files from a disk image based on file types specified  by  the  user
       using the -t switch.

       jpg     Support  for the JFIF and Exif formats including implementations used
               in modern digital cameras.

       gif

       png

       bmp     Support for windows bmp format.
 Manual page foremost(8) line 1 (press h for help or q to quit)
```

4. View the configuration file for Foremost by typing the following command: root@kali:~/forensics# **head -n 20 /etc/foremost.conf**



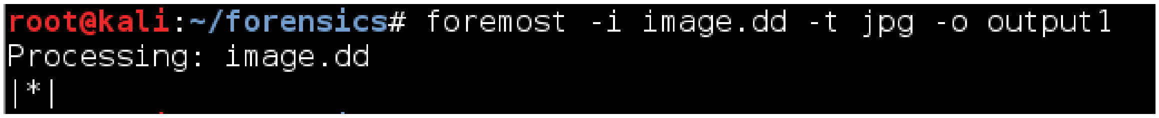
```
root@kali:~/forensics# head -n 20 /etc/foremost.conf
#
# Foremost configuration file
#-------------------------------------------------------------------------
# Note the foremost configuration file is provided to support formats which
# don't have built-in extraction functions.  If the format is built-in to foremost
# simply run foremost with -t <suffix> and provide the format you wish to extract.
#
# The configuration file is used to control what types of files foremost
# searches for. A sample configuration file, foremost.conf, is included with
# this distribution. For each file type, the configuration file describes
# the file's extension, whether the header and footer are case sensitive,
# the maximum file size, and the header and footer for the file. The footer
# field is optional, but header, size, case sensitivity, and extension are
# not!
#
# Any line that begins with a '#' is considered a comment and ignored. Thus,
# to skip a file type just put a '#' at the beginning of that line
#

# Headers and footers are decoded before use. To specify a value in
```

5. Make an output directory for the carved files by typing the following command:
   root@kali:~/forensics# **mkdir output1**



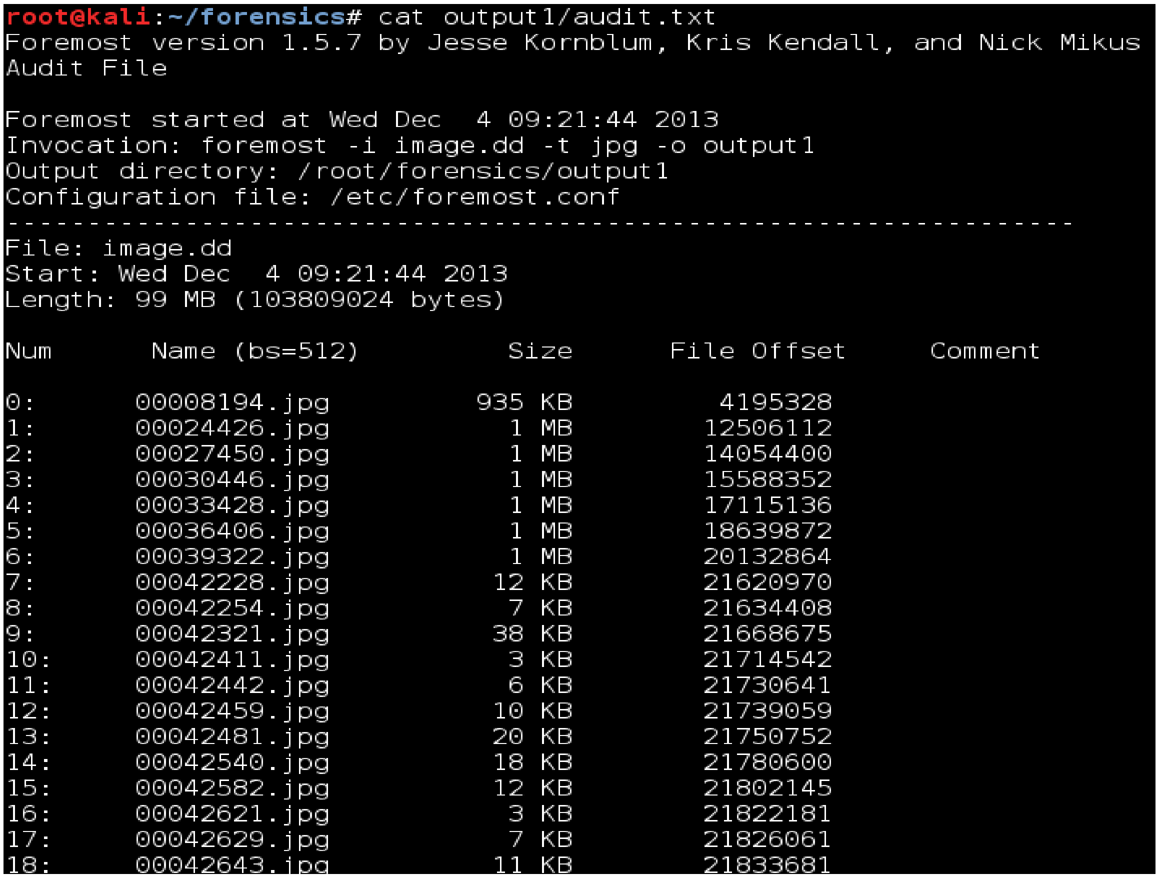
```
root@kali:~/forensics# mkdir output1
```

6. Type the following command to carve JPG files from the image file:
   kali:~/forensics# **foremost -i image.dd -t jpg -o output1**



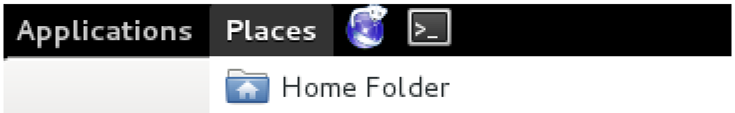
```
root@kali:~/forensics# foremost -i image.dd -t jpg -o output1
Processing: image.dd
|*|
```

7. Type the following to view the audit log for the carved jpeg files (total of 83):
   root@kali:~/forensics# **cat output1/audit.txt**

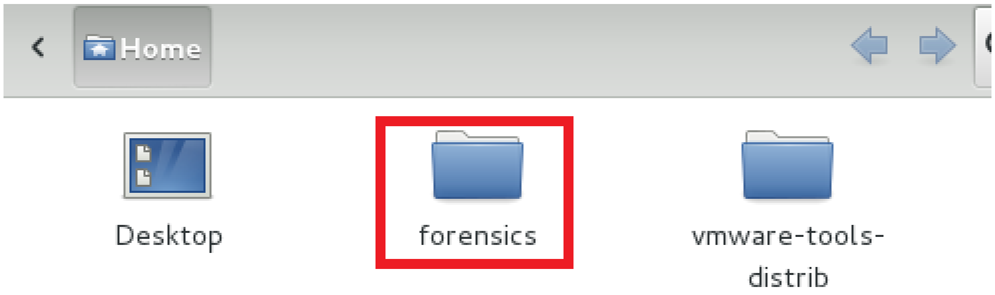


```
root@kali:~/forensics# cat output1/audit.txt
Foremost version 1.5.7 by Jesse Kornblum, Kris Kendall, and Nick Mikus
Audit File

Foremost started at Wed Dec  4 09:21:44 2013
Invocation: foremost -i image.dd -t jpg -o output1
Output directory: /root/forensics/output1
Configuration file: /etc/foremost.conf
------------------------------------------------------------------
File: image.dd
Start: Wed Dec  4 09:21:44 2013
Length: 99 MB (103809024 bytes)

Num      Name (bs=512)         Size      File Offset     Comment

0:      00008194.jpg        935 KB         4195328
1:      00024426.jpg          1 MB        12506112
2:      00027450.jpg          1 MB        14054400
3:      00030446.jpg          1 MB        15588352
4:      00033428.jpg          1 MB        17115136
5:      00036406.jpg          1 MB        18639872
6:      00039322.jpg          1 MB        20132864
7:      00042228.jpg         12 KB        21620970
8:      00042254.jpg          7 KB        21634408
9:      00042321.jpg         38 KB        21668675
10:     00042411.jpg          3 KB        21714542
11:     00042442.jpg          6 KB        21730641
12:     00042459.jpg         10 KB        21739059
13:     00042481.jpg         20 KB        21750752
14:     00042540.jpg         18 KB        21780600
15:     00042582.jpg         12 KB        21802145
16:     00042621.jpg          3 KB        21822181
17:     00042629.jpg          7 KB        21826061
18:     00042643.jpg         11 KB        21833681
```

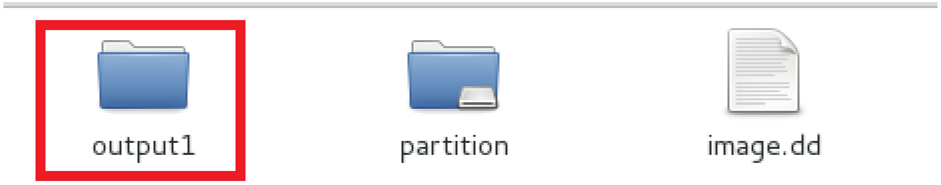
8. From the top menu bar, select **Places**, then navigate to the **Home Folder** link.



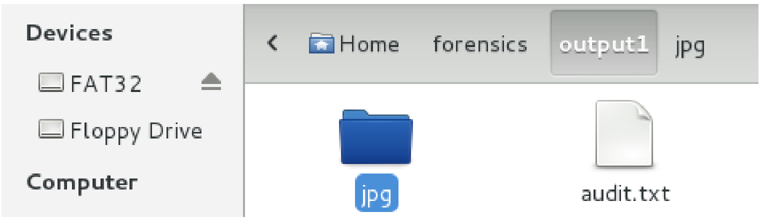
9. Double-click on the **forensics** folder within the Home folder.



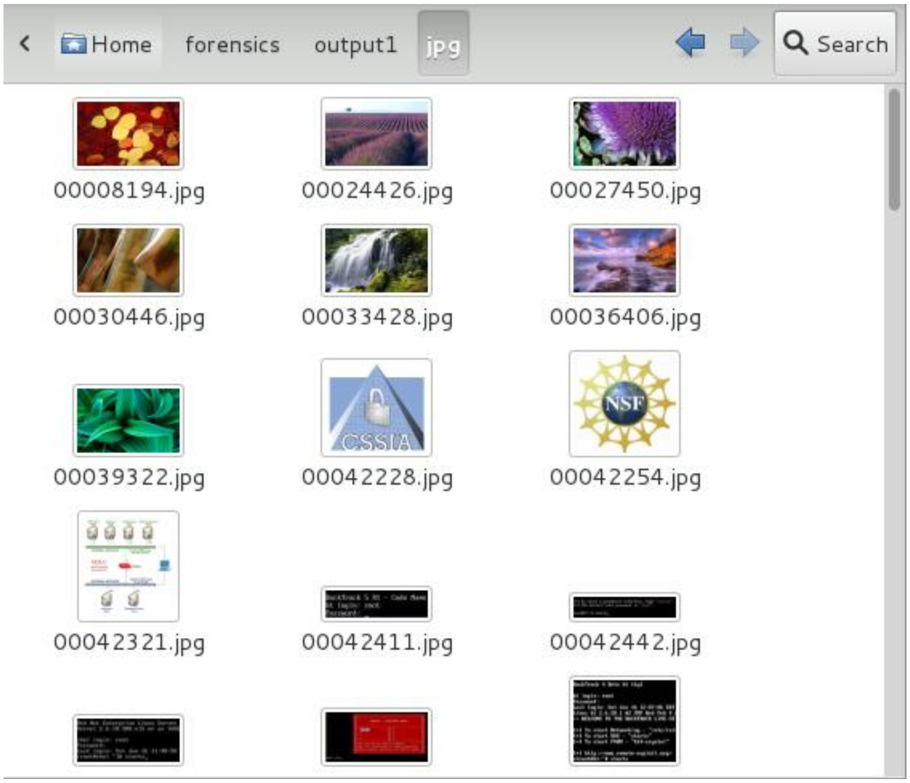
10. Double-click on the **output1** folder.



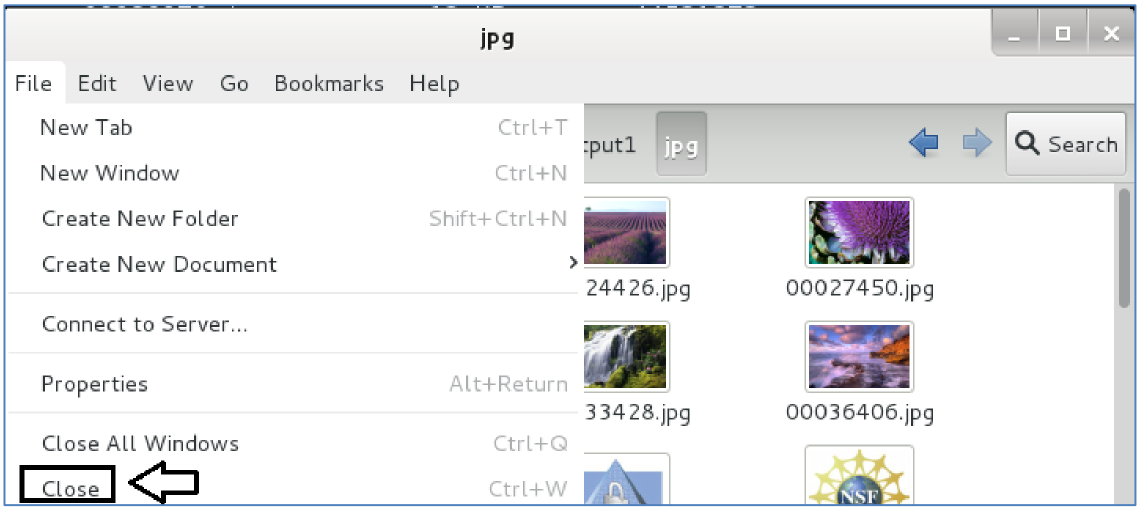
11. Double-click on the **jpg** folder, which is the location of the carved files.



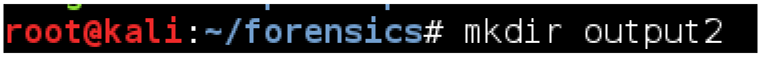
12. View the 38 files that were carved out by the Foremost utility.



13. Close the nautilus window by selecting **File** from the menu and choosing **Close** from the drop-down.

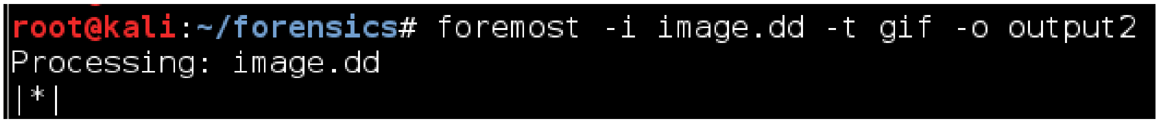


14. Make an output directory for the carved files by typing the following command:
    root@kali:~/forensics# **mkdir output2**



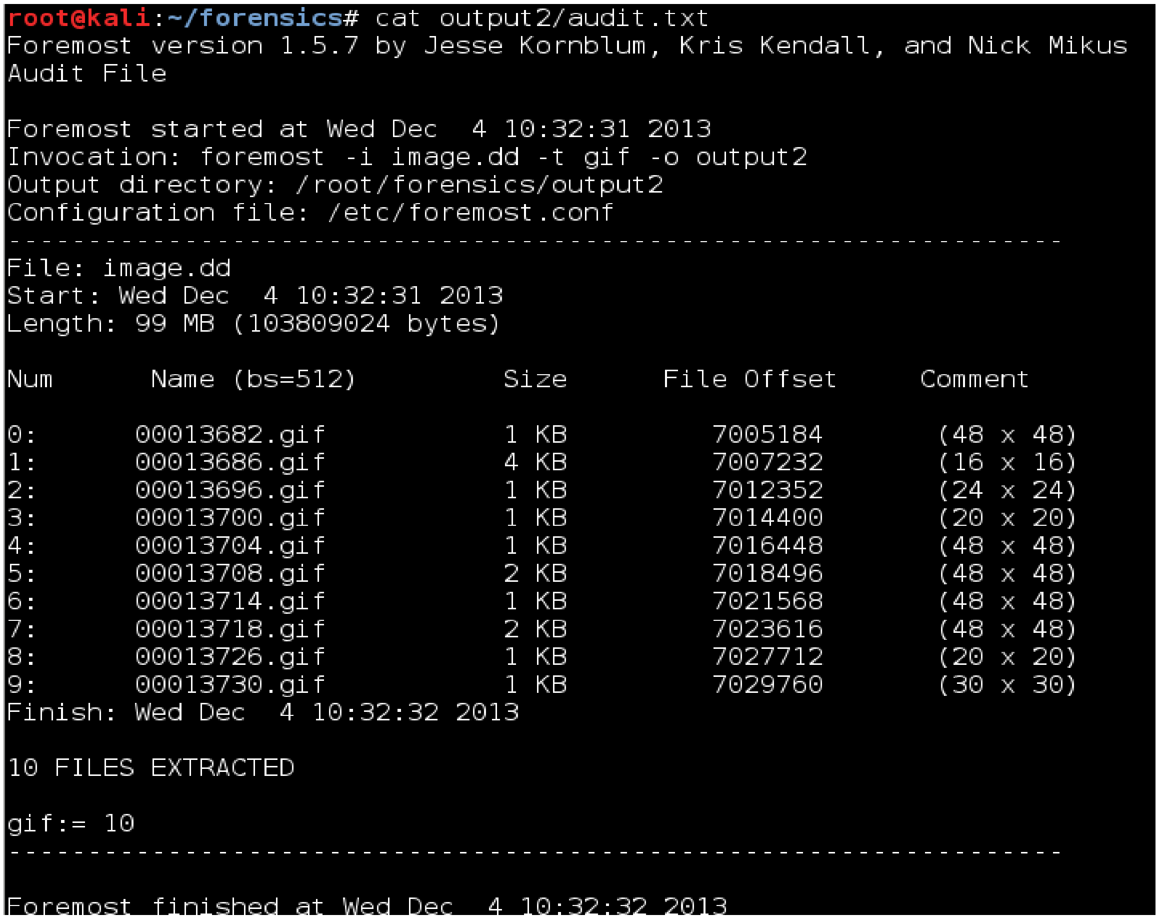
```
root@kali:~/forensics# mkdir output2
```

15. Type the following command to carve GIF files from the image file:
    kali:~/forensics# **foremost -i image.dd -t gif -o output2**



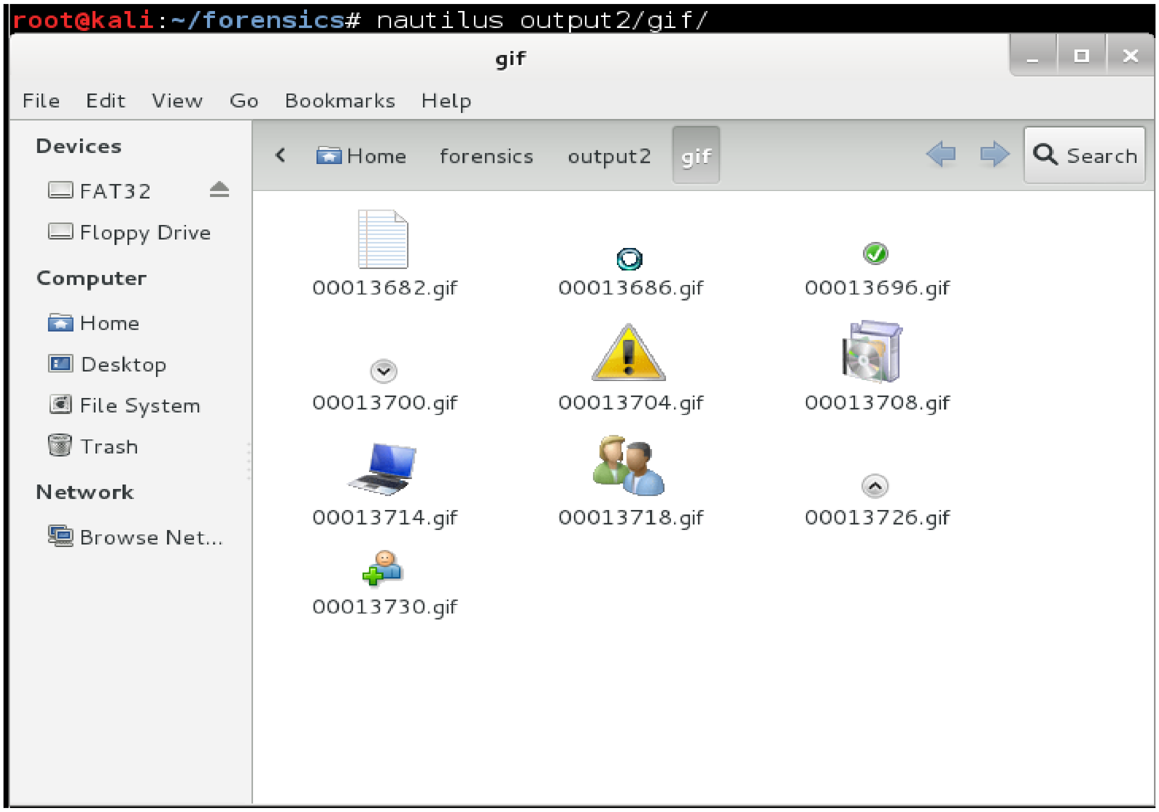
```
root@kali:~/forensics# foremost -i image.dd -t gif -o output2
Processing: image.dd
|*|
```

16. Type the following to view the audit log for the carved GIF files (total of 10):
    root@kali:~/forensics# **cat output2/audit.txt**



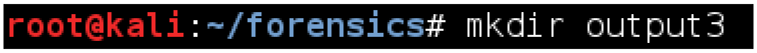
```
root@kali:~/forensics# cat output2/audit.txt
Foremost version 1.5.7 by Jesse Kornblum, Kris Kendall, and Nick Mikus
Audit File

Foremost started at Wed Dec  4 10:32:31 2013
Invocation: foremost -i image.dd -t gif -o output2
Output directory: /root/forensics/output2
Configuration file: /etc/foremost.conf
------------------------------------------------------------------
File: image.dd
Start: Wed Dec  4 10:32:31 2013
Length: 99 MB (103809024 bytes)

Num      Name (bs=512)          Size      File Offset     Comment

0:      00013682.gif           1 KB         7005184       (48 x 48)
1:      00013686.gif           4 KB         7007232       (16 x 16)
2:      00013696.gif           1 KB         7012352       (24 x 24)
3:      00013700.gif           1 KB         7014400       (20 x 20)
4:      00013704.gif           1 KB         7016448       (48 x 48)
5:      00013708.gif           2 KB         7018496       (48 x 48)
6:      00013714.gif           1 KB         7021568       (48 x 48)
7:      00013718.gif           2 KB         7023616       (48 x 48)
8:      00013726.gif           1 KB         7027712       (20 x 20)
9:      00013730.gif           1 KB         7029760       (30 x 30)
Finish: Wed Dec  4 10:32:32 2013

10 FILES EXTRACTED

gif:= 10
------------------------------------------------------------------

Foremost finished at Wed Dec  4 10:32:32 2013
```

17. Type the following to view files that were carved out by the Foremost utility:
    root@kali:~/forensics# **nautilus output2/gif/**



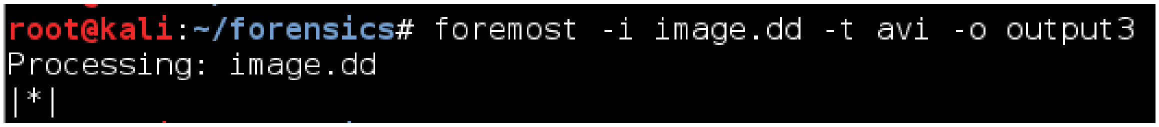
```
root@kali:~/forensics# nautilus output2/gif/
```

18. Close the nautilus window when you are finished viewing the carved out files.
19. Make an output directory for the carved files by typing the following command:
    root@kali:~/forensics# **mkdir output3**



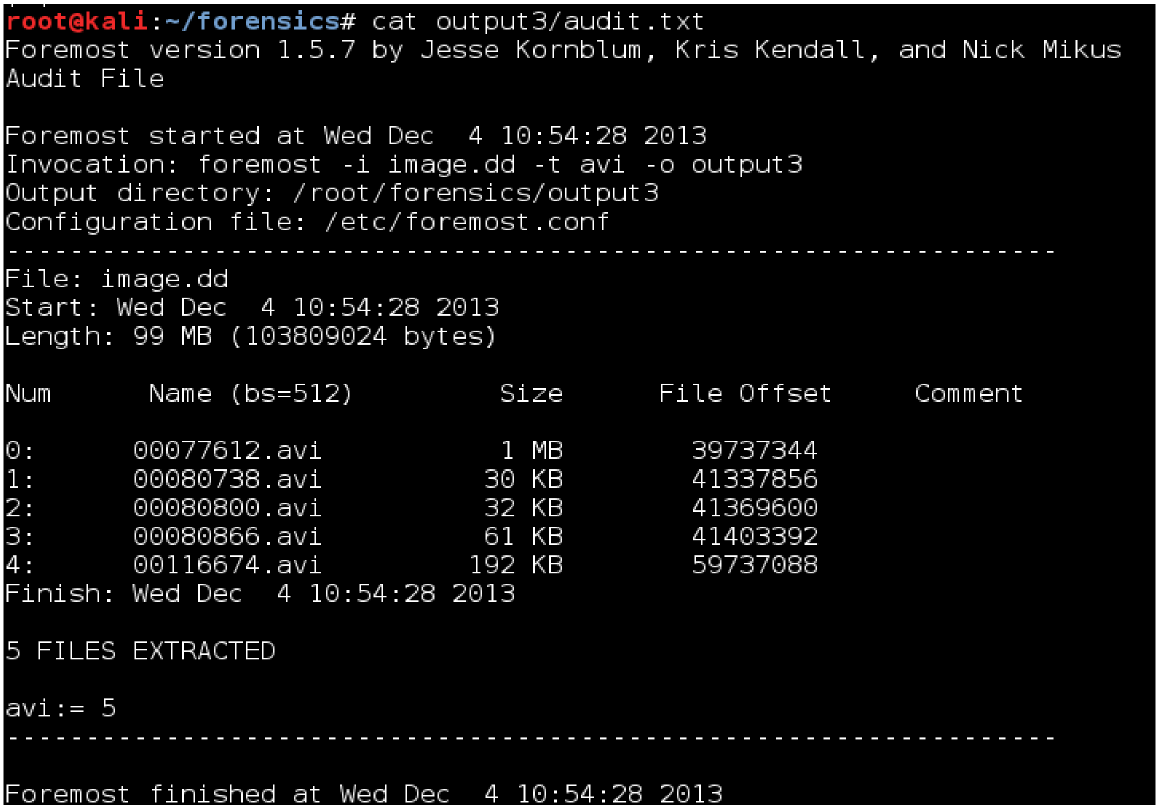
```
root@kali:~/forensics# mkdir output3
```

20. Type the following command to carve AVI files from the image file:
    kali:~/forensics# **foremost -i image.dd -t avi -o output3**



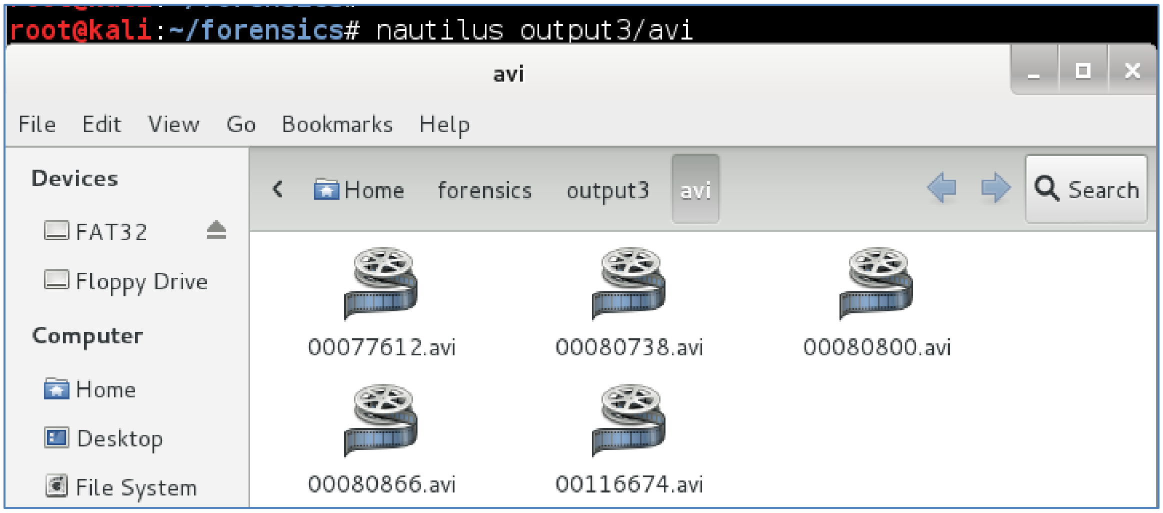
```
root@kali:~/forensics# foremost -i image.dd -t avi -o output3
Processing: image.dd
|*|
```

21. Type the following to view the audit log for the carved AVI files (total of 10):
    root@kali:~/forensics# **cat output3/audit.txt**



```
root@kali:~/forensics# cat output3/audit.txt
Foremost version 1.5.7 by Jesse Kornblum, Kris Kendall, and Nick Mikus
Audit File

Foremost started at Wed Dec  4 10:54:28 2013
Invocation: foremost -i image.dd -t avi -o output3
Output directory: /root/forensics/output3
Configuration file: /etc/foremost.conf
------------------------------------------------------------------
File: image.dd
Start: Wed Dec  4 10:54:28 2013
Length: 99 MB (103809024 bytes)

Num      Name (bs=512)         Size      File Offset     Comment

0:      00077612.avi           1 MB        39737344
1:      00080738.avi          30 KB        41337856
2:      00080800.avi          32 KB        41369600
3:      00080866.avi          61 KB        41403392
4:      00116674.avi         192 KB        59737088
Finish: Wed Dec  4 10:54:28 2013

5 FILES EXTRACTED

avi:= 5
------------------------------------------------------------------

Foremost finished at Wed Dec  4 10:54:28 2013
```

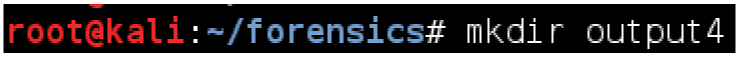
22. Type the following to view files that were carved out by the Foremost utility:
    root@kali:~/forensics# **nautilus output3/avi/**



```
root@kali:~/forensics# nautilus output3/avi
```

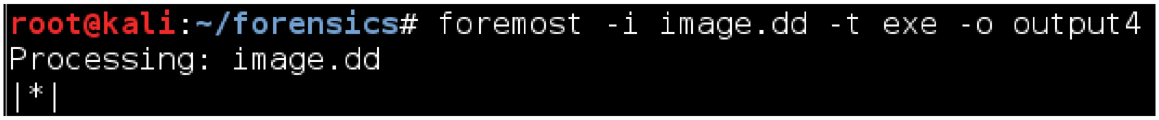
23. Close the nautilus window when you are finished viewing the carved out files.

24. Make an output directory for the carved files by typing the following command:
    root@kali:~/forensics# **mkdir output4**



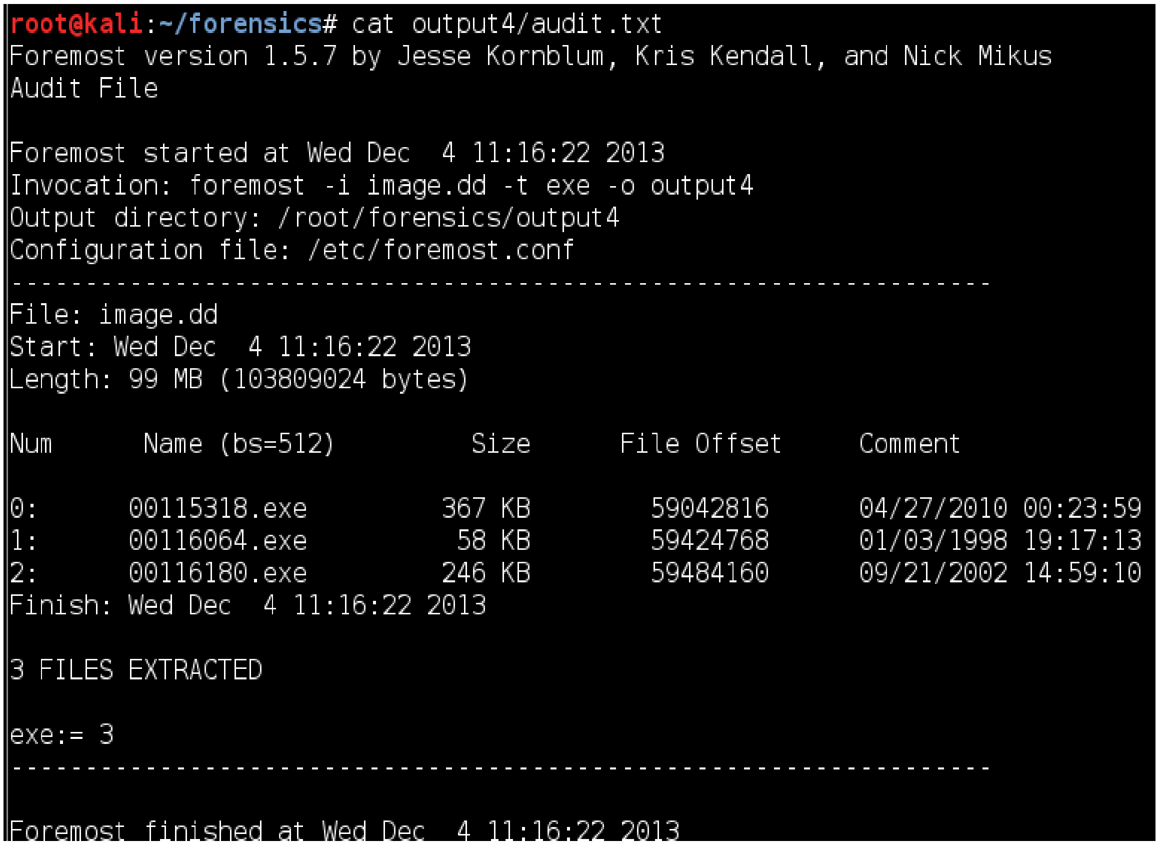
```
root@kali:~/forensics# mkdir output4
```

25. Type the following command to carve EXE files from the image file:
    kali:~/forensics# **foremost -i image.dd -t exe -o output4**



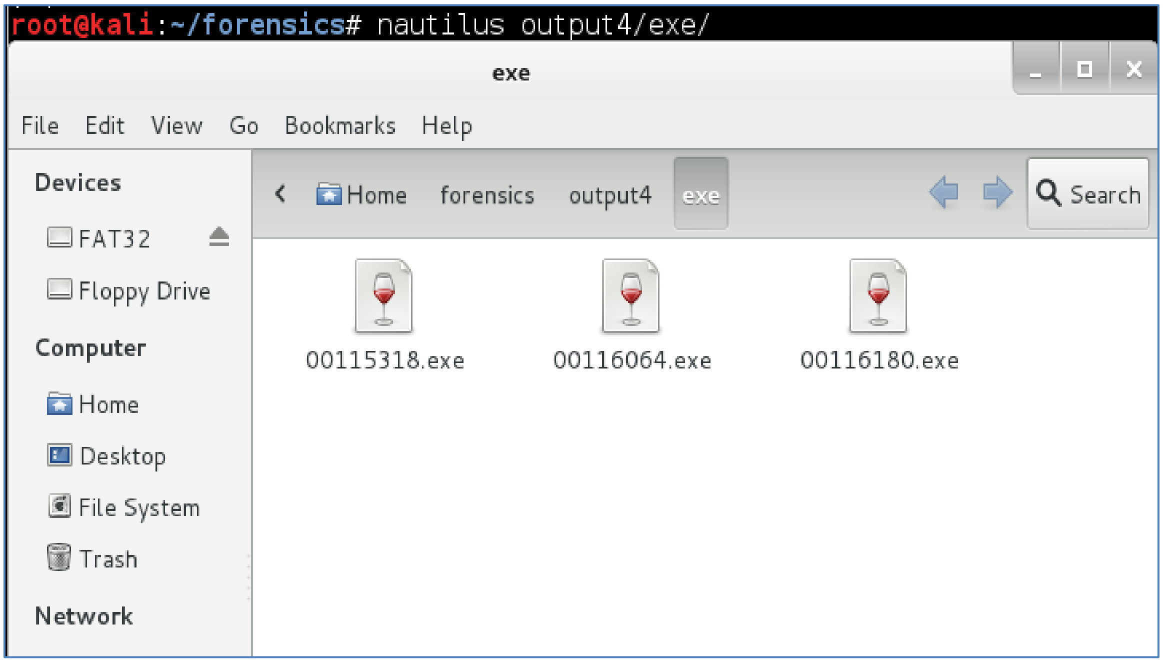
```
root@kali:~/forensics# foremost -i image.dd -t exe -o output4
Processing: image.dd
|*|
```

26. Type the following to view the audit log for the carved EXE files (total of 4):
    root@kali:~/forensics# **cat output4/audit.txt**



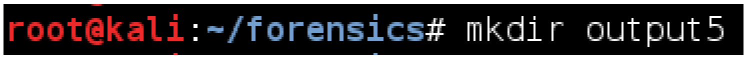
```
root@kali:~/forensics# cat output4/audit.txt
Foremost version 1.5.7 by Jesse Kornblum, Kris Kendall, and Nick Mikus
Audit File

Foremost started at Wed Dec  4 11:16:22 2013
Invocation: foremost -i image.dd -t exe -o output4
Output directory: /root/forensics/output4
Configuration file: /etc/foremost.conf
------------------------------------------------------------------
File: image.dd
Start: Wed Dec  4 11:16:22 2013
Length: 99 MB (103809024 bytes)

Num      Name (bs=512)         Size      File Offset     Comment

0:      00115318.exe         367 KB        59042816      04/27/2010 00:23:59
1:      00116064.exe          58 KB        59424768      01/03/1998 19:17:13
2:      00116180.exe         246 KB        59484160      09/21/2002 14:59:10
Finish: Wed Dec  4 11:16:22 2013

3 FILES EXTRACTED

exe:= 3
------------------------------------------------------------------

Foremost finished at Wed Dec  4 11:16:22 2013
```

27. Type the following to view files that were carved out by the Foremost utility:
    root@kali:~/forensics# **nautilus output4/exe/**



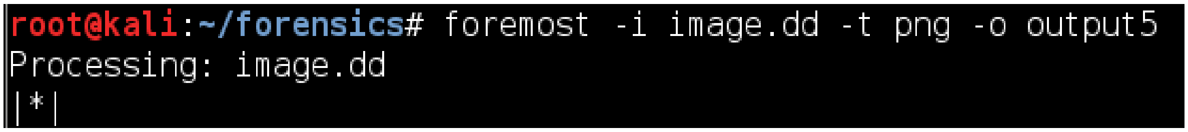
```
root@kali:~/forensics# nautilus output4/exe/
```

28. Close the nautilus window when you are finished viewing the carved out files.
29. Make an output directory for the carved files by typing the following command:
    root@kali:~/forensics# **mkdir output5**



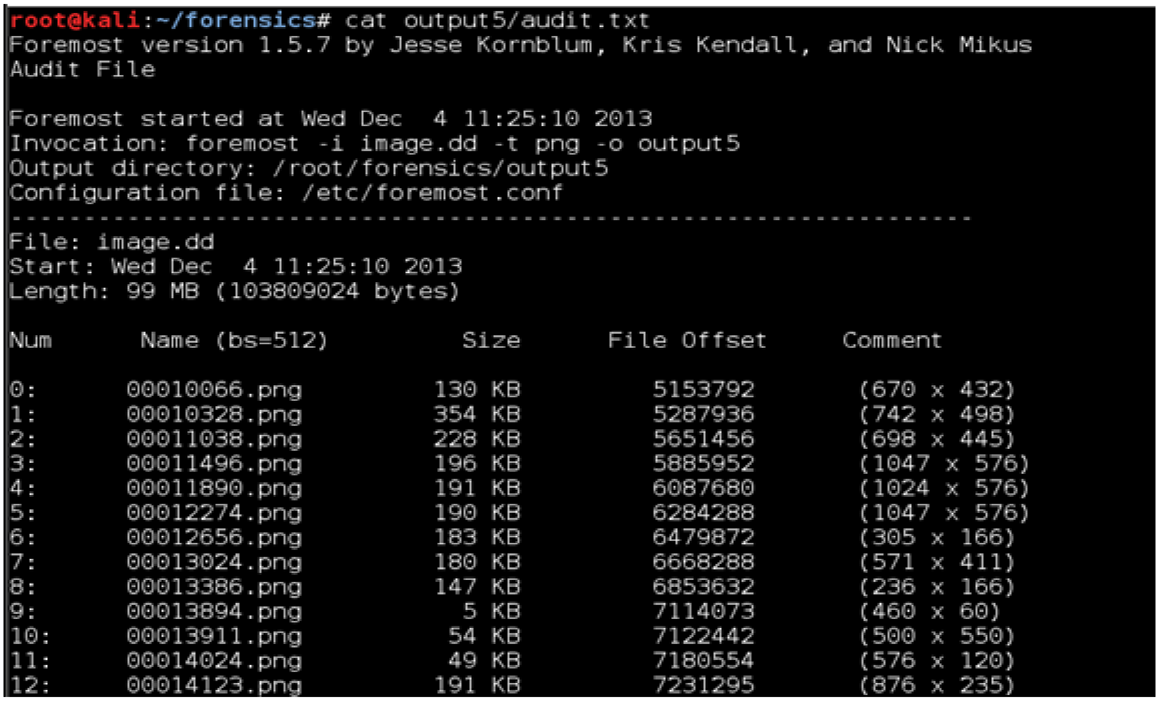
```
root@kali:~/forensics# mkdir output5
```

30. Type the following command to carve PNG files from the image file:
    kali:~/forensics# **foremost -i image.dd -t png -o output5**



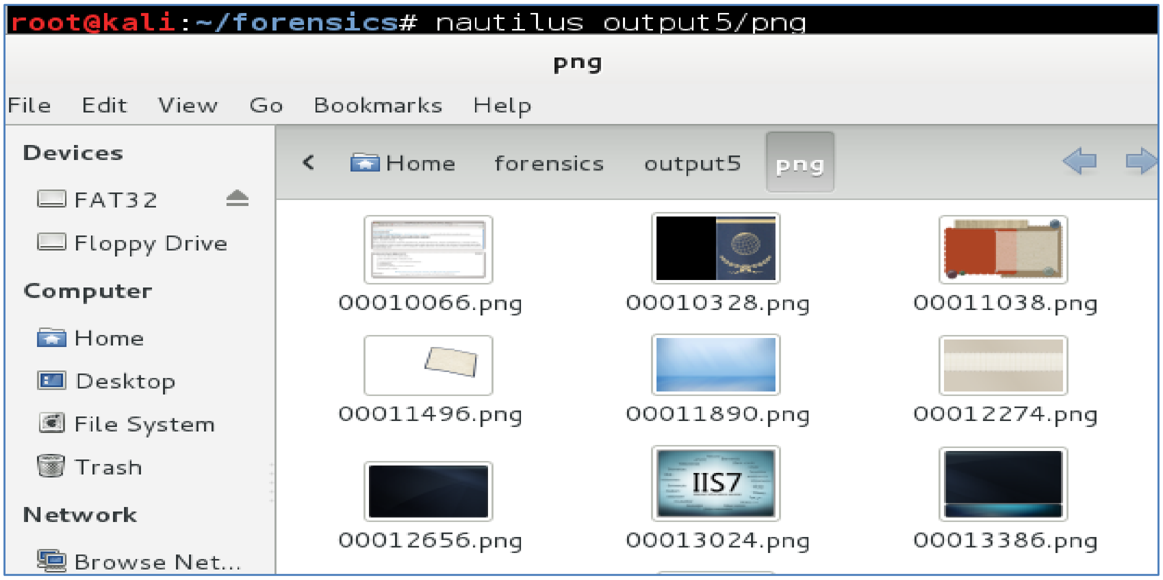
```
root@kali:~/forensics# foremost -i image.dd -t png -o output5
Processing: image.dd
|*|
```

31. Type the following to view the audit log for the carved PNG files (total of 230):
    root@kali:~/forensics# **cat output5/audit.txt**



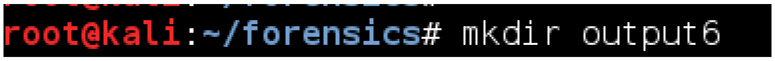
```
root@kali:~/forensics# cat output5/audit.txt
Foremost version 1.5.7 by Jesse Kornblum, Kris Kendall, and Nick Mikus
Audit File

Foremost started at Wed Dec  4 11:25:10 2013
Invocation: foremost -i image.dd -t png -o output5
Output directory: /root/forensics/output5
Configuration file: /etc/foremost.conf
------------------------------------------------------------------
File: image.dd
Start: Wed Dec  4 11:25:10 2013
Length: 99 MB (103809024 bytes)

Num      Name (bs=512)        Size      File Offset     Comment

0:      00010066.png        130 KB       5153792        (670 x 432)
1:      00010328.png        354 KB       5287936        (742 x 498)
2:      00011038.png        228 KB       5651456        (698 x 445)
3:      00011496.png        196 KB       5885952        (1047 x 576)
4:      00011890.png        191 KB       6087680        (1024 x 576)
5:      00012274.png        190 KB       6284288        (1047 x 576)
6:      00012656.png        183 KB       6479872        (305 x 166)
7:      00013024.png        180 KB       6668288        (571 x 411)
8:      00013386.png        147 KB       6853632        (236 x 166)
9:      00013894.png          5 KB       7114073        (460 x 60)
10:     00013911.png         54 KB       7122442        (500 x 550)
11:     00014024.png         49 KB       7180554        (576 x 120)
12:     00014123.png        191 KB       7231295        (876 x 235)
```

32. Type the following to view files that were carved out by the Foremost utility:
    root@kali:~/forensics# **nautilus output5/png**



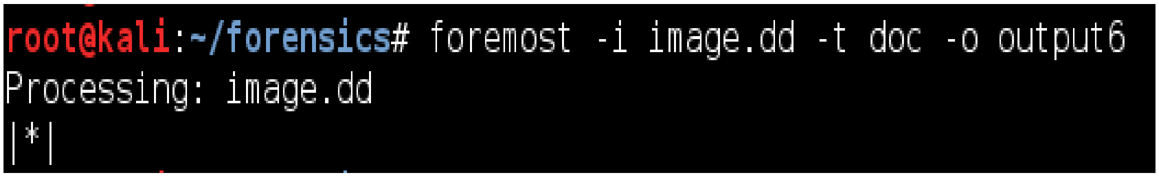
```
root@kali:~/forensics# nautilus output5/png
```

33. Close the nautilus window when you are finished viewing the carved out files.
34. Make an output directory for the carved files by typing the following command:
    root@kali:~/forensics# **mkdir output6**



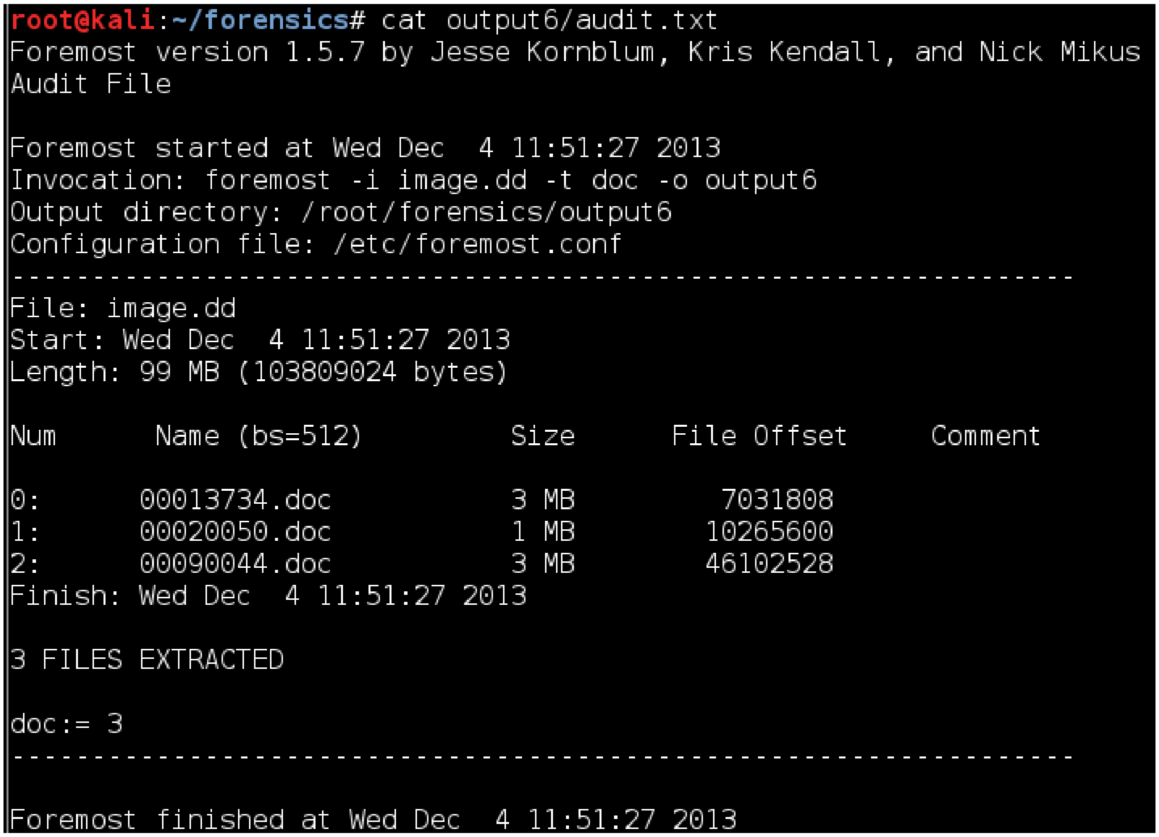
```
root@kali:~/forensics# mkdir output6
```

35. Type the following command to carve DOC files from the image file:
    kali:~/forensics# **foremost -i image.dd -t doc -o output6**



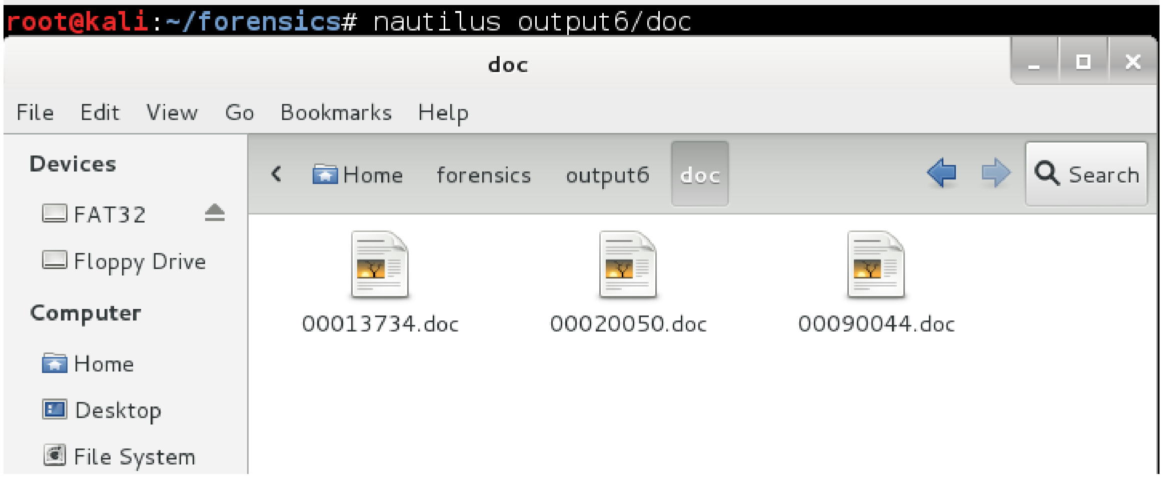
```
root@kali:~/forensics# foremost -i image.dd -t doc -o output6
Processing: image.dd
|*|
```

36. Type the following to view the audit log for the carved DOC files (total of 3):
    root@kali:~/forensics# **cat output6/audit.txt**



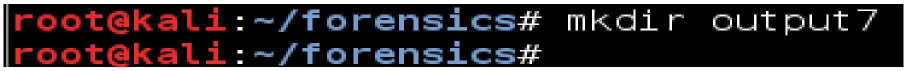
```
root@kali:~/forensics# cat output6/audit.txt
Foremost version 1.5.7 by Jesse Kornblum, Kris Kendall, and Nick Mikus
Audit File

Foremost started at Wed Dec  4 11:51:27 2013
Invocation: foremost -i image.dd -t doc -o output6
Output directory: /root/forensics/output6
Configuration file: /etc/foremost.conf
------------------------------------------------------------------
File: image.dd
Start: Wed Dec  4 11:51:27 2013
Length: 99 MB (103809024 bytes)

Num      Name (bs=512)          Size      File Offset     Comment

0:      00013734.doc           3 MB         7031808
1:      00020050.doc           1 MB        10265600
2:      00090044.doc           3 MB        46102528
Finish: Wed Dec  4 11:51:27 2013

3 FILES EXTRACTED

doc:= 3
------------------------------------------------------------------

Foremost finished at Wed Dec  4 11:51:27 2013
```

37. Type the following to view files that were carved out by the Foremost utility:
    root@kali:~/forensics# **nautilus output6/doc**



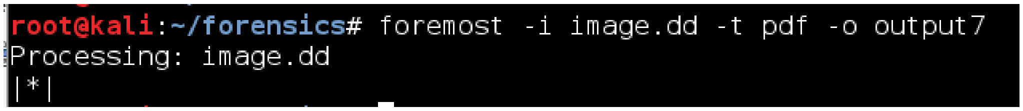
```
root@kali:~/forensics# nautilus output6/doc
```

38. Close the nautilus window when you are finished viewing the carved out files.
39. Make an output directory for the carved files by typing the following command:
    root@kali:~/forensics# **mkdir output7**



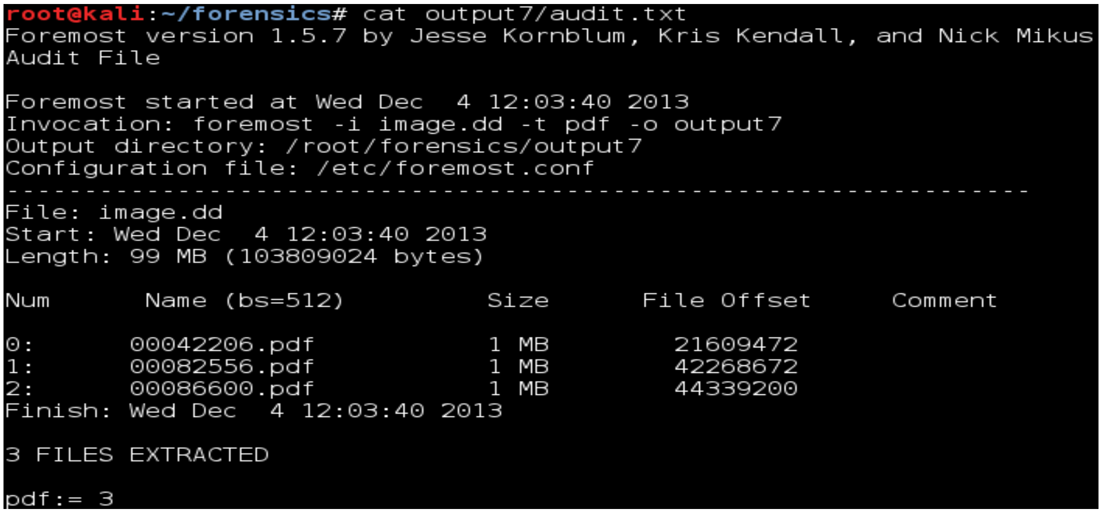
```
root@kali:~/forensics# mkdir output7
root@kali:~/forensics#
```

40. Type the following command to carve PDF files from the image file:
    kali:~/forensics# **foremost -i image.dd -t pdf -o output7**



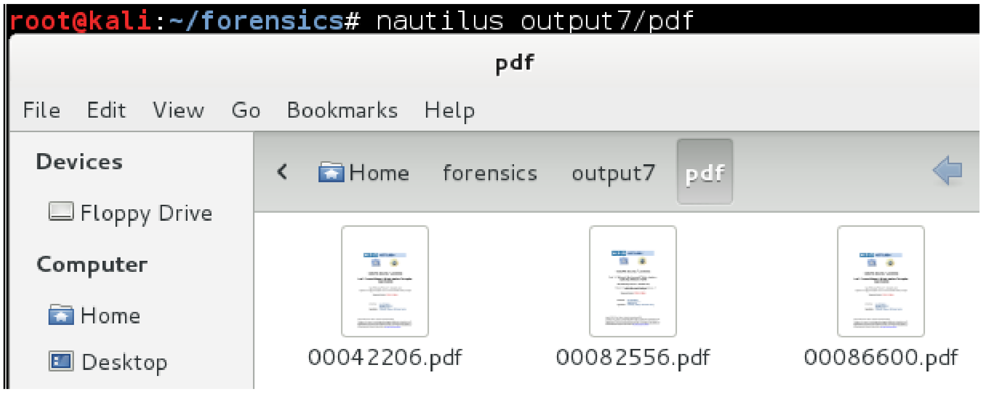
```
root@kali:~/forensics# foremost -i image.dd -t pdf -o output7
Processing: image.dd
|*|
```

41. Type the following to view the audit log for the carved PDF files (total of 3):
    root@kali:~/forensics# **cat output7/audit.txt**



```
root@kali:~/forensics# cat output7/audit.txt
Foremost version 1.5.7 by Jesse Kornblum, Kris Kendall, and Nick Mikus
Audit File

Foremost started at Wed Dec  4 12:03:40 2013
Invocation: foremost -i image.dd -t pdf -o output7
Output directory: /root/forensics/output7
Configuration file: /etc/foremost.conf
------------------------------------------------------------------
File: image.dd
Start: Wed Dec  4 12:03:40 2013
Length: 99 MB (103809024 bytes)

Num      Name (bs=512)         Size      File Offset     Comment

0:      00042206.pdf           1 MB        21609472
1:      00082556.pdf           1 MB        42268672
2:      00086600.pdf           1 MB        44339200
Finish: Wed Dec  4 12:03:40 2013

3 FILES EXTRACTED

pdf:= 3
```

42. Type the following to view files that were carved out by the Foremost utility:
    root@kali:~/forensics# **nautilus output7/pdf**



```
root@kali:~/forensics# nautilus output7/pdf
```

43. Close the nautilus window when you are finished viewing the carved out files.

3.2 Conclusion

Foremost is a tool written by Jesse Kornblum that carves files out of images. When Foremost is used, the file type that you want to carve from an image or partition must be specified. An audit.txt file will be generated with information about what was carved. The files carved are stored in a separate output folder with the name of the file type.

3.3 Discussion Questions

1. For what purpose is Foremost used?
2. What is the way to get the manual page for the foremost command?
3. What command can be utilized to get general information about Foremost?
4. What options must be specified when the foremost command is utilized?

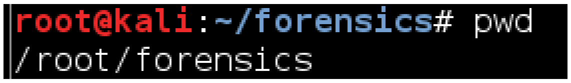
4 Using a HEX Editor

Hexadecimal is a numbering system where the numbers 0-9 and letters A-F are used. Also known as base 16, hexadecimal is commonly used in computer forensics and networking. HEX Editors are GUI or command line tools that can be utilized to analyze the hexadecimal code of files. File headers have hexadecimal signatures that are unique to a particular type of file. For example, a JPEG file has a file signature of JFIF. Tools like Foremost use signatures to carve out files from an image.

4.1 Using hexedit

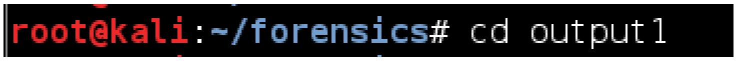
The output files created in the previous task (Task 3) are used in this task (Task 4). The previous task must be completed in order to proceed with this task.

1. Type the following to verify that you are currently in the forensics folder:
   root@kali:~/forensics# **pwd**



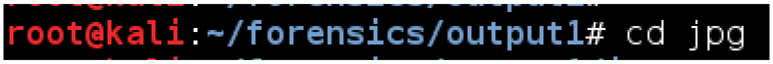
```
root@kali:~/forensics# pwd
/root/forensics
```

2. Type the following to enter the output1 directory within the forensics folder:
   root@kali:~/forensics# **cd output1**



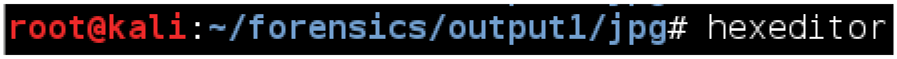
```
root@kali:~/forensics# cd output1
```

3. Type the following to enter the jpg folder within the output1 directory:
   root@kali:~/forensics/output1# **cd jpg**



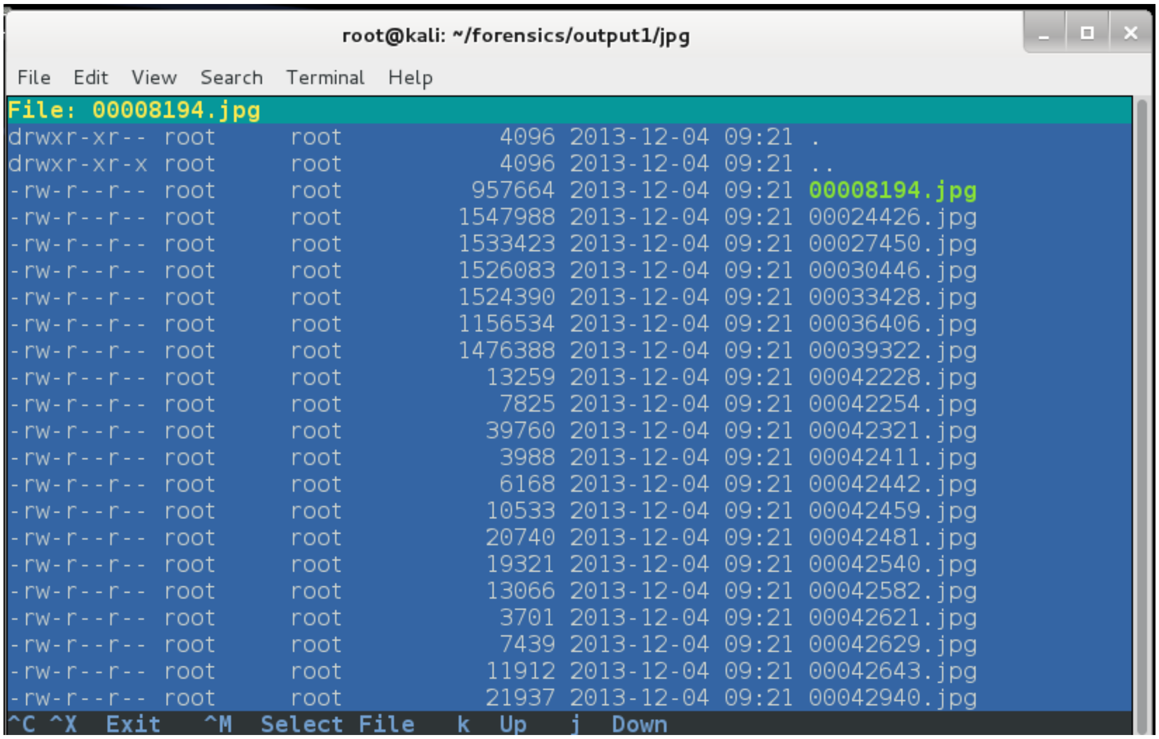
```
root@kali:~/forensics/output1# cd jpg
```

4. Type the following to use the hexeditor from the terminal:
   root@kali:~/forensics/output1/jpg# **hexeditor**



```
root@kali:~/forensics/output1/jpg# hexeditor
```

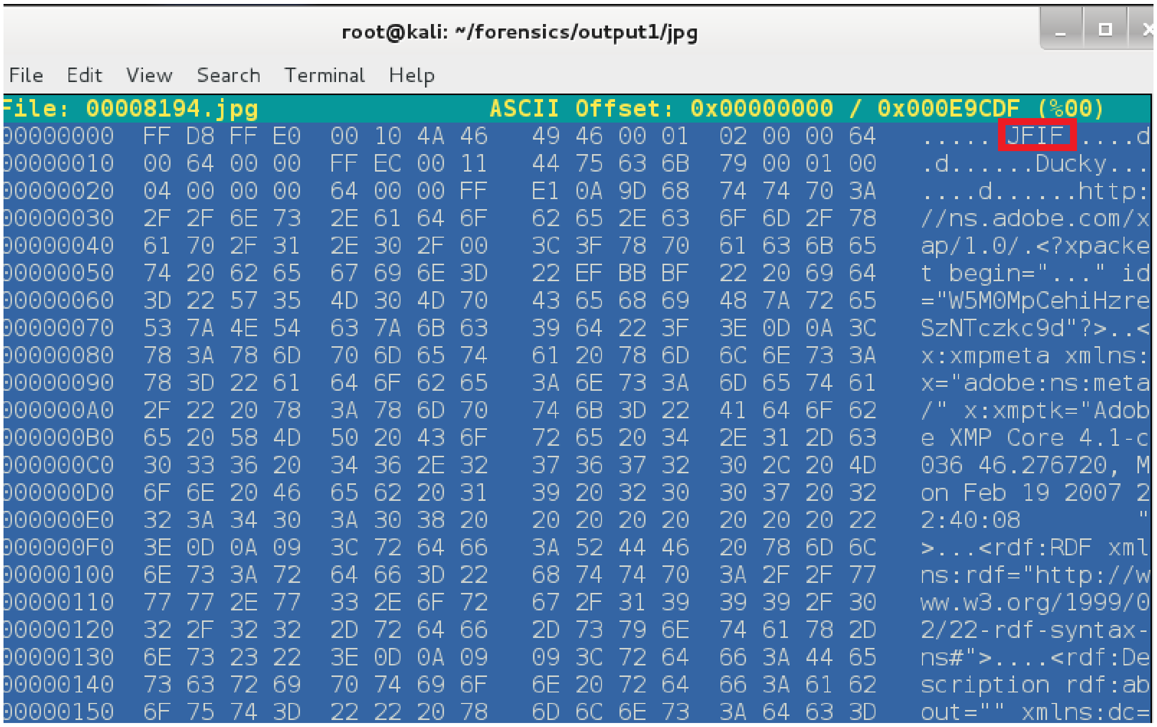
5. Scroll down to any of the recovered jpg files and press the Enter key.



root@kali: ~/forensics/output1/jpg
File Edit View Search Terminal Help

```
File: 00008194.jpg
drwxr-xr-- root      root        4096 2013-12-04 09:21 .
drwxr-xr-x root      root        4096 2013-12-04 09:21 ..
-rw-r--r-- root      root      957664 2013-12-04 09:21 00008194.jpg
-rw-r--r-- root      root     1547988 2013-12-04 09:21 00024426.jpg
-rw-r--r-- root      root     1533423 2013-12-04 09:21 00027450.jpg
-rw-r--r-- root      root     1526083 2013-12-04 09:21 00030446.jpg
-rw-r--r-- root      root     1524390 2013-12-04 09:21 00033428.jpg
-rw-r--r-- root      root     1156534 2013-12-04 09:21 00036406.jpg
-rw-r--r-- root      root     1476388 2013-12-04 09:21 00039322.jpg
-rw-r--r-- root      root       13259 2013-12-04 09:21 00042228.jpg
-rw-r--r-- root      root        7825 2013-12-04 09:21 00042254.jpg
-rw-r--r-- root      root       39760 2013-12-04 09:21 00042321.jpg
-rw-r--r-- root      root        3988 2013-12-04 09:21 00042411.jpg
-rw-r--r-- root      root        6168 2013-12-04 09:21 00042442.jpg
-rw-r--r-- root      root       10533 2013-12-04 09:21 00042459.jpg
-rw-r--r-- root      root       20740 2013-12-04 09:21 00042481.jpg
-rw-r--r-- root      root       19321 2013-12-04 09:21 00042540.jpg
-rw-r--r-- root      root       13066 2013-12-04 09:21 00042582.jpg
-rw-r--r-- root      root        3701 2013-12-04 09:21 00042621.jpg
-rw-r--r-- root      root        7439 2013-12-04 09:21 00042629.jpg
-rw-r--r-- root      root       11912 2013-12-04 09:21 00042643.jpg
-rw-r--r-- root      root       21937 2013-12-04 09:21 00042940.jpg
^C ^X  Exit    ^M  Select File    k  Up    j  Down
```

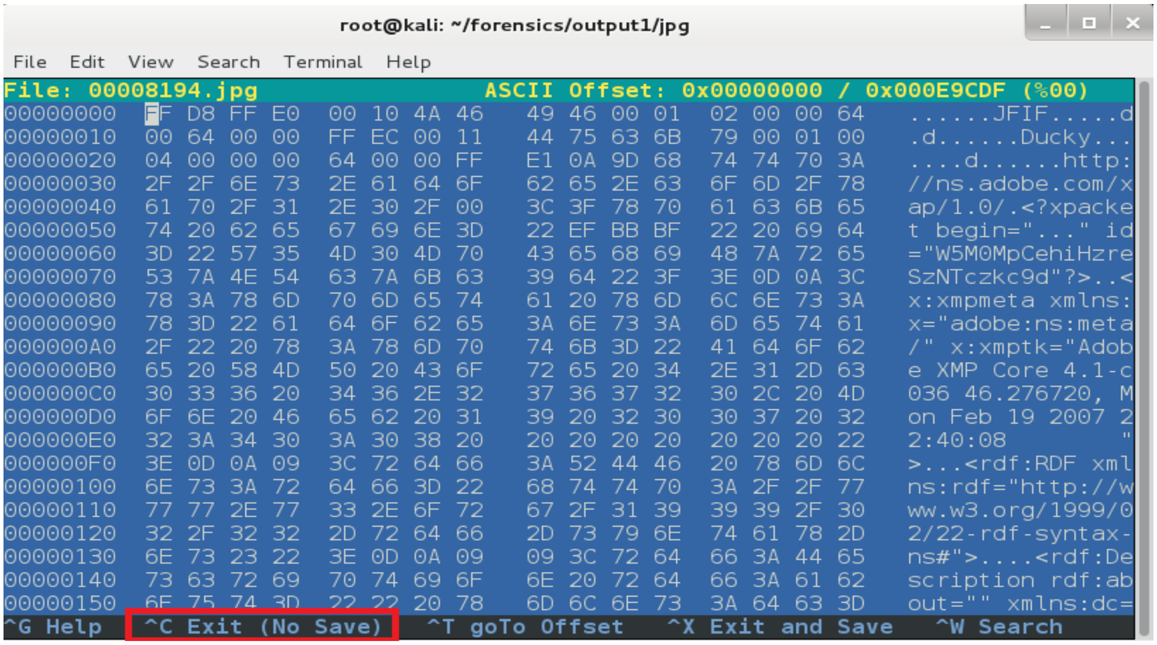
6. Look for the JFIF file signature header in the beginning of the file.



root@kali: ~/forensics/output1/jpg
File Edit View Search Terminal Help

```
File: 00008194.jpg          ASCII Offset: 0x00000000 / 0x000E9CDF (%00)
00000000  FF D8 FF E0  00 10 4A 46  49 46 00 01  02 00 00 64  ......JFIF.....d
00000010  00 64 00 00  FF EC 00 11  44 75 63 6B  79 00 01 00  .d......Ducky...
00000020  04 00 00 00  64 00 00 FF  E1 0A 9D 68  74 74 70 3A  ....d......http:
00000030  2F 2F 6E 73  2E 61 64 6F  62 65 2E 63  6F 6D 2F 78  //ns.adobe.com/x
00000040  61 70 2F 31  2E 30 2F 00  3C 3F 78 70  61 63 6B 65  ap/1.0/.<?xpacke
00000050  74 20 62 65  67 69 6E 3D  22 EF BB BF  22 20 69 64  t begin="..." id
00000060  3D 22 57 35  4D 30 4D 70  43 65 68 69  48 7A 72 65  ="W5M0MpCehiHzre
00000070  53 7A 4E 54  63 7A 6B 63  39 64 22 3F  3E 0D 0A 3C  SzNTczkc9d"?>..<
00000080  78 3A 78 6D  70 6D 65 74  61 20 78 6D  6C 6E 73 3A  x:xmpmeta xmlns:
00000090  78 3D 22 61  64 6F 62 65  3A 6E 73 3A  6D 65 74 61  x="adobe:ns:meta
000000A0  2F 22 20 78  3A 78 6D 70  74 6B 3D 22  41 64 6F 62  /" x:xmptk="Adob
000000B0  65 20 58 4D  50 20 43 6F  72 65 20 34  2E 31 2D 63  e XMP Core 4.1-c
000000C0  30 33 36 20  34 36 2E 32  37 36 37 32  30 2C 20 4D  036 46.276720, M
000000D0  6F 6E 20 46  65 62 20 31  39 20 32 30  30 37 20 32  on Feb 19 2007 2
000000E0  32 3A 34 30  3A 30 38 20  20 20 20 20  20 20 20 22  2:40:08        "
000000F0  3E 0D 0A 09  3C 72 64 66  3A 52 44 46  20 78 6D 6C  >...<rdf:RDF xml
00000100  6E 73 3A 72  64 66 3D 22  68 74 74 70  3A 2F 2F 77  ns:rdf="http://w
00000110  77 77 2E 77  33 2E 6F 72  67 2F 31 39  39 39 2F 30  ww.w3.org/1999/0
00000120  32 2F 32 32  2D 72 64 66  2D 73 79 6E  74 61 78 2D  2/22-rdf-syntax-
00000130  6E 73 23 22  3E 0D 0A 09  09 3C 72 64  66 3A 44 65  ns#">....<rdf:De
00000140  73 63 72 69  70 74 69 6F  6E 20 72 64  66 3A 61 62  scription rdf:ab
00000150  6F 75 74 3D  22 22 20 78  6D 6C 6E 73  3A 64 63 3D  out="" xmlns:dc=
```

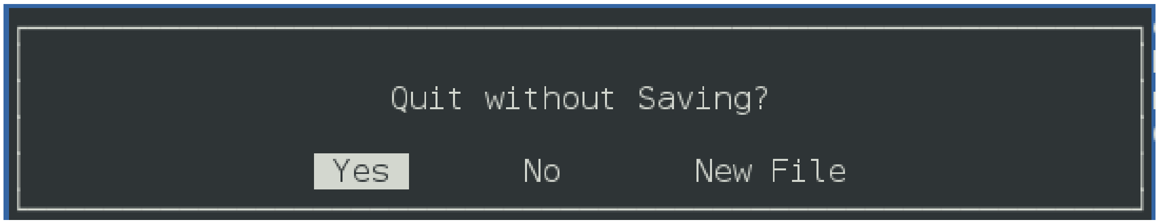
7. To quit the HEX editor, hold down the CTRL button and press C.



root@kali: ~/forensics/output1/jpg
File Edit View Search Terminal Help

```
File: 00008194.jpg                ASCII Offset: 0x00000000 / 0x000E9CDF (%00)
00000000  FF D8 FF E0  00 10 4A 46  49 46 00 01  02 00 00 64  ......JFIF.....d
00000010  00 64 00 00  FF EC 00 11  44 75 63 6B  79 00 01 00  .d......Ducky...
00000020  04 00 00 00  64 00 00 FF  E1 0A 9D 68  74 74 70 3A  ....d.......http:
00000030  2F 2F 6E 73  2E 61 64 6F  62 65 2E 63  6F 6D 2F 78  //ns.adobe.com/x
00000040  61 70 2F 31  2E 30 2F 00  3C 3F 78 70  61 63 6B 65  ap/1.0/.<?xpacke
00000050  74 20 62 65  67 69 6E 3D  22 EF BB BF  22 20 69 64  t begin="..." id
00000060  3D 22 57 35  4D 30 4D 70  43 65 68 69  48 7A 72 65  ="W5M0MpCehiHzre
00000070  53 7A 4E 54  63 7A 6B 63  39 64 22 3F  3E 0D 0A 3C  SzNTczkc9d"?>..<
00000080  78 3A 78 6D  70 6D 65 74  61 20 78 6D  6C 6E 73 3A  x:xmpmeta xmlns:
00000090  78 3D 22 61  64 6F 62 65  3A 6E 73 3A  6D 65 74 61  x="adobe:ns:meta
000000A0  2F 22 20 78  3A 78 6D 70  74 6B 3D 22  41 64 6F 62  /" x:xmptk="Adob
000000B0  65 20 58 4D  50 20 43 6F  72 65 20 34  2E 31 2D 63  e XMP Core 4.1-c
000000C0  30 33 36 20  34 36 2E 32  37 36 37 32  30 2C 20 4D  036 46.276720, M
000000D0  6F 6E 20 46  65 62 20 31  39 20 32 30  30 37 20 32  on Feb 19 2007 2
000000E0  32 3A 34 30  3A 30 38 20  20 20 20 20  20 20 20 22  2:40:08        "
000000F0  3E 0D 0A 09  3C 72 64 66  3A 52 44 46  20 78 6D 6C  >...<rdf:RDF xml
00000100  6E 73 3A 72  64 66 3D 22  68 74 74 70  3A 2F 2F 77  ns:rdf="http://w
00000110  77 77 2E 77  33 2E 6F 72  67 2F 31 39  39 39 2F 30  ww.w3.org/1999/0
00000120  32 2F 32 32  2D 72 64 66  2D 73 79 6E  74 61 78 2D  2/22-rdf-syntax-
00000130  6E 73 23 22  3E 0D 0A 09  09 3C 72 64  66 3A 44 65  ns#">....<rdf:De
00000140  73 63 72 69  70 74 69 6F  6E 20 72 64  66 3A 61 62  scription rdf:ab
00000150  6F 75 74 3D  22 22 20 78  6D 6C 6E 73  3A 64 63 3D  out="" xmlns:dc=
^G Help   ^C Exit (No Save)   ^T goTo Offset   ^X Exit and Save   ^W Search
```

8. When you are prompted to Quit without Saving, select **Yes**.



```
Quit without Saving?

Yes        No        New File
```

Close all open windows and the Kali PC Viewer.

4.2 Conclusion

A HEX editor can be used to examine the "code" of a file. Certain files have signatures in the header of the file. For example, a JPEG file has a file signature of JFIF. Even if the file extension is changed, a forensic examiner can verify the file type by looking at the header. This site provides signatures: http://www.garykessler.net/library/file_sigs.html .

4.3 Discussion Questions

1. What is hexadecimal?
2. What is a hex editor?
3. What is the file signature for a JPEG file?
4. When someone renames a file, does that affect the file signature?

References

1. Foremost:
   http://foremost.sourceforge.net/

2. Hexadecimal:
   http://en.wikipedia.org/wiki/Hexadecimal

3. MD5 Hash:
   http://en.wikipedia.org/wiki/MD5

4. SHA1 Hash:
   http://en.wikipedia.org/wiki/SHA-1

5. HEX Editors:
   http://en.wikipedia.org/wiki/Hex_editor