**Useful Linux Commands**

**Directory Listing**

ls List Files

ls –F Classifies files and directories

ls –a Show all files (hidden) \* hidden files start with ‘.’

ls –l detailed file list

cp <sourcefile> <destination file>

Copy contents of source file to destination file

clear Clear the terminal session

mv <sourcefile> <destination file>

move or rename file

cat filename Display the contents of file

rm filename remove the file

rm –r remove all files in the current directory

rmdir remove directories

find Search for files

find / -name fstab –print

The above command starts from the root directory, searches for file with name fstab and print the results to the screen

pwd print the present working directory

more filename display the contents of the file one page at a time

less filename supports scrolling in both directions

**Important Linux Utilities:**

**Pipes and Redirection:**

* Symbols > and >> are used for redirecting the output of Linux command from standard output to file
* >> appends the output of command to the file instead of “Over-Writing” it.

Example:

ls –al >> filelist.txt

Sends the output of ls command which provides detailed list of all files in the current directory to a file titled ‘Filelist.txt’

* Command Pipe ( | )in Linux are used to take the output of one command and give it as input to another command

Example:

ls –al | grep bash

Output of ls is given as input to grep command. It searches for pattern bash in the detailed output provided by ls command.

**File Hash:**

* Verify the integrity of the data before analysis. SHA /md5 are the popular algorithms used
* SHA sum of a disk image can be obtained using the following command

shasum1 able2.dd

<able2.dd> - raw image for which SHA sum is calculated

\*If md5 hash value is preferred, use md5sum instead of shasum1 in the command

**Calculating hash for all files**

find . –type f –exec sha1sum {} \; > ~/sha.filelist

The command finds (-type f) any regular file in the current directory and execute (-exec) the command sha1sum for all files found ({}).

\; is the escape sequence to terminate the exec command. The identified files and calculated hashes are written to file sha.filelist present in the root directory.

**Interpreting File Type:**

* Linux offers a facility to interpret the file type even if the file has been sabotaged or extension has been changed using file utility

**Command Usage : file filename**

Example:

file matrix3.jpg

Output of the command will be JPEG IMAGE DATA, JFIF standard 1.02

* The command compares the file’s header (first few bytes of file) with the contents of magic file and outputs description of the file

**Viewing Files:**

* To view the contents of data files (or) text files, the following commands can be used.

**cat filename (or) less filename (or) more filename**

* To view unknown files, **strings** command can be used. It is used to parse ASCII text out of any file even unidentified executables.

**strings filename | less**

\*It is better to pip

e the output of strings command through less.

**Searching Unallocated and Slack space for Text:**

* One approach would be searching for the specified pattern through grep command. grep interprets the disk image as a text file. If the text uses character set not recognizable by grep, then it will not be helpful.
* Tools like xxd which produces hexadecimal dump of the input file will be useful in such cases.

Usage of grep:

grep <search pattern/expression> <filename>

Command searches for the specified text pattern in the file mentioned and return the lines that contain the search pattern.

grep –abif searchlist.txt image.dd

It searches for all strings mentioned in the file <searchlist.txt> in the raw image <image.dd> and displays the lines with the byte offset (-b option). It also ignores upper case and lower case (-i option).

**Compression:**

* A tar.gz is referred as tar archive created using tar command in Linux. .gz indicates that gzip is used for compressing the file.
* To list the contents of tar archive , the following command can be used

**tar tzvf logs.tar.gz**

List the files (l) and decompress (z) with verbose output(v) the file (logs.tar.gz)

* To extract the contents of tar use (option x) instead of ( l ).

**tar xzvf logs.tar.gz**

**File Carving:**

* The files in unallocated spaces can be recovered if the blocks corresponding to the file are not reallocated.
* Every file created embeds few bytes of data at the beginning of the file regarding its type.
* JPEG files start with hex value ffd8 with six byte offset to string JFIF and ends with hex value ffd9.
* This information could be used for carving files from raw disk images.

**Tools:**

**dd tool for carving files**

dd if=able2.dd of=sample.jpg skip=3456 bs=1 count=4567

Copies 4567 bytes starting from the offset 3456 setting the block size to 1 (number of bytes to be considered as block)

**grep locate the file header in the hex dump of the image**

**xxd displays the hexadecimal dump of the disk image**

**bc command line calculator**

**Tasks:**

**Carve the JPEG image from the given raw image “image\_carve.raw”**

**Locate the deleted tar file holding the root kit with name lrkn.tgz**