Lab Assignment 5

Aim: To explore hashdeep tool in kali linux for generating, matching and auditing hash of files.

Lab Outcome Attainment: LO2

1. To check the version of Hashdeep - *hashdeep -V*
2. To display help about hashdeep - *hashdeep -h* or *hashdeep -hh*
3. To display the manual page of hashdeep- *man hashdeep*
4. To display the manual page of any specific hash algorithm supported by hashdeep- *man md5deep*

   By default, hashdeep generates MD5 n SHA256 hash values.

5. To hash a file - *hashdeep filename*
6. If you don't want to display the full path of file in output hash record- *hashdeep -b filename*
7. To supress any error messages- *hashdeep -s filename*
8. To apply multiple hash algorithms than default-

   hashdeep -c md5,sha1,sha256,tiger filename

9. To hash multiple files (say all text files) using md5

   *hashdeep -c md5 *.txt*

10. To hash multiple files (say all text files) using md5 and sha1

    *hashdeep -c md5,sha1 *.txt*

11. Hashing block of files- *hashdeep -c md5 -p 100 example.txt*

12. To recursively calculate hash (all files and subdirectories in a specified directory)

```
hashdeep c md5 -r /home/shachi/myfiles
```

13. To redirect the output of md5 hash of files to another file

```
md5deep *.txt>hashset.txt
hashdeep *.txt>hashtext1.txt
```

Check the content of output file-

```
cat hashset.txt
cat hashset1.txt
```

14. To display output in matching mode

```
md5deep -m hashset.txt *
hashdeep -m -k hashset1.txt *
```

15. To supress unwanted system msgs/error

```
md5deep -m hashset.txt *
hashdeep -s -m hashset1.txt *
```

No output is displayed if there is no matching hashed file is found.

16. To display all files which are negatively matching use -x option

```
Md5deep -s -x hashset.txt *
hashdeep -s -x hashset1.txt *
```

Forensic auditing can be done using hashdeep tool which means a check to determine if any files in the system are changed due to malware or any normal system operation like update patching.

17. To audit, first create a hashset file and then audit it against the files to be checked if they are modified.

```
hashdeep -c md5,sha1,sha256 -r /home/shachi/myfiles>hashset1.txt

hashdeep -a -r -k hashset1.txt  /home/shachi/myfiles
```

18. Add new file to the directory and audit. It fails.

```
touch  /home/shachi/myfiles/newfile.txt

hashdeep -a -r -k hashset1.txt  /home/shachi/myfiles
```

19. To get where it failed use the command with -v option

```
hashdeep -v -a -r -k hashset1.txt  /home/shachi/myfiles
```

20. Move one of the files to another directory and audit n see output

```
mv /home/shachi/myfiles/example.txt   /tmp

hashdeep -v -a -r -k hashset1.txt   /home/shachi/myfiles
```

21. Rename one of the files and audit n see the output

```
mv /home/shachi/myfiles/shachi.txt   /home/shachi/myfiles/shachi.bak

hashdeep -v -a -r -k hashset1.txt  /home/shachi/myfiles
```

22. For verbose output of audit

```
hashdeep -vv -a -r -k hashset1.txt  /home/shachi/myfiles

hashdeep -vvv -a -r -k hashset1.txt  /home/shachi/myfiles
```