Title: Learn how to acquire disk image
Lab Scenario: A kali linx machine to execute the imaging commands
Lab Objective : This lab provides insight into :

1) How to acquire image of a physical or logical disk
2) How to use dd, dcfldd and dc3dd commands
3) How to wipe a data safely of a disk

Lab Environment: To carry out this job you need :

- A kali Linux installed on a machine

Lab Tasks:

- Disk Imaging: dd, dcfldd, dc3dd

Listing Disks of your machine : #fdisk -l

- Dd command in kali linux :
  - dd if =/dev/sda3 of = evidence.dd bs=512 conv=noerror, sync
- Command is also used for wiping the data of a disk:
  - dd if=/dev/zero of =/dev/sda bs=1M
- dcfldd in kali linux:
  1) On the fly hashing: hashing input data as it is being transported, helping to ensure data integrity.
  2) Status output: dcfldd can update user on amount of data transferred and time to completion.
  3) Image/ wipe and verify: dcfldd can verify that a target drive is a bit for bit match of the specified input fate or pattern
  4) Multiple Outputs
  5) Split outputs
  6) Log outputs
- Dcfldd Commands
  1) #dcfldd if=/dev/sda3 hash =md5,sha256 hashwindow =1G md5log=/root/md5.txt sha 256 log = /root/sha256.txt hash conv = after conv= no error,sync of = /root/driveimage.dd
  2) The command begins with dcfldd if =/dev/sha3 designates the drive that will be copied in the disk image
  3) #chmod a -w/root/driveimage.dd in order to change the permissions to read only
- Dcfldd in kali linux
  1. #dc3dd if=/dev/sda of =/dev/sda3 hash =md5 log= /mnt/usbstick/[incident name] dc3dd.log

Conclusion:

At this point the disk image has been created, write blocked (changed to read-only) and the original drive has been hashed for comparison to make sure no changes have been made.
dcfldd is a highly functional imaging tool on kali linux

Title: study the use of md5deep hashing tool to compute the hashes of the directories and compare them to check the integrity of directories
Lab Scenario: A kali linux machine to execute the hashing commands
Lab Objective: This lab provides insight into:

1. The md5deep command shows you how to take hashes of directories recursively
2. The different modes of md5deep for hashing.

Lab Environment: A kali linux installed on a machine
Lab tasks: File integrity is no joke and system administrators know the severity of dealing corrupt files

- Md5deep is a set of programs to compute MD5, SHA-1, SHA-256. Tiger or whirlpool message digests on an arbitrary number of files.

Supported platforms: Microsoft windows, linux, mac OS X, etc
Key Features:

1. Recursive Mode
2. Comparison/ Matching mode
3. Tome estimation
4. Piecewise hashing-hash input files in arbitrary sized blocks
5. File-type mode

Basic Commands

- $md5sum: dir/*
- $md5deep -r-s/dir1 >dir1_sums
- $md5deep -r -X dir1sums/dir2

Recurrsive mode:

- $md5deep /home/my_dir/*
- Output:
  md5deep: /home/my_dir/folder2: Is a directory
- $md5deep -r*

Time Estimation mode:

- $md5deep /home/my_dir/*
- Output:
  hd1: 1MB of 47 done, 00:00:46 left
- $md5deep -e/dev/hda1
- Output:
  ca1bdfgdfglsdklfjhsdkf/dev/hda1

File size mode: $md5deep -z*

Matching mode:

a) Positive matching
    a. $md5deep -m-saved hashes.txt
    b. Md5deep -M saved-hashes.txt*
       Output: $md5deep -a3e944694593932*
    c. $md5ddeep -wM saved -hashes.txt*
b) Negative Matching
    a. $md5deep -x saved hashes.txt*
    b. $md5deep -x saved-hashes.txt*
    c. $md5deep -wX saved-hashes.txt*
c) Advanced Matching modes:
    a. Which file matched
        i. $md5deep -m saved-hashes.txt*
        ii. $md5deep -wm saved-hashes.txt*
        iii. $md5deep -wx saved-hashes.txt*
    b. Unused hashes:
        i. $md5deep -nm saved-hashes.txt*

Conclusion:

We successful used md5deep hashing tool to hash the directories and checked the integrity of the directories.

Experiment NO :3

Title : Imaging a disk using accessdata FTK imager on windows
Lab Scenario: A windows machine and AccessData FTK Imager software are provided. We have to take image of a disk so that it can be analyzed
Lab Objective: How to take image of a disk on windows machine
Lab Environment: Windows OS installed and AcessDara FTK imager
Lab tasks:

1. Launch FTK Imager by clicking on the AcessData FTK Imager icon
2. CLICK FILE -> Create disk image
3. Then select logical drive and click next.
4. Select the desired drive in the resulting ‘select drive’ window and click finish
    a. Or select the source drive and click finish
5. Create image using ‘Create Image’ window and click Add to select the image type and choose the image destination.
6. Select the desired image format.Select ‘Raw (dd)’ and click next.
7. Enter the necessary information about the case for the image.
8. Select the folder in which the image file will be placed (y:/Investigative dreive). Also,give the image file in a specific name if desired.
9. Select the ‘Image Fragment size’ field and click finish
10. Verify images after they are created and check the image summary.
11. The image after will be created. This may take some time depending on the file size.

12. Once the image is completed we verify the hash codes and MD5 and Sha1
13. Click on 'Image Summary' and verify it.

Conclusion: At this point the disk image has been created. This is essential for analyzing the contents without touching the original drive.

Experiment No:5

Title: using galletta tool to study cookies created in during browsing
LabScenario: A windows machine contains various unreadable cookie files in non-user readable format and galleta tool converting it into user readable format
Lab Objective: How to use" galleta" forensic tool to decrypt file
Lab Environment: few cookie files
Lab tasks: syntax galleta [options] <filename>
Steps:

1. Open terminal & type galleta
2. Copy the cookie file to a folder
3. Internet explorer > internet tools > browser > history >setting > view files
4. # galleta <filename (ab.cookie)> > <dest.file>

Conclusion: we used the galleta tool to view and study about

Experiment No:4.1

Title: To use a forensic tool foremost in order to recover files.
Lab Scenario: we will use a forensic tool to restore the contents of captured RAM's image and get to know what files were deleted from given RAM.
Lab Objective: This lab provides insight into How to recover deleted files from RAM image.
Lab tasks: steps to use foremost

1. Capture a RAM image
2. To view the details for forensic tool foremost use $foremost -h
3. To restore the deleted files
    a. Foremost -t <type of files format> -I <dir/img name from where the deleted files need to be restored> -o <dir/folder name where the restored files are saved>#foremost -t jpeg -i
    b. $foremost -t pdf -I usingdd.img -o out.

Conclusion : At this point it was clearly possible to recover the files using the foremost tool.

Experiment No :4.2

Title: Using Vinetto a forensic tool to analyse thubs.db files and extract data
Lab Scenario: IN this lab, we will use vinetto forensic tool to analyze the contents of folder using the Thumbs.db file present in the folder and to extract the content in the form of thumbnails and display the details
Lab Objective: How vinetto works& used to extract data
Lab tasks: Elementary/directory/file system mode
Lab environment: suspected machine Thumb.db file, kali
Lab tasks: Download kali linux and burn ISO to pendrive

1. #vinetto -h
2. #find /-name Thumbs.db
3. Vinetto -Ho Thumbs Thumbs.db
4. Vinetto -Ho <dest folder> Thumbs.db file path>

Conclusion:At this point, we were able to analyze and extract and list all the Thubnails using various thumbnails command

Experiment 6.1

1. Title: using a password forensic tool to crack zip file for password protected files
   lab scenario: A zip or rar archive is password protected which carries sensitive data using a password forensic tool we will break the password
   Lab Obective: how to use " fcrackzip","rarcrack" tool
   Lab env: kali machine lab tasks:
   1. $frackzip -B
   2. $frackzip -v -m zip6 -l 4-8 -u secret.zip
      a. -v verbose
      b. -m method
      c. -l length of password
      d. -u unzip and check for password
   3. $frackzip -v -D -u -p /usr/share/dict/words.zip
      a. -v verbose
      b. -D Dictionary
      c. -p path of file

Syntax:

1. $frack - -u -c <your password char> -p < total digits> zipfilepath
2. $frackzip -u -C1 -p aaaaaa '/root/desktop/test/.zip' -c 1-p
3. $frackzip -u -ca -p aaaa '/root/desktop/test.zip'
4. $frackzip -u -D -p <dictionary path> <zip file path>
5. $rarcrack encrypt.extension [--threads] [--type]

Experiment no 7

Title: learn art of steg
lab scenario : to hide a text message behing the image
lab env: win machine with jpg images
lab tasks: steg is the art of covered or hidden writing
steg_medium = hidden_msg + carrier + steg_key
copy /b name.jpg+ text.txt result.jpg
example: copy /b belin.jpg + groovy.txt hidden.jpg

1. Go to cmd and write commands
2. In command write ori image dest to unite what you want to hide in image + then write dest file
3. Click enter
4. Now we have 3 files
   a. Ori file/txt file/img file with embedded txt.

Conclusion: the text of some file has been hidden in the jpg file

Experiment 8.1

Title: memory forensic tool to capture RAMs [volatile memory] img
lab scenario: In this lab we will take an image of RAM memory on a machine
lab objective: how to take an image of RAM in windows and linux
lab env: win&linux machine to take memory img of RAM, kali, pendrive
lab task:

1. RAM capture is process of capturing live memory from a running computer system. RAM analysis consist of performing forensic analysis or the data gathered from the live computer
2. After conducting a memory dump on any live machine to capture RAM, the meory image can be used to determine information about running programs the operating system and overall state of a computer.
3. Until recently RAM analysis and capture was not mandatory step in investigation on even in triage solutions where analysts were attempting to gather forensic data on a file.
4. However with new tool that allow entry into locked systems and with growing importance of temporary files, RAM analysts is quickly becoming a private (and mandatory part of process)
5. Volatile memory access is useful in law enforcement situation where data would be last by powering off a suspect machine.
6. The longer the machine is off the more the data becomes lost.
7. The following can be found using RAM capture processors, network connections, open files, configs, encryption keys, open/active, registry keys, exploit related information, 0-day attacks, root kits and kernel related exloits

RAM capture tools: Dumpit and lime

Dumpit [Windows]

1. Download dumpit
2. Extract the file and click on the dumpit.exe ,pressly to proceed.

Lime [linux]

1. Step1: Download lime forensics
2. Go to downloads directory and unzip the linemaster.zip
3. Cd limemaster/src
4. Make
5. Create lime forensic image

Instructions:

1. #mkdir /urs/tmp/image
2. #insmod lime-26.24.16.screen.ko path /var/tmp/image/ram-lime-format = lime
3. #ls

Conclusion: The RAM image was captured from both windows and linux for further forensic investigation.

Experiment 8.2

Title: using a memory forensic tool analyze RAM's data [volatile memory]
lab scenario: take image of RAM on a machine. Then we will use a memory forensic tool to analyze the contents of captured RAM's image and get to know what processes were running on the machine.
Lab objective: how to take image of RAM on windows machine, what is volatility and how to use it to analyze memory images, different options that volatility tool provides for memory forensics.|
Lab env: a windows machine to take an image of a RAM, bootable pendrive, pendrive to copy the memory into it. Pendrive size should be bigger than RAM size.
RAM analysis tool: volatility: a tool capable of analyzing from a memory dump disk image.
volix: tool that provides GUI for volatility; volafox: a tool capable of analysing RAM memory image (mac)

Steps in RAM analysis

1. Capturing RAM memory image
2. Gather additional information about system using captured RAM image

Capturing memory image on windows: dumpit, FTL images
steps to analyze on a windows machine RAM image using volatility

1. Launch volatility
2. $ cd /usr/share/volatility
3. $ python vol.py -h
4. Examine captured RAM dump to volatility
5. Find basic data machine
    a. $python vol.py image info -f/root/memory.dump mon [output in a rar file]
6. To get RAM login data and the hash valve for the login password through lime list
    a. $python vol.py lime list -profile=win2008plx86 -f /root/memdump.mem
7. To extract the hashes from memory dump
    a. $python vol.py hashdump –profile=win2008sp1X86 -f /root/memd.mem -y 0x8623423 -s 0x98c33k =/root/hash.txt
8. To view process running at time of memory dump
    a. $python vol.py ps list = profile=win2008 splx86 -f /root/memdump.mem
9. To see recent console commands that has been executed
    a. $python vol.py consoles –profile=win2008 splx86 -f /root/memdump.mem
10. To see the services running at time of memory dump
    a. $python vol.py consoles –profile=win2008splx86 -f /root/memdump.mem
11. To see the services running at time of memory dump
    a. $python vol.py sv scan –profile= win2008splx86 -f /root/memdump.mem

Conclusion: The experiment conducted above was verified.