

# INDEX

S. No.	List of Experiments	Date	Remarks
1.	How to Recover Deleted Files using Forensics Tools	13-01-2025	
2.	Live Forensics Case Investigation using Autopsy	20-01-2025	
3.	Write a Program in Linux to Demonstrate Netstat Tool	27-01-2025	
4.	Write a Program in Linux to Demonstrate NMap Tool	03-02-2025	
5.	Write a Program in Linux to List all files that have been accessed, modified, changed in last 7 Days	17-02-2025	
6.	Write a Program in Linux to Demonstrate ARP (Address Resolution Protocol)	24-02-2025	
7.	How to Collect Email Evidence in Victim PC	10-03-2025	
8.	Write a Program in Linux to monitor authentication-related events in real-time	17-03-2025	
9.	Comparison of two Files for forensics investigation by Compare IT software	24-03-2025	
10.	Write a Program in Linux to Break Encryption on Password locked zip file	07-04-2025	

# EXPERIMENT - 1

**Aim:** How to Recover Deleted Files using Forensics Tools.

## Step-01: Create a File

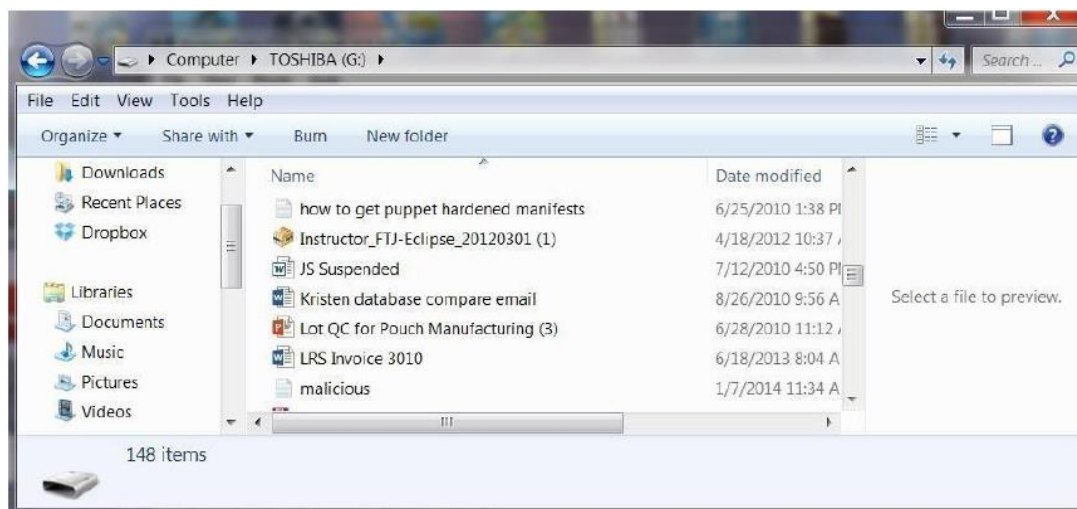
To demonstrate how to recover deleted files, let's create a malicious document. We will call this document "Malicious" and create it with Notepad in Windows.



This sounds like a sound, albeit ambitious plan.

## Step 2: Delete the File

Next, now that we have completed our plans to take over the world, let's delete the file because we no longer need it and we don't want to leave behind any evidence of our malicious plans.



Right-click on the malicious file and select delete. If you put the file in the Recycle Bin, you have made it even easier for the forensic investigator to recover. The Recycle Bin is actually simply a folder where the files are moved until you empty the Recycle Bin. Nothing is deleted until you empty the Recycle Bin.

### Step 3: Create an Image

The first step a forensic investigator will do when examining your computer is to make a bit-by-bit copy of your hard drive or in this case your flash drive.

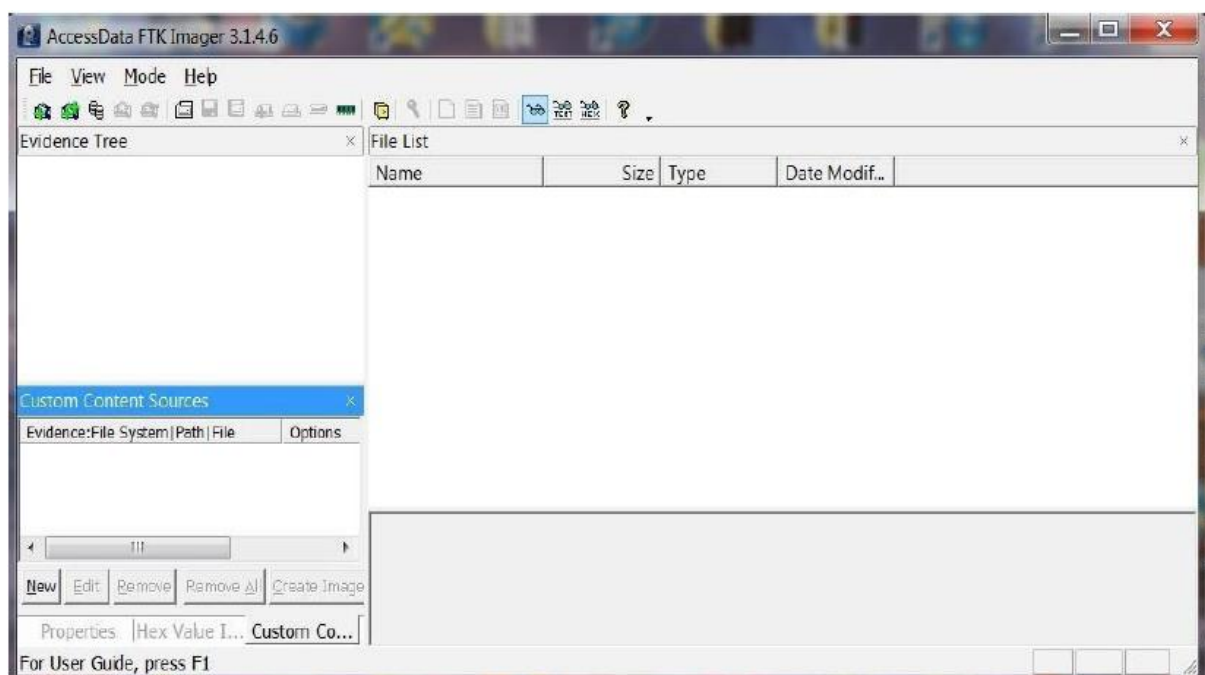
There are numerous tools that can do this and in Linux, we have the dd command that does an excellent job of making bit-by-bit copies (it's on all Linux distributions including BackTrack). File backups and copies are not forensically sound as they will not copy deleted files and folders and in many cases will actually change the data.

Most forensic investigators use commercial tools. The two most popular being Encase by Guidance Software and Forensic Tool Kit by Access Data.

FTK, as it is commonly known in the industry, has a free imager that creates a bit-by-bit copy of the drive. This imager is probably the most widely used in the industry and its price is right, so let's use it.

You can download it **here**.

Now that have downloaded the FTK imager, we need to create a bit-by-bit image of the flash drive.



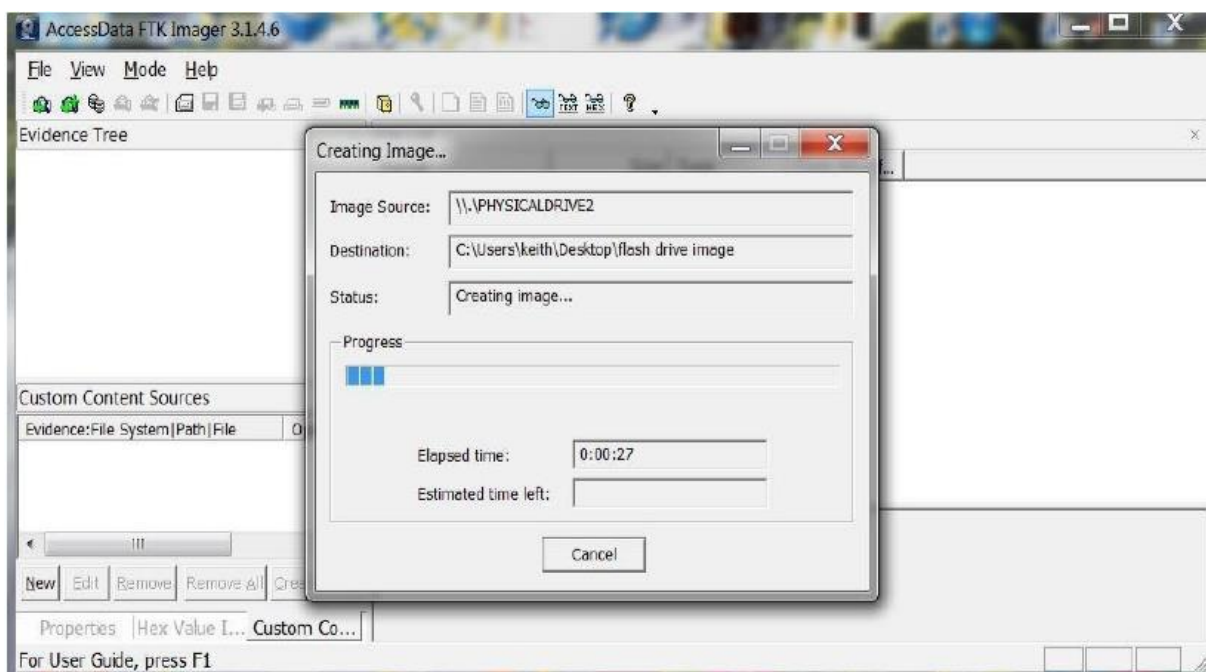
Go to the menu at the top of the application and select:

- File -> Create Image

It will open a wizard that will walk you through the process of opening a case and ask you for a case number, evidence number, examiner name, etc.

Obviously, this software was designed for law enforcement and all evidence needs to be categorized and labelled.

Finally, it will ask for a location of the physical drive you want to image, a destination directory and a name for the image file. When you are done with all these administrative tasks, FTK Imager will begin the process of creating a forensically sound bit-by-bit image of your drive.



Now that we've created an image of the flash drive, we are ready to recover the deleted files.

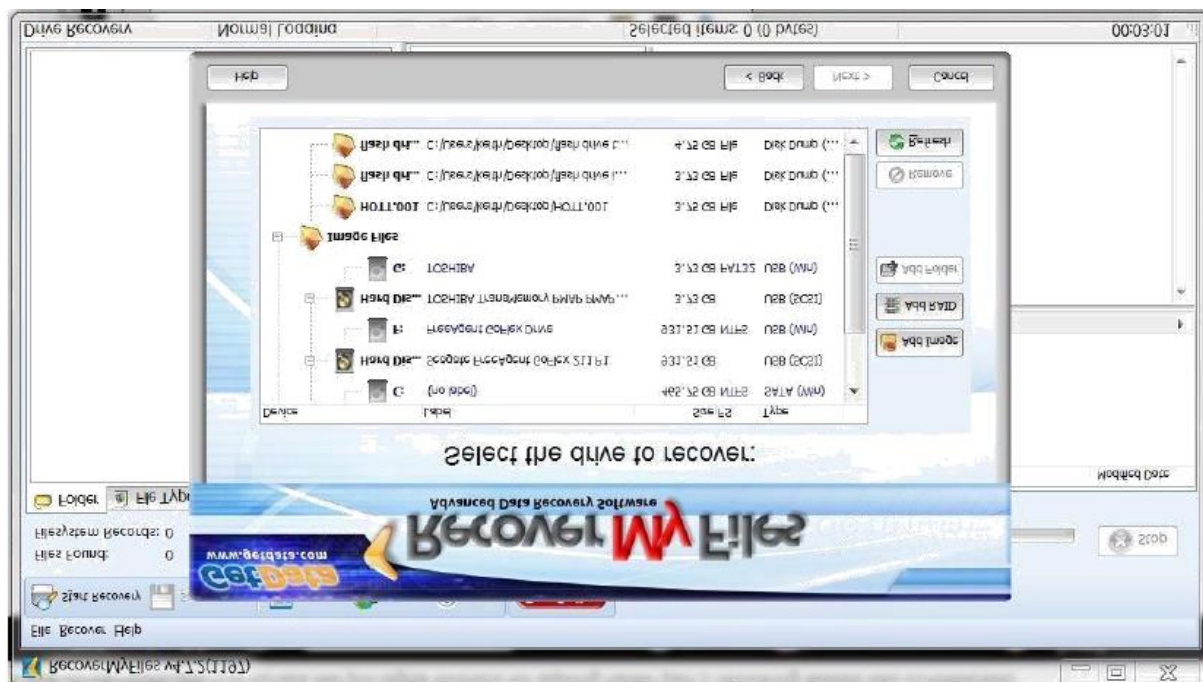
#### Step 4: Recover Deleted Files

There are many tools on the market to recover deleted files and all of them are adequate to do the job. Deleted file recovery is probably the simplest of forensic tasks. Here, I will be using a trial version of RecoverMyFiles.

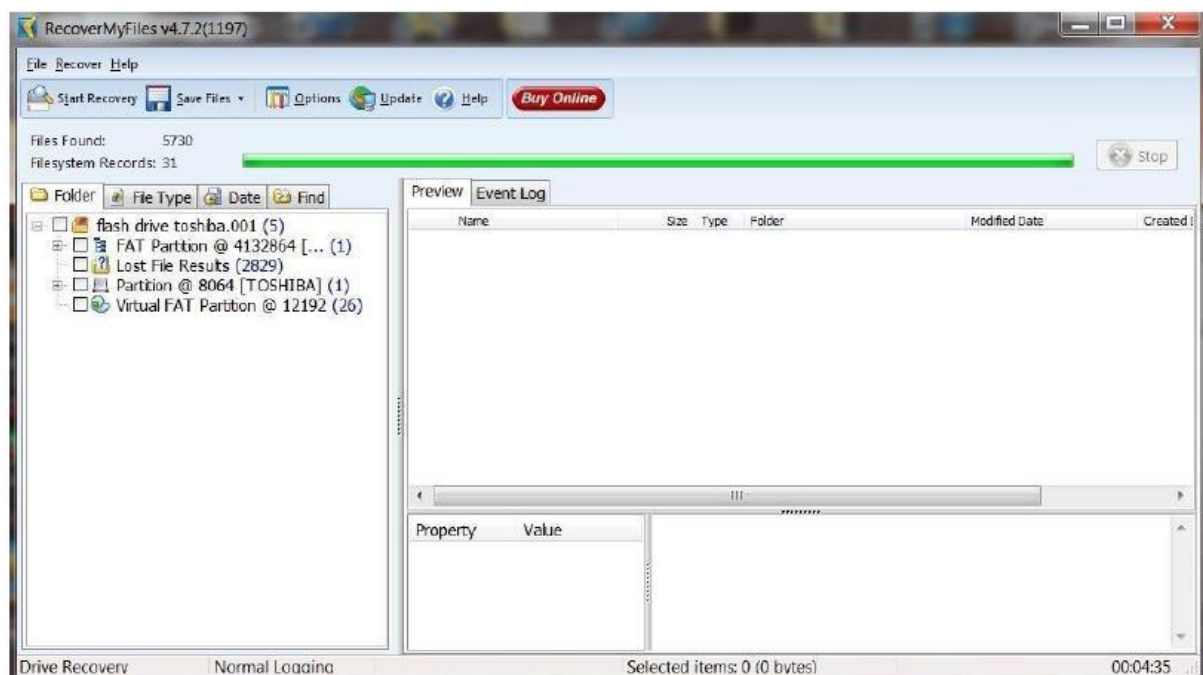
You can download a trial version **here**.

Once you have installed RecoverMyFiles, select the Start Recovery icon in the upper left corner. It will ask you to select either Recover Files or Recover Drive. Select Recover a Drive. It will then search and display all your drives like that in the screenshot below. Since we are using a forensic image, select Add Image

button to the right. You will need to provide a path to your image file created with FTK.



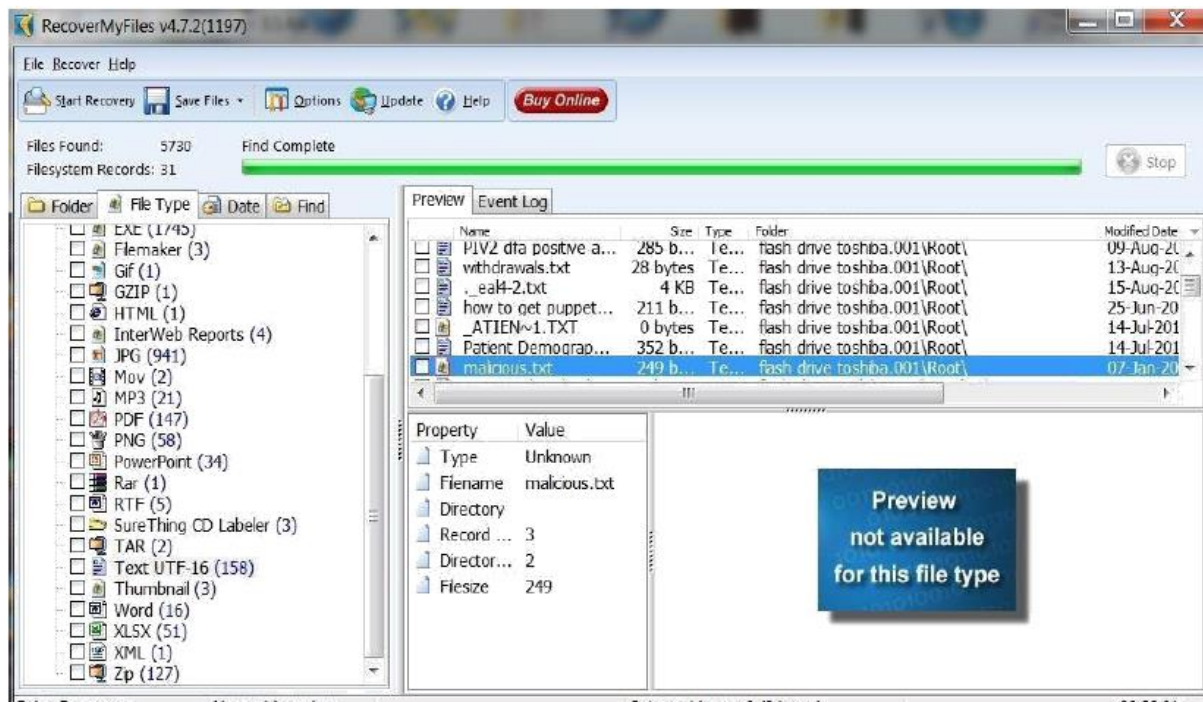
Once you select an image file, start the automatic file recovery. When the recovery is completed, you will see a screen similar to the one below.



I then selected the File Type tab above the Explorer window to categorize the files by type.

As you can see, there are numerous file types recovered from this flash drive. Since our malicious document was a .txt, I have selected the TXT UTF-16 file

type. It then puts all 158 .txt files on display in the upper right window. As you can see, it has recovered our malicious.txt file and everything on it. Busted!



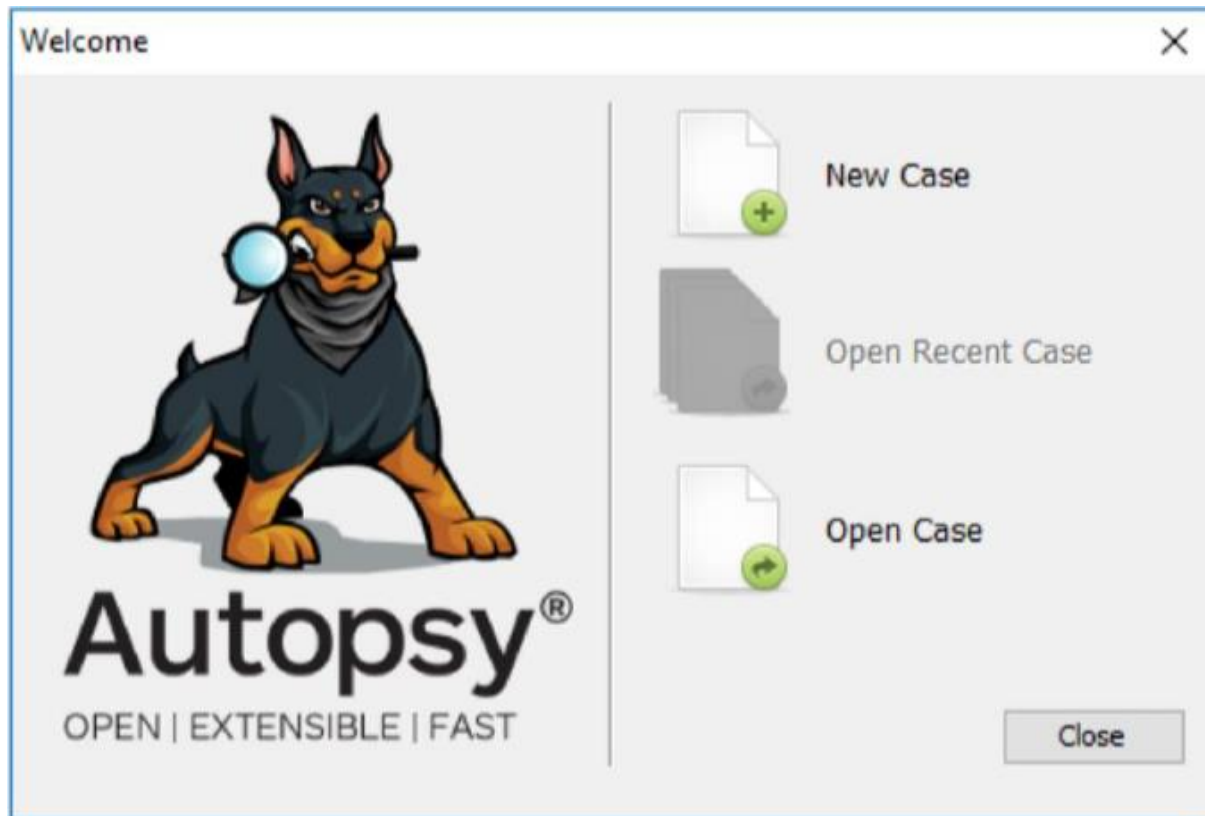
I'm hoping that this tutorial clearly showed you how simple it is for a forensic investigator to recover the files you have deleted. This should be a lesson that you need to be exceedingly cautious and when possible, overwrite any deleted files to remove evidence. In some cases, even that may not be enough to keep your files from a skilled forensic investigator.



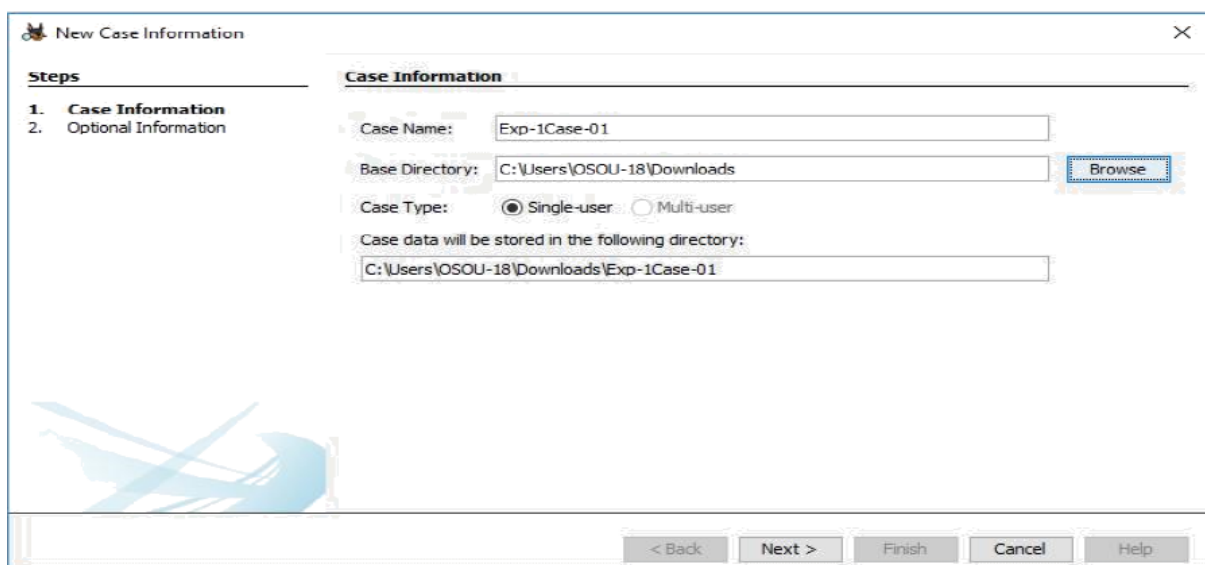
## EXPERIMENT - 2

**Aim:** Live Forensics Case Investigation using Autopsy.

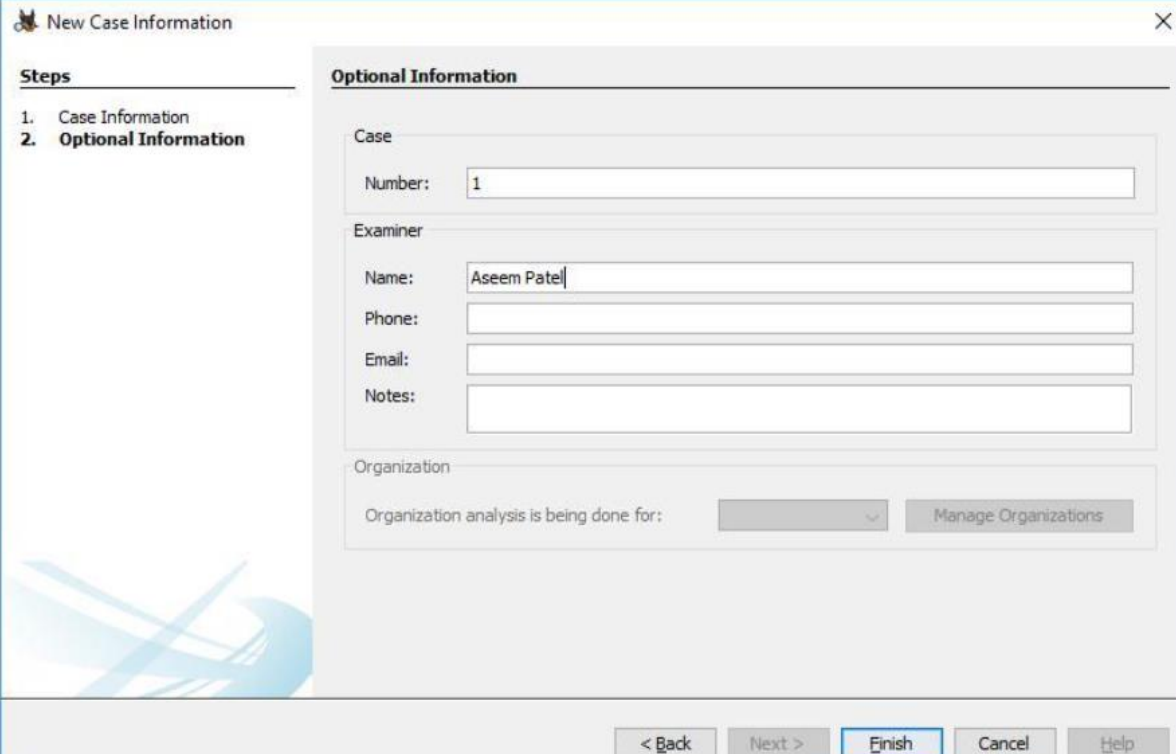
First Download autopsy from here and install in your pc. Click '**New Case**' option



A new page will open. Enter the details in '**Case Name**' and '**Base Directory**' and choose the location to save the report e.g. :Autoreport. Then click on next to proceed to the next step.

The image shows the 'New Case Information' window. On the left, there is a 'Steps' section with two items: '1. Case Information' and '2. Optional Information'. The 'Case Information' section is currently active. It contains several input fields: 'Case Name' with the value 'Exp-1Case-01', 'Base Directory' with the value 'C:\Users\OSOU-18\Downloads', and 'Case Type' with 'Single-user' selected (indicated by a filled radio button) and 'Multi-user' unselected (indicated by an empty radio button). Below these, there is a label 'Case data will be stored in the following directory:' followed by a text box containing 'C:\Users\OSOU-18\Downloads\Exp-1Case-01'. A 'Browse' button is located next to the 'Base Directory' field. At the bottom of the window, there are five buttons: '< Back', 'Next >', 'Finish', 'Cancel', and 'Help'.

Here in the next step, you have to enter the case number and Examiner details and click on finish to proceed to the next step.



**New Case Information**

**Steps**

1. Case Information
2. **Optional Information**

**Optional Information**

Case

Number:

Examiner

Name:

Phone:

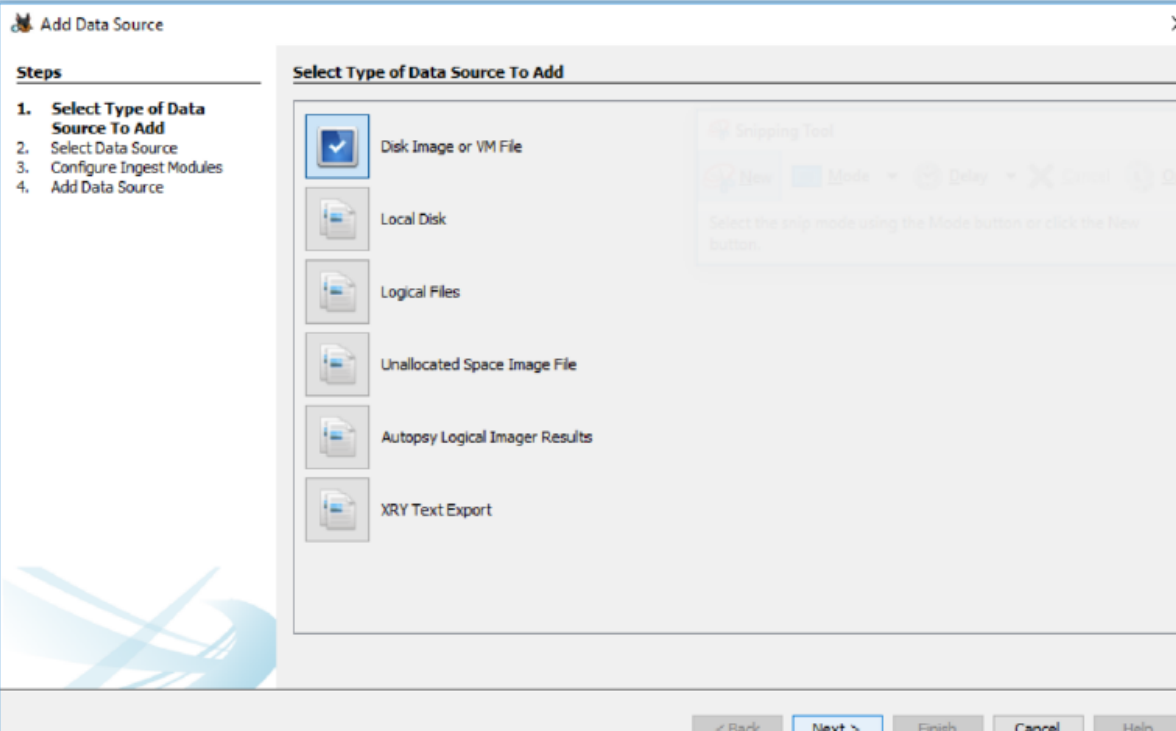
Email:

Notes:

Organization

Organization analysis is being done for:

< Back   Next >   **Finish**   Cancel   Help



**Add Data Source**

**Steps**

1. **Select Type of Data Source To Add**
2. Select Data Source
3. Configure Ingest Modules
4. Add Data Source

**Select Type of Data Source To Add**

☒ Disk Image or VM File

☐ Local Disk

☐ Logical Files

☐ Unallocated Space Image File

☐ Autopsy Logical Imager Results

☐ XRY Text Export

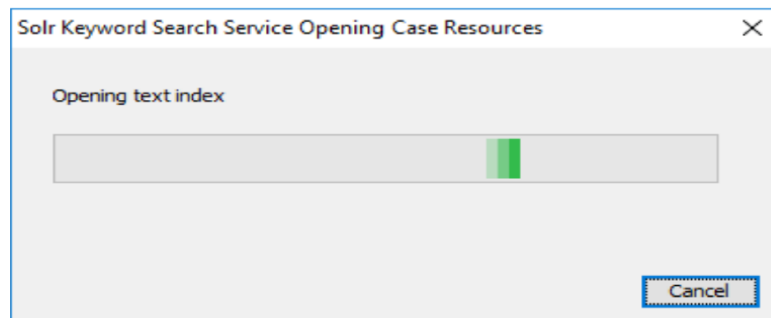
Snipping Tool

New   Mode   Delay   Cancel   Help

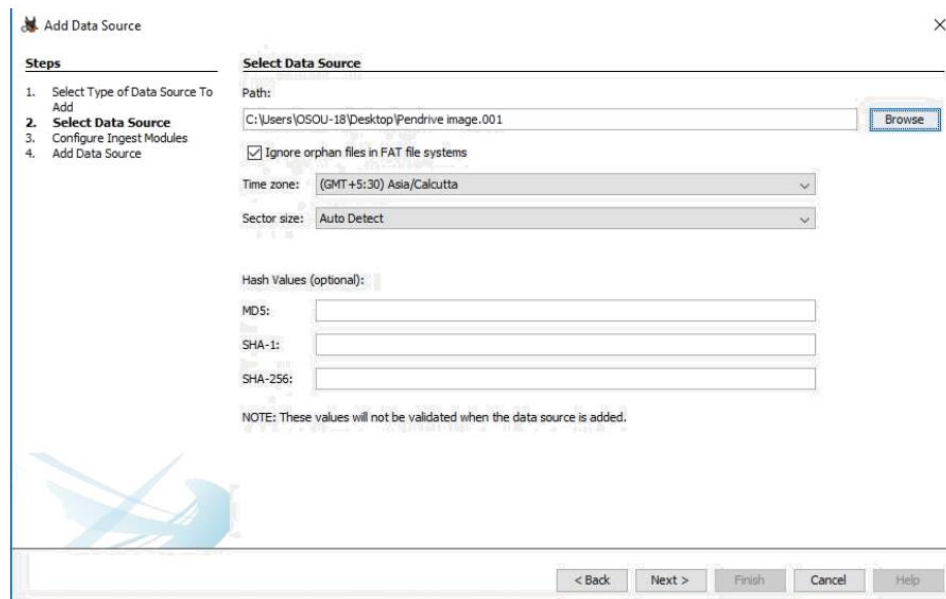
Select the snip mode using the Mode button or click the New button.

< Back   **Next >**   Finish   Cancel   Help

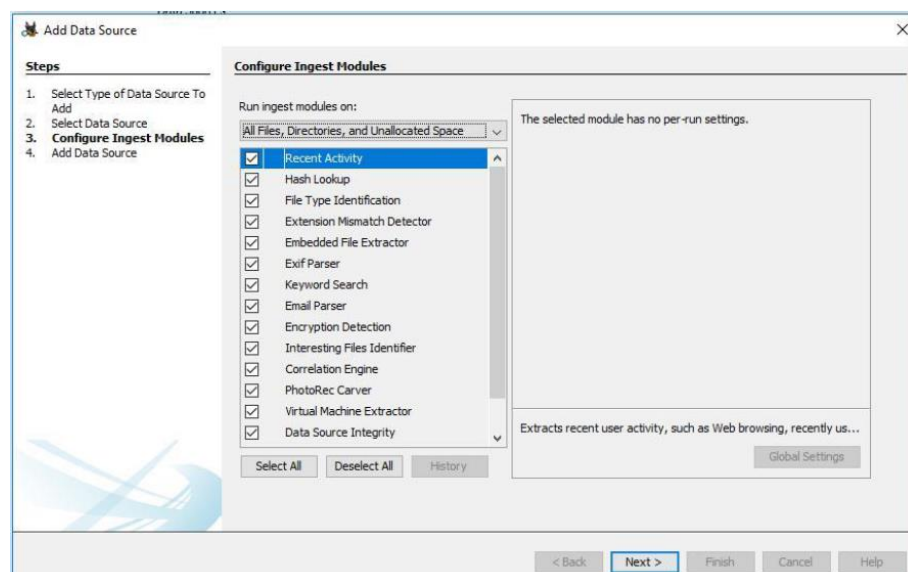




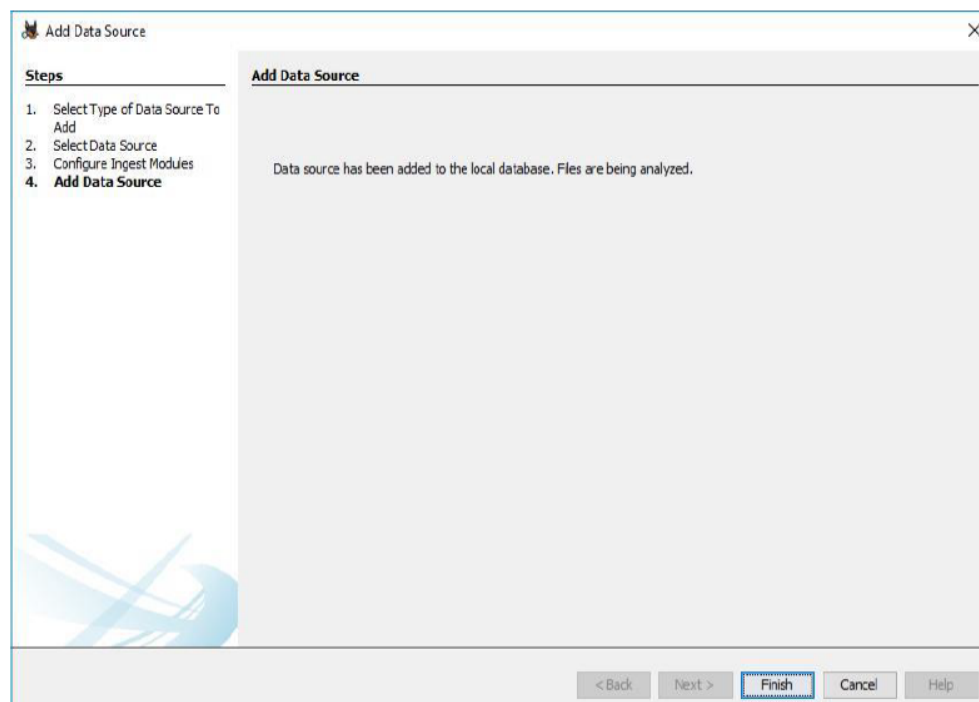
A new window will open. It will ask for the add data source in Step 1. Select source type to add & browse the file Path and click on NEXT option to proceed further.



Configure ingest Modules I have chosen all the modules as I am looking for complete information on evidence device or disk or system etc. and click next to proceed further.



In Add Data Source just click on Finish to generate the report of the device and you can perform complete investigate on the victim device or system or any other disk. It will process the data Source and add it to the local database.



After Process completion, it will show the Forensic Investigation Report. Now click on Devices Attached option, it will show the list of the attached device with the system.

Now click on EXIF Metadata (Exchangeable image file format for images, sound used by Digital Camera, Smartphone and scanner), click on Installed Programs to see the entire installed programs in the system, Click Operating System Information. It will show the entire operating system list, Now Select Operating System User Account Option. It will Display the name of all the user Accounts, Now click on Recent Documents Option, it will display the latest created or opened documents, Click Web Bookmarks Option to see all the bookmarks by system users in different browsers, To see web cookies, select web cookies option, To See Web Downloads, Click on Web Downloads option, To check internet History, click on Web History Option, To see the history of internet search, click on Web Search Option, To see the list of all email ids in the system, click on email address.

And try to explore other option in autopsy.

# EXPERIMENT - 3

**Aim:** Write a Program in Linux to Demonstrate Netstat Tool.

**Theory:** Netstat (Network Statistics) is a command-line tool that provides detailed information about: Active TCP/UDP network connections, Listening ports, Routing tables, Interface statistics, Network protocol usage

**Step 1:** Install Netstat in Linux

```
sudo apt install net-tools
```

**Step 2:**

```
netstat -ant
```

-a: Show all connections and listening ports

-n: Show numeric IP addresses (don't resolve DNS)

-t: Show only TCP connections

This command helps list all TCP connections on the system, including established, listening, or waiting states.

```
ubuntu-u1@ubuntu-u1-VirtualBox:~$ netstat -ant
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:631            0.0.0.0:*               LISTEN
tcp        0      0 192.168.122.1:53        0.0.0.0:*               LISTEN
```

**Code:**

```
GNU nano 7.2 netstat_exp.sh
#!/bin/bash
LOGFILE="network_activity.log"

echo "Scanning for active network connections" | tee $LOGFILE
netstat -tulnp | grep ESTABLISHED | tee -a $LOGFILE

echo "Checking for suspicious IPs" | tee -a $LOGFILE
SUSPICIOUS_IPS=("192.168.122.1:53" "10.0.2.15:44500" "10.0.2.15:59250")
for ip in "${SUSPICIOUS_IPS[@]}; do
    if netstat -antp | grep -q $ip; then
        echo "Alert : Suspicious IP Detected: $ip" | tee -a $LOGFILE
    fi
done
echo "Scan complete. Results saved in $LOGFILE"
```

**Output:**

```
ubuntu-u1@ubuntu-u1-VirtualBox:~$ sudo bash ./netstat_exp.sh
Scanning for active network connections
Checking for suspicious IPs
Alert : Suspicious IP Detected: 192.168.122.1:53
Scan complete. Results saved in network_activity.log
```

# EXPERIMENT - 4

**Aim:** Write a Program in Linux to Demonstrate NMap Tool.

**Theory:** Nmap is a powerful command-line network scanning tool used for discovering hosts, open ports, and services running on a system or network.

It provides detailed information about: Open TCP/UDP ports, Running services and their versions, Host availability, OS detection (advanced use)

**Step 1:** Install NMap in Linux

```
sudo apt install nmap
```

**Step 2:** Operative portion of Nmap Command used:

```
nmap -p 1-1000 --open -T3 127.0.0.1
```

-p 1-1000: Scan the first 1000 TCP ports (common ports)

--open: Show only ports that are open

-T3: Use moderate scan timing (safe for local scanning)

27.0.0.1: Localhost (can replace with other IP)

**Code:**

```
GNU nano 7.2                                nmap_exp.sh
#!/bin/bash

read -p "Enter target IP (default: 127.0.0.1): " target
target=${target:-127.0.0.1}

LOGFILE="nmap_scan_log.txt"

echo "Scanning target: $target" | tee $LOGFILE

#Scan using non-aggressive timing T3
echo "Scan for open ports"
nmap -p 1-1000 --open -T3 $target | tee -a $LOGFILE

echo "Detecting services on open ports"
nmap -sV -p 1-1000 -T3 $target | tee -a $LOGFILE

echo "Scan complete. Results saved in $LOGFILE" | tee -a $LOGFILE
```

## Output:

```
ubuntu-ui@ubuntu-ui-VirtualBox:~$ sudo ./nmap_exp.sh
Enter target IP (default: 127.0.0.1): 127.0.0.1
Scanning target: 127.0.0.1
Scan for open ports
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-11 04:24 EDT
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000050s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE
631/tcp   open  ipp

Nmap done: 1 IP address (1 host up) scanned in 0.18 seconds
Detecting services on open ports
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-11 04:24 EDT
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000040s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
631/tcp   open  ipp      CUPS 2.4

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.39 seconds
Scan complete. Results saved in nmap_scan_log.txt
```

## EXPERIMENT - 5

**Aim:** Write a Program in Linux to List all files that have been accessed, modified, changed in last 7 Days.

**Theory:** In Linux, each file maintains metadata including atime (access time), which records the last time a file was read or opened. By using the find command with the -atime flag, we can locate files that have been accessed within a given number of days.

atime - access - Detect if a hidden/sensitive file was viewed

mtime - modified - See if contents were edited (e.g., tampered logs)

ctime - change - Tracks metadata changes (e.g., permission tampering)

**Code:**

```
GNU nano 7.2                                last7days_exp.sh
#!/bin/bash

LOGFILE="file_activity_log.txt"
echo "File Forensics Report of Last 7 Days" > "$LOGFILE"
echo "Generated on: $(date)" >> "$LOGFILE"
echo "-----" >> "$LOGFILE"

#Files Accessed in Last 7 Days
echo -e "\nFiles Accessed in last 7 days (atime):" | tee -a "$LOGFILE"
find / -type f -atime -7 | tee -a "$LOGFILE"

#Files Modified in Last 7 Days
echo -e "\nFiles Modified in last 7 days (mtime):" | tee -a "$LOGFILE"
find / -type f -mtime -7 | tee -a "$LOGFILE"

#Files Metadata Changed in Last 7 Days
echo -e "\n*Files changed in last 7 days (ctime):" | tee -a "$LOGFILE"
find / -type f -ctime -7 | tee -a "$LOGFILE"

echo -e "\nScan complete. Results saved in $LOGFILE"
```

**Output:**

```
Open  ~
accessed_files.log
~/
/home/ubuntu-u1/netstat_exp.sh
/home/ubuntu-u1/Documents/mv home ubuntu u1 Downloads.txt
/home/ubuntu-u1/snap/firmware-updater/127/.config/user-dirs.locale.md5sum
/home/ubuntu-u1/snap/firmware-updater/127/.config/fontconfig/fonts.conf
/home/ubuntu-u1/snap/firmware-updater/127/.config/user-dirs.dirs
/home/ubuntu-u1/snap/firmware-updater/127/.config/user-dirs.dirs.md5sum
/home/ubuntu-u1/snap/firmware-updater/127/.config/user-dirs.locale
/home/ubuntu-u1/snap/firmware-updater/127/.last_revision
/home/ubuntu-u1/snap/snapd-desktop-integration/178/.config/user-dirs.locale.md5sum
```



# EXPERIMENT - 6

**Aim:** Write a Program in Linux to Demonstrate ARP (Address Resolution Protocol).

**Theory:** ARP (Address Resolution Protocol) is a network protocol used to find the MAC (Media Access Control) address associated with an IP address within a local network (LAN).

**Key Points:** Converts IPv4 addresses → MAC addresses. Used when a host wants to communicate with another device in the same network. Entries are stored in the ARP cache.

You can view or manipulate the ARP table using commands like:

arp -a (on Windows/macOS)

ip neighbour or arp (on Linux)

**Code:**

```
GNU nano 7.2                                arp_exp.sh *
#!/bin/bash

echo "ARP Table Entries:" > arp_output.txt
date >> arp_output.txt

echo -e "\n Listing ARP entries using 'arp':" | tee -a arp_output.txt
arp | tee -a arp_output.txt

echo -e "\n ARP table collected. Check 'arp_output.txt'."
```

**Output:**

```
ubuntu-u1@ubuntu-u1-VirtualBox:~$ sudo ./arp_exp.sh

 Listing ARP entries using 'arp':
Address          HWtype  HWaddress          Flags Mask          Iface
_gateway         ether    52:54:00:12:35:02    C                    enp0s3

 ARP table collected. Check 'arp_output.txt'.
```

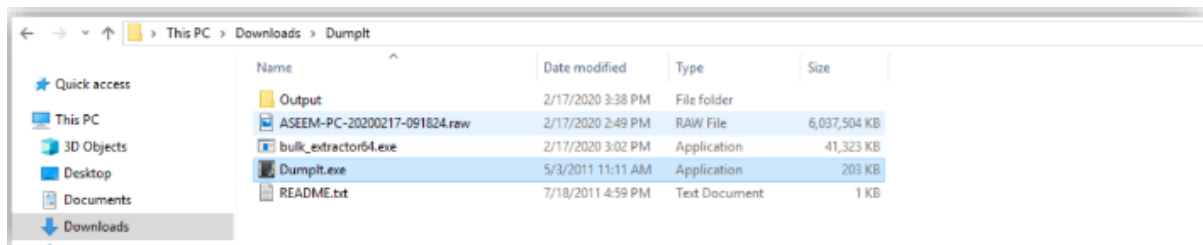
# EXPERIMENT - 7

**Aim:** How to Collect Email Evidence in Victim PC.

To collect email evidence from Victim PC the first step is to capture the victim's RAM. This can be possible using dumpit tool.

This utility is used to generate a physical memory dump of Windows machines. It works with both x86 (32-bits) and x64 (64-bits) machines. The raw memory dump is generated in the current directory, only a confirmation question is prompted before starting. Perfect to deploy the executable on USB keys, for quick incident responses needs.

Run Dumpit.exe file the raw memory dump will be generated and save to the same directory



```
C:\Users\OSOU-18\Downloads\DumpIt\DumpIt.exe

DumpIt - v1.3.2.20110401 - One click memory memory dumper
Copyright (c) 2007 - 2011, Matthieu Suiche <http://www.msuiche.net>
Copyright (c) 2010 - 2011, MoonSols <http://www.moonsols.com>

Address space size:      6182404096 bytes ( 5896 Mb)
Free space size:         59016699904 bytes ( 56282 Mb)

* Destination = \\?\C:\Users\OSOU-18\Downloads\DumpIt\ASEEM-PC-20200217-105827.raw

--> Are you sure you want to continue? [y/n]
```

```
C:\Users\OSOU-18\Downloads\DumpIt\DumpIt.exe

DumpIt - v1.3.2.20110401 - One click memory memory dumper
Copyright (c) 2007 - 2011, Matthieu Suiche <http://www.msuiche.net>
Copyright (c) 2010 - 2011, MoonSols <http://www.moonsols.com>

Address space size:      6182404096 bytes ( 5896 Mb)
Free space size:         59016699904 bytes ( 56282 Mb)

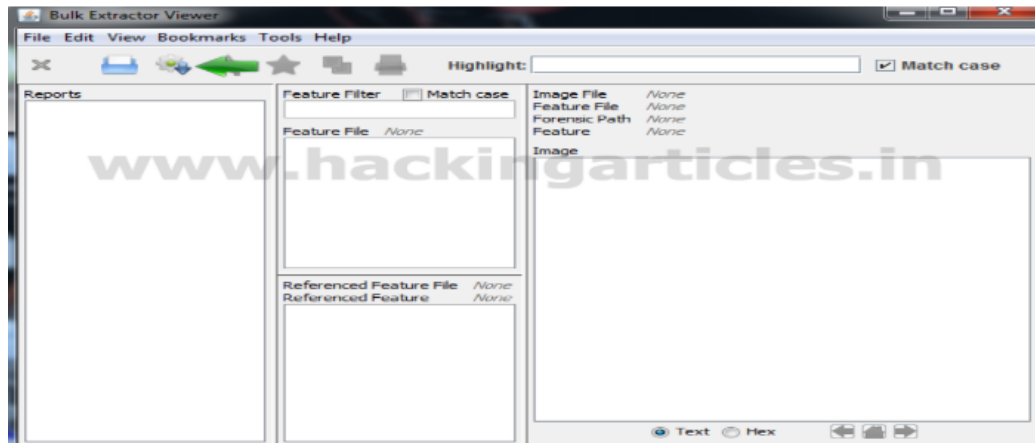
* Destination = \\?\C:\Users\OSOU-18\Downloads\DumpIt\ASEEM-PC-20200217-105827.raw

--> Are you sure you want to continue? [y/n] y
+ Processing...
```

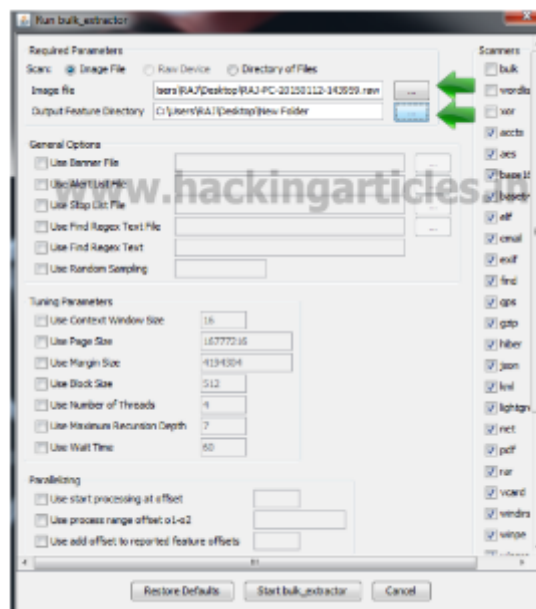
The output .RAW file will be as follows

Name	Date modified	Type	Size
Output	2/17/2020 3:38 PM	File folder	
ASEEM-PC-20200217-105827.raw	2/17/2020 4:29 PM	RAW File	6,037,504 KB
bulk_extractor64.exe	2/17/2020 3:02 PM	Application	41,323 KB
Dumplt.exe	5/3/2011 11:11 AM	Application	203 KB
README.txt	7/18/2011 4:59 PM	Text Document	1 KB

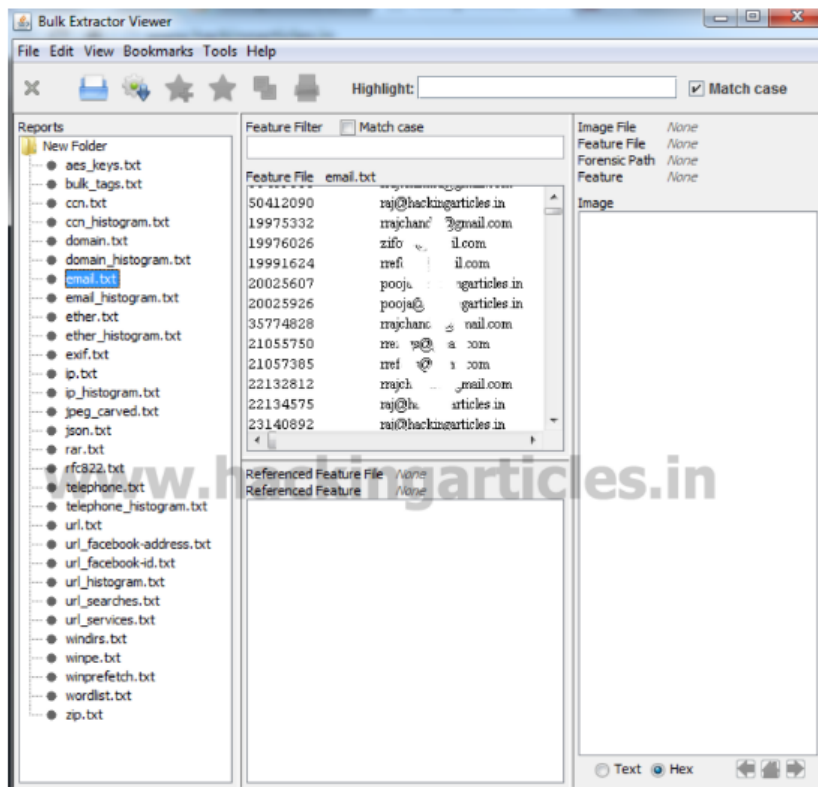
Then Download bulk extractor viewer from GitHub and install it in your PC. Now open bulk extractor viewer and click on to generate report.



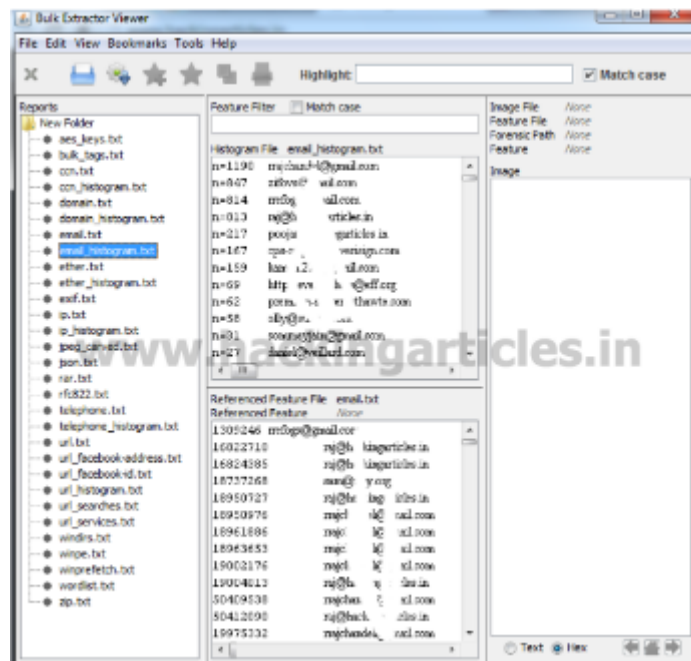
Now select the dump it image file and select an output folder for the report and click on start bulk extractor as seen below



Now in order to investigate the victim saved information of Email ID Click on email.txt as seen below



And also click on email\_histogram.txt



## EXPERIMENT - 8

**Aim:** Write a Program in Linux to monitor authentication-related events in real-time

**Theory:** The `/var/log/auth.log` file stores system authentication attempts. Monitoring it using `tail -f` enables investigators to observe login attempts, failures, and sudo usage as they happen. This is essential in detecting intrusions or suspicious activity. So, in simple terms:

- `auth.log` is the actual log file (generated by the system)
- `tail -f` just follows that file live

You're watching new login or authentication-related events as they're written (sudo usage, su commands, ssh logins (both success and failure), polkit authentication attempts, passwd usage (password changes), PAM (Pluggable Authentication Module) events etc.

**Code:** in terminal type `sudo tail -f /var/log/auth.log`

```
ubuntu-u1@ubuntu-u1-VirtualBox:~$ sudo tail -f /var/log/auth.log
[sudo] password for ubuntu-u1:
2025-04-11T09:25:01.877939-04:00 ubuntu-u1-VirtualBox CRON[10978]: pam_unix(cron:session): session closed for user root
2025-04-11T09:25:23.861199-04:00 ubuntu-u1-VirtualBox gdm-password[1]: gkr-pam: un
```

In another new terminal, type `sudo su`, and enter failed passwords to trigger `auth. failure`.

```
ubuntu-u1@ubuntu-u1-VirtualBox:~$ sudo su
[sudo] password for ubuntu-u1:
Sorry, try again.
[sudo] password for ubuntu-u1:
Sorry, try again.
[sudo] password for ubuntu-u1:
sudo: 3 incorrect password attempts
ubuntu-u1@ubuntu-u1-VirtualBox:~$
```

Failed auth. attempt is logged in and shown live on first terminal

```
ubuntu-u1@ubuntu-u1-VirtualBox:~$ sudo tail -f /var/log/auth.log
2025-04-11T09:31:36.590371-04:00 ubuntu-u1-VirtualBox sudo: ubuntu-u1 : 3 incorrect password attempts ; TTY=pts/2 ; PWD=/home/ubuntu-u1 ; USER=root ; COMMAND=/usr/bin/su
```

## EXPERIMENT - 9

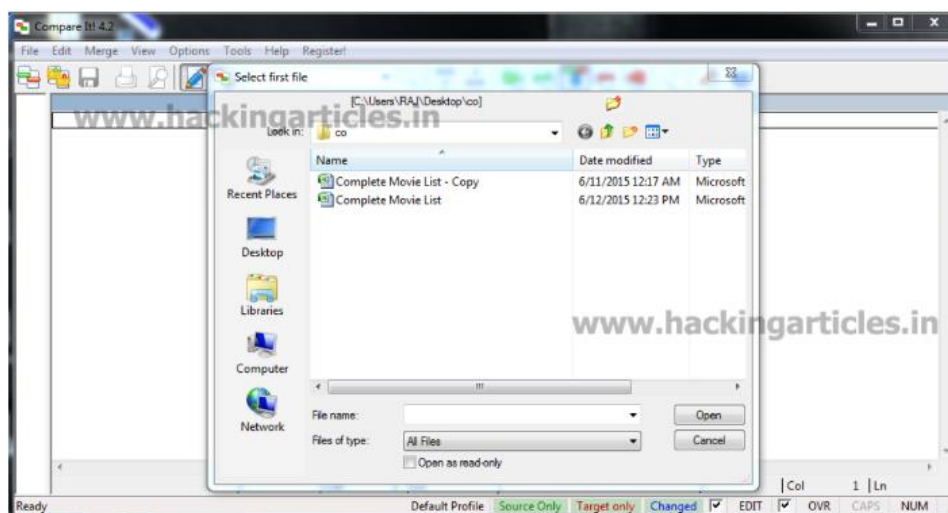
**Aim:** Comparison of two Files for forensics investigation by Compare IT software

Compare It is a software that displays 2 files side by side, with colored differences sections to simplify analyzing. You can move changes between files with a single mouse click or keystroke, and of course, you have the ability to edit files directly in comparison window. It can make colored printout of differences report, exactly as it's on the screen.

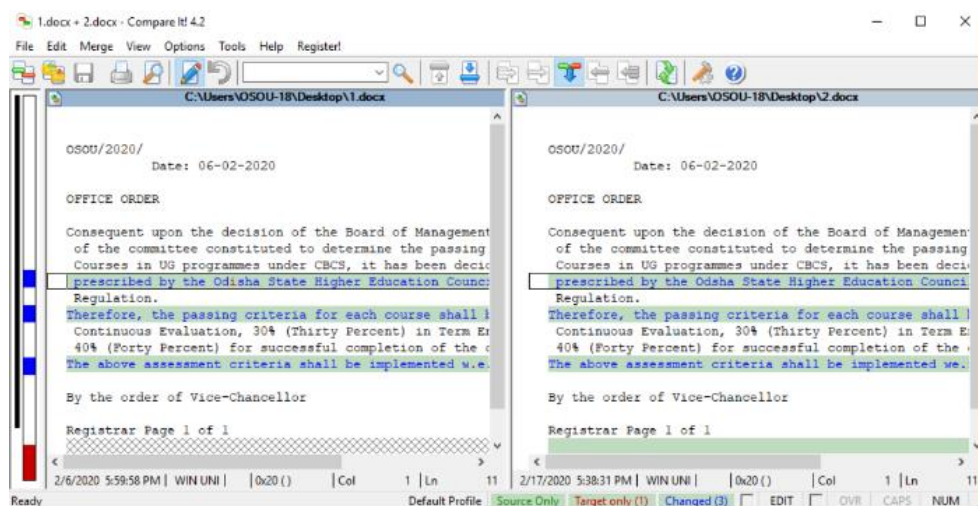
First of all, install the Compare It from the Link given below.

<http://www.grigsoft.com/wincmp3.htm> it is a 1.7 Mb Software package

Click on Compare It Tool, It will show a window to select the files to be compared. First, select the first file and click on open and then select the second file and click on open.



Now it will show us the changes in the highlighted bar.





It also gives you Print report of the difference in the file as follows

2/17/2020 5:54:13 PM

C:\Users\OSOU-18\Desktop\1.docx		C:\Users\OSOU-18\Desktop\2.docx	
1		1	
2		2	
3	OSOU/2020/	3	OSOU/2020/
4	Date: 06-02-2020	4	Date: 06-02-2020
5		5	
6	OFFICE ORDER	6	OFFICE ORDER
7		7	
8	Consequent upon the decision of the	8	Consequent upon the decision of the
9	Board of Management (BOM) and the	9	Board of Management (BOM) and the
10	resolution	10	resolution
11	of the committee constituted to	11	of the committee constituted to
12	determine the passing criteria in	12	determine the passing criteria in
13	different	13	different
14	Courses in UG programmes under CBCS	14	Courses in UG programmes under CBCS
15	, it has been decided to adopt the	15	, it has been decided to adopt the
16	criteria	16	criteria
17	prescribed by the Odisha State Higher	17	prescribed by the Odisha State Higher
18	Education Council (OSHEC) under Model	18	Education Council (OSHEC) under Model
19	Regulation.	19	Regulation.
20	Therefore, the passing criteria for	20	Therefore, the passing criteria for
21	each course shall be 40% (Forty	21	each course shall be 30% (Forty
22	Percent) in	22	Percent) in
23	Continuous Evaluation, 30% (Thirty	23	Continuous Evaluation, 30% (Thirty
24	Percent) in Term End Examination and	24	Percent) in Term End Examination and
25	overall	25	overall
26	40% (Forty Percent) for successful	26	40% (Forty Percent) for successful
27	completion of the course.	27	completion of the course.
28	The above assessment criteria shall be	28	The above assessment criteria shall be
29	implemented w.e.f. TEE-Dec 2019.	29	implemented w.e.f. TEE-Dec 2019.
30		30	
31	By the order of Vice-Chancellor	31	By the order of Vice-Chancellor
32		32	
33	Registrar Page 1 of 1	33	Registrar Page 1 of 1
34		34	

## EXPERIMENT - 10

**Aim:** Write a Program in Linux to Break Encryption on Password locked zip file

**Theory:** Password-protected ZIP archives are commonly used to secure sensitive files. However, weak or commonly used passwords can be susceptible to brute-force or dictionary-based attacks.

fcrackzip is a lightweight, command-line tool in Linux that can be used to perform dictionary or brute-force attacks on encrypted ZIP archives.

fcrackzip -v -u -b -c a -l 5-10 secret.zip

-v: Verbose mode (display all break-in attempts)

-u: Try to unzip each file (verifies password)

-b: Brute-force mode

-c a: Use lowercase letters (a set)

-l 5-10: Try passwords from length 5 to 10 (just an e.g.)

**Code:** Converting an already existing secret.txt to secret.zip file and applying a password to it

```
ubuntu-u1@ubuntu-u1-VirtualBox:~$ zip -e secret.zip secret.txt
Enter password:
Verify password:
  adding: secret.txt (deflated 3%)
ubuntu-u1@ubuntu-u1-VirtualBox:~$ fcrackzip -v -b -c a -l 3 secret.zip
found file 'secret.txt', (size cp/uc    76/    66, flags 9, chk 552d)
possible pw found: aaa ()
possible pw found: acx ()
possible pw found: awo ()
possible pw found: axv ()
```

Password found after verifying it as input to zip file

```
ubuntu-u1@ubuntu-u1-VirtualBox:~$ fcrackzip -v -u -b -c a -l 3 secret.zip
found file 'secret.txt', (size cp/uc    76/    66, flags 9, chk 552d)

PASSWORD FOUND!!!!: pw == aaa
```