

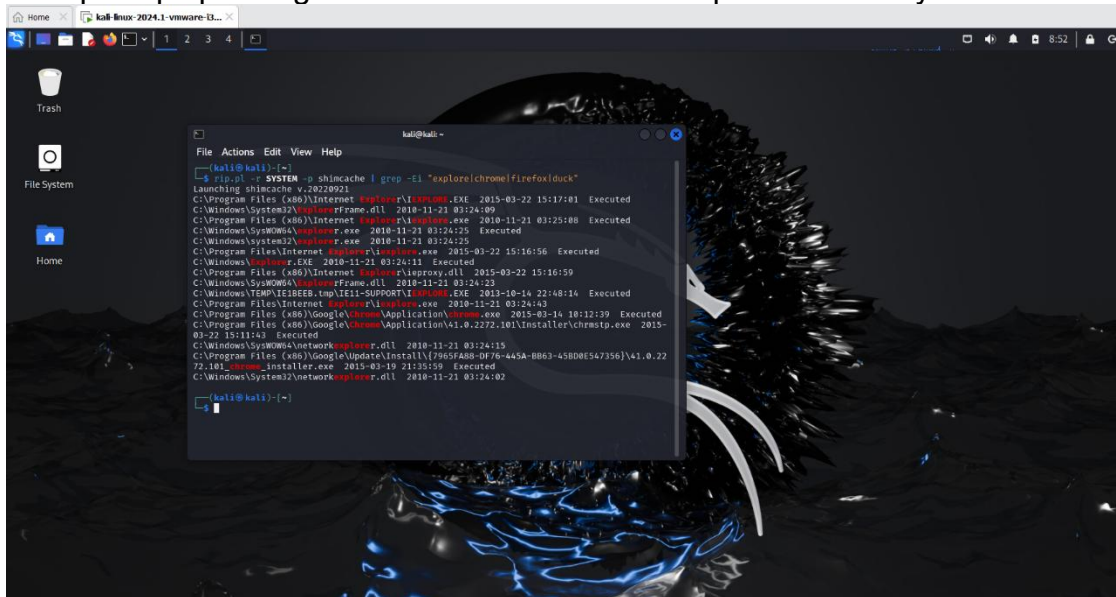
ITS60904

COMPUTER CRIME AND DIGITAL EVIDENCE

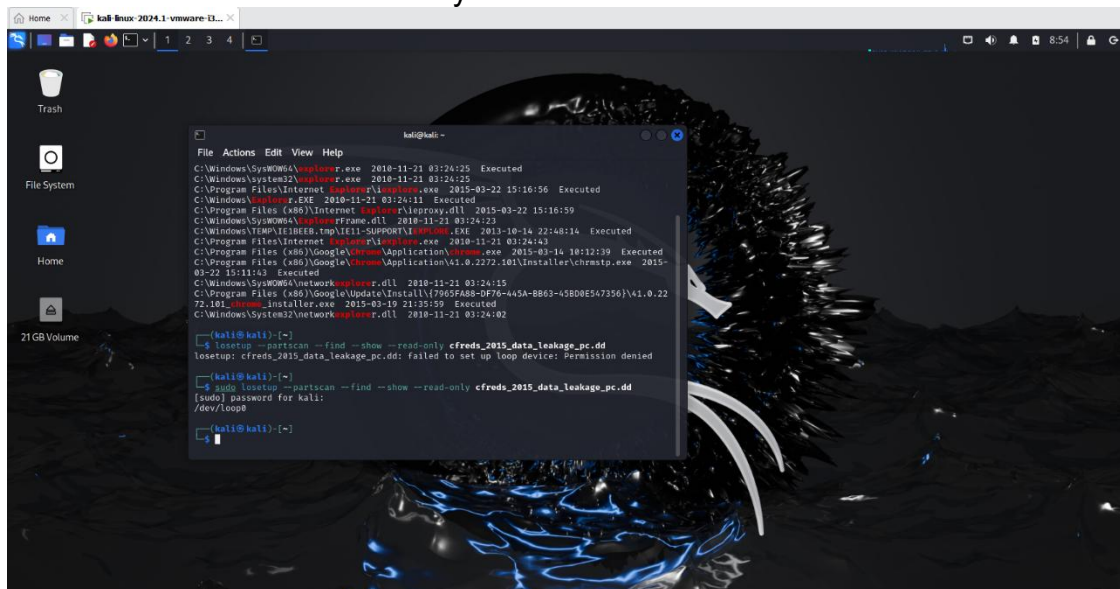
PRACTICAL 3 LAB REPORT

1. What web browsers were used?

- Find all possible browsers used by suspects including only those related to popular browsers like Chrome, Firefox, Safari, Opera, and Edge. This approach helps in pinpointing the browser activities of suspects efficiently.



- To set up a loop device, create a file to use as the virtual disk. Then, use `losetup` to associate this file with a loop device, making it accessible as a block device. There must be `sudo` as this command allows a permitted user to execute a command as the superuser or another user, providing elevated permissions temporarily. Finally, mount the loop device to make its contents visible and accessible in the filesystem.



- Monitoring log files of all versions of Internet Explorer aid in identifying potential

The screenshot shows a Kali Linux desktop environment. The background is a dark, abstract image. On the left side, there is a vertical dock with icons for Home, File System, Home, and 2TB Volume. The top of the screen displays a window manager bar with the title 'kali-linux-2024.1-vmware-ESX...' and a taskbar with icons for Home, File System, Home, and 2TB Volume. The system clock in the top right corner shows 8:57.

A terminal window is open in the center, displaying the following commands and output:

```
kali@kali:~$
File Actions Edit View Help
C:\Windows\System32\networkexplorer.dll 2010-11-21 03:12:42

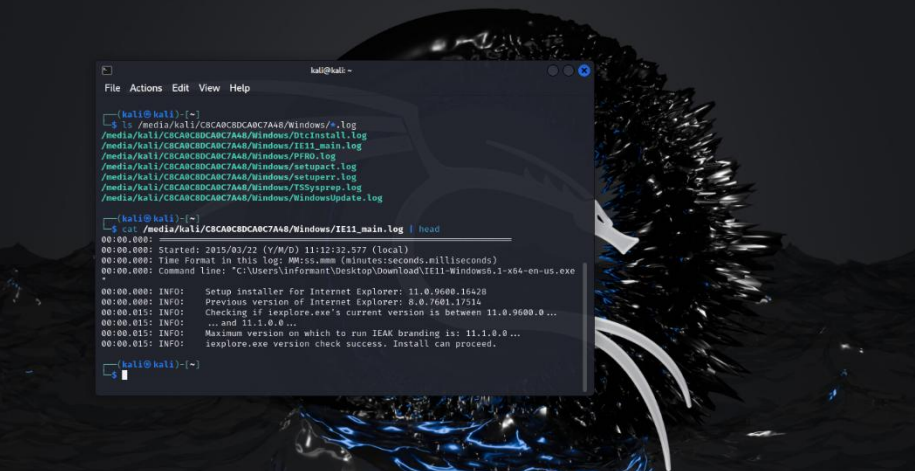
kali@kali:~$ lssetup -partscan -find -show --read-only cfreds_2015_data_leakage_pc.dd
lssetup: cfreds_2015_data_leakage_pc.dd: failed to set up loop device: Permission denied

kali@kali:~$ lssetup -partscan -find -show --read-only cfreds_2015_data_leakage_pc.dd
[udev] password for kali:
/dev/loop0

kali@kali:~$ ls -ls /media/kali/CSCA8C8DC8C7A48/Windows/*.log
ls: cannot access '/media/kali/CSCA8C8DC8C7A48/Windows/*.log': No such file or directory

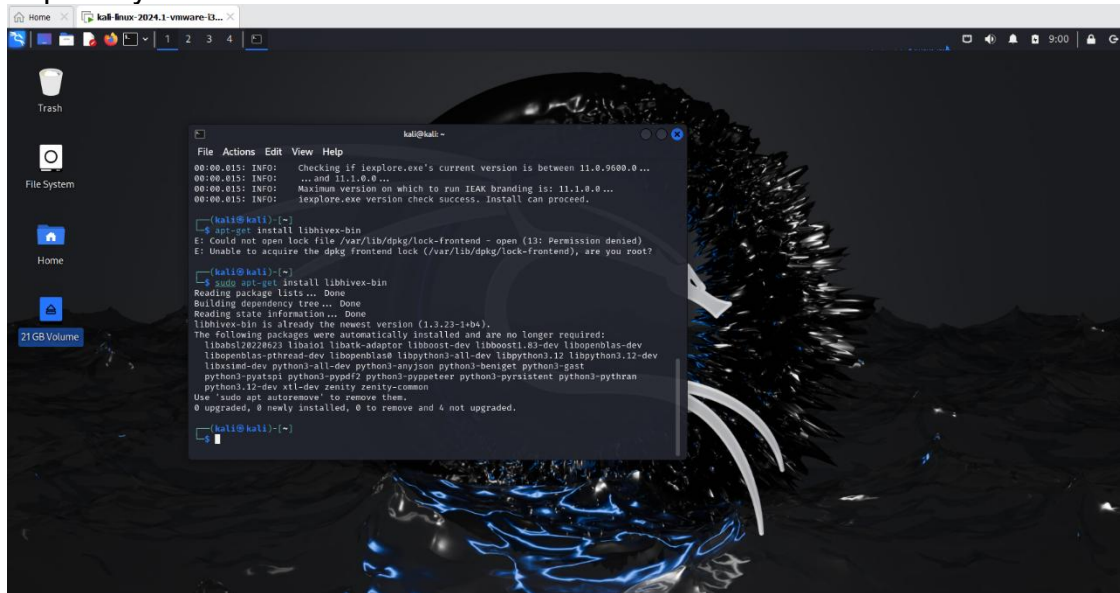
kali@kali:~$ ls -ls /media/kali/CSCA8C8DC8C7A48/Windows/*.log
/media/kali/CSCA8C8DC8C7A48/Windows/GitInstall.log
/media/kali/CSCA8C8DC8C7A48/Windows/IE11_main.log
/media/kali/CSCA8C8DC8C7A48/Windows/PPROF.log
/media/kali/CSCA8C8DC8C7A48/Windows/setupact.log
/media/kali/CSCA8C8DC8C7A48/Windows/setuperr.log
/media/kali/CSCA8C8DC8C7A48/Windows/TSSprep.log
/media/kali/CSCA8C8DC8C7A48/Windows/WindowsUpdate.log

kali@kali:~$
```

- 
- The screenshot shows a Kali Linux desktop environment. In the background, there is a dark, abstract wallpaper. The desktop has several icons: a trash can, a file system icon, a home icon, and a 2TB Volume icon. A terminal window is open in the foreground, displaying the following text:
- ```
kali@kali: ~
File Actions Edit View Help
kali@kali:~$ ls /media/kali/CSCA8C8DCARC7AAB/Windows/ -log
/media/kali/CSCA8C8DCARC7AAB/Windows/gtciinstall.log
/media/kali/CSCA8C8DCARC7AAB/Windows/IE11_main.log
/media/kali/CSCA8C8DCARC7AAB/Windows/PPROD.log
/media/kali/CSCA8C8DCARC7AAB/Windows/setupact.log
/media/kali/CSCA8C8DCARC7AAB/Windows/setuperr.log
/media/kali/CSCA8C8DCARC7AAB/Windows/TSdypprep.log
/media/kali/CSCA8C8DCARC7AAB/Windows/WindowsUpdate.log
kali@kali:~$ cat /media/kali/CSCA8C8DCARC7AAB/Windows/IE11_main.log | head
00:00:00:
00:00:00: Started: 2015/03/22 (Y/M/D) 11:12:32.577 (local)
00:00:00: Time Format in this log: MM:ss.mmm (minutes:seconds.milliseconds)
00:00:00: Command Line: "C:\Users\informant\Desktop\Download\IE11-Windows8.1-x64-en-us.exe"
00:00:00: INFO: Setup installer for Internet Explorer: 11.0.9600.16428
00:00:00: INFO: Previous version of Internet Explorer: 8.0.7601.17514
00:00:01: INFO: Checking if iexplore.exe's current version is between 11.0.9600.0...
00:00:01: INFO: ...and 11.1.0.0...
00:00:01: INFO: Maximum version on which to run IEAK branding is: 11.1.0.0...
00:00:01: INFO: iexplore.exe version check success. Install can proceed.
kali@kali:~$
```

- Install hivexsh

There must be `sudo` as this command allows a permitted user to execute a command as the superuser or another user, providing elevated permissions temporarily.

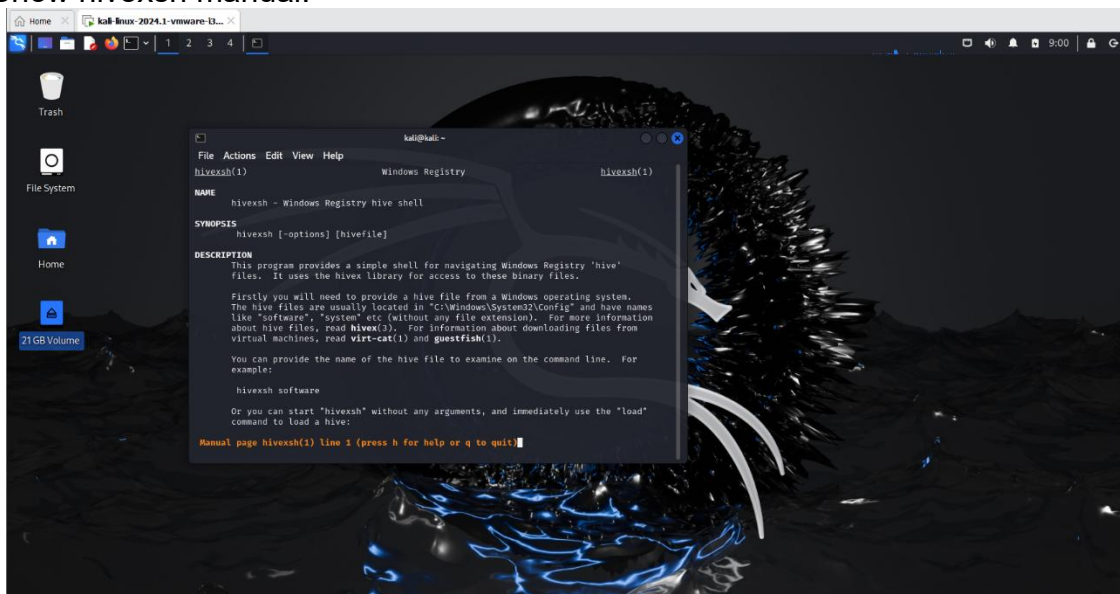


```
kali@kali:~$ sudo apt-get install libhivex-bin
E: Could not open lock file /var/lib/dpkg/lock-frontent - open (13: Permission denied)
E: Unable to acquire the dpkg frontend lock (/var/lib/dpkg/lock-frontent), are you root?

kali@kali:~$ sudo apt-get install libhivex-bin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
libhivex-bin is already the newest version (1.3.22-1+b1).
The following packages were automatically installed and are no longer required:
libabsl20220623 libasio libatk-adaptor libboost-dev libboost1.83-dev libopenblas-dev
libopenblas-pthread-dev libopenblas libpython3-all-dev libpython3.12 libpython3.12-dev
libxsimd-dev python3-all-dev python3-anyjson python3-beniget python3-gast
python3-pyatspi python3-pypdf2 python3-pyppeteer python3-pyrsistent python3-pythrane
python3.12-dev x11-dev zenity zenity-common
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 4 not upgraded.

kali@kali:~$
```

- Show hivexsh manual.



```
kali@kali:~$ hivexsh(1)
NAME
 hivexsh - Windows Registry hive shell

SYNOPSIS
 hivexsh [-options] [hivefile]

DESCRIPTION
 This program provides a simple shell for navigating Windows Registry 'hive'
 files. It uses the hivex library for access to these binary files.

 Firstly you will need to provide a hive file from a Windows operating system.
 The hive files are usually located in "C:\Windows\System32\Config" and have names
 like "software", "system" etc (without any file extension). For more information
 about hive files, read hivex(1). For information about downloading files from
 virtual machines, read virt-cat(1) and guestfish(1).

 You can provide the name of the hive file to examine on the command line. For
 example:

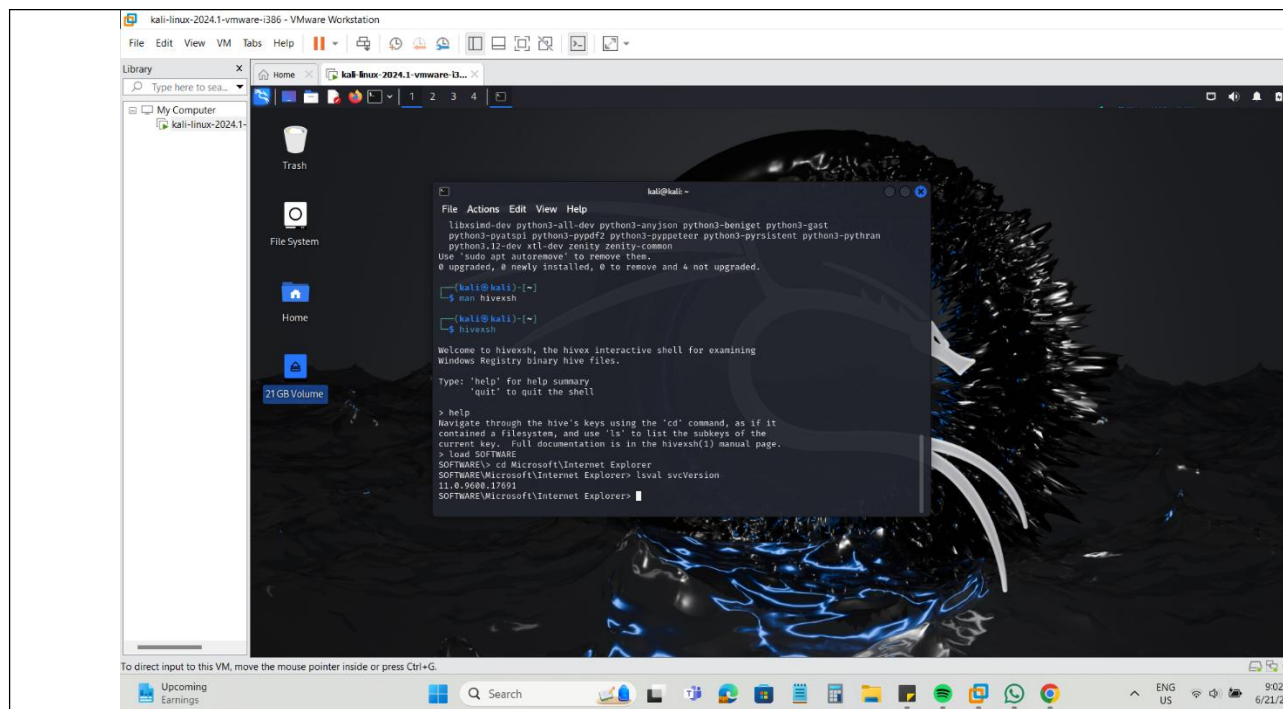
 hivexsh software

 Or you can start 'hivexsh' without any arguments, and immediately use the "load"
 command to load a hive:

 Manual page hivexsh(1) line 1 (press h for help or q to quit)
```

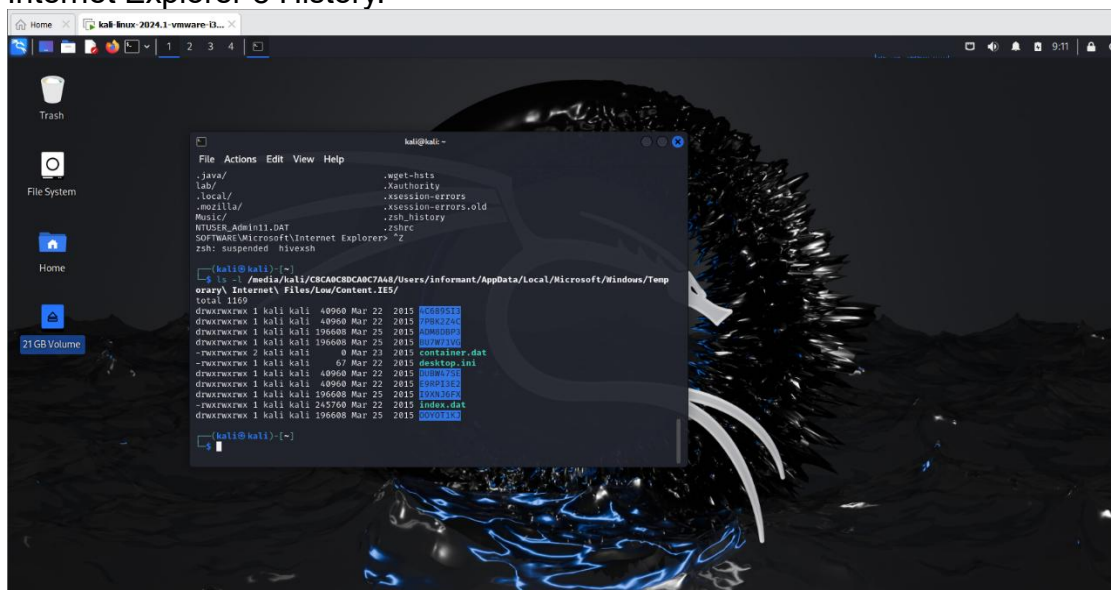
- Exam the version of IE using hivexsh  
This method is to analyze the version of Internet Explorer using `hivexsh` that would help to know specifically the type of vulnerability and whether it implements the standards of the web or not. This step is helpful in the evaluation of the security vulnerability of the system because the various IE versions come with different levels of risks and susceptibility to threats and attacks.



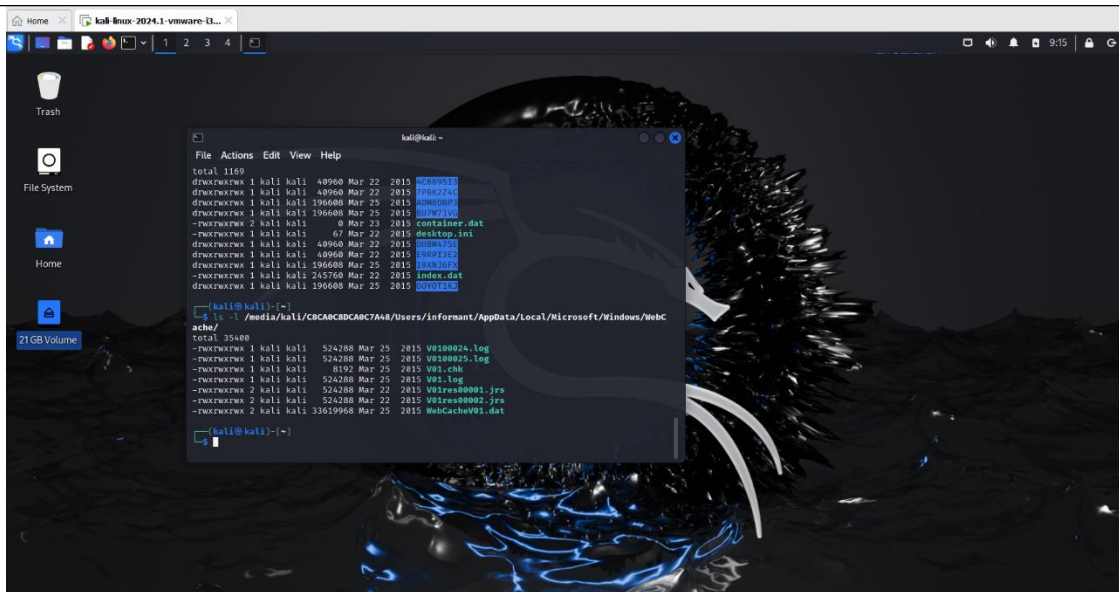


### 3. Identify directory/file paths related to the web browser history.

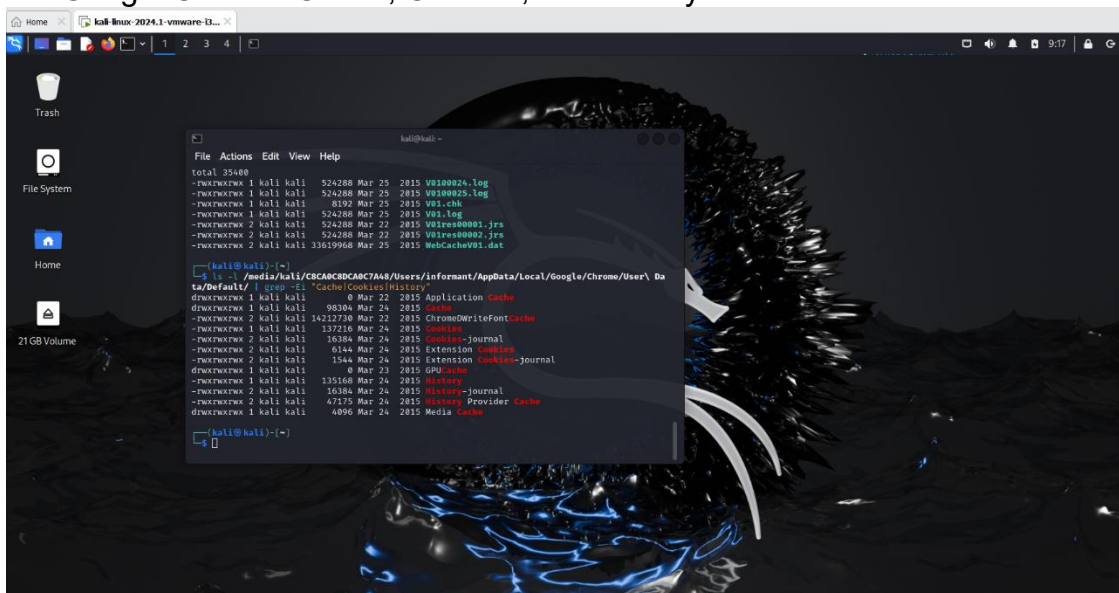
- Internet Explorer 8 History.



- Internet Explorer 11 History.

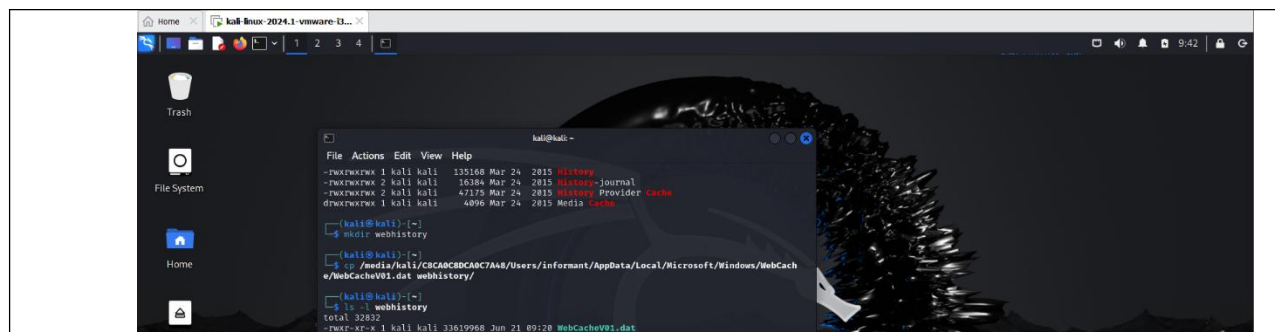


- Find Google Chrome Cache, Cookies, and History.

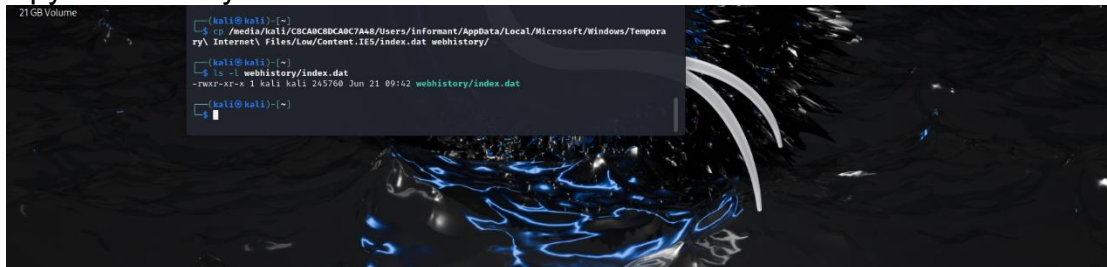


4. What websites were the suspect accessing? (Timestamp, URL...)

- Copy IE 11 History WebCacheV01.dat.



- Copy IE 8 History index.dat.

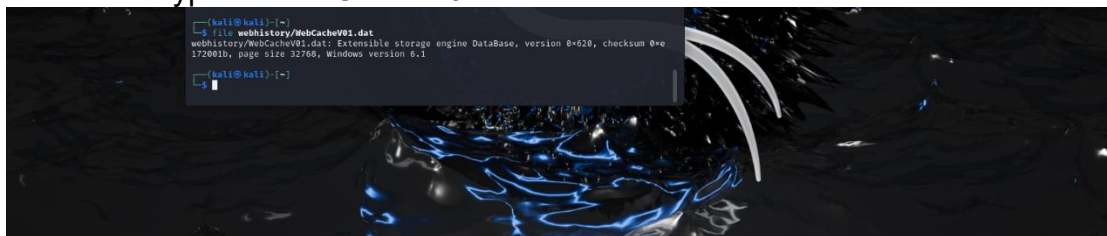


- Copy Chrome History History.



## 5. View IE 11 History Using libesedb.

- Find the file type of WebCacheV01.dat.

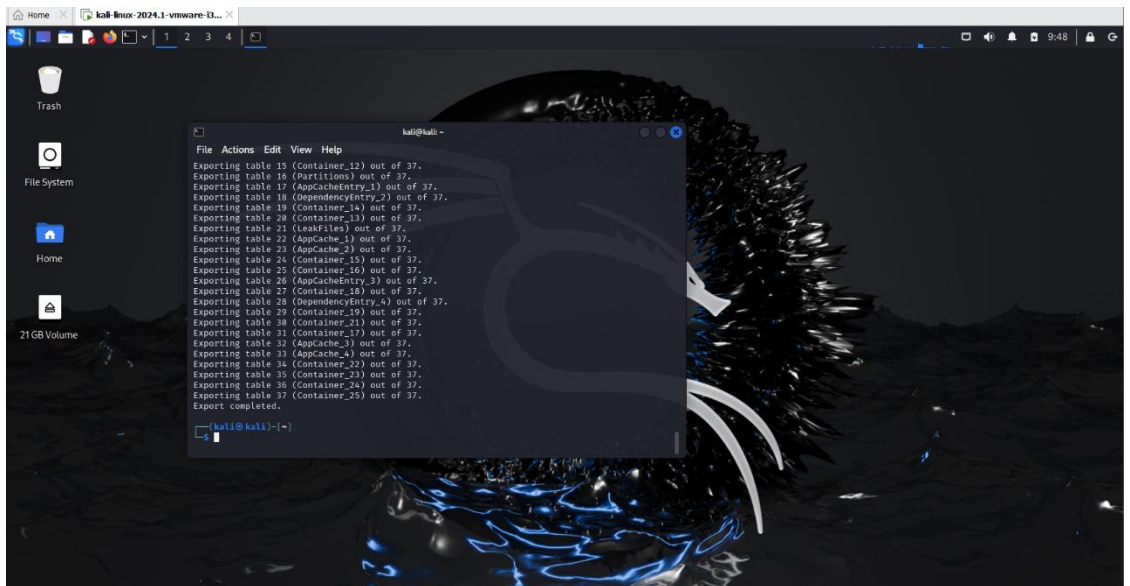
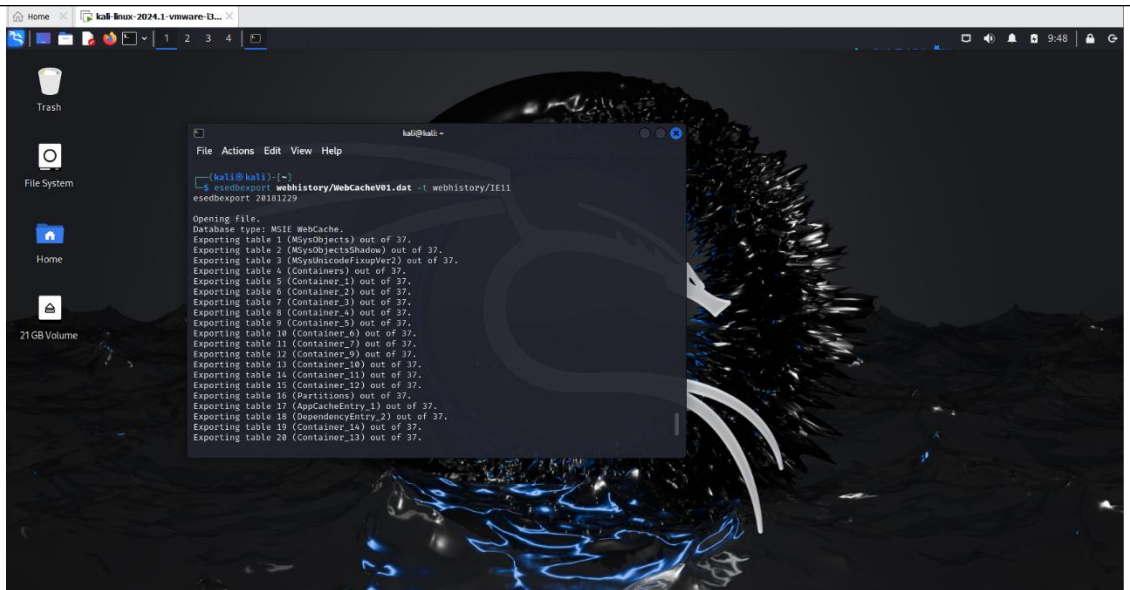


- Install libesedb

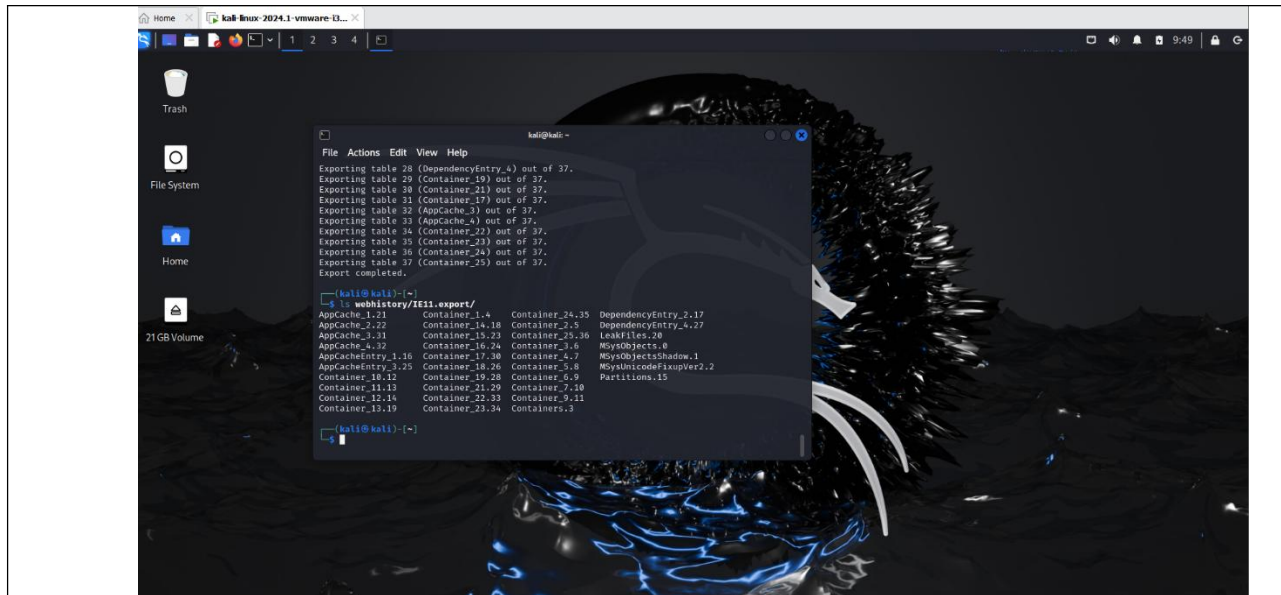
There must be `sudo` as this command allows a permitted user to execute a command as the superuser or another user, providing elevated permissions temporarily. The utilization of libesedb is helpful to gain access to the ESE Database File (.edb) format is great for getting data out of applications running on the Windows platform such as Windows Search, Windows Mail, Exchange, and Active Directory. This step is critical for data investigations, forensic and diagnostic logs, and information about a particular application's data and its activity.







- Verify the file the folder.



- Find the type of the file.



- Create three lines with three attributes. The '-e' option is used to enable interpretation of backslash escapes since we might wish to include any special character in the output. The '\n' option is to make a new line, thus enhancing the organization and legibility of the output to be generated. This step helps in producing formatted text to either scripts or command-line outputs for better arrangement of predicted structures.



- Select lines with the key word "manager".
- Select attributes 1 and 3.
- The practical operation that concerns selection is 'awk' which works by using the following syntax: '/selection\_criteria/ {action}' input-file > output-file The above operation is crucial when it comes to refining large text documents. This step creates the opportunity for selective extraction and even selective alteration of the information with the help of criteria set to make automation of analysis and reporting possible.



- Show Row Number.

```
(kali@kali)-[~]
└─$ echo -e "Frank manager 50000 \nAlex clerk 45000 \nEirc clerk 25000" | awk '{print NR, $1}'
1 Frank
2 Alex
3 Eirc
(kali@kali)-[~]
└─$
```

- Calculate sum.

```
(kali@kali)-[~]
└─$ echo -e "Frank manager 50000 \nAlex clerk 45000 \nEirc clerk 25000" | awk '{sum += $2}; END {print sum}'
120000
```

- Show field names. There must be `` for it to be able to work.

```
(kali@kali)-[~]
└─$ cat webhistory/IE11.export/Container_1.4 | head -n 1
cat: webhistory/IE11.export/Container: No such file or directory
cat: 1.4: No such file or directory
(kali@kali)-[~]
└─$ cat webhistory/IE11.export/Container_1.4 | head -n 1
EntryId ContainerId Cached UrlHash SecureDirectory FileSize ModifiedTime Type Flags Acc AccessedTim
e PostCheckTime SyncCount CreationTime ExemptionDelta Url Filename FileExtensi
on RequestHeaders ResponseHeaders RedirectUrl Group ExtraData
(kali@kali)-[~]
└─$
```

- Separate fields with tab 't' and show ModifiedTime and URL. Fields must be separated by the tab \t. The necessity of displaying ModifiedTime and URL arises due to the further correct display of data. This step is useful in simplification the comparison of large numbers both in the timestamps and URLs which is useful in such tasks as monitoring of changes, web activity, and in forensic analysis.

```
(kali@kali)-[~]
└─$ awk '{print NR, $13, $18}' FS="\t" webhistory/IE11.export/Container_1.4 | head -n 5
1 ModifiedTime Url
2 Feb 27, 2015 00:14:145.0000000000 https://technet.microsoft.com/favicon.ico
3 Mar 20, 2015 14:10:139.0000000000 http://www.wired.com/wp-content/themes/Phoenix/assets/ima
ges/favicon.ico
4 Mar 20, 2015 14:10:139.0000000000 http://www.wired.com/favicon.ico
5 Jan 01, 1401 04:00:000.0000000000 https://licenase.piriform.com/verify/?nccpr0bccccbcv5.0k
.S15101-10330Lk+C39T-37CU-SPIV-QWMB-WBEC0kx+V5NW-75UN-3WNU-QD1I-6UFT-T7TM-9AX9-8Z08-USQZ
(kali@kali)-[~]
└─$
```

- Count the number records in the file.

```
(kali@kali)-[~]
└─$ awk '{print NR, $13, $18}' FS="\t" webhistory/IE11.export/Container_1.4 | wc -l
58
(kali@kali)-[~]
└─$
```

- Count the number records in all files start with the string "Container".

```
(kali@kali)-[~]
└─$ awk '{print NR, $13, $18}' FS="\t" webhistory/IE11.export/Container* | wc -l
10248
(kali@kali)-[~]
└─$
```



## 6. View IE 8 History using pasco.

- Exam the file type.

```
(kali@kali)-[~]
└─$ file webhistory/index.dat
webhistory/index.dat: Internet Explorer cache file version Ver 5.2
└─$
```

- Install pasco.
- pasco - tool to extract information from index.dat. There must be `sudo` as this command allows a permitted user to execute a command as the superuser or another user, providing elevated permissions temporarily.

```
File Actions Edit View Help
┌─(kali@kali)-[~]
│ └─$ cat webhistory/index.dat
│ webhistory/index.dat: Internet Explorer cache file version Ver 5.2
│ └─$
│ └─$ sudo apt-get install pasco
│ [sudo] password for kali:
│ Reading package lists... Done
│ Building dependency tree... Done
│ Reading state information... Done
│ pasco is already the newest version (20040905-4).
│ The following packages were automatically installed and are no longer required:
│ libbabel1022002 libasol libatk-adaptor libboost-dev libboost1.03-dev libopenblas-dev
│ libopenblas-pthread-dev libopenblas libpython3-all-dev libpython3.12 libpython3.12-dev
│ libxsimd-dev python3-all-dev python3-anyjson python3-beniget python3-gast
│ python3-pytsapi python3-pydf2 python3-pypeteer python3-pyrsistent python3-pythran
│ python3.12-dev xtl-dev zenity zenity-common
│ Use 'sudo apt autoremove' to remove them.
│ 0 upgraded, 0 newly installed, 0 to remove and 4 not upgraded.
│ └─$
```

- Show help.

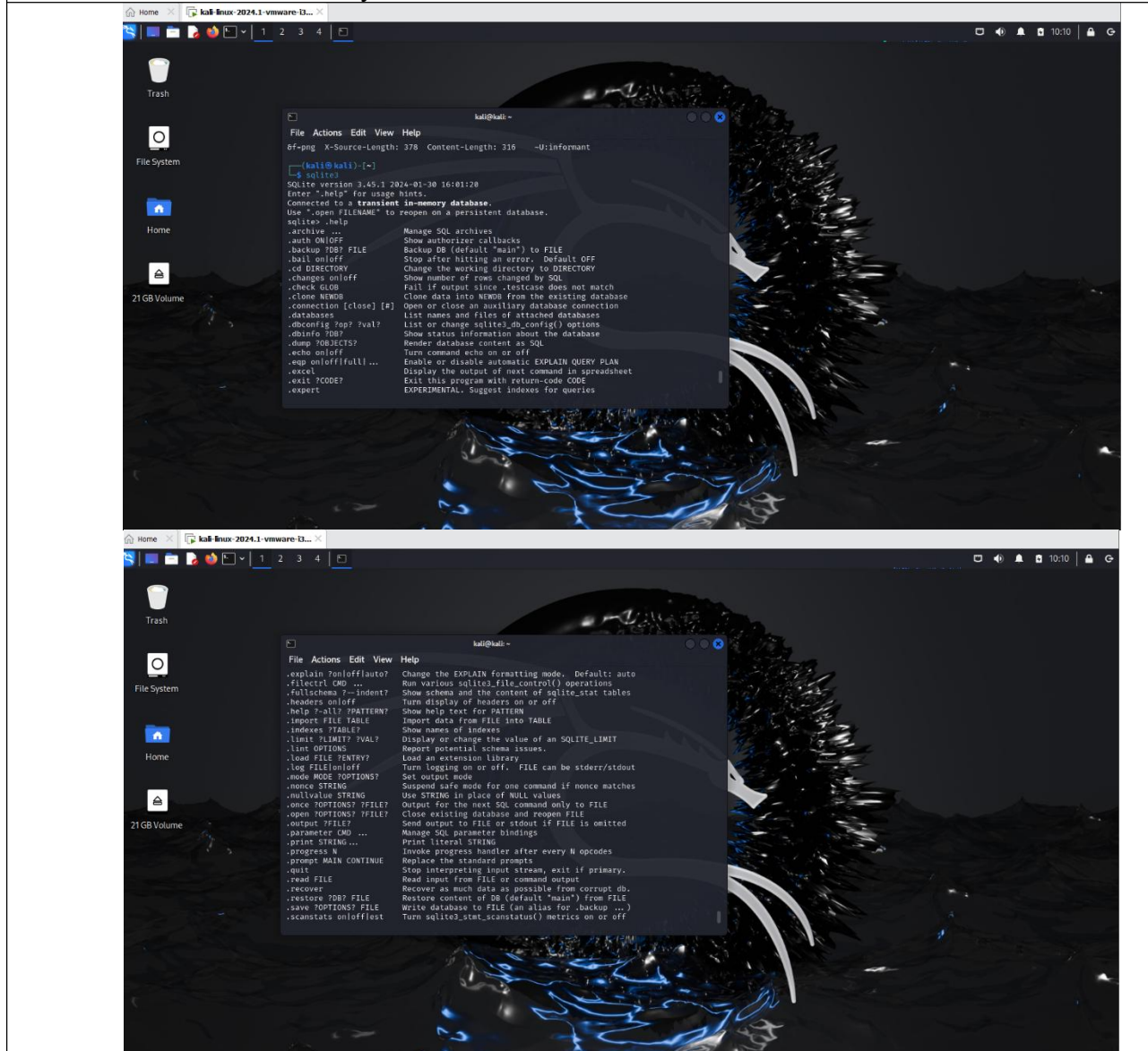
```
┌─(kali@kali)-[~]
│ └─$ pasco -h
│ ERROR - The index.dat file cannot be opened!
│
│ Usage: pasco [options] <filename>
│ -d Undelete Activity Records
│ -t Field Delimiter (TAB by default)
│ └─$
```

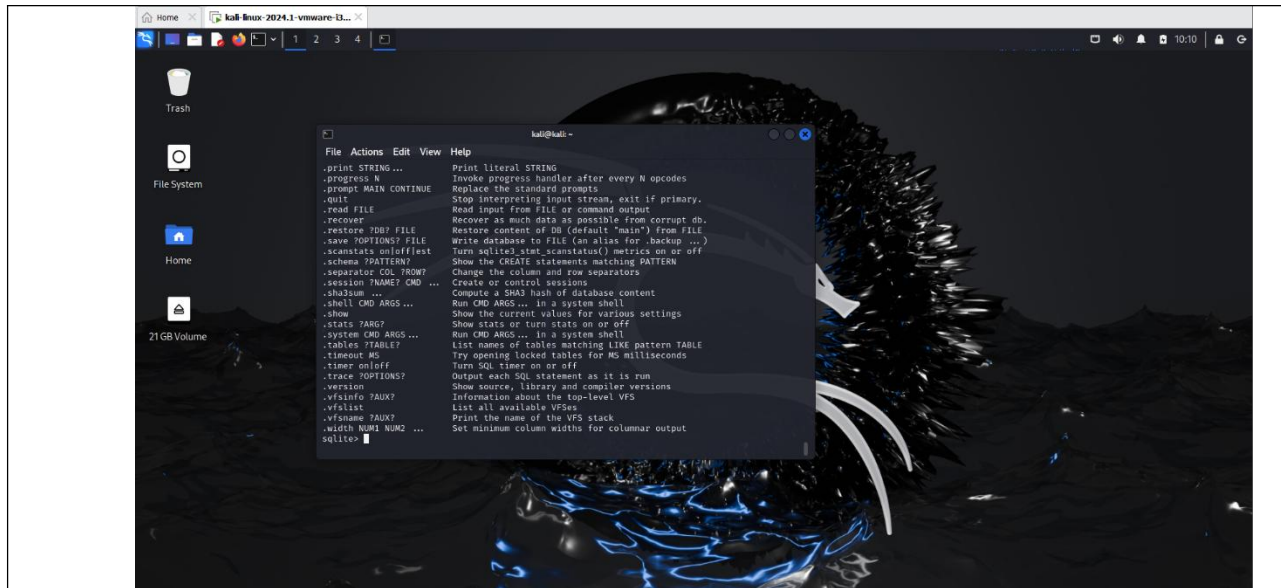
- Exam index.dat.



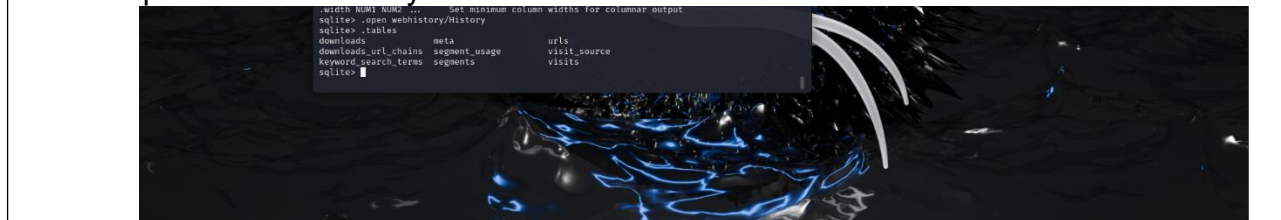


## 7. View Chrome History SQLite.

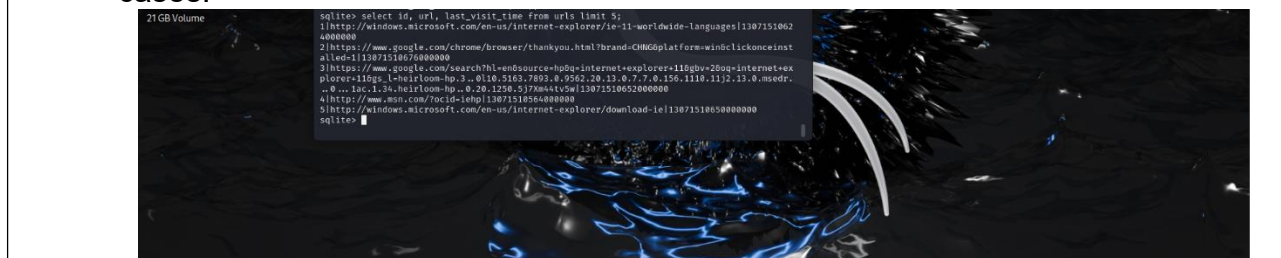




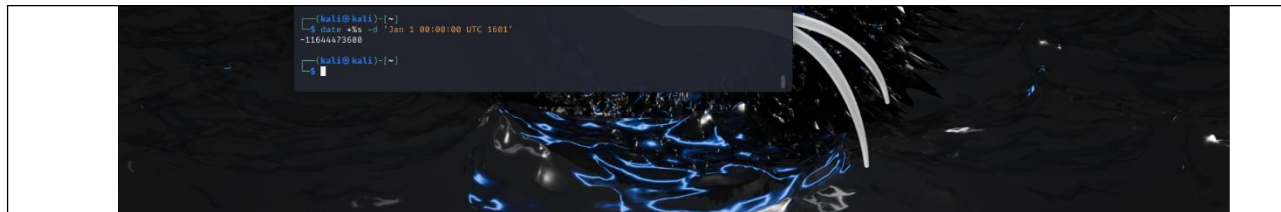
- Open Chrome history.



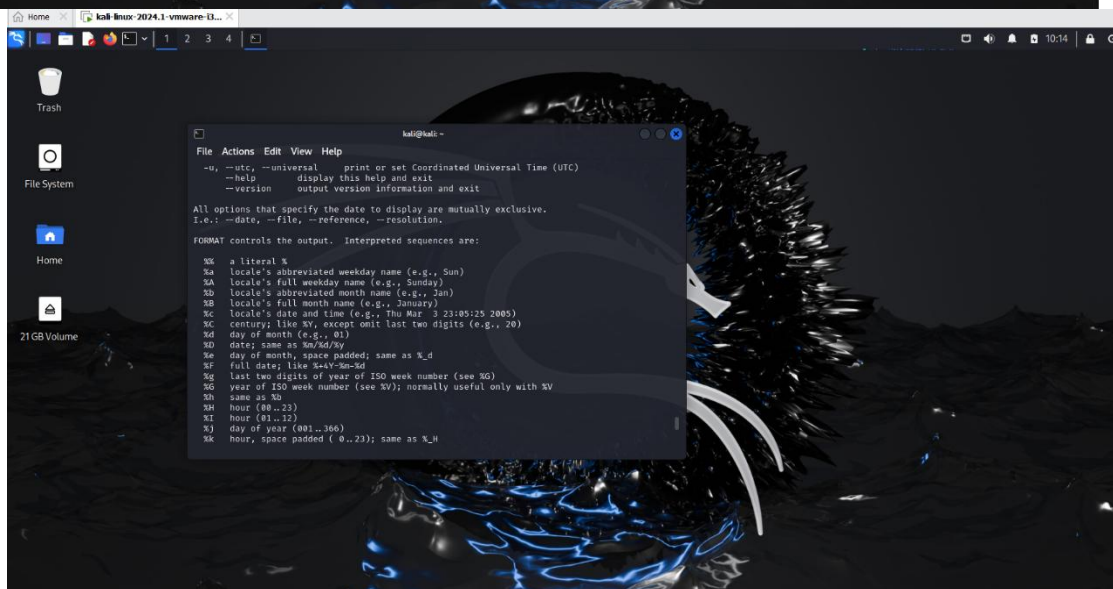
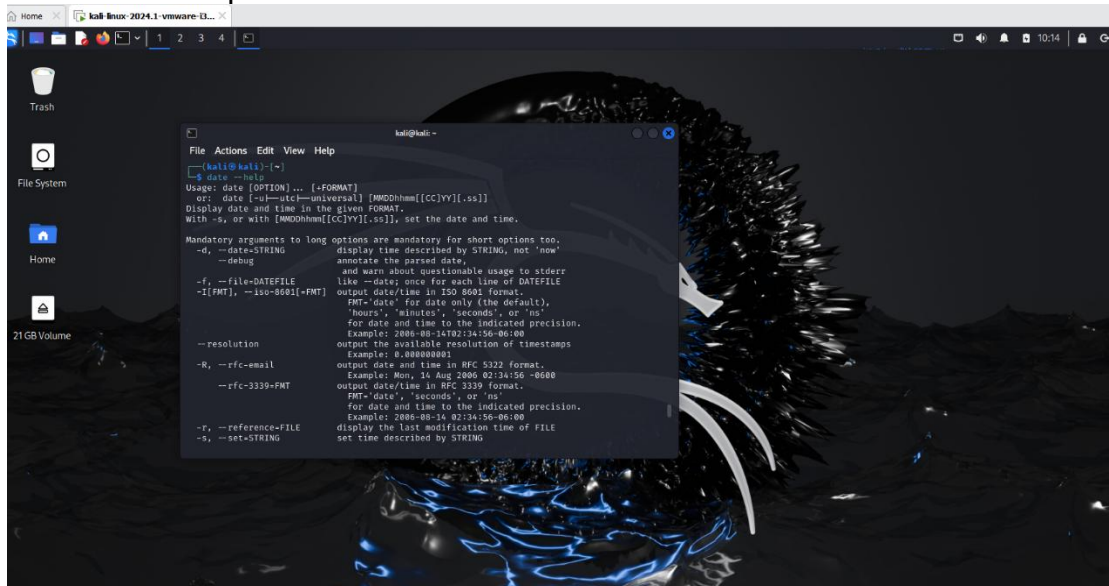
- This format of timestamp `1307151065000000` is implemented in Apple Safari (WebKit), Google Chrome, and Opera browsers (Chromium/Blink) and is hugely beneficial in timing and synchronizing web activity over a while. The time is accurately represented by 64 bits of microsecond since the year 1601-Jan-01 to facilitate logging (audit trails) which is help for debugging or to solve any crime cases.



- The number of seconds elapsed since 01/01/1970 00:00:00. %s: seconds since 1970-01-01 00:00:00 UTC.



- List of Format specifiers used with date command.

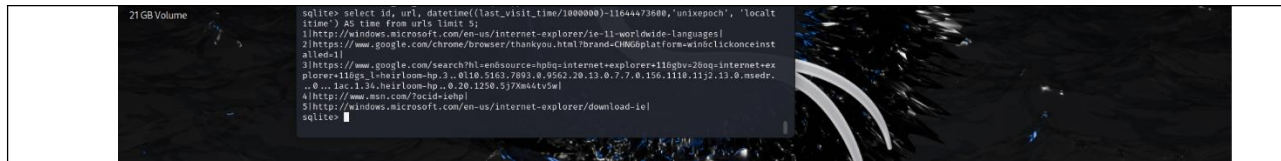






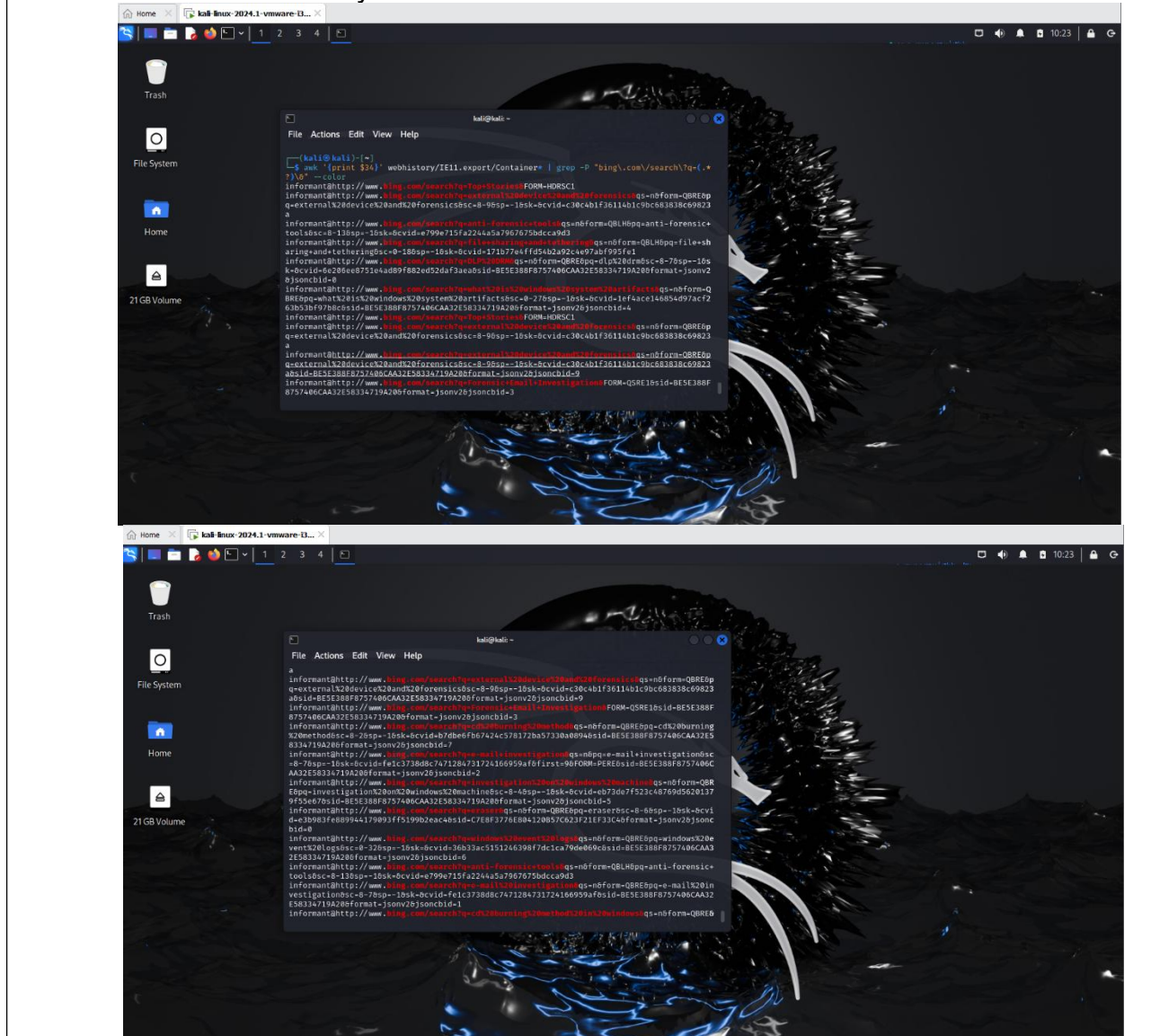
- Datetime function to convert webkit time to Unix epoch.

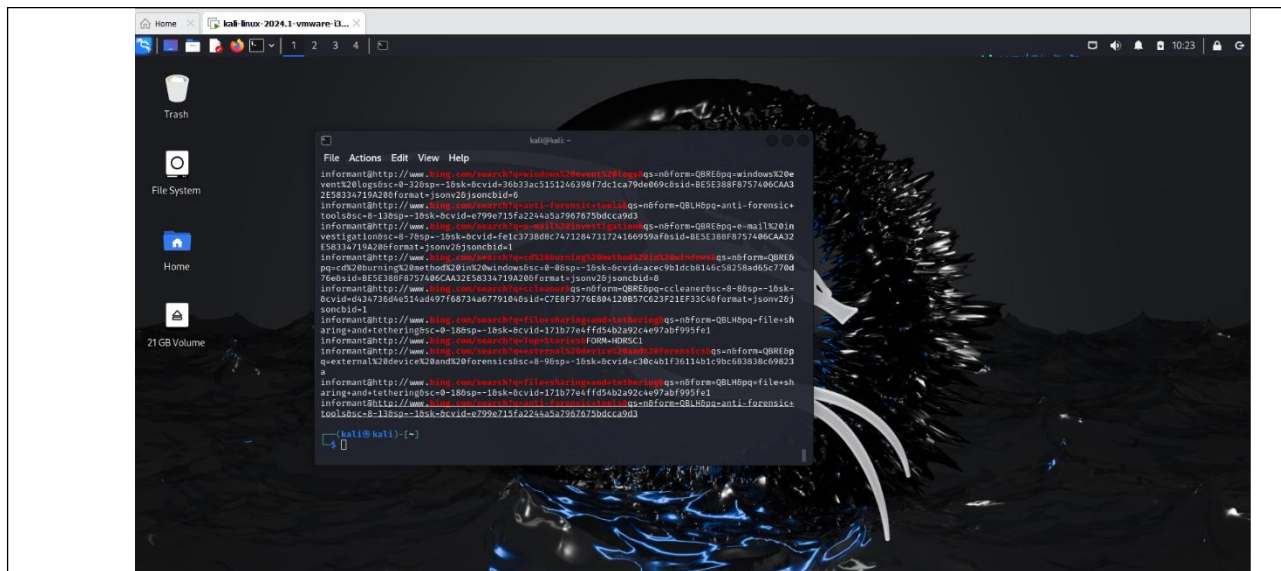




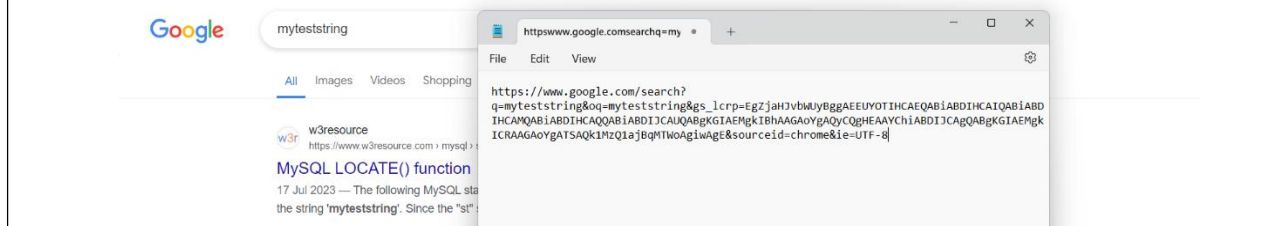
8. List all search keywords using web browsers. (Timestamp, URL, keyword...)

- List IE 11 search keywords.

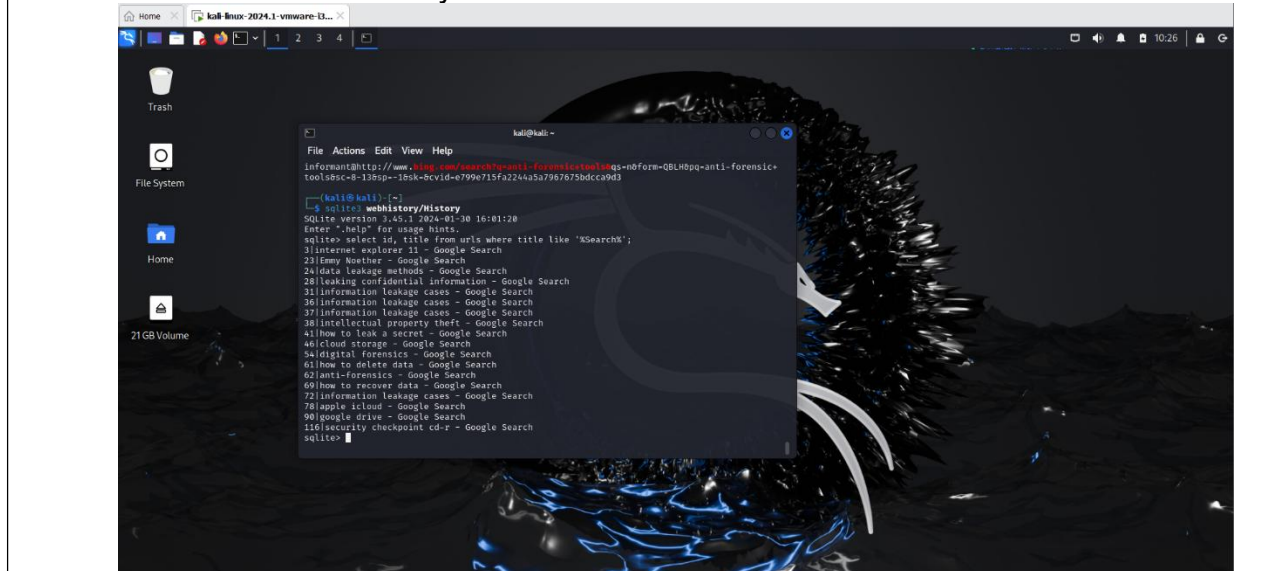




- Chrome browser.



- List Chrome search keywords.



9. List all user keywords at the search bar in Windows Explorer. (Timestamp, Keyword)

- rip.pl on NTUSER.DAT file with wordwheelquery plugin. Running `rip. pl` on an `NTUSER. A` .DAT` file, particularly generated by the `wordwheelquery` plugin, is relevant for the identification of recent search terms entered by a user on a system running on Windows. This step is of utmost importance in the digital forensic investigation, allows germinating insights toward users' details, search

history, and possibly evidence related with the case being investigated.

