Digital Forensic
Report-s5348038- Muhammad Hamza Aslam

Evidence A.

Evidence A contains **11 disk** image files that were combined into a single dd image as shown in Figure 1. The hash of the file was taken before investigating the files to ensure integrity of data throughout the process.

```
sansforensics@siftworkstation: ~/Downloads/EvidenceA
$ ls
EvidenceA.001  EvidenceA.002  EvidenceA.003  EvidenceA.004  EvidenceA.005  EvidenceA.006  EvidenceA.007  EvidenceA.008  EvidenceA.009  EvidenceA.010  EvidenceA.011
sansforensics@siftworkstation: ~/Downloads/EvidenceA
$ cat EvidenceA.001 EvidenceA.002 EvidenceA.003 EvidenceA.004 EvidenceA.005 EvidenceA.006 EvidenceA.007 EvidenceA.008 EvidenceA.009 EvidenceA.010 EvidenceA.011 > EvidenceA.dd
sansforensics@siftworkstation: ~/Downloads/EvidenceA
$ md5sum EvidenceA.dd
e2163d35fb453047af7534d12de89055  EvidenceA.dd
```

Figure 1: Combining Files and Matching Hash

I used the **img_stat** to extract information **EvidenceA.dd**. It shows that the image is type **raw** with sector size of **512 bytes**. These details are crucial for further investigation.

```
$ img_stat EvidenceA.dd
IMAGE FILE INFORMATION
--------------------------------------------
Image Type: raw

Size in bytes: 21474836480
Sector size:    512
```

Figure 2 : Result of Img_stat

I use **mmls** command to display partition structure of EvidenceA.dd. This enable me to look at the partitions having more data, and of interest partition **006**, then I calculated the offset of partition by **512 X 673792**, resulting an offset of **344293376** bytes. This step was important for targeting further analysis. After this I used the **fsstat** command to gather additional file system details. The output confirmed that the file system on partition 006 is **exfat/NTFS** having windows format, with a cluster size of 4096 bytes. As shown in the following figures

```
$ mmls EvidenceA.dd
GUID Partition Table (EFI)
Offset Sector: 0
Units are in 512-byte sectors

     Slot    Start        End          Length       Description
000:  Meta    0000000000   0000000000   0000000001   Safety Table
001:  -------  0000000000   0000002047   0000002048   Unallocated
002:  Meta    0000000001   0000000001   0000000001   GPT Header
003:  Meta    0000000002   0000000033   0000000032   Partition Table
004:  000     0000002048   0000411647   0000409600   EFI system partition
005:  001     0000411648   0000673791   0000262144   Microsoft reserved partition
006:  002     0000673792   0041940991   0041267200   Basic data partition
007:  -------  0041940992   0041943039   0000002048   Unallocated
```

Figure 3 : Partition Layout

```
$ fsstat -o 673792 EvidenceA.dd
FILE SYSTEM INFORMATION
--------------------------------------------
File System Type: NTFS
Volume Serial Number: 90B06F6AB06F562E
OEM Name: NTFS
Version: Windows XP

METADATA INFORMATION
--------------------------------------------
First Cluster of MFT: 786432
First Cluster of MFT Mirror: 2
Size of MFT Entries: 1024 bytes
Size of Index Records: 4096 bytes
Range: 0 - 113152
Root Directory: 5

CONTENT INFORMATION
--------------------------------------------
Sector Size: 512
Cluster Size: 4096
Total Cluster Range: 0 - 5158398
Total Sector Range: 0 - 41267198
```

Figure 4 :File System Information

After that, I mounted the partition using the command in Fig 5 with the calculated offset **344981504**. For integrity of the evidence, the specified partition was mounted in read-only mode to avoid modification of the info. The partition was mounted to **/mnt/windows_mount**

```
sansforensics@siftworkstation: ~/Downloads/EvidenceA
$ sudo mount -o ro,loop,offset=344981504 EvidenceA.dd /mnt/windows_mount
sansforensics@siftworkstation: ~/Downloads/EvidenceA
$
```

Figure 5 : Mounting the Partition

Question 1: Who is the owner of the computer?

To determine the owner, I used **RegRipper** to extract data from registry hive files. By using **`grep`** command and accessing hives like **SAM** and **Software** to find active user in Figure 6.

```
$ rip.pl -r /mnt/windows_mount/Windows/System32/config/SOFTWARE -p profilelist
Launching profilelist v.20200518
profilelist v.20200518
(Software) Get content of ProfileList key

Microsoft\Windows NT\CurrentVersion\ProfileList

Path      : %systemroot%\system32\config\systemprofile
SID       : S-1-5-18
LastWrite : 2019-12-07 09:16:08Z

Path      : %systemroot%\ServiceProfiles\LocalService
SID       : S-1-5-19
LastWrite : 2019-12-07 09:16:08Z

Path      : %systemroot%\ServiceProfiles\NetworkService
SID       : S-1-5-20
LastWrite : 2019-12-07 09:16:08Z

Path      : C:\Users\Alex Marshall
SID       : S-1-5-21-212117580-4225460857-3930097821-1000
LastWrite : 2024-08-27 13:19:41Z

Domain Accounts
```

```
$ rip.pl -r /mnt/windows_mount/Windows/System32/config/SAM -p samparse | grep Username
Launching samparse v.20200825
Username        : Administrator [500]
Username        : Guest [501]
Username        : DefaultAccount [503]
Username        : WDAGUtilityAccount [504]
Username        : Alex Marshall [1000]
```

Figure 6 : Extracting Usernames from

Figure 7 : User Directory Listing

```
sansforensics@siftworkstation: /mnt/windows_mount/Users
$ ls
'Alex Marshall'  'All Users'   Default  'Default User'   desktop.ini   Public
```

Figure 8 : User Directory Listing

To further confirm evidence, I use plugin **profilelist (Fig 7)** to get info about the current active or previous active accounts on the machine and also User directory shows the **Alex** is **primary** User with **ID 1000 (Fig 8)**

Question 2. What programs have been installed on the computer? What recent programs have been run?

To detect which programs were installed , I used the **RegRipper** tool with that has **uninstall** plugin, which loops through Keys in registry tracking all installed software in Control system. Following programs I found are **Microsoft Edge**, **Mozilla Firefox** , **7zip** and **VLC Media** and many more in **Fig 9**

```
$ rip.pl -r /mnt/windows_mount/Windows/System32/config/SOFTWARE -p uninstall
Launching uninstall v.20200525
uninstall v.20200525
(Software, NTUSER.DAT) Gets contents of Uninstall keys from Software, NTUSER.DAT hives

Uninstall
Microsoft\Windows\CurrentVersion\Uninstall

2024-08-27 13:12:17Z
  7-Zip 24.08 (x64) v.24.08

2024-08-27 13:07:35Z
  Mozilla Maintenance Service v.129.0.2

2024-08-27 13:05:11Z
  VMware Tools v.11.1.5.16724464

2024-08-27 13:04:44Z
  Microsoft Visual C++ 2019 X64 Additional Runtime - 14.24.28127 v.14.24.28127

2024-08-27 13:04:43Z
  Microsoft Visual C++ 2019 X64 Minimum Runtime - 14.24.28127 v.14.24.28127

2019-12-07 09:51:08Z
  WIC

2019-12-07 09:16:09Z
  AddressBook
  Connection Manager
  DirectDrawEx
  Fontcore
  IE40
  IE4Data
  IE5BAKEX
  IEData
  MobileOptionPack
  SchedulingAgent
```

```
Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall

2024-08-28 06:01:06Z
  Microsoft Edge v.92.0.902.67

2024-08-27 13:16:25Z
  VLC media player v.3.0.21

2024-08-27 13:07:35Z
  Mozilla Firefox (x86 en-US) v.129.0.2

2024-08-27 13:04:44Z
  Microsoft Visual C++ 2015-2019 Redistributable (x64) - 14.24.28127 v.14.24.28127.4

2024-08-27 13:04:41Z
  Microsoft Visual C++ 2019 X86 Minimum Runtime - 14.24.28127 v.14.24.28127
  Microsoft Visual C++ 2015-2019 Redistributable (x86) - 14.24.28127 v.14.24.28127.4
  Microsoft Visual C++ 2019 X86 Additional Runtime - 14.24.28127 v.14.24.28127

2023-05-05 12:32:49Z
  Microsoft Edge Update v.1.3.147.37

2019-12-07 09:51:08Z
  WIC

2019-12-07 09:16:08Z
  AddressBook
  Connection Manager
  DirectDrawEx
  Fontcore
  IE40
  IE4Data
  IE5BAKEX
  IEData
  MobileOptionPack
  SchedulingAgent
```

Figure 9 : List of all Installed Programs

For recent programs, I used RunMRU plugin to extract data as shown in Figure below. In figure 10, '**cmd**, **calc** '**notepad**,' and '**mspaint**' were the most recently run programs.

```
$ rip.pl -r /mnt/windows_mount/Users/Alex\ Marshall/NTUSER.DAT -p runmru
Launching runmru v.20200525
runmru v.20200525
(NTUSER.DAT) Gets contents of user's RunMRU key

RunMru
Software\Microsoft\Windows\CurrentVersion\Explorer\RunMRU
LastWrite Time 2024-08-27 13:13:05Z
MRUList = edcba
a   cmd\1
b   calc\1
c   notepad\1
d   mspaint\1
e   firefox\1
```

Question 3. Recover the content of any files in the recycle bin.

To recover the files, I navigated to **Recycle.Bin** and see following **5 files**. It has two deleted items, one is text file and other is zip file as shown in Figure 11

```
sansforensics@siftworkstation: /mnt/windows_mount/$Recycle.Bin/S-1-5-21-212117580-4225460857-3930097821-1000
$ ls
'$IAUO6YK.txt'  '$ILY0J7N.zip'  '$RAUO6YK.txt'  '$RLY0J7N.zip'   desktop.ini
```

Figure 11: Files in RecycleBin

To recover **$RAUO6YK.txt**, I use **file** command to shows its **archive** file, I unzip the file and got **notes.jpg** as shown in Figure 12 . I use built in tool **Libreoffice** to view the file. It was **Alex** Testimonial in Figure 13.

```
sansforensics@siftworkstation: ~/Downloads/EvidenceA
$ file "\$RAUO6YK.txt"
$RAUO6YK.txt: POSIX tar archive
sansforensics@siftworkstation: ~/Downloads/EvidenceA
$ unzip "\$RAUO6YK.txt"
Archive:  $RAUO6YK.txt
warning [$RAUO6YK.txt]:  1536 extra bytes at beginning or within zipfile
  (attempting to process anyway)
  inflating: notes.jpg
sansforensics@siftworkstation: ~/Downloads/EvidenceA
$ file notes.jpg
notes.jpg: Microsoft Word 2007+
```

This is my last will and testament.

If you're reading this, it means something has gone terribly wrong. I'm sorry for everything that's happened, for the pain I've caused. I want my parents to know that I love them, even though we didn't always see eye to eye. I wish I could have been a better son.

To Lily, I'm sorry for all the hurt I caused. You deserve so much better. I hope you can find it someday.

To Sophia… I don't know what to say. I never meant for things to end this way. I just wanted to find a way to be happy, but I made too many mistakes. I hope you can find peace.

All my belongings should go to my family, to help them with anything they need. I don't have much, but it's all I can offer.

Goodbye, everyone.

Alex Marshall

Figure 12,13: Recovering Files & Results

The zip file **'$RLYOJ&N.zip'** was password protected. I use **Fcrackzip** with **rockyou** dictionary to **crack** the password. The password found was "**football"**, I use the password to unzip the file as shown in Figure 14, The file name was **UniversityWarning_letter.pdf**

```
sansforensics@siftworkstation: ~/Downloads/EvidenceA
$ unzip '$RLY0J7N.zip'
Archive:  $RLY0J7N.zip
[$RLY0J7N.zip] UniversityWarning_Letter.pdf password:
   skipping: UniversityWarning_Letter.pdf  incorrect password
sansforensics@siftworkstation: ~/Downloads/EvidenceA
$ fcrackzip -u -v -D -p rockyou.txt '$RLY0J7N.zip'
found file 'UniversityWarning_Letter.pdf', (size cp/uc  63178/ 66769, flags 9, chk 7688)


PASSWORD FOUND!!!!: pw == football
sansforensics@siftworkstation: ~/Downloads/EvidenceA
$ unzip '$RLY0J7N.zip'
Archive:  $RLY0J7N.zip
[$RLY0J7N.zip] UniversityWarning_Letter.pdf password:
  inflating: UniversityWarning_Letter.pdf
```

Figure 14: Cracking zip File

PENNBROOK

March 15, 2024

To: Alex Marshall

Student ID: 734545

Department of Computer Science

Dear Mr. Marshall,

We are writing to inform you that your recent academic performance has raised significant concerns. Our records indicate that you have failed two courses in the previous semester and have missed several key assignments and exams in the current semester.

Please be advised that you are now on academic probation. If you do not demonstrate substantial improvement in your academic performance by the end of this term, you may face suspension from the university, and your scholarship may be revoked.

Additionally, we are aware of an incident last semester where you were suspected of unauthorized access to university systems. Although this case was closed due to insufficient evidence, any further infractions will be met with severe disciplinary action.

We strongly encourage you to seek academic counseling and make use of the resources available to help you succeed.

Sincerely,

Dr. Karen Thompson

Dean of Students

Pennbrook University

Figure 15 : Uni Warning letter

Question 4. Is there evidence that gives an indication of the state of mind of the owner of the computer?

During my investigation, I found multiple documents related to the owner , which helps me to deduce the state of mind of owner. Following are the documents given as

1. **Alex Dad** email, in which his dad was angry at him for doing poor in courses (Fig16) and ask him to get his life sorted out or otherwise its final warning from his side.

2. **Debit_Tracker** shows that alex was in debt with alot of money of **$7450** and there is big amount of $5000 dollar from **unknown** source causing stress to him (Fig 17)
3. **Uni Warning_letter** shows that Alex **failed** in two courses and giving a warning to go on probation and no continual of scholarship. It is also mention he is involved in suspicious activity but case close due to less evidence (Figure 15)
4. In **Journal1**, Alex mentions he is in big debt and under stress due to studies. He also feeling guilty due to selling exam answers and feeling scared if get caught. (Fig 18)
5. In **Journal2**, Alex mentions that his credit **card maxed out** and owe money to people, he is **unwilling** to share problem and situation with parents (Fig 20)
6. **Mum_Email** provides insights that alex isn't contact with her mom and she is worried about him and ask him to share his problem (Figure 19)
7. **InternshipOffer** letter shows that alex received internship but it needs good academic results to join, highlighting that it put more stress on Alex (Figure 21)
8. **Alex Testament,** the biggest clue, in which alex mention he's lifes is messed up due to poor decisions and saying sorry to his **parents**, **lily**, **sophia** and further mention that his belongings handed to his parents and overall goodbye to everyone (Fig 13)

From: Oliver Marshall <oliver.marshall@email.com>

To: Alex Marshall <alex.marshall@university.edu>

Date: March 10, 2024

Subject: Your Future

Alex,

I've just received your latest report card, and I'm deeply disappointed. This is not the level of performance I expect from my son. You have all the potential in the world, but you're throwing it away by not applying yourself.

I've warned you before that you're running out of chances. If you don't get your act together, I will have no choice but to cut off your financial support. You need to understand that the Marshall family has a reputation to uphold, and you're not living up to it.

You need to focus on your studies and start making decisions that reflect the kind of man you want to become. I didn't work my entire life for you to squander these opportunities.

Consider this your final warning.

Dad

Figure 16: Dad_Email_March.pdf

April 5, 2024

Dear Alex,

I hope you're doing well. I've been thinking about you a lot lately and wanted to write you a quick note to check in.

I know things have been tough with school, and your father can be hard on you. He just wants what's best for you, but I understand it can be overwhelming. I want you to know that I'm here for you, no matter what.

If you ever feel like you need someone to talk to, please don't hesitate to reach out. I'm worried about you, and I hope you're taking care of yourself. Life can be challenging, but you're strong, and I know you'll find your way.

Remember that we love you very much, and we're always here for you.

Love,
Mum

Figure 19: Mum_Email.jpg

A	B	C	D	E
Creditor	Amount	Owed	Due Date	Notes
Credit Card 1	1200	2024-04-15	Maxed out	
Credit Card 2	800	2024-04-20	Maxed out	
Lily Parker	300	2024-04-10	Personal loan, urgent	
Private Loan	5000	2024-04-30	From unknown source, urgent	
Friends	150	2024-04-25	Various small loans	
Total	7450			

journal1.txt

1 February 28, 2024
2
3 Money's been tighter than I expected this semester. Between rent, utilities, and trying to keep up appearances, I'm running out of cash faster than I thought. I've been looking for a job, but with the course load and everything else, I just don't have the time.
4
5 I started selling exam answers a few weeks ago—just a couple of assignments here and there. I know it's wrong, but I need the money. Every time I think about stopping, I remember how empty my bank account is. It's a risk, but what choice do I have?
6
7 The guilt is starting to get to me, though. I'm always looking over my shoulder, waiting for someone to figure it out. What if I get caught? What if I lose everything? But I can't back out now. I'm in too deep.

Figure 17,18: Debit_Tracker,Journal1.txt

journal2.txt

1 April 5, 2024
2
3 I'm drowning in debt. The credit cards are maxed out, and I still owe Lily for that loan she gave me last month. I hate asking for help, especially from her, but I didn't have a choice. I don't know how I'm going to pay her back.
4
5 Dad's been on my case again, telling me I need to get my life together. Easy for him to say—he's not the one trying to balance classes, relationships, and this mess of a life. He doesn't understand the pressure I'm under, and I can't tell him about the money problems. He'd lose it.
6
7 I've thought about dropping out, but where would I go? What would I do? I can't let everyone down, but I don't know how much longer I can keep this up.

Figure 20: Journal2.txt

Dear Alex,

I am pleased to offer you an internship position with TechVision, a leading company in the field of artificial intelligence and cybersecurity. This opportunity is extended to you based on your demonstrated skills and potential in computer science.

The internship is a highly competitive program that could be a pivotal step in your career. However, I must stress the importance of staying focused and avoiding any distractions that may impede your progress.

This internship will require full commitment and excellence in both your academic and professional pursuits. I trust you will take this opportunity seriously and make the most of it.

Please confirm your acceptance of this offer by April 10, 2024.

Best regards,
Professor Mark Reynolds
Department of Computer Science
Pennbrook University

Figure 21: Internship_letter

Evidence B.

EvidenceB was provided in the zip file, after extracting I got .pcap file. I use Wireshark tool to monitor the network traffic and do the following questions

Question 5. Who are the people communicating in the transmission? When does the first transmission begin and the last transmission finish?

To get data, I follow the tcp stream by clicking on packet selecting option follow > TCP stream as shown in the Figure 22

I get the following data from tcp streams **0**, **3**, **272**, **371** as shown below in the following images below

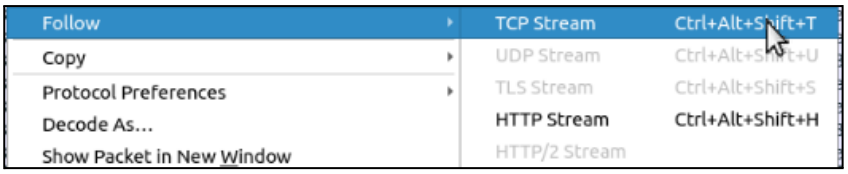


Figure 22 :Following Tcp stream

The people who were communicating in the transmission are **AlexM21**, **PartyDude**, **DormKing**, **BookWorm**, **LilyLaw** and **ArtLover99** as shown in Figure 23, 24 and 25.

To get the starting and ending time of Transmission I check the first and last packet so it begins at **Sep 1, 2024 07:28:33 UTC** and ended at **Sep 1, 2024 08:51:43 UTC**
And **conversation** started at time **07:57:11** and ended at **08:46:20**

Figure 23 : Stream 272, output

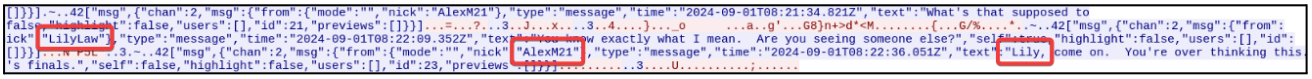


Figure 24: stream 371 output

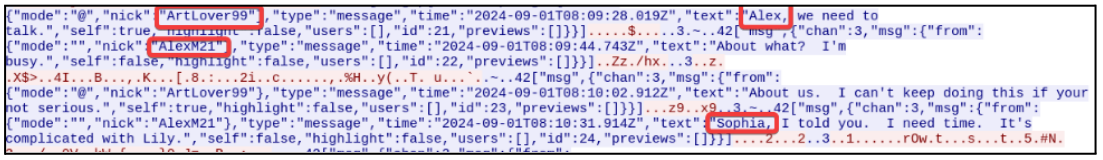


Figure 25: stream 0 output

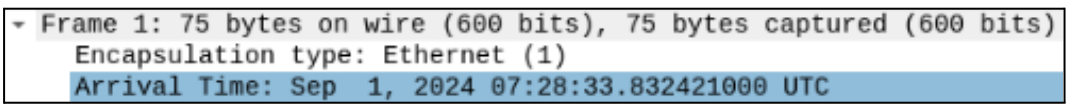


Figure 26: Starting of Transmission

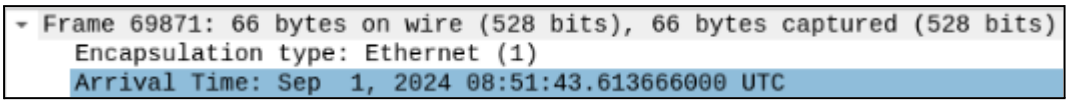


Figure 27: Ending of Transmission

6. What browsers, operating systems, and IP addresses are used by the communication endpoints?

To find the IP addresses, I use the tool **Network Miner** to extract the IP address and get more info about the operating system as shown in figures below. By following TCP stream, I found **User-Agent** header in stream 4 which shows **Mozilla/5.0**. I further confirm from **user-agents.net** and get the following data about **browser** and **operating system** in Figure 28 which is Firefox/129.0 and Ubuntu

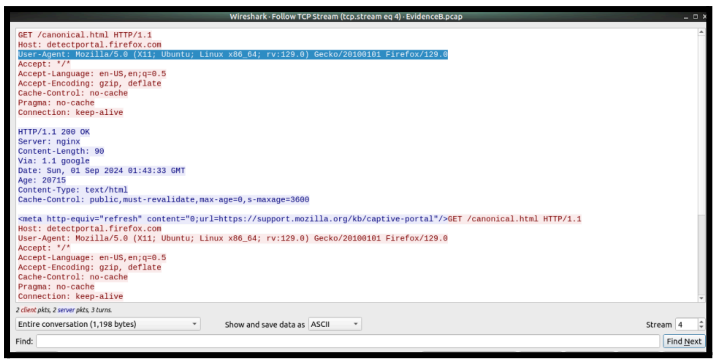


Figure 28: User Agent Found in Stream 4

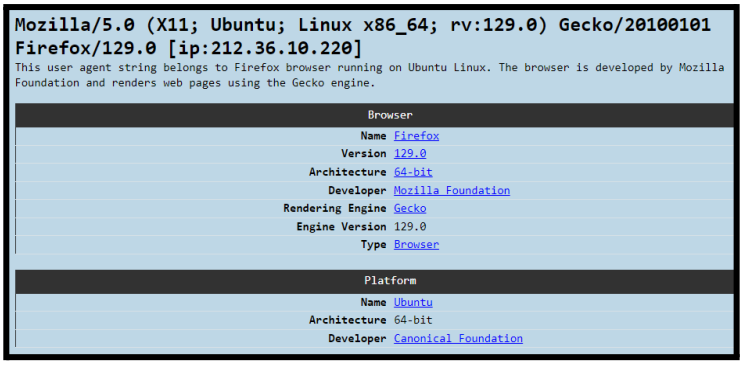


Figure 29 : Confirming User agent from Source

Following IP address found through network miner are given in Table 1 below

10.10.10.1(Firefox/129.0)	10.10.10.22(Firefox/129.0)
10.10.10.33(Firefox/129.0)	10.10.10.44(Firefox/15.0.1)
10.10.10.56(Firefox/129.0)	10.10.10.254

Table 1 : IP Address

For confirmation IP's addressed displayed in the miner tool is given as

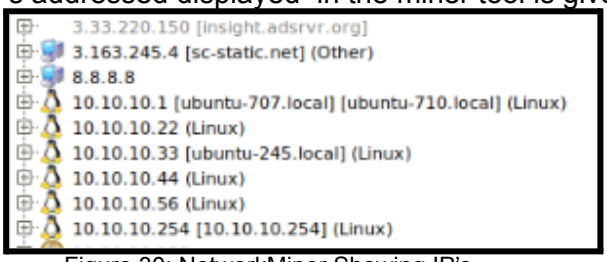


Figure 30: NetworkMiner Showing IP's

Question 7. What files were transmitted on the local network?

I filter packets using ftp and found out that sslkey file is shared by FTP in Fig 31. I used ftp-data filter to get SSL Key and recovered it in raw form. Then I choose option Edit > preferences and add the raw key to TLS protocol. This decrypt the TLS conversations and as result I was able to get the hhtp2 protocol and I was able to extract more valuable data as shown below.

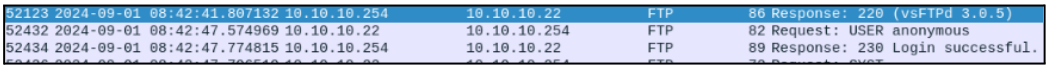


Figure 31 : FTP Data Transfer Found

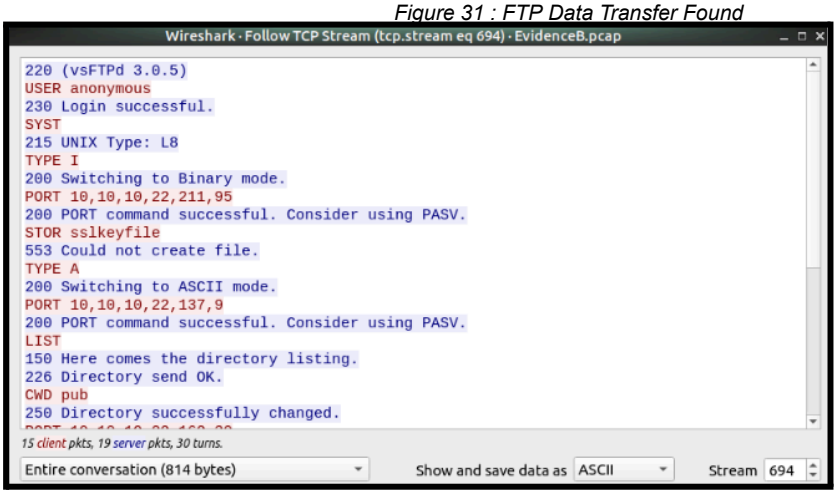
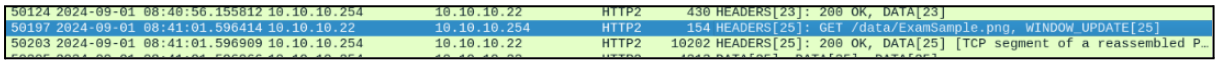


Figure 32 : Sign of SSLKeyFile Transfer in stream 694

```
# SSL/TLS secrets log file, generated by NSS
CLIENT_HANDSHAKE_TRAFFIC_SECRET 7eb7aa304379b70109738e04a08b548856f1efaf0d245966fb4dd24f8d9d7d42 10b7a548166d4d0030ef819588efee47d85e4fbc1b307670bbdefe28d8aea790
SERVER_HANDSHAKE_TRAFFIC_SECRET 7eb7aa304379b70109738e04a08b548856f1efaf0d245966fb4dd24f8d9d7d42 2b4c0eecdc27e54df3c8159034d2f13a76e50caf2864bc209e1e79c81d9348d4
CLIENT_TRAFFIC_SECRET_0 7eb7aa304379b70109738e04a08b548856f1efaf0d245966fb4dd24f8d9d7d42 50d9e26b5d9d51c42d51f13707674fc14c8ecb342b64885530ec8ad1307ad5d4
SERVER_TRAFFIC_SECRET_0 7eb7aa304379b70109738e04a08b548856f1efaf0d245966fb4dd24f8d9d7d42 5292a58dafe774317fcd05d6d64f286d1593bfc39803221ecff844f47f3d14ce
EXPORTER_SECRET 7eb7aa304379b70109738e04a08b548856f1efaf0d245966fb4dd24f8d9d7d42 552cfd619a3ddd6213374f8329c65b90fcfee98155c5687f7a2c9195e2c31c7c
CLIENT_HANDSHAKE_TRAFFIC_SECRET db75f172d0879ab8d42c7efae0e3b40091560f7be596022ab448eac447970500 818f2273e0c5bfcbf2b167c4b0fff8b52c9f34d3666cc17c26d53cbba5c2c97e
SERVER_HANDSHAKE_TRAFFIC_SECRET db75f172d0879ab8d42c7efae0e3b40091560f7be596022ab448eac447970500 80a8110cdee88f02a8b8eaa732a7e35cc69160a10ebffeee11f824294cbbc2db
CLIENT_TRAFFIC_SECRET_0 db75f172d0879ab8d42c7efae0e3b40091560f7be596022ab448eac447970500 d823923398b872af8fbcefe4699d29f6b26bb2e106cfb69452417715bf56fa04
SERVER_TRAFFIC_SECRET_0 db75f172d0879ab8d42c7efae0e3b40091560f7be596022ab448eac447970500 39983d3505c846a1a05118e094b455d61f92c6e08ef77009a46d0694dfb7a037
EXPORTER_SECRET db75f172d0879ab8d42c7efae0e3b40091560f7be596022ab448eac447970500 cfdacc43bf6248e6a508a12e6b8e6f226b434eeac6d6f79742d89f72c1582bb6
CLIENT_RANDOM 38016bc94ae0826412f45f2a7e0d826697cf1a8e396df56fb4634e9534bfcff5 201ccdb82cb83a8cba3e8cd8323055d135a20b68c1107f78f6610821eb10ca506669c3b0b5199a29be5b5412a7386460
CLIENT_RANDOM 72d3bab0ca2aedac7cd4b842b1d3d0410a0e36428c7823e590e80e3c9e747644 f83e24701a8f89ee658104ed422c878565e24b8e12b857046e044e1a99d29296b688cf6bfde704cdc930cc749cf10da5
CLIENT_HANDSHAKE_TRAFFIC_SECRET 87bdc31f6fca9c3618ad60038ae29462e02f35613624745ca0a3e1ffc65fd789 5ec3b1cd3a472a21cce88f1d59b594d01bbca8a13d30680c341d9b26f01d5682
SERVER_HANDSHAKE_TRAFFIC_SECRET 87bdc31f6fca9c3618ad60038ae29462e02f35613624745ca0a3e1ffc65fd789 8214a82c8000acdcac3c223995b3c879d78095167c5505dcf61d444c2152a7dc
CLIENT_HANDSHAKE_TRAFFIC_SECRET bebbf13481fa86e2c7842648a0cf05f3ff257eac015125d2c044b4bee17a6870 754413ac10cef125c1a3ddf4b90f3630df497fde0709b79ef84d9104b359e995
SERVER_HANDSHAKE_TRAFFIC_SECRET bebbf13481fa86e2c7842648a0cf05f3ff257eac015125d2c044b4bee17a6870 12c6ef79f260c8a87eaf13e08a80ec0042af324b881dda060c4998e0d4bb3321
CLIENT_TRAFFIC_SECRET_0 87bdc31f6fca9c3618ad60038ae29462e02f35613624745ca0a3e1ffc65fd789 117fa0ce5c14fd43867ead620b62513b6bc144c52692d178656f8e1ce260e965
SERVER_TRAFFIC_SECRET_0 87bdc31f6fca9c3618ad60038ae29462e02f35613624745ca0a3e1ffc65fd789 f1285e29eeea294df91109dea79028a8e83dc6b652a5be10d61dacb86214bd53
EXPORTER_SECRET 87bdc31f6fca9c3618ad60038ae29462e02f35613624745ca0a3e1ffc65fd789 fea2139978e1111b44272ff7ea7df4206a72c3038e591bd2fa7570fc52375fe7
```

Figure 33 : SSL/TLS log file in Stream 701

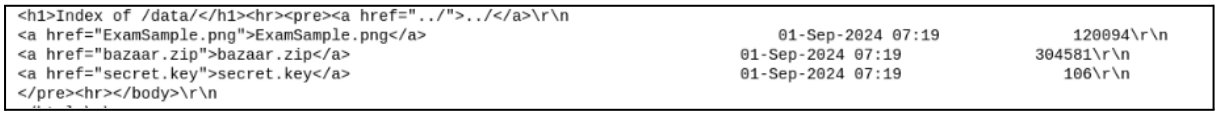
After gaining access to http2 protocol, I gained new insights as shown in figure 34



```
50124 2024-09-01 08:40:56.155812 10.10.10.254   10.10.10.22    HTTP2   430 HEADERS[23]: 200 OK, DATA[23]
50197 2024-09-01 08:41:01.596414 10.10.10.22    10.10.10.254   HTTP2   154 HEADERS[25]: GET /data/ExamSample.png, WINDOW_UPDATE[25]
50203 2024-09-01 08:41:01.596909 10.10.10.254   10.10.10.22    HTTP2   10202 HEADERS[25]: 200 OK, DATA[25] [TCP segment of a reassembled P…
```

Figure 34: Access to HTTP2 protocol

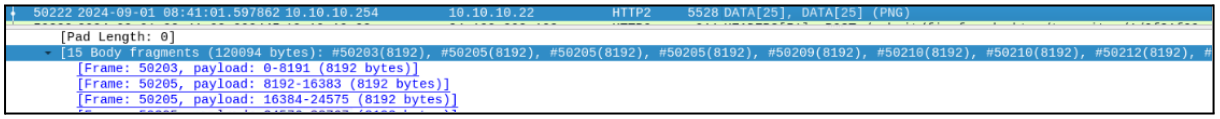
Further checking the packets details and bytes I found that following files has been shared on this stream which include **ExamSample.png, bazaar.zip and secret.key** as shown in the figure. Which shows these files can be extracted out .



```
<h1>Index of /data/</h1><hr><pre><a href="../">../</a>\r\n
<a href="ExamSample.png">ExamSample.png</a>                    01-Sep-2024 07:19          120094\r\n
<a href="bazaar.zip">bazaar.zip</a>                        01-Sep-2024 07:19          304581\r\n
<a href="secret.key">secret.key</a>                        01-Sep-2024 07:19             106\r\n
</pre><hr></body>\r\n
```

Figure 35 : Packet bytes shows Sharing of files

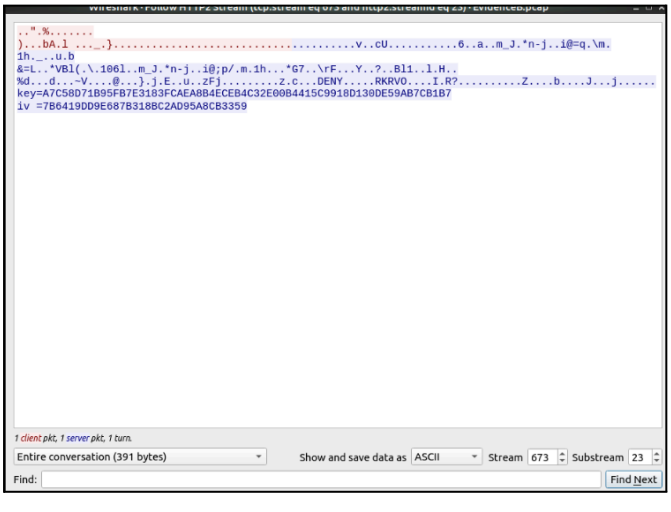
After that I looked ExamSample.png by searching png in field and got the following PNG stream. I investigated further by checking fragments bytes and found the Exam image



```
50222 2024-09-01 08:41:01.597862 10.10.10.254   10.10.10.22    HTTP2   5528 DATA[25], DATA[25] (PNG)
    [Pad Length: 0]
  [15 Body fragments (120094 bytes): #50203(8192), #50205(8192), #50205(8192), #50205(8192), #50209(8192), #50210(8192), #50210(8192), #50212(8192), #
    [Frame: 50203, payload: 0-8191 (8192 bytes)]
    [Frame: 50205, payload: 8192-16383 (8192 bytes)]
    [Frame: 50205, payload: 16384-24575 (8192 bytes)]
```

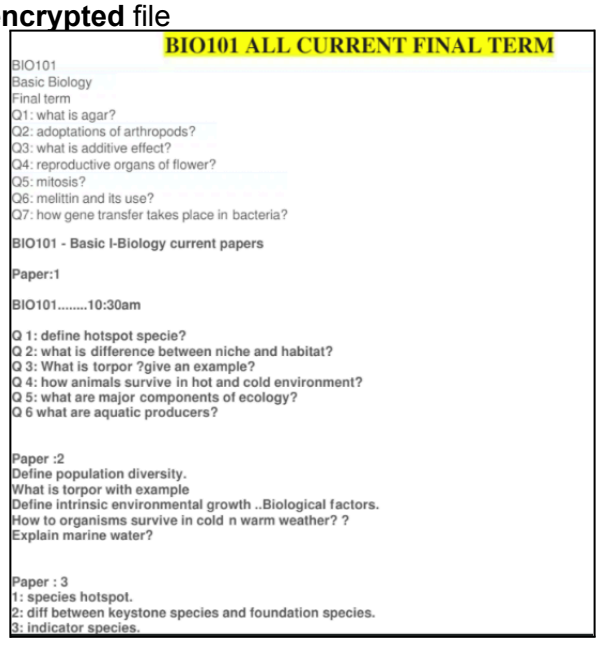
Figure 36 :ExamSample.png Fragments Found

I use the option show packet bytes and it display Bio101 exam image as shown below. I also further extracted Secret key which hints there is **aes encrypted** file



```
..".%.....
)...bA.1 ....}.....v..cU..........6..a..m_J.*n-j..i@=q.\m.
1h._..u.b
&=L..*VBl(.\.106l..m_J.*n-j..i@;p/.m.1h...*G7..\rF...Y..?..Bl1..l.H..
%d...d...~V...@...}.j.E..u..zFj.........z.c...DENY.....RKRVO....I.R?..........Z....b....J...j.....
key=A7C58D71B95FB7E3183FCAEA8B4ECEB4C32E00B44415C9918D130DE59AB7CB1B7
iv =7B6419DD9E687B318BC2AD95A8CB3359

1 client pkt, 1 server pkt, 1 turn.
Entire conversation (391 bytes)   Show and save data as ASCII   Stream 673   Substream 23
Find:                                                          Find Next
```



BIO101 ALL CURRENT FINAL TERM

BIO101
Basic Biology
Final term
Q1: what is agar?
Q2: adoptations of arthropods?
Q3: what is additive effect?
Q4: reproductive organs of flower?
Q5: mitosis?
Q6: melittin and its use?
Q7: how gene transfer takes place in bacteria?

BIO101 - Basic I-Biology current papers

Paper:1

BIO101.....10:30am

Q 1: define hotspot specie?
Q 2: what is difference between niche and habitat?
Q 3: What is torpor ?give an example?
Q 4: how animals survive in hot and cold environment?
Q 5: what are major components of ecology?
Q 6 what are aquatic producers?

Paper :2
Define population diversity.
What is torpor with example
Define intrinsic environmental growth ..Biological factors.
How to organisms survive in cold n warm weather? ?
Explain marine water?

Paper : 3
1: species hotspot.
2: diff between keystone species and foundation species.
3: indicator species.

Figure 37: Secret Key

Figure 38 : BIO101 Exam

I further investigate more and follow tcp streams, in which stream **699** display few files that were present in the directory. These files were shared previously in **2021** present in **pub directory** highlighting alex shared academic documents before with these guys. Files names are **test.txt** and **upload.txt**

```
-rw-rw-r--    1 ftp      ftp             5 Aug 31  2021 test.txt
-rw-r--r--    1 ftp      ftp             5 Aug 31  2021 upload.txt
```

Figure 39 : Files present in Public Directory

All above results shows that the files I have recovered and shared on local network are **ExamSample.png** and **secret.key**

Question 8. What is the relationship of the people communicating in the network capture? How are they related to the victim?

According to above Data, 6 people were found in the communicating(Fig 23,24,25). Data shows that these people were friends with victim(Alex) and shared same university studies

Following are the people involved in transmission.

DormKing(Figure 23): He is involved in group **Dorm** to get the exam answer and was not willing to pay $200 to victim. Also instructed PartyDude to get the sessionkey file from victim system.

PartyDude(Figure 23): He is also involved in **Dorm** in getting exam answers and heavily dependent on victim awnsers insights. He tried to steal the session key from victim room, so he couldn't pay for the Bioexam insights.

BookWorm(Figure 23) : He is also part of the group Dorm and warned others about the integrity theft of the exam and mention he would not be a part of it as he was scared but observe the situation as spectator.

LilyLaw(Figure 24): Lily is girlfriend of victim and she is quite angry on situation and suspecting that victim is seeing someone else and warned him to sort this out.

ArtLover(Figure 25): Artlover is Sophia, who is in a romantic relationship with the victim and asking him to leave Lily and choose between one of them as soon as possible.

All of the above data highlights how the conversation and scenario play out the friends involved.

Evidence C.

The Evidence C was provided in **.vmem** format, to examine **RAM** memory dump I use **Volatility** tool, which is use in forensic to extract important info from these memory dump. To start the Process I use **imageinfo** plugin to identify the windows profiles which is **Win7SP1x64** and it also shows timestamps in Figure 40.

Figure 40 : Memory Dump of Profile

Question 9. What applications are running on the memory dump computer?

To know which applications are running I use **plugin pstree** which shows the hierarchical and parent-child view of the process as shown in Figure 41. It lists all the User Applications which are **Windows Explorer**, **Mozilla Firefox**, **Thunderbird**, **Notepad++** and other applications and services can be seen in the figure below.

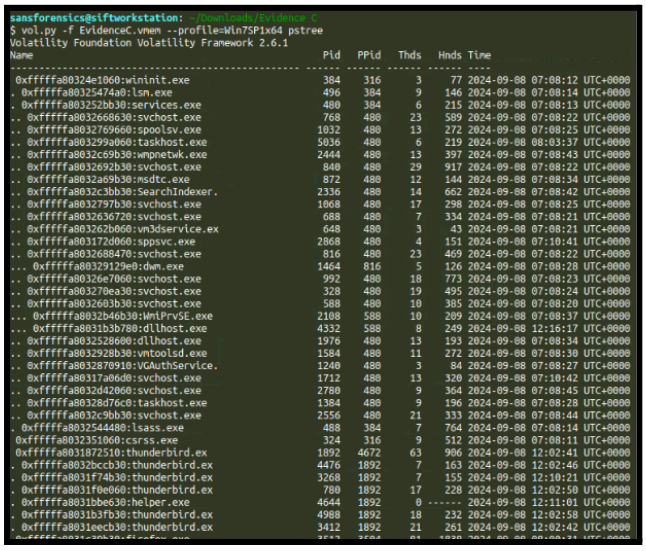
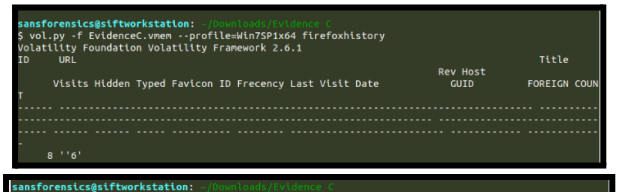
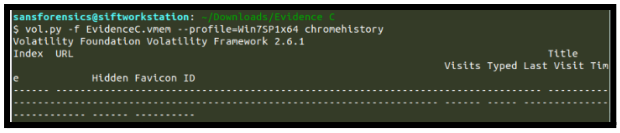


Figure 41: List of Running Applications & Services

10. What web pages has the memory dump computer visited recently?

To check which web pages were visited, I started my investigation by using the plugin **firefoxhistory** and **chromehistory** to check recently visited pages. I didn't get any data from it as shown in Fig 42. Then I used the plugin **iehistory**, which extracted the data but it was unfiltered and not useful for investigation as shown in Fig 43.





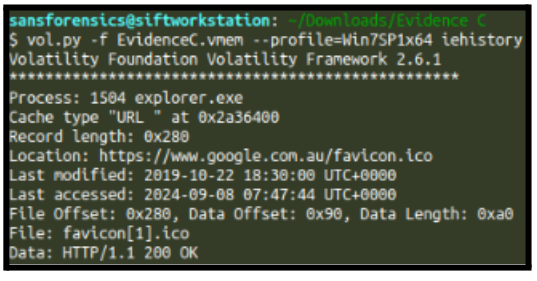


Figure 42 : Result of firefoxhistory & chromehistory

Figure 43 : Result of iehistory

To investigate further I do memory dump of **firefox.exe** files by grabbing the PId **3512** and using the following command as shown in Fig 44

```
sansforensics@siftworkstation: ~/Downloads/Evidence C
$ vol.py -f EvidenceC.vmem --profile=Win7SP1x64 dumpfiles -n -p 3512 -D FirefoxData/
Volatility Foundation Volatility Framework 2.6.1
DataSectionObject 0xfffffa8031c50a20   3512   \Device\HarddiskVolume1\Windows\Registration\R000000000006.clb
DataSectionObject 0xfffffa8031b30b70   3512   \Device\HarddiskVolume1\Users\Sophia Bennett\AppData\Roaming\Mozilla\Firefox\Profiles\v2333frx.default-esr\storage\default
ps%2Cquora.com%29\ls\data.sqlite
DataSectionObject 0xfffffa8032c5ae20   3512   \Device\HarddiskVolume1\Windows\System32\en-US\MMDevAPI.dll.mui
DataSectionObject 0xfffffa8031c257a0   3512   \Device\HarddiskVolume1\Users\Sophia Bennett\AppData\Roaming\Mozilla\Firefox\Profiles\v2333frx.default-esr\storage.sqlite
DataSectionObject 0xfffffa8031c8d320   3512   \Device\HarddiskVolume1\Users\Sophia Bennett\AppData\Roaming\Mozilla\Firefox\Profiles\v2333frx.default-esr\cert9.db
DataSectionObject 0xfffffa8031c59d60   3512   \Device\HarddiskVolume1\Users\Sophia Bennett\AppData\Roaming\Mozilla\Firefox\Profiles\v2333frx.default-esr\key4.db
```

Figure 44 : Firefox memory Dump

Further, I found a file name file **formhistory.sqlite.dat,** opened through **sqlite** database software and found the result in **formhistory** database which shows search **keywords**

id	fieldname	value	timesUsed	firstUsed	lastUsed	guid
Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	searchbar-history	download thunderbird 115	1	172578246...	172578246...	NorhaZ4LRoGMGdBB
2	searchbar-history	How to stop someone from leaving you	1	172579828...	172579828...	8xMxmGvJS4inUXsN
3	searchbar-history	Signs someone is cheating	1	172579829...	172579829...	8JjrLyERRQ69WUMn
4	searchbar-history	How to get away with self-defense	1	172579829...	172579829...	uk5POvm7SNSE2LX5
5	searchbar-history	Emergency therapy services	1	172579831...	172579831...	CmT5RtqRTHG1w2gY

Figure 45: Search History Found in Database

After this I found **places.sqlite.dat** file, which is valuable as it holds data about the website visited plus bookmarks. After running the command **Strings filename | grep** https give me the following result in Fig 46. The web pages searches data from file confirm and matches with formhistory database which shows the recently visited pages

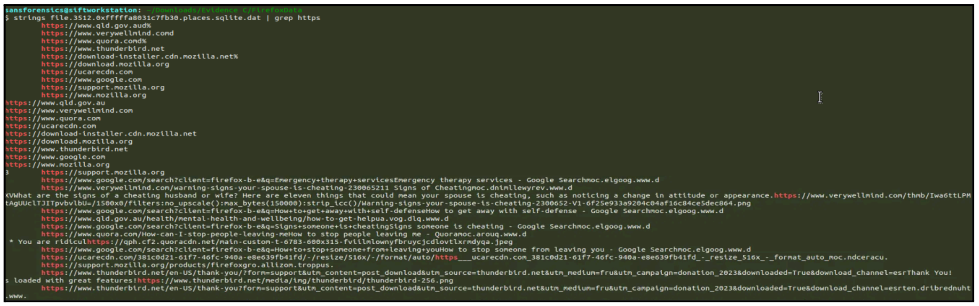


Figure 46: WebPage Searches From places.sqlite.dat

By all the above evidence these are webpages url and keyword history give in Fig 45,46

1. https://www.quora.com/How-can-I-stop-people-leaving-me
2. https://www.google.com/search?client=firefox-b-e&q=Emergency+therapy+services
3. https://download-installer.cdn.mozilla.net/pub/thunderbird/releases/115.15.0/win64/en-US/Thunderbird%20Setup%20115.15.0.exe
4. https://www.verywellmind.com/warning-signs-your-spouse-is-cheating-2300652
5. https://www.google.com/search?client=firefox-b-e&q=How+to+get+away+with+self-defense

Question 11. What is email address of the owner of the memory dump computer and how are they connected to the case?

To get email address, I use Volatility plugin **dumpregistry** to dump registries and hivelist by command in Figure 47. Further it was analysed by **RegRipper** to extract important data.
I saw a file name **registry.0xfffff8a000b98010.SOFTWARE.reg** and use plugin **lastloggedon** to extract owner name, which is **Sophia Bennett** as shown in Figure 48

```
sansforensics@siftworkstation: ~/Downloads/Evidence C
$ vol.py -f EvidenceC.vmem --profile=Win7SP1x64 dumpregistry -D SystemReg/
Volatility Foundation Volatility Framework 2.6.1
**************************************************
Writing out registry: registry.0xfffff8a000b80410.BCD.reg

**************************************************
**************************************************
Writing out registry: registry.0xfffff8a000024010.SYSTEM.reg

**************************************************
**************************************************
Writing out registry: registry.0xfffff8a000061010.HARDWARE.reg

**************************************************
**************************************************
Writing out registry: registry.0xfffff8a000dd7010.SECURITY.reg
```

Figure 47 : Registry Dump

```
sansforensics@siftworkstation: ~/Downloads/Evidence C/SystemReg
$ rip.pl -r registry.0xfffff8a000b98010.SOFTWARE.reg -p lastloggedon
Launching lastloggedon v.20200517
lastloggedon v.20200517
(Software) Gets LastLoggedOn* values from LogonUI key

LastLoggedOn
Microsoft\Windows\CurrentVersion\Authentication\LogonUI
LastWrite: 2024-09-08 07:08:26Z

LastLoggedOnUser    = .\Sophia Bennett
LastLoggedOnSAMUser = WIN-H3NARMQ1QIU\Sophia Bennett
```

Figure 48 : Extracting Owner Name

To further confirm the owner of the laptop I use the **Samparse plugin** on file **registry.0xfffff8a000e43010.SAM.reg**, which contains account & security information. I use grep Username to extract the owner name as shown in Figure 49

```
sansforensics@siftworkstation: ~/Downloads/Evidence C/SystemReg
$ rip.pl -r registry.0xfffff8a000e43010.SAM.reg -p samparse | grep Username
Launching samparse v.20200825
Username        : Administrator [500]
Username        : Guest [501]
Username        : Sophia Bennett [1000]
Username        : HomeGroupUser$ [1002]
```

Figure 49: Extracting Username From Sam registry

To confirm the email address of the owner I the following Strings commands in figure 50,52 to extract the data about the emails and how its related to victim.

```
sansforensics@siftworkstation: ~/Downloads/Evidence C
$ strings  EvidenceC.vmem | grep "gmail.com"
```

Figure 50 : Grabbing All Mails

```
To: sr8640171@gmail.com
From: Sophia Bennett <vb9945311@gmail.com>
Subject: Missed You Today
Content-Type: text/plain; charset=UTF-8; format=flowed
Content-Transfer-Encoding: 8bit
Hey Alex,
```

Figure 51: Owner Email Extracted

By using grep, I found owner email address which is **vb9945311@gmail.com** (Figure 51)
Further running grep Alex command show me Alex email which is sr8640171@gmail.com (Fig 52) and It also shows more data which is conversation between Sophia and Alex.

```
vb9945311@gmail.com
sr8640171@gmail.com
To: sr8640171@gmail.com
From: Sophia Bennett <vb9945311@gmail.com>
1@gmail.com;
```

Figure 52 :Alex Email Detected

```
Subject: You and Me
Content-Type: text/plain; charset=UTF-8; format=flowed
Content-Transfer-Encoding: 8bit
Hi Alex,
It felt so good to see you the other day. I could tell you
re feeling
torn, and I get it
I really do. But you know you don
```

Figure 53: Subject You and Me

```
Subject: Missed You Today
Content-Type: text/plain; charset=UTF-8; format=flowed
Content-Transfer-Encoding: 8bit
Hey Alex,
I missed you in class today. I kept looking over at your usual spot,
hoping you
d walk in with that goofy smile of yours. It
s strange how
^C
```

Figure 54: Subject Missed You Today

```
Subject: Feeling Overwhelmed
To: vb9945311@gmail.com
Content-Type: multipart/alternative; boundary="000000000000f7b5bd06219aa924
--000000000000f7b5bd06219aa924
Content-Type: text/plain; charset="UTF-8"
Content-Transfer-Encoding: quoted-printable
Hey Soph,
Thanks for checking in. I=E2=80=99ve been feeling a bit overwhelmed with
```

Figure 55: Subject Feeling Overwhelmed

Following emails with subject **You and Me**, **Missed You Today** and **Feeling Overwhelmed** shows that **owner(Sophia)** connected with case chatting with Alex and Alex chatting about his situation with owner. From emails it is deduced that owner loves Alex and want to help him as they are good friends in Figure 53, 54, 55.

Question 12. What is password of the memory dump computer?

To get the password of dump system, I use volatility plugin **lsadump** as it can extract info about **security logs** including passwords and authentications. Result is shown in Fig 56. Password Extracted is **'Alexisbeautiful'.** I get the same result by using **hashdump** (Fig 57)

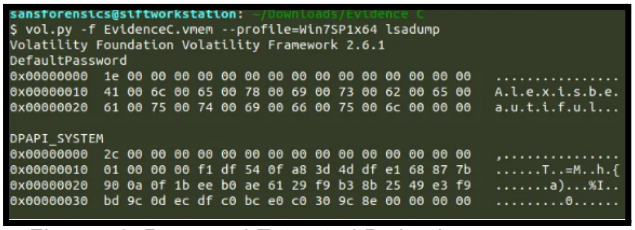


Figure 56: Password Extracted By lsadump

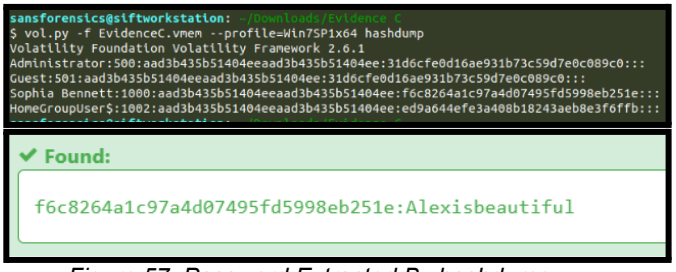


Figure 57: Password Extracted By hashdump

And when I decode the hash to get the same password

Evidence D.

When I extracted EvidenceD I get 10 disk files. By using **ls -la** I get info about disk size and started my investigation from largest file which were **dm-1** and **vdc.** As shown in Figure 58. Further I run **File** & **mmls** command to check the details about the image. Dm-1 is linux file with no partitions and vdc is data file with no partitions in figure 59

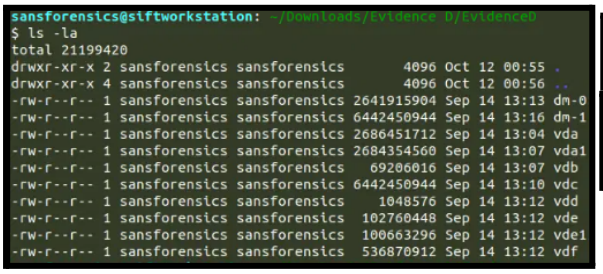


Figure 58 : List of Disks & Size

Figure 59: Details about Disk & Partitions

From above detail it can be deduce that **vdc** is corrupted version of **dm-1** as they have the same size to the **bits** which is **rare**, Now applying this info , I mounted the dm-1 without **offset** and **readonly** settings to avoid data tampering by following command **(Figure 60)**

Figure 60 : Mounting dm-1

Question 13. What are the non-stock applications installed on the phone?

I found the non-stock applications in app directory and found 5 apps as shown(Figure 61)

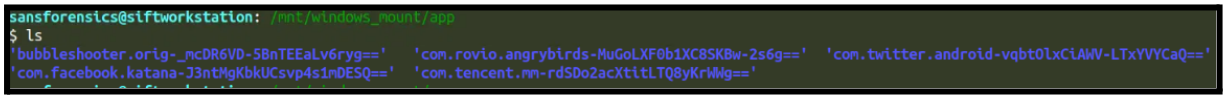


Figure 61: List of non-stock Applications

I identify the app as in android phones apps name start with .com. The non stock-apps are **facebook**, **bubbleshooter**, **angrybirds** , **twitter** , **tencent** as **Wechat.**

Question 14. Who is in the contacts list? What messages and calls have been sent and received by the phone?

To get contact list, I search for all logs and databases by using the command Find. After getting the list I found important databases named **calllogs.db**, **contact.db**, **contact2.db** and **mmssms.db**. To examine these files, I gain **root** access(Andriod) to copy these files and to other folder and then change **ownership(sansforensic)** to open these files by **sqlite software** as shown in figure 62 and resetting ownership using **chown** in Figure 63

```
$ sudo su
root@siftworkstation:/mnt/windows_mount#
root@siftworkstation:/mnt/windows_mount# cp /mnt/windows_mount/data/com.android.providers.contacts/databases/contacts2.db /home/sansforensics/Downloads/EvidenceD
root@siftworkstation:/mnt/windows_mount# cp /mnt/windows_mount/data/com.android.providers.contacts/databases/calllog.db /home/sansforensics/Downloads/EvidenceD
root@siftworkstation:/mnt/windows_mount# cp /mnt/windows_mount/data/com.android.providers.telephony/databases/mmssms.db /home/sansforensics/Downloads/EvidenceD
```

Figure 62: Copying Database Files as Root User

```
sansforensics@siftworkstation: ~/Downloads/Data
$ sudo chown sansforensics:sansforensics contacts2.db
sansforensics@siftworkstation: ~/Downloads/Data
$ sudo chown sansforensics:sansforensics calllog.db
sansforensics@siftworkstation: ~/Downloads/Data
$ sudo chown sansforensics:sansforensics mmssms.db
```

Figure 63: Setting privileges

In calllogs.db and in table **calls**, I found that alex received **5 calls** in which **4 declined** calls from **lily** and one **70** seconds call from **Sophie** as showin in Figure 64.

Table: calls

_id	number	duration	type	subscription_id	name	numbertype	numberlabel
1	+61449857236	0	5	890141032...	Lily Parker	2	NULL
2	+61449857236	0	5	890141032...	Lily Parker	2	NULL
3	+61449857236	0	5	890141032...	Lily Parker	2	NULL
4	+61449857236	0	5	890141032...	Lily Parker	2	NULL
5	+61417692485	70	2	890141032...	Sophie ...	2	NULL

Figure 64 : Calls Received & Declined in calllogs.db database.

Further, in contacts2.db I found 4 contacts(Fig 65) with following names and phone numbers

1 Lily Parker +61 449 857 236 **2 Sophia Bennett +61 417 692 485**

3 Mum +61 438 179 564 **4 Dad +61 467 315 782**

Database Structure | Browse Data | Edit Pragmas | Execute SQL

Table: data

_id	mimetype_id	raw_contact_id	is_read_only	data_version	data1	data2	data3	data4	data5	data6
16	11	4	0	1	1	NULL	NULL	NULL	NULL	NULL
18	5	4	0	1	+61 449 857 236	2	NULL	+61449857...	NULL	NULL
19	7	4	0	2	Lily Parker	Lily	Parker	NULL	NULL	NULL
20	3	4	0	0	NULL	1	NULL	NULL	NULL	NULL
21	12	4	0	0	NULL	NULL	NULL	NULL	NULL	NULL
22	11	5	0	1	1	NULL	NULL	NULL	NULL	NULL
24	5	5	0	1	+61 417 692 485	2	NULL	+61417692...	NULL	NULL
25	7	5	0	2	Sophie Bennett	Sophie	Bennett	NULL	NULL	NULL
26	11	6	0	1	1	NULL	NULL	NULL	NULL	NULL
28	5	6	0	1	+61 438 179 564	2	NULL	+61438179...	NULL	NULL
29	7	6	0	2	Mum	Mum	NULL	NULL	NULL	NULL
30	3	5	0	0	NULL	1	NULL	NULL	NULL	NULL
31	12	5	0	0	NULL	NULL	NULL	NULL	NULL	NULL
32	3	6	0	0	NULL	1	NULL	NULL	NULL	NULL
33	12	6	0	0	NULL	NULL	NULL	NULL	NULL	NULL
34	11	7	0	1	1	NULL	NULL	NULL	NULL	NULL
36	5	7	0	1	+61 467 315 782	2	NULL	+61467315...	NULL	NULL
37	7	7	0	2	Dad	Dad	NULL	NULL	NULL	NULL
38	3	7	0	0	NULL	1	NULL	NULL	NULL	NULL
39	12	7	0	0	NULL	NULL	NULL	NULL	NULL	NULL

Figure 65 : Contact list in Contact2.db

Continuing my investigation, I examine **mmssms.db** and found the following conversation

_id	ead	address	ersc	date	date_sent	otoc	eac	tatu	ype	ath_	ibje	body
F...	F...	Filter	F...	Filter	Filter	F...	F...	F...	F...	F...	F...	Filter
1	3	+61449857236	4	1726318296...	1726318296000	0	1	-1	1	0	...	Hey, can we talk tonight? We really need to sort this out.
2	3	+61449857236	...	1726318315...	0	...	1	-1	2	...	...	Not tonight, Lily. I'm dealing with a lot right now.
3	3	+61449857236	4	1726318328...	1726318329000	0	1	-1	1	0	...	Alex, this is important. I can't keep going on like this, wondering where we stand.
4	3	+61449857236	...	1726318345...	0	...	1	-1	2	...	...	I know, I know. But I just can't tonight. Tomorrow?
5	3	+61449857236	4	1726318358...	1726318358000	0	1	-1	1	0	...	Tomorrow might be too late, Alex. If you don't end this with her, I will.
6	3	+61449857236	...	1726318370...	0	...	1	-1	2	...	...	Lily, please don't do anything rash. I'll figure it out, I promise.
7	3	+61449857236	4	1726318383...	1726318384000	0	1	-1	1	0	...	You better. I'm done waiting.

Figure 66 : Conversation Between Lily and Alex

4	+61417692485	5	1726318414...	1726318414000	0	1	-1	1	0	...	Alex, I'm outside your dorm. Can we talk?	
4	+61417692485	...	1726318436...	0	...	1	-1	2	...	...	Sophia, now's not a good time. I'm swamped.	
4	+61417692485	5	1726318448...	1726318448000	0	1	-1	1	0	...	It'll only take a minute. Please, I really need to see you.	
4	+61417692485	...	1726318462...	0	...	1	-1	2	...	...	Fine, come up. But I don't have long.	
4	+61417692485	5	1726318476...	1726318476000	0	1	-1	1	0	...	Thank you. I just... I need to know where we stand. I can't keep feeling like this.	
4	+61417692485	...	1726318495...	0	...	1	-1	2	...	...	We'll talk when you get here.	
4	+61417692485	5	1726318507...	1726318507000	0	1	-1	1	0	...	Okay, I'm on my way.	
4	+61417692485	5	1726318688...	1726318689000	0	1	-1	1	0	...	I'm just outside. I can't take this anymore, Alex. You promised you'd choose me.	NULL
4	+61417692485	...	1726318705...	0	...	1	-1	2	...	...	Sophia, please, I need time. We'll talk, but not like this.	NULL
4	+61417692485	5	1726318716...	1726318716000	0	1	-1	1	0	...	No, I need to know now. It's either me or her.	NULL
4	+61417692485	...	1726318729...	0	...	1	-1	2	...	...	Come up, we'll talk. But you need to calm down.	NULL
4	+61417692485	5	1726318740...	1726318740000	0	1	-1	1	0	...	I'm calm, but I don't think you understand what this means for me. You can't just toss me aside.	NULL
4	+61417692485	...	1726318761...	0	...	1	-1	2	...	...	I'm not tossing you aside. Let's talk when you get here.	NULL

Figure 67: Conversation Between Sophie and Alex

5	+61458230941	...	1726318530...	1726318531000	0	1	-1	1	0	...	You're overdue on the payment. We agreed you'd have it by tonight.	NULL
5	+61458230941	...	1726318551...	0	...	1	-1	2	...	...	I'm working on it. I just need a bit more time.	NULL
5	+61458230941	...	1726318563...	1726318564000	0	1	-1	1	0	...	Time's up, Alex. You don't want to see what happens if you keep stalling.	NULL
5	+61458230941	...	1726318574...	0	...	1	-1	2	...	...	I'll have it by tomorrow. Please, just one more day.	NULL
5	+61458230941	...	1726318588...	1726318588000	0	1	-1	1	0	...	Fine. One day. But after that, we're done talking.	NULL
5	+61458230941	...	1726318602...	0	...	1	-1	2	...	...	Thank you. I won't let you down.	NULL

Figure 68 : Conversation Between Alex and Unknown Number (+61458230941)

As by above data, Alex was having conversation with 3 people which included **Sophia** , **Lily** and **Unkown person(+61458230941).** Deleted_contacts db was also found.

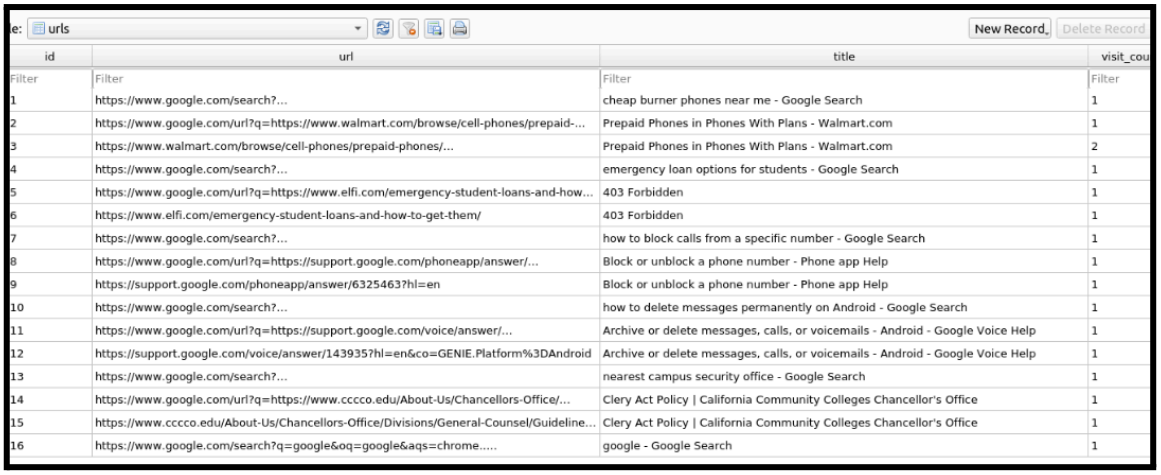
Question 15. What Internet searches has the owner of the phone made?

To find web searches of the owner, I run the command **Find** to get valuable logs and database. I searched keyword history and get the following results.

```
root@siftworkstation:/mnt/windows_mount# find /mnt/windows_mount -iname "history"
/mnt/windows_mount/data/com.android.chrome/app_chrome/Default/History
root@siftworkstation:/mnt/windows_mount# find /mnt/windows_mount -iname "*history*"
/mnt/windows_mount/data/com.android.chrome/app_chrome/Default/History
/mnt/windows_mount/data/com.android.chrome/app_chrome/Default/History-journal
/mnt/windows_mount/data/com.google.android.gms/databases/google_account_history.db
root@siftworkstation:/mnt/windows_mount#
```

Figure 67 : Checking Histroy Files

I copied all the files and reset the ownership as previously. Then I examined all the files by sqlite and found all websearches in the **History** file in table **urls**(Figure 68)



id	url	title	visit_cou
Filter	Filter	Filter	Filter
1	https://www.google.com/search?...	cheap burner phones near me - Google Search	1
2	https://www.google.com/url?q=https://www.walmart.com/browse/cell-phones/prepaid-...	Prepaid Phones in Phones With Plans - Walmart.com	1
3	https://www.walmart.com/browse/cell-phones/prepaid-phones/...	Prepaid Phones in Phones With Plans - Walmart.com	2
4	https://www.google.com/search?...	emergency loan options for students - Google Search	1
5	https://www.google.com/url?q=https://www.elfi.com/emergency-student-loans-and-how...	403 Forbidden	1
6	https://www.elfi.com/emergency-student-loans-and-how-to-get-them/	403 Forbidden	1
7	https://www.google.com/search?...	how to block calls from a specific number - Google Search	1
8	https://www.google.com/url?q=https://support.google.com/phoneapp/answer/...	Block or unblock a phone number - Phone app Help	1
9	https://support.google.com/phoneapp/answer/6325463?hl=en	Block or unblock a phone number - Phone app Help	1
10	https://www.google.com/search?...	how to delete messages permanently on Android - Google Search	1
11	https://www.google.com/url?q=https://support.google.com/voice/answer/...	Archive or delete messages, calls, or voicemails - Android - Google Voice Help	1
12	https://support.google.com/voice/answer/143935?hl=en&co=GENIE.Platform%3DAndroid	Archive or delete messages, calls, or voicemails - Android - Google Voice Help	1
13	https://www.google.com/search?...	nearest campus security office - Google Search	1
14	https://www.google.com/url?q=https://www.cccco.edu/About-Us/Chancellors-Office/...	Clery Act Policy | California Community Colleges Chancellor's Office	1
15	https://www.cccco.edu/About-Us/Chancellors-Office/Divisions/General-Counsel/Guideline...	Clery Act Policy | California Community Colleges Chancellor's Office	1
16	https://www.google.com/search?q=google&oq=google&aqs=chrome.....	google - Google Search	1

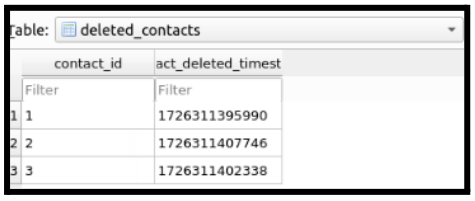
Figure 68 : Internet Searches Data

Owner made the following Searches which include
1 . Nearest campus security office
2. How to delete messages permanently on Android
3.Block or unblock a phone number - Phone app Help
4. Cheap burner phones near me
5 Emergency loan options for students
And some other random searches can be seen in above Figure 68

Question 16. Is there other evidence on the phone that might indicate the role of the owner in Alex's death?

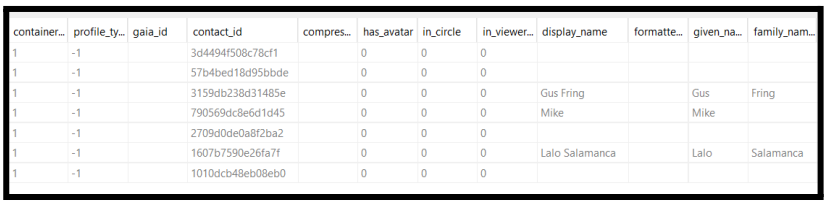
There are many other evidence which I found in **dm-1** . After finding a table **deleted_contacts** in **contacts2.db** I noticed 3 numbers that have been deleted within 1 minute as shown in Figure 69.

After investigating more, I found a database name **pluscontacts.db** and in one of the table name **ac_container** I found 3 rare Names **Gus Fring**, **Mike** and **Lalo Salamanca** (Figure 70). After this I use strings command to extract the information related to Gus Fring person. I use command strings and grep as shown in Figure 71 and it extracted out valuable information highlighting there Mike and lalo salamanca including there phone numbers (Figure 71).



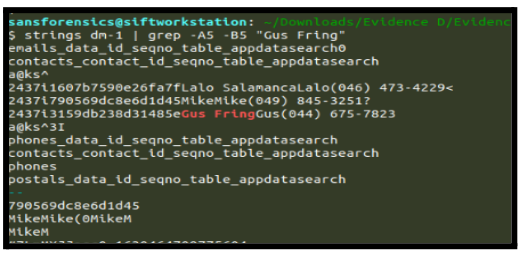
contact_id	act_deleted_timest
Filter	Filter
1	1726311395990
2	1726311407746
3	1726311402338

Figure 69 Deleted Contacts



container...	profile_ty...	gaia_id	contact_id	compres...	has_avatar	in_circle	in_viewer...	display_name	formatte...	given_na...	family_nam...
1	-1		3d4494f508c78cf1		0	0	0				
1	-1		57b4bed18d95bbde		0	0	0				
1	-1		3159db238d31485e		0	0	0	Gus Fring		Gus	Fring
1	-1		790569dc8e6d1d45		0	0	0	Mike		Mike	
1	-1		2709d0de0a8f2ba2		0	0	0				
1	-1		1607b7590e26fa7f		0	0	0	Lalo Salamanca		Lalo	Salamanca
1	-1		1010dcb48eb08eb0		0	0	0				

Figure 70: New names shown in plusdatabase



```
sansforensics@siftworkstation: ~/Downloads/Evidence D/Evidenc
$ strings dm-1 | grep -A5 -B5 "Gus Fring"
emails_data_id_seqno_table_appdatasearch0
contacts_contact_id_seqno_table_appdatasearch
a@ks^
2437i1607b7590e26fa7fLalo SalamancaLalo(046) 473-4229<
2437i790569dc8e6d1d45MikeMike(049) 845-3251?
2437i3159db238d31485eGus FringGus(044) 675-7823
a@ks^3I
phones_data_id_seqno_table_appdatasearch
contacts_contact_id_seqno_table_appdatasearch
phones
postals_data_id_seqno_table_appdatasearch
--
790569dc8e6d1d45
MikeMike(0MikeM
MikeM
```

Figure 7 1: Run Strings Command

I also find a strange file in **/media/0/Download**, which is r**ecording.aes**. This file is encrypted with aes algorithm and holds valuable data in figure 72

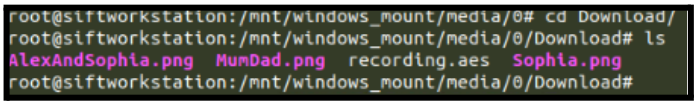


Figure 72 : recording.aes File

Question 17. Conduct a timeline analysis of the pieces of evidence.

For **evidence A** timeline analysis, I use **fls** tool from sleuth kit to list all the metadata of Evidence A from starting flag. By this command I received the output evidencetimefile.txt which include **Modified, Accessed, Changed** and **Created** timestamps in figure 73. After this I use **mactime** tool, which organize the data chronologically and shows files created or modified. File formate set to **csv** for easy analyzation as shown in Figure 74

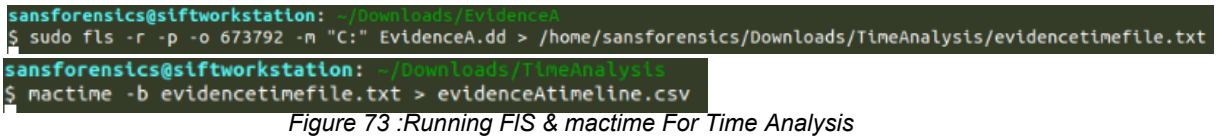


Figure 73 :Running FIS & mactime For Time Analysis

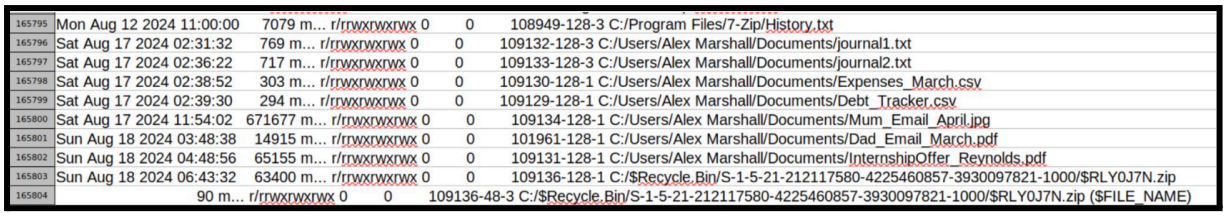


Figure 74 :Running FIS & mactime For Time Analysis

For **evidence B** timeline analysis, I capture timestamps of the network capture through tool wireshark. As **wireshark** highlight the timestart and endingtime and separating different protocols making it easier to do analysis. I also capture valuable info timestamps manually as shown in Figure 23, 24 and 25.

For **evidence C** timeline analysis, I use **volatility** and mactime tools to extract metadata. Firstly I use the plugin **timeliner**, which gather **timestamps** data of creation, access, modifiing and give output in .body file. After getting output I use **mft parser** plugin to extract the meta data of activity and getting the output in .body format again in Figure 75

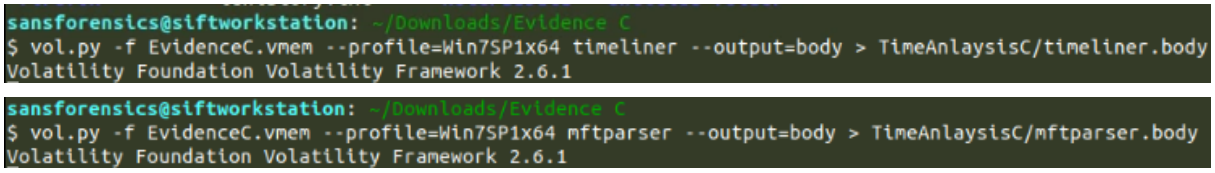


Figure 75 : Running Plugin Timeliner & Mftparser

After this I concatenate the data of both files to create detailed data by using **cat** (Figure 75) command and get the final output as shown in the figure . After this I use mactime tool again to give ascending order to the metadata of whole file and check sequence(Figure 76)

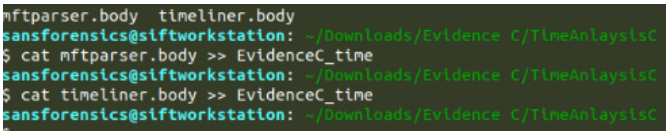


Figure 76: Concatenating Data

27259	Sun Aug 25 2024 07:18:20	0	macb	---a-------	0	0	511 [MFT FILE_NAME] Users\SOPHIA~1\AppData\Roaming\MICROS~1\INTERN~1\QUICKL~1\Shows Desktop.lnk (Offset: 0x7ee53c00
27260	Sun Aug 25 2024 07:18:20	0	.a.b	---a-------	0	0	511 [MFT STD_INFO] Users\SOPHIA~1\AppData\Roaming\MICROS~1\INTERN~1\QUICKL~1\SHOWSD~1.LNK (Offset: 0x7ee53c00)
27261	Sun Aug 25 2024 07:18:20	0	macb	-hsa-------	0	0	512 [MFT FILE_NAME] Users\SOPHIA~1\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Accessories\System Tools\Desktc
27262	Sun Aug 25 2024 07:18:20	0	.a.b	-hs-------	0	0	512 [MFT STD_INFO] Users\SOPHIA~1\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Accessories\System Tools\Desktop
27263	Sun Aug 25 2024 07:18:20	0	macb	---a-------	0	0	513 [MFT FILE_NAME] Users\SOPHIA~1\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Accessories\Accessibility\Narrator
27264	Sun Aug 25 2024 07:18:20	0	.a.b	---a-------	0	0	513 [MFT STD_INFO] Users\SOPHIA~1\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Accessories\Accessibility\Narrator.l
27265	Sun Aug 25 2024 07:18:20	0	macb	-hsa-------	0	0	514 [MFT FILE_NAME] Users\SOPHIA~1\NTUSER.DAT{016888bd-6c6f-11de-8d1d-001e0bcde3ec}.TMContainer00000000000000000000

Figure 78 : Timline of Evidence C output

For **Evidence D**, I used similar process as **Evidence A**. I used **fls** and **mactime** tools. Firstly I use **fls** command to extract the metadata of disk, here **-r** mean to list directories and subdirectories . The **-m** shows that root path is set as system and the output is save as **evD.txt file**. After this I run **mactime** to arrange the all the metadata in sequential format that is easy to analyse and save the output in csv file for easy visualisation in Figure 78. After this I combined all the important events in separate csv file to do proper time analysis and to recreate the narrative and detect activities of the people for accurate analyization as shown in figure 79 & 80 and also link for file is given as Sorted Data Of All Evidence .

Date	Time	Action
8/12/2024	11:00	C:/Program Files/7-Zip/History.txt
8/17/2024	2:36	C:/Users/Alex Marshall/Documents/journal2.txt
8/17/2024	2:38	C:/Users/Alex Marshall/Documents/Expenses_March.csv
8/17/2024	2:39	C:/Users/Alex Marshall/Documents/Debt_Tracker.csv
8/17/2024	2:31	C:/Users/Alex Marshall/Documents/journal1.txt
8/17/2024	11:54	C:/Users/Alex Marshall/Documents/Mum_Email_April.jpg
8/18/2024	3:48	C:/Users/Alex Marshall/Documents/Dad_Email_March.pdf
8/18/2024	4:48	C:/Users/Alex Marshall/Documents/InternshipOffer_Reynolds.pdf
8/18/2024	6:43	C:/$Recycle.Bin/S-1-5-21-212117580-4225460857-3930097821-1000/$RLY0J7N.zip
8/25/2024	4:42	C:/Users/Alex Marshall/Pictures/DayOne.png
8/25/2024	12:38	C:/$Recycle.Bin/S-1-5-21-212117580-4225460857-3930097821-1000/$RAUO6GYK.txt
8/25/2024	4:42	C:/Users/Alex Marshall/Pictures/DormC.png
8/25/2024	4:42	C:/Users/Alex Marshall/Pictures/CampusfromTower.png
8/25/2024	5:57	C:/Users/Alex Marshall/Pictures/Alex and Lily.png
8/25/2024	5:58	C:/Users/Alex Marshall/Pictures/Alexand Lily 2.png
8/25/2024	5:59	C:/Users/Alex Marshall/Pictures/lily.png
9/1/2024	8:41:01	Bio101 file found in network packets
9/1/2024	08:09:28 -	Chatting between Alex and Sophia
9/1/2024	07:57:11 -	Conversation between AlexM21,DormKing,PartDude,BookWarm
9/1/2024	07:57:11 -	Deal for Bio101 answers
9/1/2024	08:16:09 -	DormKing Manipulates PartyDude
9/1/2024	08:44:49 -	DormKing accuses PartDude
9/1/2024	8:43:17	SSL/TLS found in the network packets
9/1/2024	08:19:53 -	Conversation between LillyLaw and Alex
9/8/2024	07:08:41 -	Thunderbird session
9/8/2024	08:00:31 -	Firefox session
9/8/2024	12:15:06 -	Notepad session
9/8/2024	8:01:02	download thunderbird 115 - Sophia web search
9/8/2024	12:23:22	How to stop someone from leaving you - Sophia web search
9/8/2024	12:24:52	Signs someone is cheating - Sophia web search
9/8/2024	12:24:59	How to get away with self-defense - Sophia web search
9/8/2024	12:25:13	Emergency therapy services - Sophia web search
9/14/2024	12:58:49	Come up, weâ€™ll talk. But you need to calm down. (From: +61417692485)
9/14/2024	10:58:56	Call Log: Lily Parker (+61449857236)
9/14/2024	10:58:53	Call Log: Lily Parker (+61449857236)
9/14/2024	10:59:22	Call Log: Sophie Bennett (+61417692485)

9/8/2024	12:24:59	How to get away with self-defense - Sophia web search
9/8/2024	12:25:13	Emergency therapy services - Sophia web search
9/14/2024	12:58:49	Come up, weâ€™ll talk. But you need to calm down. (From: +61417692485)
9/14/2024	10:58:56	Call Log: Lily Parker (+61449857236)
9/14/2024	10:58:53	Call Log: Lily Parker (+61449857236)
9/14/2024	10:59:22	Call Log: Sophie Bennett (+61417692485)
9/14/2024	12:51:36	Hey, can we talk tonight? We really need to sort this out. (From: +61449857236)
9/14/2024	12:52:09	Alex, this is important. I canâ€™t keep going on like this, wondering where we stand. (From: +61449857236)
9/14/2024	12:52:25	I know, I know. But I just canâ€™t tonight. Tomorrow? (From: +61449857236)
9/14/2024	12:52:38	Tomorrow might be too late, Alex. If you donâ€™t end this with her, I will. (From: +61449857236)
9/14/2024	12:52:55	Lily, please donâ€™t do anything rash. Iâ€™ll figure it out, I promise. (From: +61449857236)
9/14/2024	12:53:03	You better. Iâ€™m done waiting. (From: +61449857236)
9/14/2024	12:53:34	Alex, Iâ€™m outside your dorm. Can we talk? (From: +61417692485)
9/14/2024	12:54:01	Sophia, nowâ€™s not a good time. Iâ€™m swamped. (From: +61417692485)
9/14/2024	12:54:08	Itâ€™ll only take a minute. Please, I really need to see you. (From: +61417692485)
9/14/2024	12:54:22	Fine, come up. But I donâ€™t have long. (From: +61417692485)
9/14/2024	12:54:36	Thank you. I justâ€¦ I need to know where we stand. I canâ€™t keep feeling like this. (From: +61417692485)
9/14/2024	12:54:55	Weâ€™ll talk when you get here. (From: +61417692485)
9/14/2024	12:55:07	Okay, Iâ€™m on my way. (From: +61417692485)
9/14/2024	12:55:51	Iâ€™m working on it. I just need a bit more time. (From: +61458230941)
9/14/2024	12:56:03	Timeâ€™s up, Alex. You donâ€™t want to see what happens if you keep stalling. (From: +61458230941)
9/14/2024	12:56:14	Iâ€™ll have it by tomorrow. Please, just one more day. (From: +61458230941)
9/14/2024	12:56:28	Fine. One day. But after that, weâ€™re done talking. (From: +61458230941)
9/14/2024	12:56:42	Thank you. I wonâ€™t let you down. (From: +61458230941)
9/14/2024	12:58:08	Iâ€™m just outside. I canâ€™t take this anymore, Alex. You promised youâ€™d choose me. (From: +61417692485)
9/14/2024	12:58:25	Sophia, please, I need time. Weâ€™ll talk, but not like this. (From: +61417692485)
9/14/2024	12:58:36	No, I need to know now. Itâ€™s either me or her. (From: +61417692485)
9/14/2024	10:58:59	Call Log: Lily Parker (+61449857236)
9/14/2024	12:59:00	Iâ€™m calm, but I donâ€™t think you understand what this means for me. You canâ€™t just toss me aside. (From: +61417692485)
9/14/2024	10:59:02	Call Log: Lily Parker (+61449857236)
	11:23:00	Not tonight, Lily. Iâ€™m dealing with a lot right now. (From: +61449857236)
	11:57:02	Youâ€™re overdue on the payment. We agreed youâ€™d have it by tonight. (From: +61458230941)

Figure 79, 80 : Time analysis of all Evidence Part 1 and Part 2

Question 18. Provide a brief final analysis of the evidence and your conclusions.
Evidence A:

The analysis of this part shows that Alex was facing intense pressure due to financial debt, academic stress and complex relationship with Sophia and Lily. His journal entries from Feb and April 2024 shows he was in guilt for **selling** exam papers and feared of getting caught. Another major factor was in file **debit_tracker** in which he was under debit of **$7450** in which there was a mysterious **$5000** loan on April 30, 2024 from unknown source added more pressure in his situation. Furthermore, University letter on March 15 informed Alex failing two course which put his **academic** and **internship** offer in future received on April 1st. In March His **Father,** Oliver Marshall expressed disappointment in mail and asked Alex to work hard or otherwise he would cut off his credit card. In the end, He wrote a testament in which he expressed his regret and message for people he loved . The main highlight of this Evidence

A is the urgent loan of **$5000** from an **unknown** source as this bit is mentioned in the future evidence and can be the reason for Alex **death**.

Evidence B:

The Analysis of evidence B shows that Alex was involved in selling the exam answers, which was likely to coverup the debit and other financial issues. This conversation took place on **September 1, 2024**, in the very first streams Alex was seen negotiating with Drom King and Party dude for **Bio101 exam** answers. They agree on the price of **200$** and Alex did mention that these answers will get them a passing grade and that's how they wont get in trouble. The whole conversation shows Alex engaging in unethical behaviour. After the deal , **DromKing** and **Party Dude** chatted in separate channels and agreed not to pay Alex and instead plan to steal the **session files** from Alex's system when he's gonna go outside. Party Dude with little understanding uploaded the key to Dorm King and then failed to go in Alex 's room as he was talking with some girl. Furthermore, it mentions that the Bio101 exam is uploaded on **08:41:01** time.

Furthermore there are snippets of conversation in the network packets and some of the stream at time **08:19:53**, where **Lily** accused Alex of seeing someone and is very **suspicious** of his activities , whereas on the other hand **Sophia** is asking Alex to choose one of us and make a final decision.

This network packet **highlights** the pressure on Alex to payback the money he owned and to do this he is willing to **unethical** activities shows his desperation which ties with narrative of evidence A, where Alex was feared and overwhelmed by **debit** and **guilt** and I think it was the only way out of this situation.

Evidence C:

The analysis of evidence C consists of a **memory** dump of **Sophia's laptop** in **Vmem format.** The laptop was present in the **Alex** room . As from previous data, it can be deduced that there was possible a **romantic relationship** between alex and sophia . Further we found email and web searches on the system.

After using **volatility** tool kit to scan memory dump, and **"pslist"** plugin to check active services and programs, which include mozilla ,**thunderbird** that play a significant role in uncovering the communication between Alex and Sophia **overwhelmed** by his circumstances and was struggling to cope with several ongoing issues.

While inspecting the data, I found **4 emails** in total, in which two of them were sent by sophia to Alex, where she expressed her feelings to Alex and conveyed her desire to be with him. Furthermore, there was one **rare** email which was sent from Sophia email(vb9945311@gmail.com) but Name **Victor Ivona was used** to **Professor Kane (**Figure 81), where the email was set as **warning** and **advice** to him to be in his lane . Finally, there was an email sent by Alex to sophia where he mentioned that he is in a bad **situation** and overwhelmed by his situation and was struggling to process ongoing issues. The **main narrative** promote the Alex **mental health** and need for support during this time.

Moving Foward, I checked some content of **db files** named **places.sqlite.dat (Fig 82)** and found out the searches made by sophia in which she was searching , "**how to stop someone from leaving**", "**Emergency therapy**" and "**self defence**". These searches highlight that sophia was going through an abdomen period and she was **possessive** for Alex.

While scanning more I found some journals written by sophia, name **Journal 1.txt** and **Journal 2.txt** that couldn't be accessed through memory forensic. In the last I did run plugin

Isa command to get hashdump and password was released of sophia which was **Alexisbeautiful.** Overall this evidence shows that Sophia was pretty possessive about alex and her mostly data was related to him, where as Alex was not in good mental health and mention his condition is getting worse.

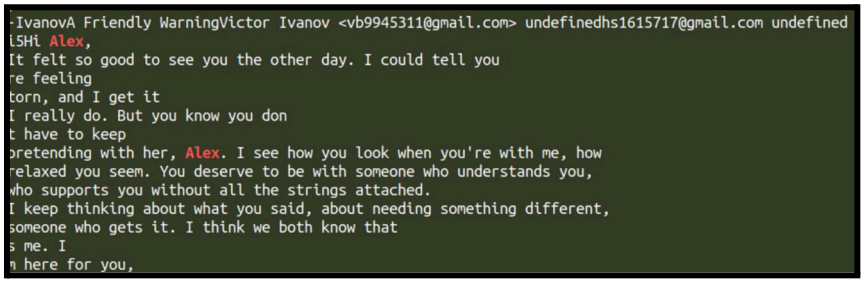


Figure 81: Rare Email to Professor Kane

id	fieldname	value	timesUsed	firstUsed	lastUsed
	Filter	Filter	Filter	Filter	Filter
	searchbar-history	download thunderbird 115	1	1725782462716000	1725782462716000
	searchbar-history	How to stop someone from leaving you	1	1725798280204000	1725798280204000
	searchbar-history	Signs someone is cheating	1	1725798292305000	1725798292305000
	searchbar-history	How to get away with self-defense	1	1725798299826000	1725798299826000
	searchbar-history	Emergency therapy services	1	1725798313777000	1725798313777000

Figure 82:: Web Searches Made By Sophia

Evidence D:

The analysis of **evidence D** provides more valuable information about the unknown sources. The evidence D has **10 disks** in which **dm-1** has the most data. After mounting the disk, It shows that the mobile belongs to Alex and has information about his **call logs, sms** and **contacts** numbers. Going through a database of call logs shows that **lily** tried to make multiple calls and for Alex **declined** all the calls, in the meantime there was a call with **Sophia** about **70** seconds in figure 64. Further moving to the contacts database , there were 4 contacts **Mum, Dad, Sophia** and **Lily** as mentioned in figure 65. Furthermore Alex sent sms to **3 different people**, one was lily in which lily asked him to end with sophia or otherwise she will take some strict action. Second with sophia in which sophia asked Alex if they can **meet** and she is outside his home, eventually Alex told her to come upstairs and it relates to the Evidence C and why sophia laptop was present in Alex's room. Third conversation was with Unknown source, as a person in the msg mentioned that if Alex doesn't pay then they are done talking and something bad will happen. The suspicious thing was found in Alex internet Search history in database **History** , in which he went through first checking for **security** near his home to search for an instant loan. After that he searches for buying a **burner phone** and how to block or **delete numbers** permanently.

After going through internet history, I checked for **deleted numbers** and I found deleted contact tables in database **contact2.db.** The timeframe of deleting those numbers was just 1 minute, which shows Alex removed these numbers in a hurry. Upon further investigation, I found a database which highlighted 3 **new people** in the case (Figure 69, 70 ,71). After checking more databases like places pluscontacts.db and icing-wal.db I found the names of these people. Which were **Gus**, **lalo salamanca** and **Mike.** This information changed the whole narrative and pointed the whole focus on these people. Further there was a recording.aes file in the download that I still couldn't decrypt . The **secret.key** was found in evidence B but still key IV has one value missing and making it difficult to crack.

To conclude the whole scene, I deduce the whole scenario revolving around **Alex murder** is that Alex was under **debt** from his activities and he took a big loan of **$5000** from these **unknown sources** (3 guys). To overcome his debt he tried to sell the exam papers and do unethical things to get some money but couldn't succeed as the whole operation was sabotaged by **DormKing** and **Party Dude**. Relating to evidence C , it is hinted multiple times that Alex mentions he is in big trouble and it's hard for him to go back to normal. As mentioned in the Task detail, there was no **force of entry** to open the door but the whole room was in a **mess**, which shows whether Alex knew the **killer** or the door was kept open. There are multiple **searches** in alex phone about trying to **run away** from these guys , he specifically **deleted 3 numbers** and was planning to buy **Burner phone** or looking for a security office to hide and save his life, furthermore he encrypted **recording.aes** file to hide the very important information from some people. The way I the picture this scene according to the time analysis done previously, before murder took place Sophia and Alex met in the room, due to this the room door kept open and these guys **(Gus, Mike and lalo)** enters the Alex house, there was no force entry as he knows these guys but due to **payment issue** these guys made mess in the room by use of force and killed Alex . In my deduction, all the evidence we have found narrows down the investigation to these killers: Gus, Lalo and Mike.

Question 19. Provide advice to your forensics team about the identification and collection methods for each specific evidence item.

Evidence A : Alex System Identification and Collection Method

First step is to Identify the **version** of the system for example which **operating system**, **hardware specifications** and the time system was **last active**. Furthermore, next step is to locate the **critical** files, go through **registry** hives for instance **SAM**, **SYSTEM**, **SOFTWARE** log files and check all the event logs along with the active process.
Prepare detail documentation of the running services, scheduled tasks and open network ports of the system.

For **collection of data** from Alex system , it is advice to use **write blocking tools** to prevent modification of disk. Secondly, while doing disk imaging it is recommended to use tools like **FTK Imager** and **DD (Sahin, 2023)** to copy the data in detailed (bit streaming) from storage drive (HDD) and its quite necessary to match the **hashes** (Md5sum, SHA-1) before and after making the image of the specific disks. For live forensic, it is recommended to use tools like Belkasoft Capture (Cloud Tips, 2023) before powering off the system and document all the steps in detailed to maintain authority.

Evidence B: PCAP File of Network Transmission

For identification of network transmission, it is mostly recommended to identify the **source** and **destination** IP communication **protocols** first for example (TCP,HTTP, FTP). After this step, analyst should look for patterns of different streams and detect suspicious behaviour, files exchange , traffic spikes and hidden protocols. Analyst can also use **ARP** spoofing, DNS **tunnelling** or do port scanning to detect **malicious** things in the network. Lastly traffic, such as **TCP**, **UDP**, **HTTP**, **DNS**, and **SSL/TLS** must be reviewed for deeper inspection.

For **collection** of **network traffic**, it is recommended to use **Wireshark** or **tcpdump** to capture the timestamps for accurate data filtering. We can use another powerful tool like **NetworkMiner** to extract files, emails and hidden data from the PCAP file. Also It is quite

important to capture this type of data directly from **router** or **IDS/IPS** and **firewall** systems which ensures that no packets are dropped during the process of collecting the data from network.

Evidence C : Memory Dump of Sophia Laptop

When doing memory analysis, it is highly recommended to start identifying the **running process** first as it gives clear idea which **applications**, programs and process were active (browser, email, notepad). After analysing the programs, investigator should examine all the **network socket**s to trace communication with external and internal enviroment. Furthermore, analyst should check for **credentials** and **passwords** often include plain text and **decrypted** applications **sessions**. While identifying data, **memory areas** that are not active should be also examined thoroughly to collect data like **cache**, **history files** and **artefacts**.

To **collect** data from memory dump tools like **Volatility** and **Rekall** are used to capture the running process. Also it is quite important to check the state of the machine as well as if system is still running then tools like FTK imager is best to take the snapshot of the memory as it collect the data in detail form matching each bit. Also it is quite important to run a live memory forensic to document all open logs and process before powering down the system. According to traditional way turning off the pc is not recommended as it clears all the cache and close the logs, analyst should alway pull the pug out to freez the current state and then gather the data. Once everything is done and data is collected, analyst should hash the files using SHA-256 to maintain the integrity and document each individual who handle the evidence

Evidence D: Mobile Android Data

In android device analyst need to narrow down there search to main data points which are listed below

Alex's mobile contain **call logs**, **SMS/MMS messages**, **emails**, and **location data**, which contains database and provide insights. Analyst should go through all the messaging app data (from apps like WhatsApp or Telegram), media files, and **browser history** that are critical for understanding person details

For collection mobile devices should be treated with care. First step is to create a **disk image** of the mobile device. Before power on the device or extracting data it should be connected to a **write blocking** device to prevent any data modification. Analyst can use tools like **Cellebrite UFED** or **MSAB XRY** to capture all available data, including deleted files, app data, and metadata. By following these steps they can maintain integrity of the data and prevent accidental overwriting or loss.

Identification and collection of evidence are essential for maintaining data integrity in forensic analysis. Alex's Windows system, the PCAP file of network communications, Sophia's laptop memory dump, and Alex's mobile phone image each hold critical information, such as logs, messages, and communication details. Proper collection ensures the evidence remains untampered, enabling a thorough investigation and legally sound conclusions.

TASK 2

Introduction

Capturing and analysing stored data in **Random Access Memory** (RAM) is known as Memory Forensics. It is one of the vital areas within digital forensics which mainly focuses on the analysis of **volatile** memory to uncover evidence of unethical and anomalous behaviour in our systems. The data stored in volatile memory doesn't last for a long time making it **ephemeral** when the system is powered off. Forensic analysts need to be quick to retrieve valuable data that may not be available in traditional storage devices which is quite essential for capturing and analyzing the memory of devices. This data can include **active processes**, network transmission connections, open **files** system, and **encryption** keys, all of which can provide valuable insights into the state of the system (Hamid & Rahman, 2024).

Memory Forensic provides valuable insights into **incident response** and **malware analysis**. In incident response it is quite stressful to detect **malicious** code and software in hard disks but memory forensics allows to capture of a real-time snapshot of the system's volatile data which highlights malware in the running process list. **Snapshot** is also known as a memory dump which extracts vital information such as which users logged in to the system, process details, browsing activity, recently opened files, and memory data applications which is stored in **mem** and **vmem** extensions (Li, Sun, & Xu, 2018).

Memory Forensics is useful in situations where **quick** output is required for example in certain scenarios where hard drive data is too big and the size of RAM is only up to **16GB** (Gaur & Chhikara, 2020). Taking a memory dump will be much faster to transfer the data to be analysed and create a timeline for indicators of compromise. Furthermore, memory forensics is often used in solving **cyber cases**, where it detects **hidden** evidence that helps to reconstruct an attacker’s activities and identify ways to **decrypt** the system’s defenses. Therefore, memory forensics is a critical tool for modern digital investigations, relating to the **Alex case**, the memory dump was done on Sophia's laptop which can aid analysts in the detection of malicious activities and setting up the timeline.

Evidence Acquisition

Evidence acquisition in digital forensics requires a detailed process and systematic approach to maintain absolute authenticity and integrity. According to the international (UNODC, 2023) standard, the important phases for handling digital data include **identification**, **collection**, **acquisition,** and **preservation**. These guidelines and processes were applied in the case of Alex's murder as well to ensure that digital data was kept safe throughout the investigation.

Following the **protocols** of acquiring **digital artifacts** and evidence, **First responders** are the ones who arrive first and take **videos**, and **photos** and **create documentation** of the scene. Their role is to deal with all the digital artifacts including hard drives, computer systems, and laptops, and have to make sure these devices are in secure condition. According to **RFC** (IETF, 2020) guidelines in the process of collecting volatile data in **memory** acquisition, the first step is to **duplicate** or capture memory also known as making **Image of RAM** into different systems to ensure the safety of data. This process is vital to collect the state of current processes(registries) and running apps(logs & cache) as data is

volatile and will be removed or destroyed if the device is powered off. To ensure accuracy, we can use tools like **FTK Imager**, **Fmem**, **volatility,** and **DD**. Each tool has its own pros and cons, for instance, DD and fmem always import their modules to make an image of the RAM, which eventually alters the data. Whereas FTK imager uses the process of **bitstream**(bit by bit) to form a copy of original data and evidence isn't tampered with. Usually, Analysts before processing evidence collection from the system, identify the nature of the device, whether the drive is **encrypted** or **physical** as some evidence is not able to process memory dump, they often require other tools such as for physical use fmem is good and for encrypted we use writer block without interfering the data.

Another tool for **taking images** of drivers in bit-by-bit form is a **write blocker** as this tool is quick and used for preserving integrity and recommended to ensure data remain persistent when transferring data from nonvolatile sections of the system for analysis labs while making sure no **tampering** is made. Forensic analysts prepare USB before for writing dumps and they don't use the same USB for multiple devices as it can alter the data (Li, Sun, & Xu, 2018). Similarly, when extracting a memory dump it should be done on separate files instead of the targeted machine.

Maintaining the **chain of custody** is a core part of digital forensics, it makes sure that each step is documented, has proof of integrity, and ensures data admissibility in court.Detailed documentation is necessary as **the recording** of evidence **condition**, **metadata**, version **name**, **serial issue,** and **people** involved in handling evidence is to make sure no tampering is done. The use of the **hashing** (Rohit *et al.*, 2019) process gives a unique identifier to the memory file. By using algorithms **md5sum**, **sha256,** and **sha512** to generate unique strings that act as **digital fingerprints** and ensure that evidence integrity and no data is changed. For instance, even a single **byte** change in evidence will generate a new hash value and investigators can easily **flag** (Rohit *et al.*, 2019) the tampering as mentioned in Alex's case. Furthermore, in a live system, forensic analysts avoid **shutting down** the machines as it wipes the **cache** of the files, and data will be tampered with or removed. The traditional way to do this process is to pull out the power plug or freeze the system state, these steps won't change the data and the system will **freeze** the current state and data can be derived from this. After collecting all the data and the evidence is ready to be transferred, there is a certain process to follow for example it needs to be stored in **tamper-evident bags** and properly **labeled** with information so it can be used for tracking information which includes the details about the evidence transfer path, which include details about the people who handed the data and accounted for it. By following these **protocols, techniques,** and **proper tools**, investigators ensure data is preserved and can maintain the **chain of custody** which lab analysts can sign off the form to verify integrity at each step to sustain scrutiny in the court..

Forensic Analysis

Memory forensic analysis provides in depth details of volatile data and it goes through **methodical analysis** of the content from **RAM** to separate out the critical information related to the case (Sahin, 2023). Similar to Alex case, analysis is done to **identify patterns** and check **malicious** activities like **DLL injection** or **API hooking** to reconstruct the scenario once the memory content is acquired. To analyse **memory images**, the **volatility** framework is used on **Kali Linux** and it goes through the following process which involves **memory**

craving, **analysis & detection**, **network artefact** access and **registry** examination (Sahin, 2023).

In memory **craving technique** we extract deleted or hidden files from tools like **Bulk extractor** . It **scan** all the memory dump files to look out for the **headers**, **undetected logs** and **corrupted data** as well. It helps investigators to **trace malware**, **timestamps**, **ID's** and valuable information (Cloud Tips, 2023).
Further, analysts do analysis and detection in which they **scan** all the running processes by using Volatility plugins **"pslist"** ,**"psscan"** and **"pstree"**. These plugins display all the **rouge processes** that don't appear in regular listings. The same process is implemented in Sophia's laptop to collect insights of the running process (Cloud Tips, 2023). **Pstree plugin** is also useful in **constructing** the relation between primary process and secondary process in the form of tree, which makes it easier to analyse as it provides good **visualisation**.
For **network** connection artefact, **Rekall** is used to provide valuable insights of web searches connections in real-time. It displays all the details of active and close connection. The **"netscan"** command in volatility gathers network related data and is used to detect virtual and network **anomalies** in communication patterns. Similarly for the **Windows registry** and **logs registry** allow investigators to go through **hives** and extract deleted content including **encrypted keys** (Li, Sun, & Xu, 2018)**.**

Despite being a resourceful tool it faces several **challenges** related to the data being **time sensitive**. Volatile data **loses** its content when its powered off and adds delay as it loses a certain amount of evidence making the data corrupt and unusable. Even in certain stages using **FTK imager** doesn't work as the system reset the memory and valuable metadata and **cache** is already removed.
Another major issue is that cybercriminal use **anti forensic** techniques such as **Obfuscation**, **AES encryption** and **rootkits** to hide their malicious activities. These methods make the data much **harder** to detect and crack files for example if some person uses the encrypted memory regions all across the system then it will delay the process and investigators have to go through manual inspection or required advance **decryption techniques** making the process **costly.**
With today's advancement, If RAM consists of **large amounts** of data it will be a challenging task for the analyst to process memoric forensic on it. As mentioned earlier inorder to process proper **filtering** of data, it needs to be collected bit by bit. There are some tool and plugins like Autopsy and Rekall that can help to narrow down the investigation or pointing towards relevant headers to process valuable data but it will require **more time** and output is usually **corrupted**.
Overall Memory forensic analyzation is important component of investigation, for better practices it is recommended that Analyst should try to prepare **early memory acquisition** using tool like **WinPMem** and **FTK** to create proper **snapshot** of system, analyst should do analysis under multiple tools like **Rekall** and **Bulk extractor** to do cross checking and ensure accuracy of content and use of **write blockers** is necessary to maintain **authenticity** and **integrity** of the system for legal purpose (Hamid & Rahman, 2024).

Forensic Tools

There are multiple tools available for conducting memory forensics on the volatile memory. Analyst interpret the memory content to dig deeper in the investigation to acquire and separate out valuable content that can further analyse. For investigation we use some of the

memory analyzation tools like **Volatility**, **Bulk Extractor**, **Rekall** and **ISO Helix** (Sahin, 2023). Relating to Alex case I used the following tools to acquire data.

Volatility : One of the most recommended memory analysis framework as its an combination of multiple libraries and uses third party access from other tools to collect information. This tools has multiple switches and plugins that makes job easier to sort through complex data and retrieve important information from each memory dumps. Some of the most used plugins are given as
Hashdump : Extract out the hash of each user, that can be decoded to access user password
Hivedump: It prints out all the hives present in the system, helpful in narrowing down the information.
Pstree: This plugin provide the visual display of the primary process in the tree form
Clipboard: By this command analyst can extract data from Windows clipboard programs
Plist : It will list all the running commands and display there Ids and how much memory they are accessing.
Investigator should use following commands to extract out valuable data
vol.py -f memdump.mem – profile= WinSP7X12 pslist
Use for analyzing the process running on the system and time of capture
vol.py -f memdump.mem --profile=WinXPSP2x86 svcscan
Shows the current status of the memory dump and help to detec malicious services
vol.py -f memdump.mem --profile=WinXPSP2x86 filescan
This command helps to detect the location of the files and use to track which files are accessed or modified. It can also do time analysis using plugins like timeliner and shellbags to provide output of all the process in sorted order.
Issues related with volatility is that it is not for beginners as it totally operates on command line and has no **GUI** (Sahin, 2023). Furthermore there are a lot of plugins to understand to perform detailed analysis.

Rekall: It's a memory forensic tool, which is used to analyse the memory images, operating system and extract useful information of networks as well
Cross Platform Support : Can be integrate into different OS environments
Process analysis : Display all the running services in sequential way
Modular: contain a library of plugins that can useful to analyse different memory dumps
Rekal -f memdump.img pslist
Lists processes running during memory dump.
Rekal -f memdump.img netstat
Lists all sockets and IPs of network connections.
Issues related to Rekall’s interface as it is for less experienced analysts due to its command-line structure, and although it's versatile, it lacks a dedicated GUI, making it slightly less user-friendly than some alternatives like WinDbg.

Magnet RAM Capture: Its a popular forensic tool that captures memory of a system. It helps investigators analyse live memory for detecting malware, network connections, and running processes (Sahin, 2023).

Cross-Platform: Compatible with Windows and Linux systems for live memory acquisition.
Process & Memory Analysis: Help to identify running processes, network activity, and

volatile data. Overall it has Simple interface good for beginner and advanced users.as it allows a single-click events, making it accessible for less-experienced analysts.

Issues with **MRC** that is not as flexible for deep-dive analysis as Volatility or Rekall, which offer more customizable commands and plugins for memory examination.Other tools for memory forensics such as **Redline**, **PTK Forensics** etc. Each tools have their own pros and cons and analyst should choose tools based on the case complexity

Conclusion

Analysis of memory forensics was instrumental in the **Alex case**, where volatile memory data from Sophia's laptop was captured and analyzed. Using tools like **Volatility**, **Rekall,** and **Magnet RAM Capture**, forensic analysts retrieved critical information such as active processes, network connections, and system logs. This provided a comprehensive view of the laptop's state during key moments of the investigation. The memory forensics **techniques** uncovered vital evidence, including **malware** traces and user activities, allowing investigators to build a timeline of events and identify malicious behaviors like **DLL injection** and unauthorized process execution (UNODC, 2023) . Furthermore, by leveraging the capabilities of plugins like **"pslist"** and **netscan** (Li, Sun, & Xu, 2018), investigators were able to detect and map suspicious processes and network activities, reinforcing the hypothesis of external tampering. Despite the inherent challenges of working with volatile data, such as its susceptibility to loss and manipulation upon shutdown, the tools used in this case provided sufficient reliability and precision. The acquisition and preservation methods followed strict protocols to ensure evidence integrity, with hashing algorithms confirming data authenticity. Moving forward, the field of memory forensics will need to evolve to manage larger memory sets and combat advanced **anti-forensic** techniques (Sahin, 2023) like obfuscation and encryption. As cyber threats grow more complex, advancements in tool efficiency and analyst training will be crucial in maintaining the accuracy and effectiveness of memory forensics in future digital investigations.

References

1. Hamid, I., & Rahman, M. M. H. (2024). A Comprehensive Literature Review on Volatile Memory Forensics. Electronics, 13(3026). https://doi.org/10.3390/electronics13153026

2. UNODC. (2023). Handling of Digital Evidence. Available at UNODC Cybercrime Module.

3. Internet Engineering Task Force (IETF). (2002). *Guidelines for evidence collection and archiving* (RFC 3227). https://datatracker.ietf.org/doc/html/rfc3227

4. Gaur, S., & Chhikara, R. (2020). *Memory Forensics: Tools and Techniques*. Indian Journal of Science and Technology, 9(48), 1–12. https://doi.org/10.17485/ijst/2016/v9i48/105851

5. Li, S., Sun, Q., & Xu, X. (2018). Forensic analysis of digital images over smart devices and online social networks. In *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)* (pp. 1015-1021). IEEE. https://doi.org/10.1109/HPCC/SmartCity/DSS.2018.00168

6. Rohit, S. Kamra, M. Sharma and A. Leekha, "Secure Hashing Algorithms and Their Comparison," *2019 6th International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, India, 2019, pp. 788-792. keywords: {SHA-1;SHA-256},https://ieeexplore.ieee.org/document/8991234

7. Sahin, A. (2023, October 10). *Investigating memory forensic processes: DLLs, consoles, process memory, and networking*. Medium. https://alpbatursahin.medium.com/investigating-memory-forensic-processes-dlls-consoles-process-memory-and-networking-7277689a09b7

8. Cloud Tips. (2023, October 5). *Memory forensics tools*. Medium. https://medium.com/@cloud_tips/memory-forensics-tools-123e32387adb