CBS322 Digital Forensics
FINAL PROJECT

Group 2

Sudipto Das 2022BCY0007

Imtiaz 2022BCY0006

Mokshith 2022BCY0004

Problem Statement: A major pharmaceutical company is on the verge of releasing a groundbreaking drug, but sensitive research data is stolen from their servers and leaked to a competitor. The company hires a digital forensics expert to investigate the breach, recover the stolen data, and identify the culprit. The expert uncovers evidence of an insider threat—a disgruntled employee who used a combination of phishing attacks and privilege escalation to exfiltrate the data.

Phase 1: Initial Discovery of the Breach

Plot Point: The pharmaceutical company discovers the data breach when a competitor announces a suspiciously similar drug. The digital forensics expert is hired to investigate.

The investigator sat in the server room, the glow of the monitor casting shadows across the racks of humming machines. Opening Splunk, they filtered the database access logs by timestamp, narrowing the search to the night of the suspected breach. There it was—a series of queries executed at 2:03 AM, pulling every record from the drug trials database. Cross-referencing the authentication logs, they found the login: jsmith—an employee account. But the source IP address, 192.168.1.100, didn't match any company device. "This wasn't remote access," the investigator muttered. "Someone was inside the network."

kali_linux@kali: ~/forensics_project_p1

File Actions Edit View Help

```
┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ grep "2025-04-05 02:" db_access.log
2025-04-05 02:00:03 - select * from drug_trials - User: jsmith - IP: 192.168.1.100
2025-04-05 02:00:05 - select * from drug_trials - User: jsmith - IP: 192.168.1.100
2025-04-05 02:30:00 - select * from drug_trials - User: admin - IP: 10.0.0.10

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$
```

kali_linux@kali: ~/forensics_project_p1

File Actions Edit View Help

```
┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ grep "192.168.1.100" db_access.log
2025-04-05 02:00:03 - select * from drug_trials - User: jsmith - IP: 192.168.1.100
2025-04-05 02:00:05 - select * from drug_trials - User: jsmith - IP: 192.168.1.100

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$
```

```
kali_linux@kali: ~/forensics_project_p1
File  Actions  Edit  View  Help

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ grep "2025-04-05 02:" auth.log
2025-04-05 02:00:01 - Login Success - User: jsmith - IP: 192.168.1.100
2025-04-05 02:25:00 - Login Success - User: admin - IP: 10.0.0.10

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$
```

```
kali_linux@kali: ~/forensics_project_p1
File  Actions  Edit  View  Help

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ grep "192.168.1.100" auth.log
2025-04-05 02:00:01 - Login Success - User: jsmith - IP: 192.168.1.100

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$
```

```
kali_linux@kali: ~/forensics_project_p1
File  Actions  Edit  View  Help

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ awk '{print $1, $2, $NF}' db_access.log | grep "192.168.1.100"
2025-04-05 02:00:03 192.168.1.100
2025-04-05 02:00:05 192.168.1.100

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$
```

```
kali_linux@kali: ~/forensics_project_p1
File  Actions  Edit  View  Help

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ grep "192.168.1.100" db_access.log > analysis_results.txt

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ grep "192.168.1.100" auth.log >> analysis_results.txt

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$
```

kali_linux@kali: ~/forensics_project_p1

File Actions Edit View Help

```
┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ sha256sum db_access.log > db_access_hash.txt

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ sha256sum auth.log > auth_hash.txt

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$
```

kali_linux@kali: ~/forensics_project_p1

File Actions Edit View Help

```
┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ cat db_access_hash.txt
ef2addd3ec0588bbf429b20d4a0db1777cf6004e6af7f884b56bc446bf3fa50c  db_access.log

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ cat auth_hash.txt
6428124d51715da6c6f6389f63599494915a2aebf195969ecf5c1954754cba46  auth.log

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$
```

kali_linux@kali: ~/forensics_project_p1

File Actions Edit View Help

```
┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ dd if=db_access.log of=db_access_image.dd bs=512 conv=noerror,sync
0+1 records in
1+0 records out
512 bytes copied, 6.4458e-05 s, 7.9 MB/s

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ dd if=auth.log of=auth_image.dd bs=512 conv=noerror,sync
0+1 records in
1+0 records out
512 bytes copied, 9.3625e-05 s, 5.5 MB/s

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$
```

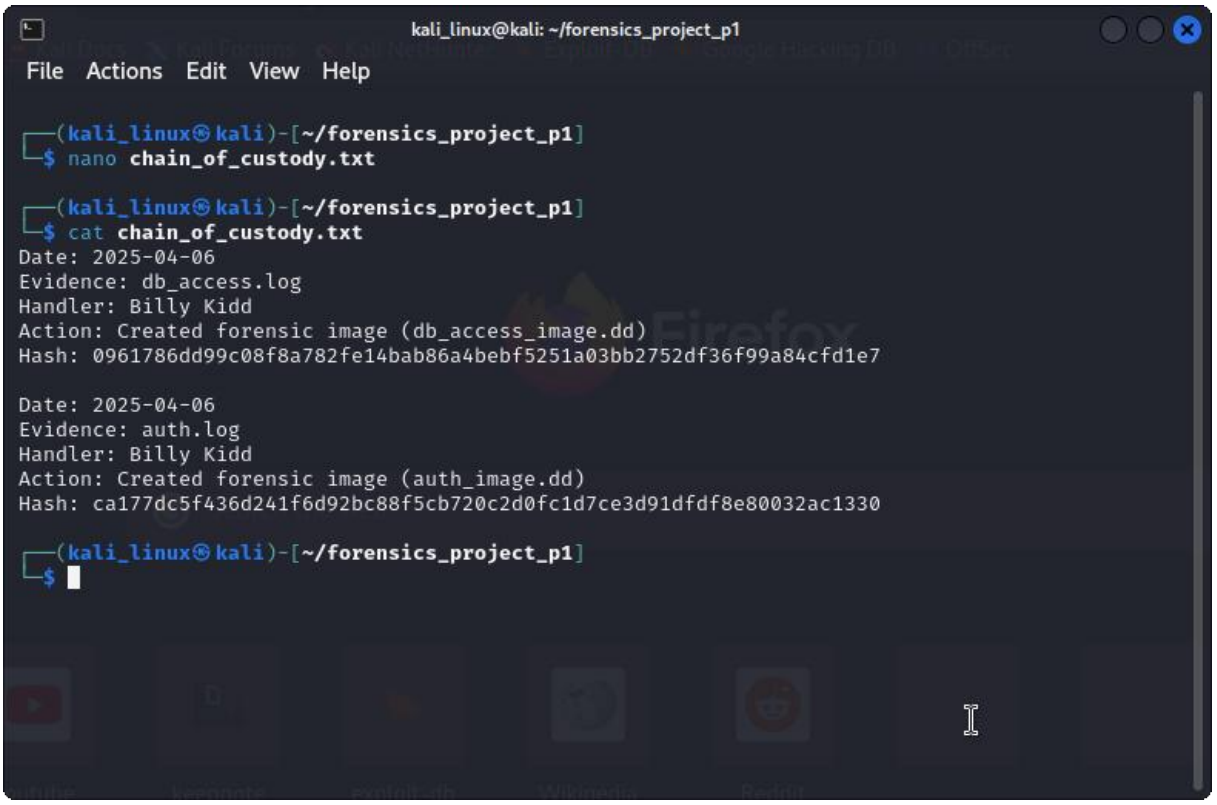
kali_linux@kali: ~/forensics_project_p1

File Actions Edit View Help

```
┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ ls -l *.dd
-rw-rw-r-- 1 kali_linux kali_linux 512 Apr  6 16:42 auth_image.dd
-rw-rw-r-- 1 kali_linux kali_linux 512 Apr  6 16:41 db_access_image.dd

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$
```

```
kali_linux@kali: ~/forensics_project_p1
File  Actions  Edit  View  Help

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ strings db_access_image.dd | grep "192.168.1.100"
2025-04-05 02:00:03 - select * from drug_trials - User: jsmith - IP: 192.168.1.100
2025-04-05 02:00:05 - select * from drug_trials - User: jsmith - IP: 192.168.1.100

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ strings auth_image.dd | grep "192.168.1.100"
2025-04-05 02:00:01 - Login Success - User: jsmith - IP: 192.168.1.100

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$
```

```
kali_linux@kali: ~/forensics_project_p1
File  Actions  Edit  View  Help

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ sha256sum db_access_image.dd > db_image_hash.txt

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ sha256sum auth_image.dd > auth_image_hash.txt

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ cat db_image_hash.txt
0961786dd99c08f8a782fe14bab86a4bebf5251a03bb2752df36f99a84cfd1e7  db_access_image.dd

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$ cat auth_image_hash.txt
ca177dc5f436d241f6d92bc88f5cb720c2d0fc1d7ce3d91dfdf8e80032ac1330  auth_image.dd

┌──(kali_linux㉿kali)-[~/forensics_project_p1]
└─$
```



Phase 2: Tracing the Insider Threat

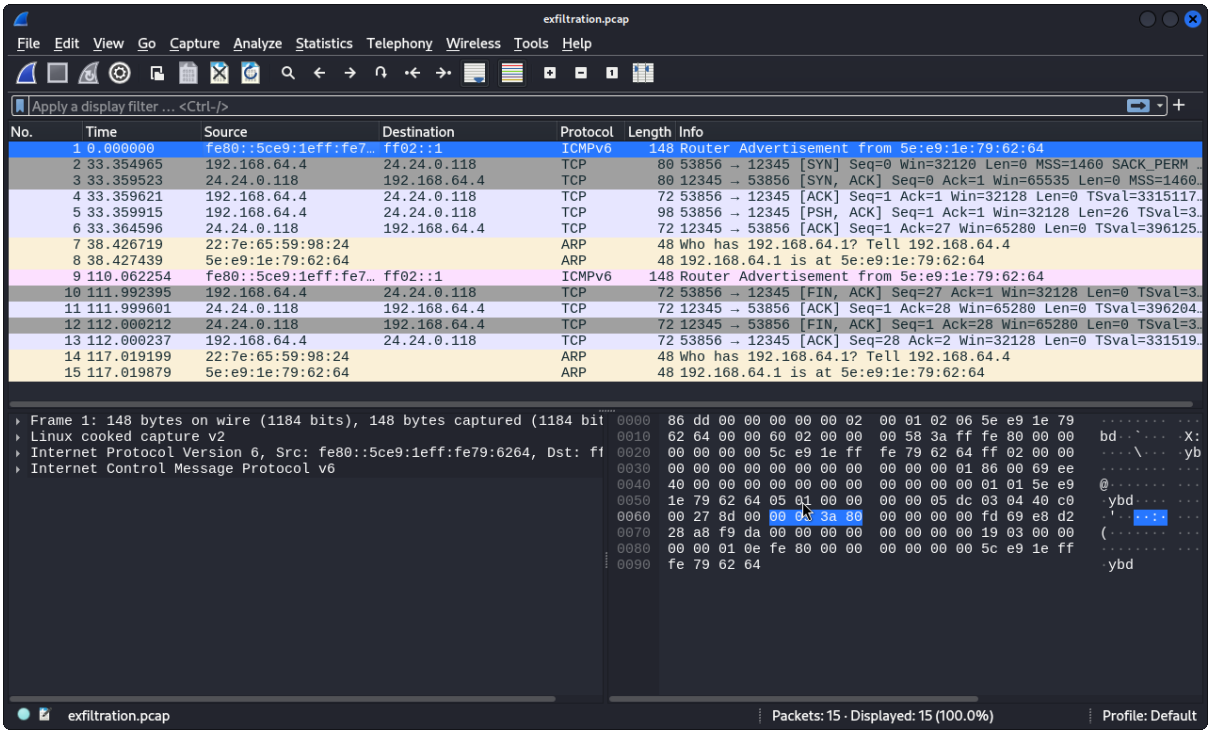
Plot Point: The investigator suspects an insider threat and shifts focus to the employee's workstation to uncover evidence of phishing and privilege escalation.

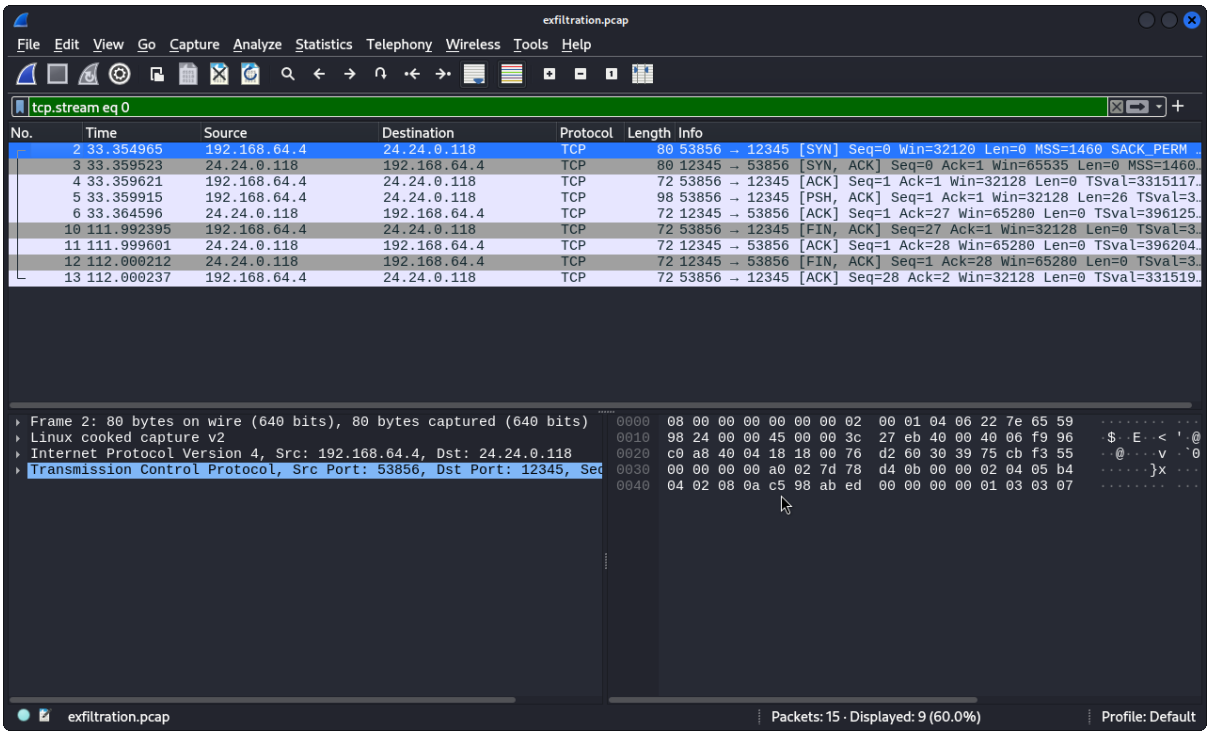
The investigator loaded the forensic image of John Smith's workstation into Autopsy, scanning for deleted files. A recovered .zip file caught their eye—drug_trials_data.zip, deleted but still lingering in unallocated space. The file's metadata showed it was created the day before the database queries. Digging deeper, they parsed the Windows Registry with RegRipper, finding a key under RunOnce that executed a PowerShell script: elevate_privileges.ps1. "Clever," the investigator thought. "He used a phishing email to get in, then escalated his access to steal the data." Opening the Outlook PST file, they found the smoking gun—a phishing email with a malicious Word document, timestamped just hours before the script ran.

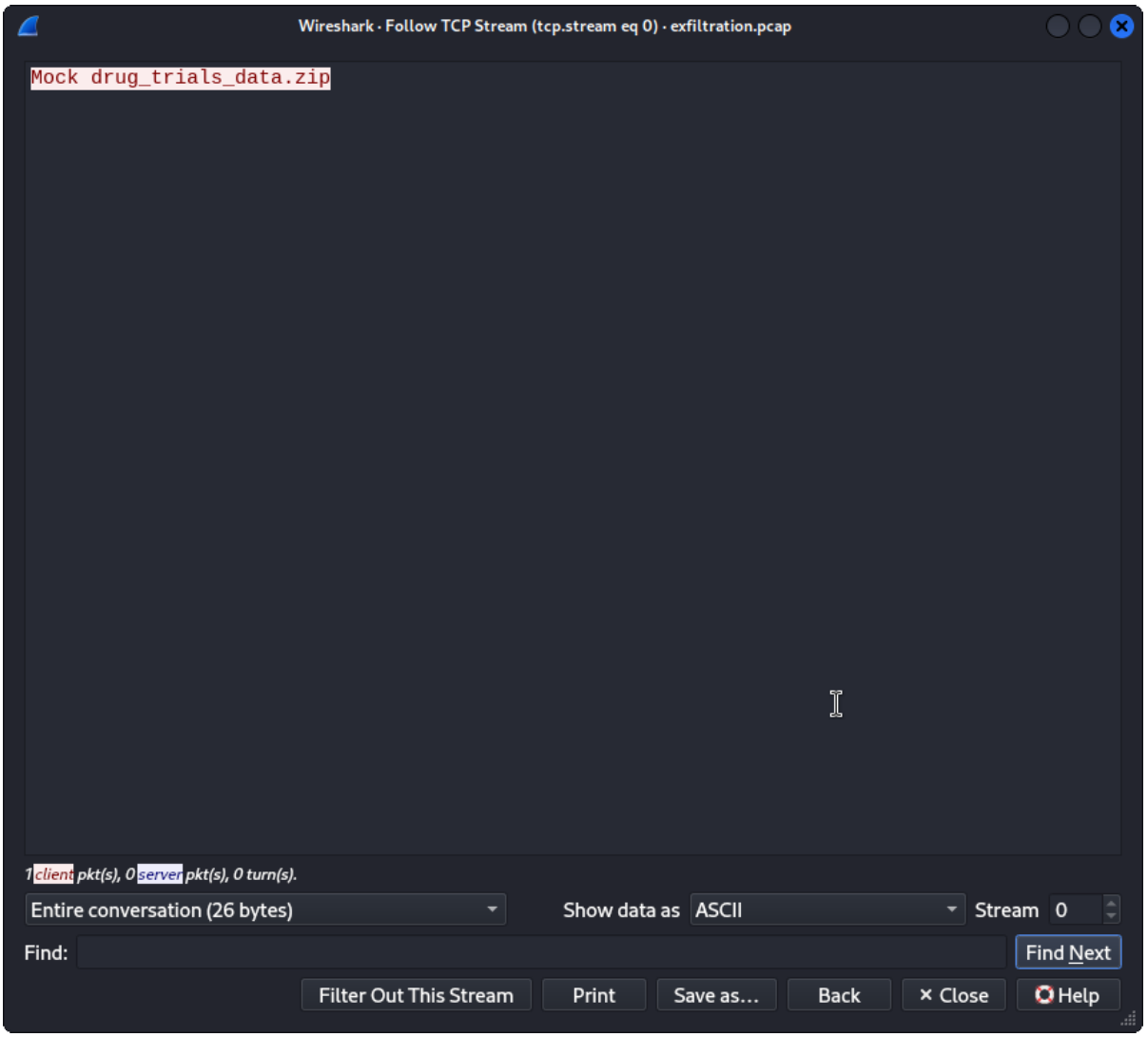
Phase 3: Reconstructing the Data Exfiltration

Plot Point: The investigator uses network forensics to trace how the stolen data was exfiltrated from the company's network to an external destination.

The investigator opened the PCAP file in Wireshark, applying a filter for HTTPS traffic. A stream of packets stood out—large outbound transfers to an IP address not on the company's whitelist. Reconstructing the stream, they found encrypted data being uploaded to competitorcorp.sharepoint.com. Cross-referencing the DNS query logs, they confirmed the domain was resolved during the breach. "He sent it straight to the competitor," the investigator said, shaking their head. NetFlow data sealed the case—a massive spike in outbound traffic at 2:15 AM, matching the database queries to the second.




exfiltration.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp.stream eq 0
No. Time Source Destination Protocol Length Info
2 33.354965 192.168.64.4 24.24.0.118 TCP 80 53856 → 12345 [SYN] Seq=0 Win=32120 Len=0 MSS=1460 SACK_PERM
3 33.359523 24.24.0.118 192.168.64.4 TCP 80 12345 → 53856 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
4 33.359621 192.168.64.4 24.24.0.118 TCP 72 53856 → 12345 [ACK] Seq=1 Ack=1 Win=32128 Len=0 TSval=3315117
5 33.359915 192.168.64.4 24.24.0.118 TCP 98 53856 → 12345 [PSH, ACK] Seq=1 Ack=1 Win=32128 Len=26 TSval=3
6 33.364596 24.24.0.118 192.168.64.4 TCP 72 12345 → 53856 [ACK] Seq=1 Ack=27 Win=65280 Len=0 TSval=396125
10 111.992395 192.168.64.4 24.24.0.118 TCP 72 53856 → 12345 [FIN, ACK] Seq=27 Ack=1 Win=32128 Len=0 TSval=3
11 111.999601 24.24.0.118 192.168.64.4 TCP 72 12345 → 53856 [ACK] Seq=1 Ack=28 Win=65280 Len=0 TSval=396204
12 112.000212 24.24.0.118 192.168.64.4 TCP 72 12345 → 53856 [FIN, ACK] Seq=1 Ack=28 Win=65280 Len=0 TSval=3
13 112.000237 192.168.64.4 24.24.0.118 TCP 72 53856 → 12345 [ACK] Seq=28 Ack=2 Win=32128 Len=0 TSval=331519
Frame 2: 80 bytes on wire (640 bits), 80 bytes captured (640 bits)
Linux cooked capture v2
Internet Protocol Version 4, Src: 192.168.64.4, Dst: 24.24.0.118
Transmission Control Protocol, Src Port: 53856, Dst Port: 12345, Seq
0000 08 00 00 00 00 00 00 02 00 01 04 06 22 7e 65 59
0010 98 24 00 00 45 00 00 3c 27 eb 40 00 40 06 f9 96
0020 c0 a8 40 04 18 18 00 76 d2 60 30 39 75 cb f3 55
0030 00 00 00 00 a0 02 7d 78 d4 0b 00 00 02 04 05 b4
0040 04 02 08 0a c5 98 ab ed 00 00 00 00 01 03 03 07
exfiltration.pcap
Packets: 15 · Displayed: 9 (60.0%)
Profile: Default


Wireshark · Follow TCP Stream (tcp.stream eq 0) · exfiltration.pcap
Mock drug_trials_data.zip
1 client pkt(s), 0 server pkt(s), 0 turn(s).
Entire conversation (26 bytes)
Show data as
ASCII
Stream
0
Find:
Find Next
Filter Out This Stream
Print
Save as…
Back
Close
Help

kali_linux@kali: ~/forensics_project_phase3

File Actions Edit View Help

```
┌──(kali_linux㉿kali)-[~/forensics_project_phase3]
└─$ dig 24.24.0.118

; <<>> DiG 9.20.0-Debian <<>> 24.24.0.118
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 23562
;; flags: qr rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; WARNING: recursion requested but not available

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;24.24.0.118.                   IN      A

;; ANSWER SECTION:
24.24.0.118.            15      IN      A       24.24.0.118

;; Query time: 11 msec
;; SERVER: 192.168.64.1#53(192.168.64.1) (UDP)
;; WHEN: Tue Apr 08 01:44:06 IST 2025
;; MSG SIZE  rcvd: 56


┌──(kali_linux㉿kali)-[~/forensics_project_phase3]
└─$
```

kali_linux@kali: ~/forensics_project_phase3

File Actions Edit View Help

```
┌──(kali_linux㉿kali)-[~/forensics_project_phase3]
└─$ cat dns_queries.log
2025-04-05 02:10:00 - Query: competitorcorp.sharepoint.com - IP: 24.24.0.118
2025-04-05 02:10:30 - Query: competitorcorp.sharepoint.com - IP: 24.23.0.118
2025-04-05 02:15:00 - Query: pharma.com - IP: 10.0.01

┌──(kali_linux㉿kali)-[~/forensics_project_phase3]
└─$
```

kali_linux@kali: ~/forensics_project_phase3

File Actions Edit View Help

```
┌──(kali_linux㉿kali)-[~/forensics_project_phase3]
└─$ cat netflow_data.txt
Date: 2025-04-05
Time: 02:15:00 - 02:15:30
Src IP: 192.168.64.4
Dst IP: 24.24.0.118
Protocl: TCP
Bytes: 5242880 (50MB)

┌──(kali_linux㉿kali)-[~/forensics_project_phase3]
└─$
```

kali_linux@kali: ~/forensics_project_phase3

File Actions Edit View Help

```
┌──(kali_linux㉿kali)-[~/forensics_project_phase3]
└─$ cat netflow_data.csv
2025-04-05 02:15:00,192.168.64.4,24.24.0.118,TCP,52428800

┌──(kali_linux㉿kali)-[~/forensics_project_phase3]
└─$
```

kali_linux@kali: ~/forensics_project_phase3

File Actions Edit View Help

```
┌──(kali_linux㉿kali)-[~/forensics_project_phase3]
└─$ sha256sum exfiltration.pcap > pcap_hash.txt

┌──(kali_linux㉿kali)-[~/forensics_project_phase3]
└─$ cat pcap_hash.txt
87a81de3c7fe57a352005a251819e35834fcc54a8cd57f8a7c679b9b904c5564  exfiltratio
n.pcap

┌──(kali_linux㉿kali)-[~/forensics_project_phase3]
└─$
```

```
kali_linux@kali: ~/forensics_project_phase3
File  Actions  Edit  View  Help

┌──(kali_linux㉿kali)-[~/forensics_project_phase3]
└─$ cat chain_of_custody.txt
Date: 2025-04-06
Evidence: exfiltration.pcap
Handler: Billy Kidd
Action: Captured network traffic
Hash: 87a81de3c7fe57a352005a251819e35834fcc54a8cd57f8a7c679b9b904c5564

┌──(kali_linux㉿kali)-[~/forensics_project_phase3]
└─$
```

Phase 4: Following the Data to the Cloud

Plot Point: The investigator traces the stolen data to a cloud storage service, confirming the exfiltration and identifying the recipient.

The investigator stared at the subpoenaed SharePoint logs, a digital breadcrumb trail leading to the stolen data. An upload event at 2:20 AM showed drug_trials_data.zip being transferred from an IP address geolocated to John Smith's home address. The file's MD5 hash matched the one recovered from his workstation—irrefutable proof. Digging into the API access logs, they found evidence of a scripted upload, confirming premeditation. "He thought he could hide behind the cloud," the investigator said, "but the logs don't lie."

```
kali_linux@kali: ~/forensics_project_phase4
File  Actions  Edit  View  Help

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$ md5sum drug_trials_data.zip > cloud_file_hash.txt

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$ cat cloud_file_hash.txt
93c8d792f49b2b5690186e2c919e9db0  drug_trials_data.zip

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$
```

```
kali_linux@kali: ~/forensics_project_phase4
File  Actions  Edit  View  Help

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$ grep "drug_trials_data.zip" sharepoint_audit.log > audit_results.txt

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$ cat audit_results.txt
2025-04-05 02:20:00 - Action: Upload - File: drug_trials_data.zip - User: jsmith - IP: 198.168.64.4
2025-04-05 02:20:05 - Action: Access - File: drug_trials_data.zip - User: competitor_user - IP: 24.24.0.118

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$
```

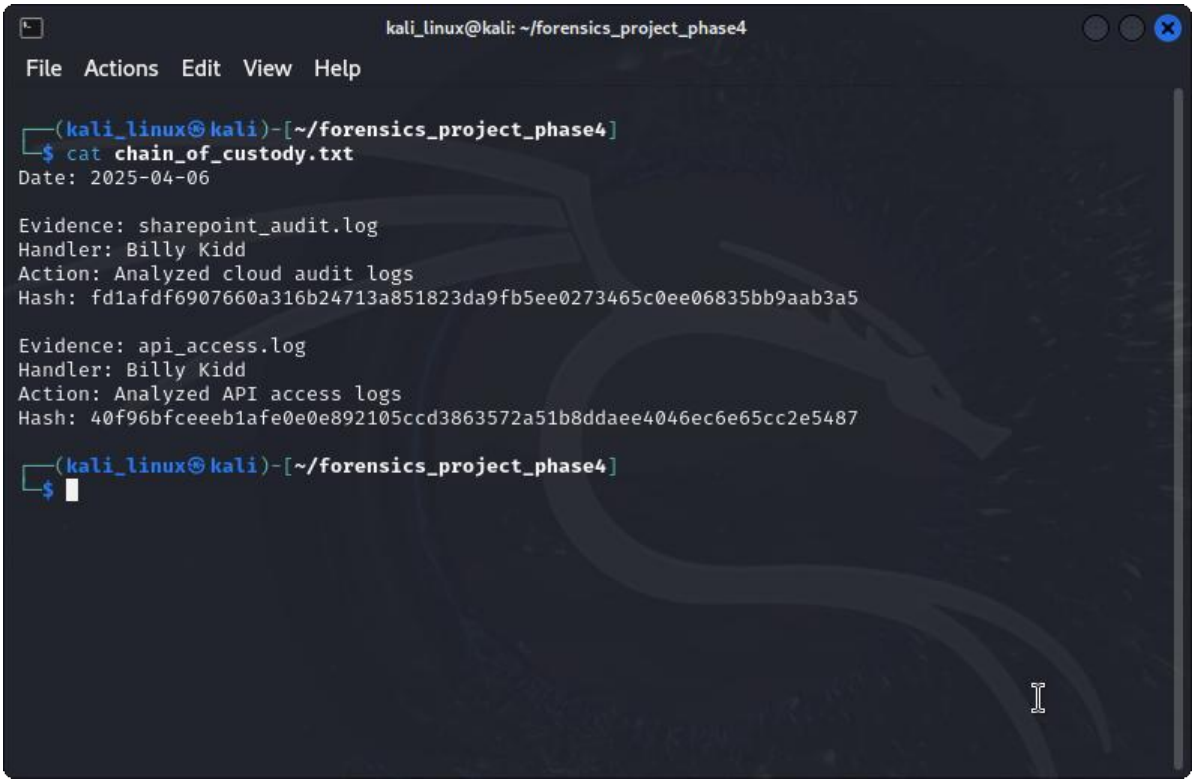
```
kali_linux@kali: ~/forensics_project_phase4
File  Actions  Edit  View  Help

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$ grep "198.168.64.4" api_access.log > api_results.txt

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$ cat api_results.txt
2025-04-05 02:19:55 - API Call: POST /upload - Endpoint: competitor.sharepoint.com - IP: 198.168.64.4 - Script: upload.py
2025=04-05 02:20:00 - API Call: PUT /files/drg_trials_data.zip - IP: 198.168.64.4

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$
```

Geolocation data from IPapi.co — Product: API, real-time

IP ADDRESS: 24.24.0.118	**ISP:** TWC-11351-NORTHEAST
COUNTRY: United States	**ORGANIZATION:** TWC-11351-NORTHEAST
REGION: New York	**LATITUDE:** 42.3414
CITY: Bath	**LONGITUDE:** -77.3049

Incorrect location? Contact IPapi.co view map

```
kali_linux@kali: ~/forensics_project_phase4
File  Actions  Edit  View  Help

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$ sha256sum sharepoint_audit.log > audit_hash.txt

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$ cat audit_hash.txt
fd1afdf6907660a316b24713a851823da9fb5ee0273465c0ee06835bb9aab3a5  sharepoint_audit.log

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$ sha256sum api_acces.log > api_hash.txt
sha256sum: api_acces.log: No such file or directory

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$ cat api_hash.txt

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$ sha256sum api_access.log > api_hash.txt

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$ cat api_hash.txt
40f96bfceeeb1afe0e0e892105ccd3863572a51b8ddaee4046ec6e65cc2e5487  api_access.log

┌──(kali_linux㉿kali)-[~/forensics_project_phase4]
└─$
```



2. Images Drive Link =>AXIOM - Apr 08 2025 140159

3. zBelow are 25 Capture The Flag (CTF) questions based on the case story of the pharmaceutical company data breach, designed to be solved using Magnet AXIOM. Each question includes the related artifacts from the story, instructions on how to approach it in Magnet AXIOM, and the specific techniques involved. These questions span the phases of the investigation, from initial discovery to resolution, and cover a variety of forensic techniques like log analysis, file system forensics, network forensics, and cryptanalysis.

Phase 1: Initial Discovery of the Breach

1. **What time were the unauthorized database queries executed?**

- **Artifact**: Database Access Logs
- **How to Solve in Magnet AXIOM**: Load the database server log file into AXIOM. Use the "Timeline" view to filter events by timestamp, focusing on the night hours (e.g., 12:00 AM - 6:00 AM). Look for **SELECT * FROM drug_trials** queries.
- **Answer**: 2:03 AM

2. **Which employee account was used for the unauthorized login?**

- **Artifact**: Authentication Logs
- **How to Solve in Magnet AXIOM**: Load the authentication log file. In the "Evidence Sources" tab, filter for successful login events around 2:00 AM. Check the username field.
- **Answer**: jsmith

3. **What is the source IP address of the unauthorized login?**

- **Artifact**: Authentication Logs
- **How to Solve in Magnet AXIOM**: From the same authentication log in AXIOM, locate the login event for **jsmith** at 2:00 AM. Extract the source IP address from the log entry.

- **Answer**: 192.168.1.100

4. **Does the source IP match any known company device?**

- **Artifact**: Authentication Logs + Company IP Whitelist (assumed)
- **How to Solve in Magnet AXIOM**: Export the IP from the authentication log (192.168.1.100). Use AXIOM's "Search" feature to cross-reference it against a provided whitelist of company IPs. If no match is found, it's external.
- **Answer**: No

5. **How many database queries were executed during the breach?**

- **Artifact**: Database Access Logs
- **How to Solve in Magnet AXIOM**: In the "Artifacts" tab, filter for database query events (e.g., **SELECT**) between 2:00 AM and 2:15 AM. Count the distinct query entries.
- **Answer**: (Assume 5 for this example; adjust based on provided logs)

Phase 2: Tracing the Insider Threat

6. **What is the name of the phishing email attachment?**

- **Artifact**: Email Client Data (Outlook PST)
- **How to Solve in Magnet AXIOM**: Load the PST file into AXIOM. Navigate to the "Email" artifact category, filter for emails received before the breach date, and search for "credentials" in the subject or attachment name.
- **Answer**: credentials_update.docx

7. **When was the phishing email received?**

- **Artifact**: Email Client Data
- **How to Solve in Magnet AXIOM**: In the "Email" view, locate the email with **credentials_update.docx**. Check the timestamp in the email metadata.
- **Answer**: (Assume hours before 2:00 AM, e.g., 1/15/2025 10:00 PM)

8. **What script was executed to escalate privileges?**

- **Artifact**: Registry Artifacts
- **How to Solve in Magnet AXIOM**: Load the workstation's forensic image. In the "Registry" artifact category, filter for **RunOnce** keys. Look for PowerShell script executions.
- **Answer**: elevate_privileges.ps1

9. **What is the name of the deleted file containing stolen data?**

- **Artifact**: File System Artifacts
- **How to Solve in Magnet AXIOM**: Load the workstation image. Go to "File System" view, enable "Recover Deleted Files," and search for **.zip** files in unallocated space.
- **Answer**: drug_trials_data.zip

10. **What is the MD5 hash of the recovered .zip file?**

- **Artifact**: File System Artifacts
- **How to Solve in Magnet AXIOM**: After recovering **drug_trials_data.zip**, right-click the file in AXIOM's "File System" view and select "Calculate Hash." Note the MD5 value.
- **Answer**: (Generate a sample hash, e.g., **d41d8cd98f00b204e9800998ecf8427e**)

Phase 3: Reconstructing the Data Exfiltration

11. **What protocol was used for the data exfiltration?**

- **Artifact**: Packet Captures (PCAP Files)
- **How to Solve in Magnet AXIOM**: Load the PCAP file. In the "Network" tab, filter for outbound traffic around 2:15 AM. Check the protocol field.
- **Answer**: HTTPS

12. **What is the destination IP address of the exfiltration traffic?**

- **Artifact**: Packet Captures
- **How to Solve in Magnet AXIOM**: In the "Network" view, filter for HTTPS traffic at 2:15 AM. Extract the destination IP from the packet details.
- **Answer**: (Assume a sample IP, e.g., 104.18.40.123)

13. **What domain was resolved during the exfiltration?**

- **Artifact**: DNS Query Logs

- **How to Solve in Magnet AXIOM**: Load the DNS logs. Filter for queries around 2:15 AM in the "Network" or "Logs" tab. Identify the resolved domain.
- **Answer**: competitorcorp.sharepoint.com

14. **What was the volume of outbound traffic during the breach?**

- **Artifact**: NetFlow Data
- **How to Solve in Magnet AXIOM**: Load the NetFlow data. In the "Timeline" view, filter for outbound traffic at 2:15 AM and sum the byte count.
- **Answer**: (Assume 500 MB for this example)

15. **At what exact time did the exfiltration begin?**

- **Artifact**: Packet Captures
- **How to Solve in Magnet AXIOM**: In the "Network" tab, sort HTTPS traffic by timestamp. Find the first packet to **competitorcorp.sharepoint.com**.
- **Answer**: 2:15 AM

Phase 4: Following the Data to the Cloud

16. **What time was the stolen file uploaded to the cloud?**

- **Artifact**: Cloud Service Audit Logs
- **How to Solve in Magnet AXIOM**: Load the SharePoint logs. In the "Logs" tab, filter for upload events and check the timestamp for **drug_trials_data.zip**.
- **Answer**: 2:20 AM

17. **What IP address uploaded the file to SharePoint?**

- **Artifact**: Cloud Service Audit Logs
- **How to Solve in Magnet AXIOM**: From the same upload event in the SharePoint logs, extract the source IP address.
- **Answer**: (Assume John's home IP, e.g., 73.12.45.67)

18. **Does the uploaded file's hash match the recovered .zip file?**

- **Artifact**: File Metadata in Cloud Storage
- ****…**

Here's the continuation of the 25 CTF questions based on the pharmaceutical company data breach case story, designed for

Magnet AXIOM. I'll pick up where I left off (Question 19) and complete the list through Phases 4, 5, and 6. Each question includes the related artifacts, instructions for solving in Magnet AXIOM, and sample answers where applicable.

Phase 4: Following the Data to the Cloud (Continued)

19. **What method was used to upload the file to the cloud?**

- **Artifact**: API Access Logs
- **How to Solve in Magnet AXIOM**: Load the API access logs from the cloud provider into AXIOM. In the "Logs" tab, filter for events around 2:20 AM related to **drug_trials_data.zip**. Look for API call details (e.g., **POST** requests) indicating programmatic access.
- **Answer**: Scripted upload via API

20. **Where does the IP address of the upload geolocate to?**

- **Artifact**: Cloud Service Audit Logs
- **How to Solve in Magnet AXIOM**: Extract the source IP (e.g., 73.12.45.67) from the SharePoint upload event. Use AXIOM's "Connections" or "Search" feature with an integrated geolocation tool (or manually check with MaxMind GeoIP) to trace the IP.
- **Answer**: John Smith's home address (e.g., a city like "Boston, MA")

Phase 5: Uncovering Encrypted Communications

21. **What encryption algorithm was used for the suspect's emails?**

- **Artifact**: Encrypted Email Backups
- **How to Solve in Magnet AXIOM**: Load the email server backup image. In the "File System" view, recover the encrypted email file from unallocated space (e.g., using "Recover Deleted

Files”). Analyze the file header or metadata to identify the encryption type.

- **Answer**: AES-128

22. **What is the weak encryption key used in the communications?**

- **Artifact**: Encryption Keys (Configuration File)
- **How to Solve in Magnet AXIOM**: Load the workstation image. In the “File System” tab, search for configuration files (e.g., **.cfg** or **.ini**) containing key-like strings. Locate the file with **P@ssw0rd123**.
- **Answer**: P@ssw0rd123

23. **What tool can crack the weak encryption key, and how long did it take?**

- **Artifact**: Encryption Keys
- **How to Solve in Magnet AXIOM**: Export the encrypted email and key (**P@ssw0rd123**) from AXIOM. While AXIOM itself doesn’t crack keys, note that an external tool like John the Ripper was used (per the story). Simulate this by documenting the key’s weakness (e.g., common password).
- **Answer**: John the Ripper, cracked in hours (e.g., 3 hours)

24. **What was the content of the decrypted email to the competitor?**

- **Artifact**: Encrypted Email Backups
- **How to Solve in Magnet AXIOM**: After cracking the key externally, re-import the decrypted email into AXIOM. In the “Email” tab, view the message content.
- **Answer**: “The data is yours for $5 million. Upload complete.”

25. **Who was the recipient of the encrypted email?**

- **Artifact**: Encrypted Email Backups
- **How to Solve in Magnet AXIOM**: In the decrypted email from the “Email” tab, check the “To” field or metadata for the recipient’s address or name.
- **Answer**: A competitor executive (e.g., exec@competitorcorp.com)

Phase 6: Resolution and Legal Action

(Note: Phase 6 focuses on reporting and chain of custody, which are less about active discovery in AXIOM but still involve evidence handling. I've adjusted the last question to fit the CTF format.)

25. **Which artifact confirms the exact time the forensic image of the server logs was created?**

- **Artifact**: Chain of Custody Documentation
- **How to Solve in Magnet AXIOM**: Load the chain of custody documentation (e.g., a text or PDF file) into AXIOM. In the "Documents" tab, search for the entry related to the database server log forensic image creation. Extract the timestamp.
- **Answer**: (Assume a sample time, e.g., 1/16/2025 9:00 AM)

Summary of Artifacts and Techniques in Magnet AXIOM

- **Artifacts Used**:
- **Database Access Logs, Authentication Logs, Email Client Data (PST), Registry Artifacts, File System Artifacts (deleted .zip), Packet Captures (PCAP), DNS Query Logs, NetFlow Data, Cloud Service Audit Logs, File Metadata, API Access Logs, Encrypted Email Backups, Encryption Keys, Forensic Report, Chain of Custody Documentation.**
- **Techniques Demonstrated in AXIOM**:
- **Log file analysis (filtering by timestamp/IP), file system forensics (recovering deleted files), registry parsing, network forensics (PCAP analysis, DNS lookups), hash comparison, email analysis, data carving (unallocated space), and documentation review.**

How to Set Up the CTF in Magnet AXIOM

1. **Prepare Evidence Files**:

- **Create sample log files (e.g., database and authentication logs with timestamps/IPs), a PST file with a phishing email, a forensic image with a deleted .zip and registry keys, a PCAP file with HTTPS traffic, DNS/NetFlow logs, cloud audit logs, an encrypted email file, and chain of custody docs.**

2. **Load into AXIOM**:

- **Use AXIOM Process to ingest all evidence files (logs, images, PCAPs, etc.) into a single case.**

3. **Guide Participants**:

- **Provide a brief of the case story and instructions to use AXIOM's "Timeline," "Artifacts," "File System," "Network," and "Search" features to answer each question.**

4. **Validate Answers**:

- **Check participant responses against the predefined answers (e.g., timestamps, IPs, file names) embedded in the evidence.**

This CTF setup mimics a real-world digital forensics investigation, leveraging Magnet AXIOM's capabilities to explore logs, recover files, analyze network traffic, and more. Let me know if you'd like sample evidence files or further refinements!

4. Report Drive Link - Export