

Forensic Investigation Report

Name: Farah Mae Sumajit
Date Completed: April 14, 2025
Case Title: Abnormal File Encryption and Data Movement
System: workstation01.domain.local
Log File: Log.xml
Tools Used:

- JSONFormatter XML Viewer
- Event Viewer in Windows 8 VM
- PowerShell [for simulation & manual log writing]
- Wireshark [packet analysis in simulation]
- Kali Linux [controlled test environment]

1. Incident Summary

A security alert was triggered for potential data exfiltration and unauthorized user actions. The forensic analysis included structured log inspection using **JSONFormatter** and simulated **manual log import and analysis in Windows 8 Event Viewer**. Supplementary validation was conducted through a controlled simulation in a Kali Linux virtual environment. Logs confirm suspicious file access, external network activity, account creation, and command-line reconnaissance.

2. Key Evidence and Analysis

A. File Access: sensitive_file.docx via Notepad

Time: 2025-03-15T14:45Z
User: Administrator
Process: notepad.exe
JSON Tree View:

```
EventID: 4663
ProcessName: C:\Windows\System32\notepad.exe
ObjectName: C:\Users\Public\Documents\sensitive_file.docx
Accesses: ReadData, WriteData
```

Findings:

- Encrypted File Access:** The file **sensitive_file.docx** was accessed with write permissions using **notepad.exe**, which could signal **unauthorized encryption or data manipulation**.
- This suspecting the file may have been **manually manipulated, altered, or prepared for unauthorized use**, such as **encryption or data exfiltration**.
- The access included both **ReadData** and **WriteData**, confirming that the file was not only opened, but **actively modified**.
- The timestamp of **14:45 UTC (2:45 PM)** could fall **outside typical usage hours**, depending on the organization's working schedule, increasing suspicion.
- A follow-up investigation on the system found that the file **sensitive_file.docx was no longer present**, and no encrypted file extensions [such as **.enc**, **.locked**, or **.crypted**] were detected.

Conclusion:

The access to **sensitive_file.docx** represents a **highly suspicious activity**. It was accessed using **Notepad**, a tool not designed to open **.docx** files, and modified under the **Administrator account**. The file's absence on disk further indicates potential **intentional deletion post-access**, likely to **hide unauthorized changes or stage the file for exfiltration**. These indicators suggest **unauthorized manipulation or preparation of sensitive data**, consistent with insider threats or early-stage data breaches.

B. Network Activity : Log-Based Evidence (Windows Security Logs)

Event: Outbound HTTPS Connection to Unknown IP
Time: 2025-03-15T14:50Z
Destination IP: 203.0.113.45
Local IP: 192.168.1.100 [the affected system]
Port: 443
Process: svchost.exe
JSON Tree View:

```
EventID: 5156
DestinationAddress: 203.0.113.45
SourceAddress: 192.168.1.100
Destination Port: 443
Process Name: C:\Windows\System32\svchost.exe
Protocol: TCP
```

Findings:

- Suspicious Timing:** The connection occurred just **5 minutes after the sensitive file [sensitive_file.docx]** was accessed and likely modified.
- Encrypted Channel:** The use of **port 443 (HTTPS)** may indicate **attempted encrypted exfiltration** of sensitive data.
- Suspicious Process Use:** While **svchost.exe** is a legitimate system process, it is **commonly abused by malware** to perform stealthy actions like data transmission.
- Unfamiliar External IP:** The IP **203.0.113.45** does not match any known internal or corporate IP addresses and is **not associated with approved domains or services**.
- No Whitelisting Evidence:** No records were found in allowed domains/IP list for this destination.

Conclusion:

The forensic evidence indicates that the host system made an outbound **encrypted connection (HTTPS)** to an **unfamiliar external IP address [203.0.113.45]** using the **svchost.exe** process. Given the proximity in time to the access of the sensitive document and the nature of the process used, this behavior is highly suggestive of **data exfiltration activity**.

🔍 This outbound connection should be treated as a critical indicator of compromise (IoC), and further threat hunting or memory analysis is advised to determine if the svchost.exe instance was maliciously spawned.

Event: Simulated Packet Capture – Controlled Reproduction

To validate how such encrypted outbound activity appears in real-time, a simulation was performed using **Kali Linux**.

Setup

- Tool Used:** Wireshark
- Command Executed:** curl -v https://example.com
- Capture Interface:** eth0
- Capture Filter:** tcp port 443

Observed Traffic: Frame 261 shows TLSv1.3 Client Hello — the client initiating a secure connection. Frame 264 shows the Server Hello and Change Cipher Spec, where the server responds and agrees on encryption settings. Frames 267 to 284 contain TLSv1.3 Application Data — this is the encrypted payload being transmitted. Frames 285 to 287 show TCP session teardown using FIN and RST flags.

🔄 This mirrors the real-world case where svchost.exe made an outbound connection over port 443

C. New User Account Created – newuser01

Time: 2025-03-15T15:00Z
Account Name: newuser01
Created By: Administrator
JSON Tree View:

```
EventID: 4720
TargetUserName: newuser01
SubjectUserName: Administrator
```

Findings:

- Persistence Mechanism:** A new account (**newuser01**) was created right after suspicious activity. This could be used to maintain access.
- A new user account (**newuser01**) was created shortly after the suspicious network connection.
- The account was created by an administrative user.
- No business justification or prior activity is associated with this user.

Conclusion:

The new user account may have been created as a **backdoor or persistence mechanism** to regain access later.

D. Suspicious Command Execution: cmd.exe /c netstat -an

Time: 2025-03-15T15:05Z
Command: cmd.exe /c netstat -an
Executed By: Administrator via svchost.exe
JSON Tree View:

```
EventID: 4688
CommandLine: cmd.exe /c netstat -an
CreatorProcessName: C:\Windows\System32\svchost.exe
```

Findings:

- Reconnaissance Activity:** Use of **cmd.exe** with **netstat -an** suggests mapping of network connections, a common step in lateral movement or preparation for data exfiltration.
- The command executed (**netstat -an**) checks for active network connections.
- It was launched from **cmd.exe**, which itself was triggered by **svchost.exe** — a **non-standard parent** for **cmd**.
- This action occurred shortly after file access and network activity.

Conclusion:

The user was likely **verifying exfiltration success or scouting open connections**, which is a typical **post-exfiltration recon behavior**.

Screenshots Proof of Log File Opened in Event Viewer and PowerShell

Click each item below to view the corresponding screenshots:

- Evidence of manual log loading**
- A. opened it using Windows powershell
- B. Simulated Packet Capture
- C. opened it using windows powershell

- a. log file found
- a. opened eventviewer

E. Limitations and Analysis Constraints

Only an XML-format event log (**log.xml**) was provided for this investigation. While it contained detailed security event records, it **did not include the actual files** [such as **sensitive_file.docx**] or a system image.

Problem with Log.xml Alone:

The file includes:

- Event metadata (usernames, filenames, timestamps, actions)
- No access to the **actual files** that were accessed or modified

What Is Needed for Complete Forensic Analysis:

Option 1: A Forensic Disk Image (e.g., .E01, .dd, .vhd)

With a full image, tools like Autopsy or FTK Imager could be used to:

- Recover **sensitive_file.docx**
- Examine file content, hashes, and encryption status

Option 2: Access to the Original Machine or VM

What Was Achieved with log.xml:

- Identified Accessed Files
- Confirmed User Activity and Timestamps
- Revealed Suspicious Behavior Patterns
- Built a Forensic Timeline
- Linked Events to Potential Threat Scenario

3. Final Conclusion

Based on the **log.xml** analysis, the evidence clearly indicates:

- Unauthorized access to a sensitive file
- Possible staging and exfiltration of that file
- Creation of a secondary user account likely for persistence
- Network scanning activity to validate open connections

Even in the absence of full disk access or file recovery, the available logs demonstrate a **coordinated, high-risk security incident**, likely involving data theft or insider misuse.

4. Recommendations

