Trend Analysis on Kimsuky Group’s Attacks Using AppleSeed - ASEC

By ATCP

Published: 2023-12-21 · Archived: 2026-04-05 14:26:06 UTC

Known to be supported by North Korea, the Kimsuky threat group has been active since 2013. At first, they attacked North Korea-related research institutes in South Korea before attacking a South Korean energy corporation in 2014. Since 2017, attacks targeting countries other than South Korea have also been observed. **[1]** The group usually launches spear phishing attacks against national defense, defense industries, media, diplomacy, national organizations, and academic sectors. Their attacks aim to steal internal information and technology from organizations. **[2]**

While the Kimsuky group typically uses spear phishing attacks for initial access, most of their recent attacks involve the use of shortcut-type malware in LNK file format. Although LNK malware comprise a large part of recent attacks, cases using JavaScripts or malicious documents are continuing to be detected.

Such attack cases that use JavaScript-type malware usually involve the distribution of AppleSeed which was covered in a past report titled “Analysis Report on Kimsuky Group’s APT Attacks (AppleSeed, PebbleDash)”. **[3]** This report was published in November 2021, but the Kimsuky group is still using AppleSeed in their attacks. In addition to JavaScript, Excel macro malware are also used to install AppleSeed. **[4]**

A notable point about attacks that use AppleSeed is that similar methods of attack have been used for many years with no significant changes to the malware that are used together. Another point of interest is that the group still uses the same Infostealer and RDP Patch malware files that were first identified in 2022, which are used after the group takes control over an infected system.

This post will cover the characteristics of malware used in recent attack cases in comparison to the past report. For example, while the same AppleSeed is still being used, arguments are checked to obstruct analysis and a variant of AppleSeed named AlphaSeed is being used. Another notable fact is that while in the past the group typically used RDP to control the infected system after installing AppleSeed, they are often observed installing Chrome Remote Desktop in recent cases. **[5]**

1. AppleSeed

AppleSeed is a backdoor that can receive the threat actor’s commands from the C&C server and execute the received commands. The threat actor can use AppleSeed to control the infected system. It also offers features such as a downloader that installs additional malware, keylogging and taking screenshots, and stealing information by collecting files from the user system and sending them.

Like in past attack cases, AppleSeed is frequently distributed via a JavaScript dropper. The JavaScript dropper is responsible for installing AppleSeed while simultaneously creating and opening document files such as HWP and

PDF. Due to this, ordinary users are deceived into thinking that a legitimate document file has been opened.

While the installed AppleSeed is similar to the one from the past, since early 2022, AppleSeed has been created by a dropper instead of being installed by JavaScript malware. Not only was a dropper added to the installation process but also a feature that checks the arguments upon malware execution. AppleSeed, which is in DLL format, is installed via the Regsvr32 process, during which the "/i" option is used to pass an argument. AppleSeed checks this argument and proceeds with installation only when it matches a certain string; otherwise, it deletes itself. Because of this, the AppleSeed DLL alone cannot perform malicious behaviors in a sandbox environment.

- **AppleSeed execution argument – example**: regsvr32.exe /s /n /i:1qa2ws4rf "C:\Users\{UserName}\AppData\Roaming\FoxitReader\Service\FoxitReaderUpdate.db"

Period	List of Arguments
Past	**123qweasdzxc** **123qweASDTYU** **12345QWERTY** **1q2w3e4r!** **2wsx!QAZ3edc** **$%ERT345ert**
Recent	**12qw3ed** **1qa2ws4rf**

Table 1. Arguments used when installing AppleSeed

AppleSeed is installed in the "%APPDATA%" or "%PROGRAMDATA%" path. The specific folder and file name are disguised to look like a legitimate program or file such as Antivirus, Chrome, and Adobe. While AppleSeed was often installed in the "%PROGRAMDATA%" path in the past, recently, "%APPDATA%" has been used frequently. The following table is a summary of the various paths AppleSeed was installed in. Paths used in attacks in the past several months were sorted separately.

Period	Installation Path
Past	**%APPDATA%\EastSoft\Control\Service\EastSoftUpdate.dll** **%APPDATA%\ESTsoft\AlLUpdate\AlCommon.dll** **%APPDATA%\ESTsoft\Common\ESTCommon.dll** **%APPDATA%\ESTsoft\Common\ko-kr.dll** **%APPDATA%\ESTsoft\updat\ESTCommon.dll** **%APPDATA%\Microsoft\Windows\Defender\AutoUpdate.dll** **%APPDATA%\Microsoft\Windows\Defender\patch.dll** **%PROGRAMDATA%\Firmware\ESTsoft\Common\ESTCommon.dll** **%PROGRAMDATA%\Firmware\Microsoft\Windows\Defender\AutoUpdate.dll** **%PROGRAMDATA%\Software\Ahnlab\Service\AutoService.dll** **%PROGRAMDATA%\Software\ControlSet\Service\ServiceScheduler.dll**

	%PROGRAMDATA%\Software\Defender\Windows\Update\AutoUpdate.dll **%PROGRAMDATA%\Software\ESTsoft\Common\ESTCommon.dll** **%PROGRAMDATA%\Software\Microsoft\AvastAntiVirus\AvastUpdate.dll** **%PROGRAMDATA%\Software\Microsoft\Avg\AvgSkin.dll** **%PROGRAMDATA%\software\microsoft\iecleaner\cpature\iecaptureclean.dll** **%PROGRAMDATA%\Software\Microsoft\Network\NetworkService.dll** **%PROGRAMDATA%\Software\Microsoft\Printer\PrinterService.dll** **%PROGRAMDATA%\Software\Microsoft\Service\TaskScheduler.dll** **%PROGRAMDATA%\Software\Microsoft\Windows\AutoDefender\UpdateDB.dll** **%PROGRAMDATA%\Software\Microsoft\Windows\AutoPatch\patch.dll** **%PROGRAMDATA%\Software\Microsoft\Windows\Chrome\GoogleUpdate.dll** **%PROGRAMDATA%\Software\Microsoft\WIndows\Defender\AutoCheck.dll** **%PROGRAMDATA%\Software\Microsoft\Windows\Defender\AutoUpdate.dll** **%PROGRAMDATA%\Software\Microsoft\Windows\Defender\update.dll** **%PROGRAMDATA%\Software\Microsoft\Windows\Explorer\FontChecker.dll** **%PROGRAMDATA%\Software\Microsoft\Windows\FontChecker.dll** **%PROGRAMDATA%\Software\Microsoft\Windows\MDF\WDFSync\WDFSync.dll** **%PROGRAMDATA%\Software\Microsoft\Windows\MetaSec\MetaSecurity.dll** **%PROGRAMDATA%\Software\Microsoft\Windows\Patch\patch.dll** **%PROGRAMDATA%\Software\Microsoft\Windows\Protect\ProtectUpdate.dll** **%PROGRAMDATA%\Software\Microsoft\Windows\Secrity\AutoCheck.dll**
Recent	**%APPDATA%\Abode\Service\AdobeService.dll** **%APPDATA%\Acrobatreader\Service\AcrobatReaderUpdate.db** **%APPDATA%\chrome\Service\updategoogle.dll** **%APPDATA%\EastSoft\Control\Service\EastSoftUpdate.dll** **%APPDATA%\FoxitReader\Service\FoxitReaderUpdate.db** **%APPDATA%\ProtectSoft\Update\Service\ProtectSoftUpdate.db**

Table 2. AppleSeed’s installation paths

2. AlphaSeed

AlphaSeed is a malware developed in Golang and supports similar features to AppleSeed such as command execution and infostealing. Due to these similarities and the path name contained in the binary, S2W named this malware AlphaSeed. **[6]**

Though most of its features are similar to those of AppleSeed, there are some differences as well. AlphaSeed was developed in Golang and uses ChromeDP for communications with the C&C server. When receiving commands from the threat actor or stealing collected information, AppleSeed generally used the HTTP protocol or email (SMTP and IMAPS). AlphaSeed also uses email protocols to communicate with the C&C, but instead of directly sending an email, it uses a tool called ChromeDP. The login process is also different: instead of using an ID and password, it uses cookie values to log into certain accounts.

AlphaSeed has been used in attacks since at least October 2022 if not before. Like AppleSeed, AlphaSeed attacks use a JavaScript dropper. Because the binary itself is in DLL format which runs using the Regsvr32 process, the actual installation process is also similar to that of AppleSeed.

The threat actor sometimes installs AlphaSeed and AppleSeed together in the same target system. Although the initial distribution stage in the following case has not been identified, seeing from the fact that AlphaSeed and AppleSeed were installed at almost the same point in time and that certutil.exe was used, it seems that like in most cases, the two malware were installed by a JavaScript dropper.

Process	Module	Behavior	Data
regsvr32.exe	FoxitReaderUpdate.db	Loads DLL	Library Dynamic FoxitReaderUpdate.db
regsvr32.exe	N/A	Creates executable file	Target l9QAEXK.xPVj
certutil.exe	N/A	Creates executable file	Target l9QAEXK.xPVj
regsvr32.exe	N/A	Creates executable file	Target FoxitReaderUpdate.db
certutil.exe	N/A	Creates executable file	Target r166axD.gSHr

The AlphaSeed identified around October 2022 had the path name “E:/golang/src/naver_crawl/” in the binary, while the binary in the version used in attacks from around May 2023 until recently contained the path “E:/Go_Project/src/alpha/naver_crawl_spy/”.

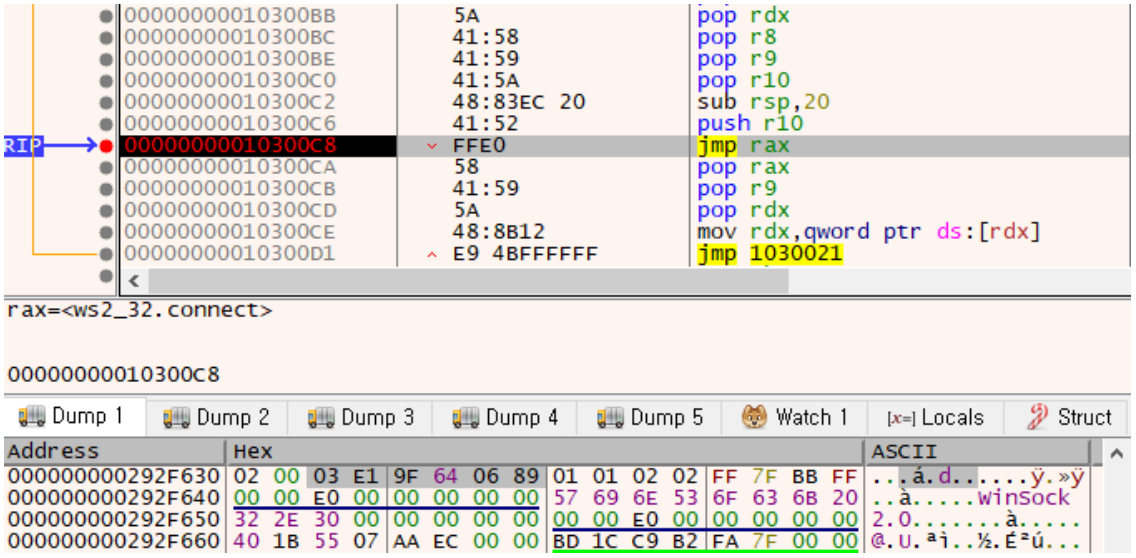
Address	Length	Type	String
.rdata:000000020172A21A	0000002F	C	E:/golang/src/naver_crawl/lib/logger/logger.go
.rdata:000000020172A618	0000002F	C	E:/golang/src/naver_crawl/lib/funcs/command.go
.rdata:000000020172A647	0000002B	C	E:/golang/src/naver_crawl/lib/funcs/enc.go
.rdata:000000020172A672	0000002A	C	E:/golang/src/naver_crawl/lib/funcs/fs.go
.rdata:000000020172A69C	0000002D	C	E:/golang/src/naver_crawl/lib/funcs/funcs.go
.rdata:000000020172A6C9	0000002C	C	E:/golang/src/naver_crawl/lib/funcs/pack.go
.rdata:000000020172A6F5	0000002C	C	E:/golang/src/naver_crawl/lib/funcs/time.go
.rdata:000000020172A721	0000002B	C	E:/golang/src/naver_crawl/lib/funcs/zip.go
.rdata:000000020172A74C	0000002A	C	E:/golang/src/naver_crawl/lib/def/info.go
.rdata:000000020172A776	0000002A	C	E:/golang/src/naver_crawl/lib/def/dirs.go
.rdata:000000020172DC40	00000034	C	E:/golang/src/naver_crawl/agent/chromium/browser.go
.rdata:000000020172DC74	0000002D	C	E:/golang/src/naver_crawl/lib/wapi/user32.go
.rdata:000000020172E33C	00000030	C	E:/golang/src/naver_crawl/agent/agent/action.go
.rdata:000000020172E36C	00000031	C	E:/golang/src/naver_crawl/agent/agent/klogger.go
.rdata:000000020172E39D	0000002F	C	E:/golang/src/naver_crawl/agent/agent/agent.go
.rdata:000000020172E3CC	0000002F	C	E:/golang/src/naver_crawl/agent/agent/setup.go
.rdata:000000020172E3FB	0000002D	C	E:/golang/src/naver_crawl/agent/agent/cmd.go
.rdata:000000020172E428	0000002F	C	E:/golang/src/naver_crawl/agent/agent/email.go
.rdata:000000020172E482	00000032	C	E:/golang/src/naver_crawl/agent/agent/sshotter.go
.rdata:000000020172E69A	00000028	C	E:/golang/src/naver_crawl/agent/main.go
.rdata:000000020172E6C2	00000027	C	E:/golang/src/naver_crawl/agent/dll.go

Address	Length	Type	String
.rdata:000000020173E6F4	0000003D	C	E:/Go_Project/src/alpha/naver_crawl_spy/lib/logger/logger.go
.rdata:000000020173EB00	0000003D	C	E:/Go_Project/src/alpha/naver_crawl_spy/lib/funcs/command.go
.rdata:000000020173EB3D	00000039	C	E:/Go_Project/src/alpha/naver_crawl_spy/lib/funcs/enc.go
.rdata:000000020173EB76	00000038	C	E:/Go_Project/src/alpha/naver_crawl_spy/lib/funcs/fs.go
.rdata:000000020173EBAE	0000003B	C	E:/Go_Project/src/alpha/naver_crawl_spy/lib/funcs/funcs.go
.rdata:000000020173EC37	0000003A	C	E:/Go_Project/src/alpha/naver_crawl_spy/lib/funcs/pack.go
.rdata:000000020173EC71	0000003A	C	E:/Go_Project/src/alpha/naver_crawl_spy/lib/funcs/time.go
.rdata:000000020173ECAB	00000039	C	E:/Go_Project/src/alpha/naver_crawl_spy/lib/funcs/zip.go
.rdata:000000020173ECE4	00000038	C	E:/Go_Project/src/alpha/naver_crawl_spy/lib/def/info.go
.rdata:000000020173ED1C	00000038	C	E:/Go_Project/src/alpha/naver_crawl_spy/lib/def/dirs.go
.rdata:000000020174266B	00000042	C	E:/Go_Project/src/alpha/naver_crawl_spy/agent/chromium/browser.go
.rdata:000000020174426AD	0000003B	C	E:/Go_Project/src/alpha/naver_crawl_spy/lib/wapi/user32.go
.rdata:0000000201742DDB	0000003E	C	E:/Go_Project/src/alpha/naver_crawl_spy/agent/agent/action.go
.rdata:0000000201742E19	0000003F	C	E:/Go_Project/src/alpha/naver_crawl_spy/agent/agent/klogger.go
.rdata:0000000201742E58	0000003D	C	E:/Go_Project/src/alpha/naver_crawl_spy/agent/agent/agent.go
.rdata:0000000201742E95	0000003D	C	E:/Go_Project/src/alpha/naver_crawl_spy/agent/agent/setup.go
.rdata:0000000201742ED2	0000003B	C	E:/Go_Project/src/alpha/naver_crawl_spy/agent/agent/cmd.go
.rdata:0000000201742F0D	0000003D	C	E:/Go_Project/src/alpha/naver_crawl_spy/agent/agent/email.go
.rdata:0000000201742F4A	00000040	C	E:/Go_Project/src/alpha/naver_crawl_spy/agent/agent/sshotter.go
.rdata:000000020174317C	00000036	C	E:/Go_Project/src/alpha/naver_crawl_spy/agent/main.go
.rdata:0000000201743182	00000035	C	E:/Go_Project/src/alpha/naver_crawl_spy/agent/dll.go

3. Meterpreter

Metasploit is a penetration testing framework. They are tools that can be used to inspect security vulnerabilities for networks and systems of companies and organizations, providing various features for each penetration test stage. Meterpreter is a backdoor provided by Metasploit and is used to control infected systems.

The Kimsuky group has often used Meterpreter in attack processes involving AppleSeed. **[7]** In the first half of 2023, Meterpreter Stager developed in Golang was identified. **[8]** However, the recently distributed version of Meterpreter was self-developed using C++ instead of Golang.



4. VNC – TightVNC, HVNC (TinyNuke)

Aside from using RDP, the Kimsuky group also develops VNC malware to control the infected system. [9] There are two types that have been used since the initial discovery: TightVNC and HVNC.

TightVNC is an open-source VNC utility, and the threat actor customizes it to use it. The Kimsuky group distributes TightVNC which is customized to allow the Reverse VNC feature to be used independently in the infected environment without installing a service. As such, simply running tvnserver will allow the attacker to access tvnviewer that operates on the C&C server and gain control of the screen of the infected system.

TinyNuke, also known as Nuclear Bot, is a banking malware discovered in 2016. It includes features such as HVNC (HiddenDesktop/VNC), reverse SOCKS4 proxy, and form grabbing. As its source code was disclosed in 2017, TinyNuke is used by various threat actors, and out of its features, the HVNC and reverse SOCKS4 proxy features are partially borrowed by other malware such as AveMaria and BitRAT.

Among the various features offered by TinyNuke, the Kimsuky group only enables the HVNC feature before distributing it. TinyNuke uses the string “AVE_MARIA” for verification when establishing an HVNC communication session between server and client. The Kimsuky group either uses this string without modification or uses the string “LIGHT’S BOMB” instead. Since the first half of 2022, the string “Alpha’s nuke” has been used, which was also found in recently identified versions.

```
if ( SetThreadDesktop(hDesktop) )
{
  if ( send(v1, "Alpha's nuke", 13, 0) > 0 )
  {
    *(_DWORD *)buf = 0;
    if ( send(v1, buf, 4, 0) > 0 )
    {
      SetProcessDPIAware();
      Sleep(0x64u);
      v2 = recv;
      while ( recv(v1, v8, 4, 0) > 0 )
      {
        if ( v2(v1, v10, 4, 0) <= 0 )
```

5. Conclusion

The Kimsuky threat group is constantly launching spear phishing attacks against South Korean users. The group usually distributes malware disguised as document files attached to emails. When users run these attachments, they may lose control over their system.

The Kimsuky threat group uses AppleSeed, Meterpreter, and VNC malware to seize control over infected systems, and even abuses the RDP remote desktop service included in Windows. Recently, the group has also been observed using the remote desktop feature in Google Chrome.

Users must carefully check the senders of emails and refrain from opening files from unknown sources. Users should also apply the latest patch for OS and programs such as internet browsers, and update V3 to the latest version to prevent malware infection in advance.

File Detection
– Backdoor/Win.AppleSeed.C5565172 (2023.12.21.00)
– Backdoor/Win.AppleSeed.R626582 (2023.12.04.02)
– Malware/Win.Agent.R628198 (2023.12.18.02)
– Trojan/Win.VNC.C5563987 (2023.12.18.03)
– Trojan/Win.TinyNuke.C5563988 (2023.12.18.03)
– Backdoor/Win.AppleSeed.C5563985 (2023.12.18.03)
– Backdoor/Win.AlphaSeed.R628550 (2023.12.21.03)
– Malware/Win.Agent.R628198 (2023.12.18.02))
– Backdoor/Win.AlphaSeed.R628552 (2023.12.21.03)
– Backdoor/Win.Iedoor.R626024 (2023.11.29.02)
– Backdoor/Win.Iedoor.R625563 (2023.11.27.03)
– Backdoor/Win.AppleSeed.R625539 (2023.11.27.02)
– Dropper/Win.AppleSeed.R625538 (2023.11.27.02)
– Backdoor/Win.AppleSeed.R624029 (2023.11.24.00)
– Backdoor/Win.AppleSeed.R625553 (2023.11.27.03)
– Backdoor/Win.AppleSeed.C5502219 (2023.10.08.03)

Behavior Detection
– Execution/MDP.Regsvr32.M4470

MD5

02843206001cd952472abf5ae2b981b2

0cce02d2d835a996ad5dfc0406b44b01

153383634ee35b7db6ab59cde68bf526

1f7d2cbfc75d6eb2c4f2b8b7a3eec1bf

232046aff635f1a5d81e415ef64649b7

Additional IOCs are available on AhnLab TIP.

URL

http[:]//104[.]168[.]145[.]83[:]993/

http[:]//107[.]148[.]71[.]88[:]993/

http[:]//159[.]100[.]6[.]137[:]993/

http[:]//38[.]110[.]1[.]69[:]993/

http[:]//45[.]114[.]129[.]138[:]33890/

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/60054/