Warning Against Distribution of Malware Disguised as Research Papers (Kimsuky Group) - ASEC

By ATCP

Published: 2025-06-11 · Archived: 2026-04-05 19:52:28 UTC

Recently, the AhnLab SEcurity intelligence Center (ASEC) confirmed the phishing email attack case where the Kimsuky group disguised their attack as a request for paper review from a professor. The email prompted the recipient to open a HWP document file with a malicious OLE object attachment. The document was password-protected, and the recipient had to enter the password provided in the email body to view the document. Upon opening the document, six files were automatically created in the **%TEMP% (temporary folder)** path. To further prompt the user to check the content, the document body included a **"More…"** phrase, which contained a hyperlink that executed the **"peice.bat"** file, one of the six files created. The table below shows the list of files created upon opening the document.

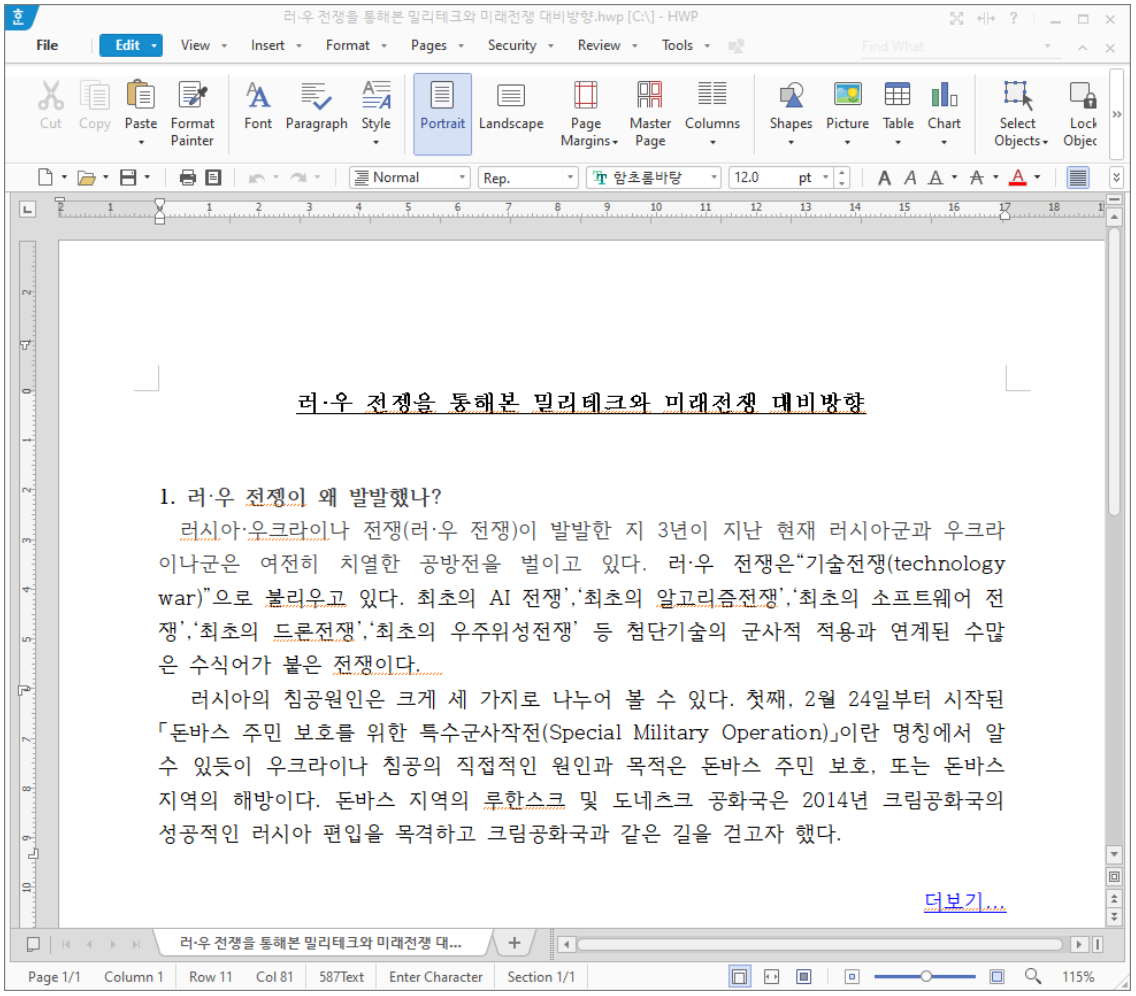

Figure 1. HWP document file containing malicious OLE object
(The content of the HWP file describes main causes and purposes of the Russo-Ukraine War.)

No.	File Name	Feature
1	app.db	EXE File with a Valid Signature
2	get.db	PowerShell script that collects process lists, installed AV information, and downloads additional files
3	hwp_doc.db	Legitimate bait Korean document file
4	mnfst.db	Configuration file read by File 1 (app.db)
5	sch_0514.db	XML scheduler file that executes get.db twice every 12 minutes
6	peice.bat	Performing tasks on files 1 to 5

Table 1. Created files

When the **“peice.bat”** file is executed, it performs the behavior of copying the created files to a specific path. The exact behavior is as follows:

- Delete a HWP document file with a malicious OLE object
- Change the name of the file 3 to **“Military Technology and Future War Direction Seen Through the Russo-Ukrainian War.hwp”** and open the file.
- Register the file 5 as a scheduler under the name **“GoogleTransltatorExtendeds”**
- Copy File 1 as “cool.exe” to the “C:\Users\Public\Music\” directory
- Copy the file 4 to the path “C:\Users\Public\Music\” and rename it to **“cool.exe.manifest”**
- Copy the file 2 as “template.ps1” to the “C:\Users\Public\Music\” directory.

The file 5, “sch_0514.db,” is a scheduler XML file configured to execute the file 1, “cool.exe,” every 12 minutes. When “cool.exe” is executed by the scheduler, the executable reads the fourth file, “cool.exe.manifest.” This file contains BASE64-encoded data located between the “<!–BEGIN_VBSEDIT_DATA” and “END_VBSEDIT_DATA–>” strings. This data is extracted, decoded, and then executed. This process involves a VBScript that executes the file 2, “C:\Users\Public\Music\template.ps1.”

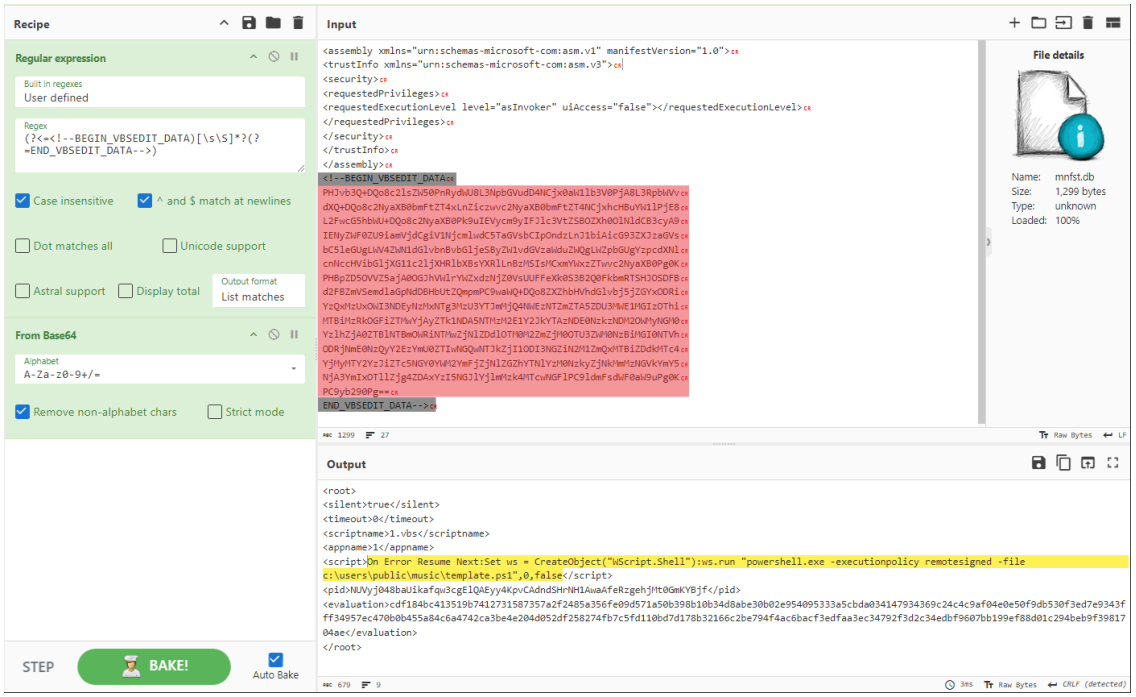

Figure 2. (Top) Original configuration file (Bottom) Decoded configuration file

“template.ps1” collects the process list and installed antivirus (AV) information on the user’s system, saves them in the format **“park_year_month_day_hour_minute_info.ini”**, and sends it to the threat actor’s Dropbox. The file **“park_test.db_sent”** is then downloaded to the path **“C:\Users\Public\Music\pol.bat”** and executed. However, the file was not available for analysis.

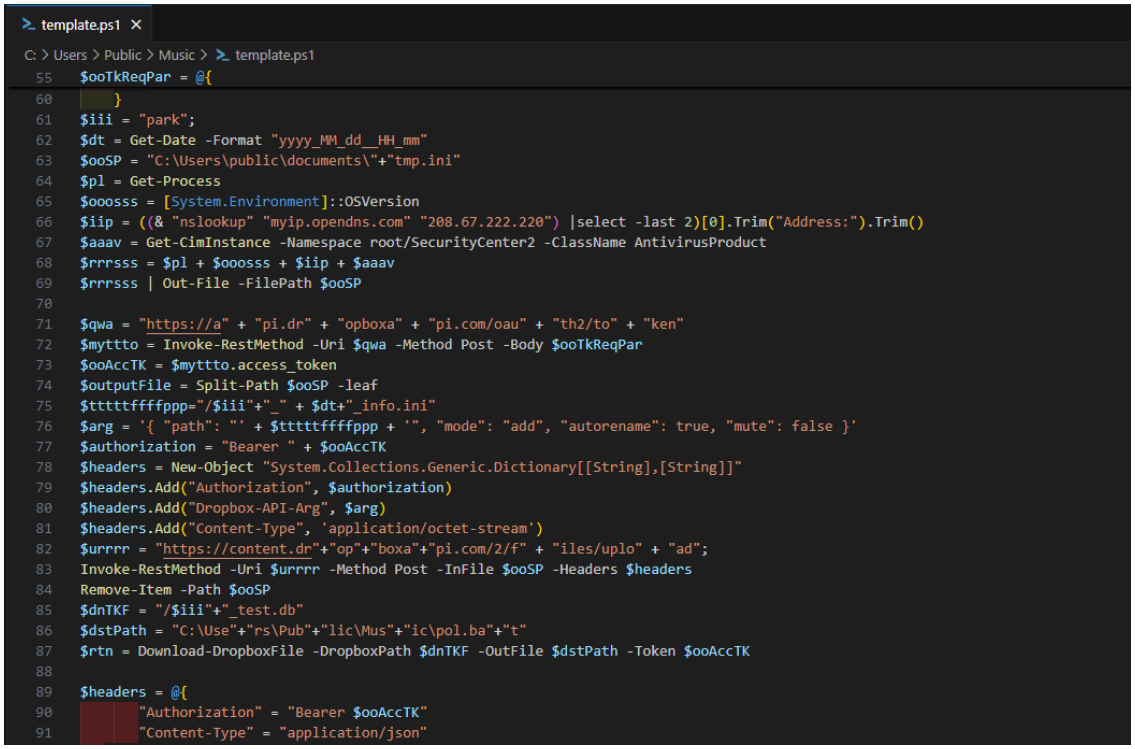

Figure 3. Part of the template.ps1 code

A file named **“template.ps1”** was collected, which is different from the file used in the above case. This file downloads the file named **“jsg_test.db_sent”** and saves it in the path **“C:\Users\Public\Music\1.bat”.** The analysis revealed that the file named **“1.bat”** was successfully downloaded. The C2 then performed additional file download and copy behaviors. For more information, refer to Tables 2 and 3 below.

No.	Behavior
1	**Download 6 files named “myapp, mnfst, attach, sch_0, vpost, bimage” from C2 using curl**
2	Register the **“bimage”** file to the scheduler under the name **User_Feed_Synchronization-{0DDC1BD9-E733-425C-B92B-ABAC149AB11232}”**
3	Copy the files **“myapp, mnfst, attach”** to the following paths with the following names: **%APPDATA%\microsoft\windows** myapp” : **default_an.vbs** mnfst” : **default_an.ps1** attach” : **default_an.exe**

No.	Behavior
4	Copy the files **“sch_0”** and **“vpost”** to the **%APPDATA%\AnyDesk** path with the following names sch_0″ : service.conf vpost” : system.conf

Table 2. Features of “1.bat”

Number	File Name (Changed File Name)	Feature
1	myapp **(default_an.vbs)**	VBScript that executes **C:\Users\Public\Videos\default_an.ps1**
2	mnfst **(default_an.ps1)**	Execute default_an.exe every 5 seconds and hide the window and tray icon
3	attach **(default_an.exe)**	Normal AnyDesk file
4	sch_0 **(service.conf)**	Configuration file containing the key and salt required for AnyDesk connection and the hashed password with a hash technique
5	vpost **(system.conf)**	File containing AnyDesk connection ID and settings
6	bimage	Scheduler XML file that executes default_an.vbs

Table 3. Features by file

Although all files downloaded by the “1.bat” file are copied to the %APPDATA%\Microsoft\Windows path, the downloaded files are actually set to the C:\Users\Public\Videos path and are not executed flexibly. This suggests that the file was created incorrectly according to the threat actor’s intentions. If all file paths are assumed to be set identically, when the “default_an.ps1” file is executed by the scheduler, it executes the legitimate AnyDesk executable file, “default_an.exe”. At this point, the script hides the AnyDesk tray icon and window to prevent the user from noticing it.

When AnyDesk is launched, it creates configuration files such as “service.conf” and “system.conf” in the “%APPDATA%\Anydesk” folder. The threat actor’s intention seems to be to replace these configuration files with their own files in order to access the user’s system.

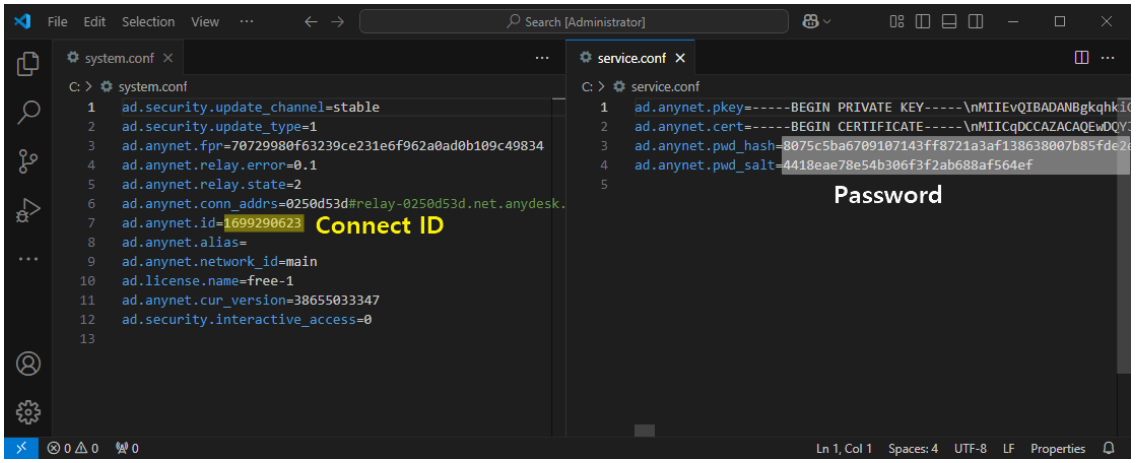

Figure 4. AnyDesk configuration files

Even when AnyDesk is running normally, the tray icon and window are hidden by the PowerShell script, making it difficult for users to realize that remote control is taking place without directly checking the process list. The left image in Figure 5 below shows the tray icon being displayed, and the right image shows the tray icon, connection window, and AnyDesk main screen not being displayed.

※ In the image below, the connection ID in the configuration file is set to “1 699 290 623”, but the password is unknown. Therefore, the image shows the connection after changing the password arbitrarily.

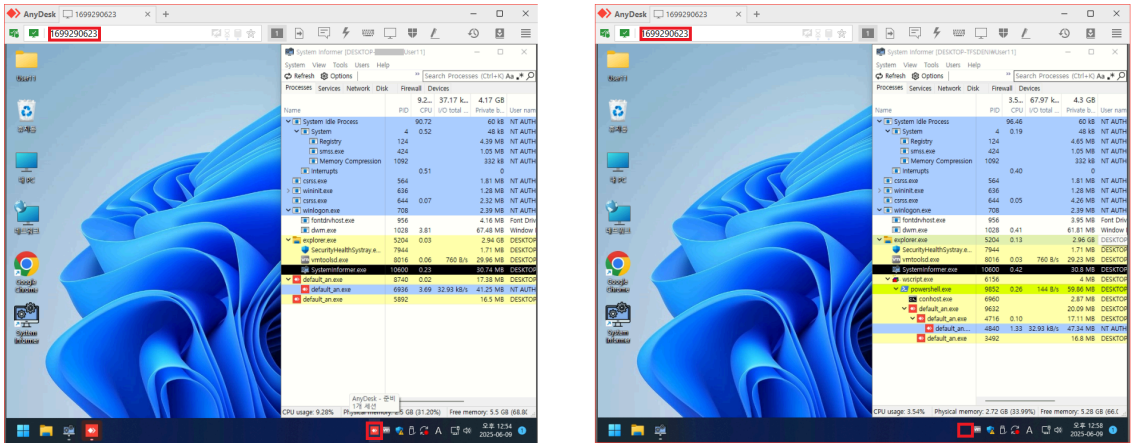

Figure 5. (Left) AnyDesk connection screen normally displayed (Right) Hidden AnyDesk connection screen

The Kimsuky group has been continuously launching APT attacks, impersonating others to target specific individuals. Recently, there has been a growing trend of threat actors exploiting legitimate software in their attacks or using shared drives like Google and Dropbox as C2 (Command and Control) storage. As shown in this case, these APT attacks are often disguised as topics related to the victim’s work or interests, increasing the risk of malware infection to users. For this reason, users are advised to refrain from opening files from unknown sources and always check the file extension before opening them.

MD5

50d4e3470232d90718d61e760a7a62fb

6a84a14dd79396f85abd0e7a536d97fc

7183295e6311ebaaea7794d8123a715e

79573759208d78816316546a9c1f0aec

873579b92d618bf2ed3f67b7a01d7f7a

Additional IOCs are available on AhnLab TIP.

URL

http[:]//103[.]149[.]98[.]230/pprb/0220_pprb_man_1/an/d[.]php?newpa=myapp

http[:]//103[.]149[.]98[.]230/pprb/0220_pprb_man_1/an/d[.]php?newpa=myappfest

https[:]//niva[.]serverpit[.]com/anlab/d[.]php?newpa=attach

https[:]//niva[.]serverpit[.]com/anlab/d[.]php?newpa=bimage

https[:]//niva[.]serverpit[.]com/anlab/d[.]php?newpa=mnfst

Additional IOCs are available on AhnLab TIP.

FQDN

Additional IOCs are available on AhnLab TIP.

IP

103[.]130[.]212[.]116

103[.]149[.]98[.]230

Additional IOCs are available on AhnLab TIP.
