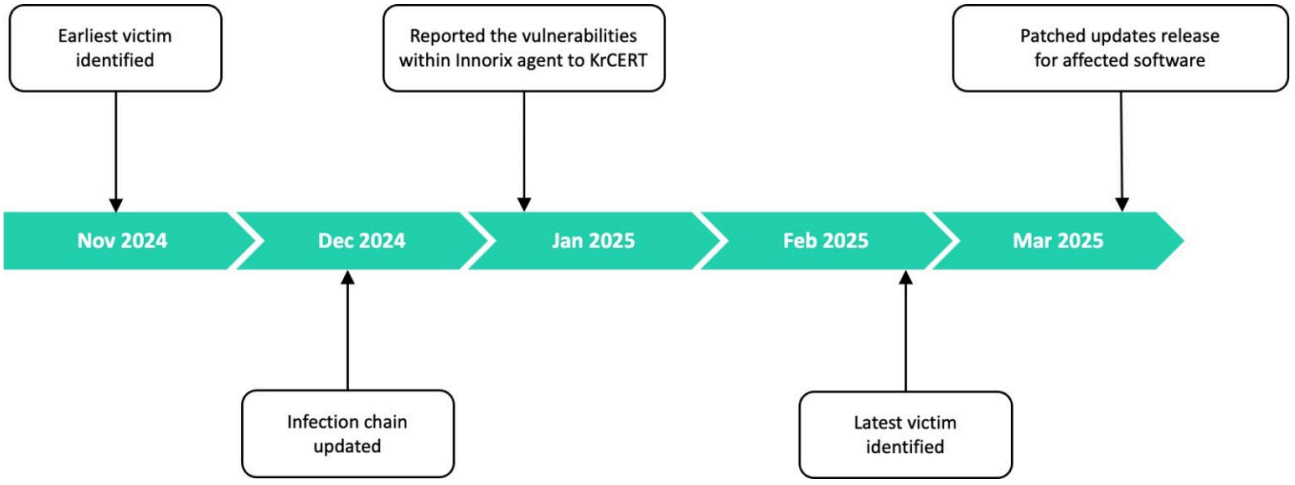
Operation SyncHole: Lazarus APT goes back to the well

By Sojun Ryu

Published: 2025-04-24 · Archived: 2026-04-05 19:47:10 UTC

We have been tracking the latest attack campaign by the Lazarus group since last November, as it targeted organizations in South Korea with a sophisticated combination of a watering hole strategy and vulnerability exploitation within South Korean software. The campaign, dubbed “Operation SyncHole”, has impacted at least six organizations in South Korea’s software, IT, financial, semiconductor manufacturing, and telecommunications industries, and we are confident that many more companies have actually been compromised. We immediately took action by communicating meaningful information to the Korea Internet & Security Agency (KrCERT/CC) for rapid action upon detection, and we have now confirmed that the software exploited in this campaign has all been updated to patched versions.



Timeline of the operation

Our findings in a nutshell:

- At least six South Korean organizations were compromised by a watering hole attack combined with exploitation of vulnerabilities by the Lazarus group.
- A one-day vulnerability in Innorix Agent was also used for lateral movement.
- Variants of Lazarus’ malicious tools, such as ThreatNeedle, Agamemnon downloader, wAgent, SIGNBT, and COPPERHEDGE, were discovered with new features.

Background

The initial infection was discovered in November of last year when we detected a variant of the ThreatNeedle backdoor, one of the Lazarus group’s flagship malicious tools, used against a South Korean software company. We found that the malware was running in the memory of a legitimate SyncHost.exe process, and was created as a subprocess of Cross EX, legitimate software developed in South Korea. This potentially was the starting point for

the compromise of further five organizations in South Korea. Additionally, according to a recent security advisory posted on the KrCERT website, there appear to be recently patched vulnerabilities in Cross EX, which were addressed during the timeframe of our research.

In the South Korean internet environment, the online banking and government websites require the installation of particular security software to support functions such as anti-keylogging and certificate-based digital signatures. However, due to the nature of these software packages, they constantly run in the background to interact with the browser. The Lazarus group shows a strong grasp of these specifics and is using a South Korea-targeted strategy that combines vulnerabilities in such software with watering hole attacks. The South Korean National Cyber Security Center published its own security advisory in 2023 against such incidents, and also published additional joint security advisories in cooperation with the UK government.

Cross EX is designed to enable the use of such security software in various browser environments, and is executed with user-level privileges except immediately after installation. Although the exact method by which Cross EX was exploited to deliver malware remains unclear, we believe that the attackers escalated their privileges during the exploitation process as we confirmed the process was executed with high integrity level in most cases. The facts below led us to conclude that a vulnerability in the Cross EX software was most likely leveraged in this operation.

- The most recent version of Cross EX at the time of the incidents was installed on the infected PCs.
- Execution chains originating from the Cross EX process that we observed across the targeted organizations were all identical.
- The incidents that saw the Synchost process abused to inject malware were concentrated within a short period of time: between November 2024 and February 2025.

In the earliest attack of this operation, the Lazarus group also exploited another South Korean software product, Innorix Agent, leveraging a vulnerability to facilitate lateral movement, enabling the installation of additional malware on a targeted host of their choice. They even developed malware to exploit this, avoiding repetitive tasks and streamlining processes. The exploited software, Innorix Agent (version 9.2.18.450 and earlier), was previously abused by the Andariel group, while the malware we obtained targeted the more recent version 9.2.18.496.

While analyzing the malware's behavior, we discovered an additional arbitrary file download zero-day vulnerability in Innorix Agent, which we managed to detect before any threat actors used it in their attacks. We reported the issues to the Korea Internet & Security Agency (KrCERT) and the vendor. The software has since been updated with patched versions.

Installing malware through vulnerabilities in software exclusively developed in South Korea is a key part of the Lazarus group's strategy to target South Korean entities, and we previously disclosed a similar case in 2023, as did ESET and KrCERT.

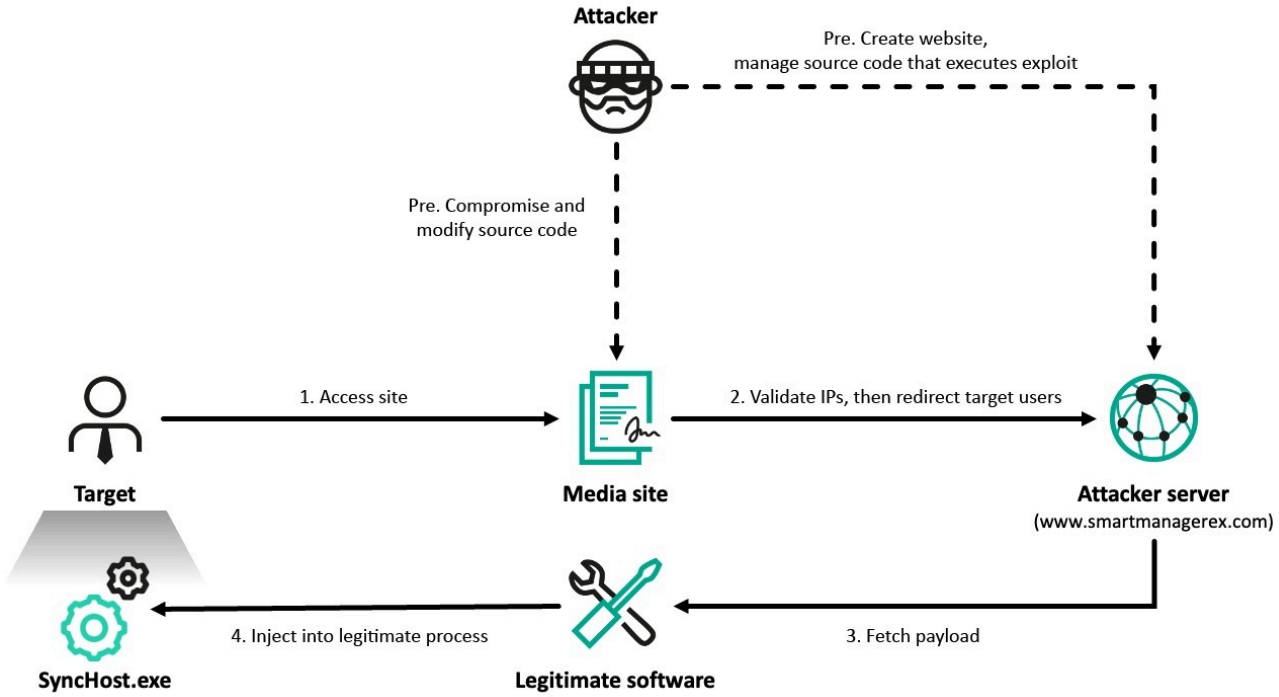
Initial vector

The infection began when the user of a targeted system accessed several South Korean online media sites. Shortly after visiting one particular site, the machine was compromised by the ThreatNeedle malware, suggesting that the

site played a key role in the initial delivery of the backdoor. During the analysis, it was discovered that the infected system was communicating with a suspicious IP address. Further examination revealed that this IP hosted two domains (T1583.001), both of which appeared to be hastily created car rental websites using publicly available HTML templates.

Appearance of www.smartmanagerex[.]com

The first domain, www.smartmanagerex[.]com, seemed to be masquerading as software provided by the same vendor that distributes Cross EX. Based on these findings, we reconstructed the following attack scenario.

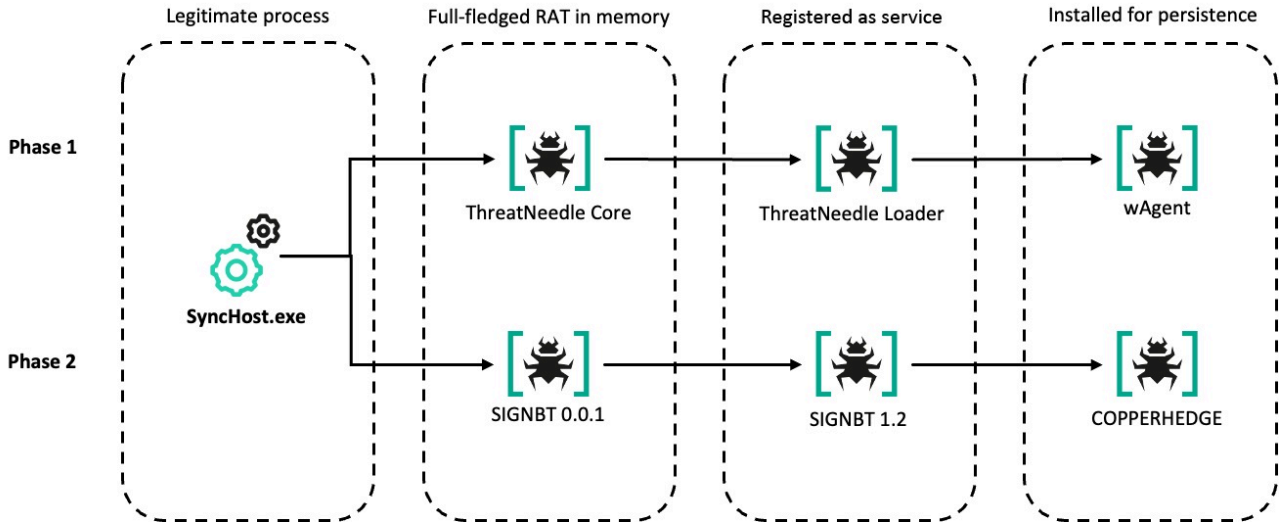


Attack flow during initial compromise

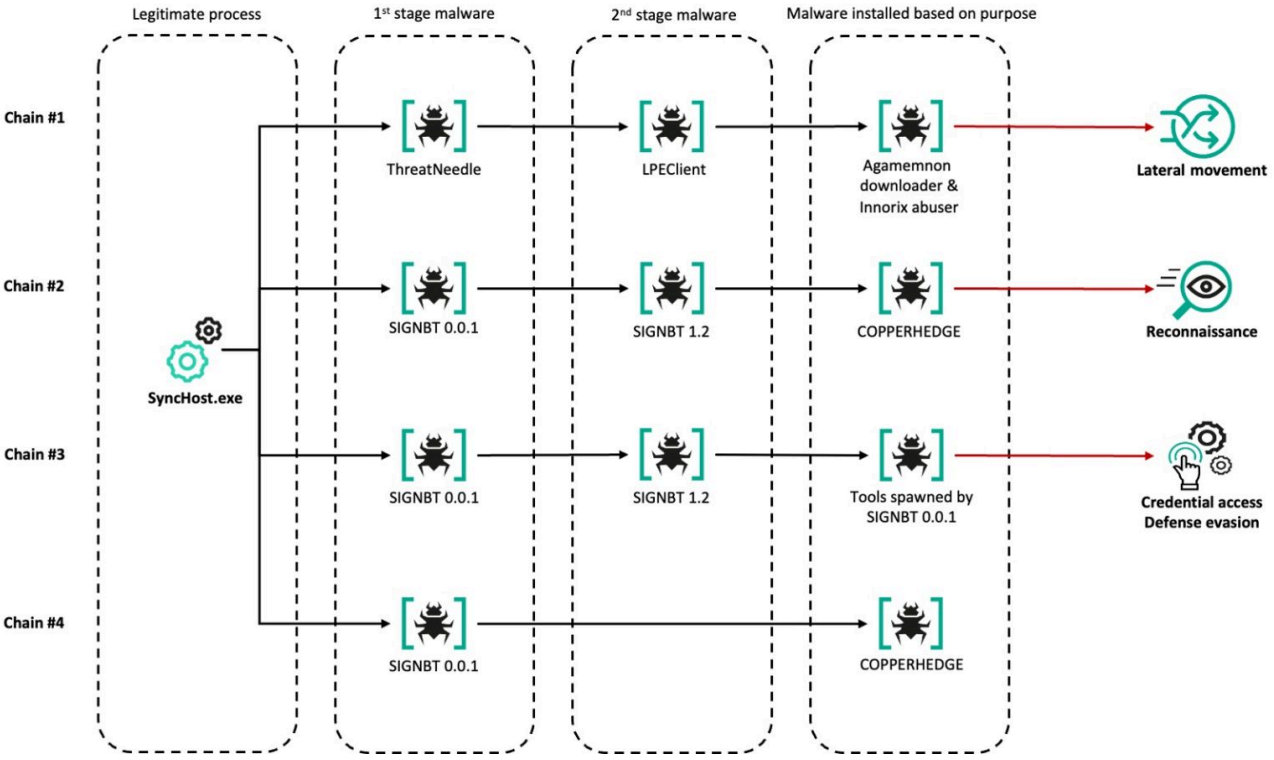
Given that online media sites are typically visited quite frequently by a wealth of users, the Lazarus group filters visitors with a server-side script and redirects desired targets to an attacker-controlled website (T1608.004). We assess with medium confidence that the redirected site may have executed a malicious script (T1189), targeting a potential flaw in Cross EX (T1190) installed on the target PC, and launching malware. The script then ultimately executed the legitimate SyncHost.exe and injected a shellcode that loaded a variant of ThreatNeedle into that process. This chain, which ends with the malware being injected into SyncHost.exe, was common to all of the affected organizations we identified, meaning that the Lazarus group has conducted extensive operations against South Korea over the past few months with the same vulnerability and the same exploit.

Execution flow

We have divided this operation into two phases based on the malware used. The first phase focused primarily on the execution chain involving ThreatNeedle and wAgent. It was then followed by the second phase which involved the use of SIGNBT and COPPERHEDGE.



We derived a total of four different malware execution chains based on these phases from at least six affected organizations. In the first infection case, we found a variant of the ThreatNeedle malware, but in subsequent attacks, the SIGNBT malware took its place, thus launching the second phase. We believe this is due to the quick and aggressive action we took with the first victim. In subsequent attacks, the Lazarus group introduced three updated infection chains including SIGNBT, and we observed a wider range of targets and more frequent attacks. This suggests that the group may have realized that their carefully prepared attacks had been exposed, and extensively leveraged the vulnerability from then on.



Chains of infection across the operation

First-phase malware

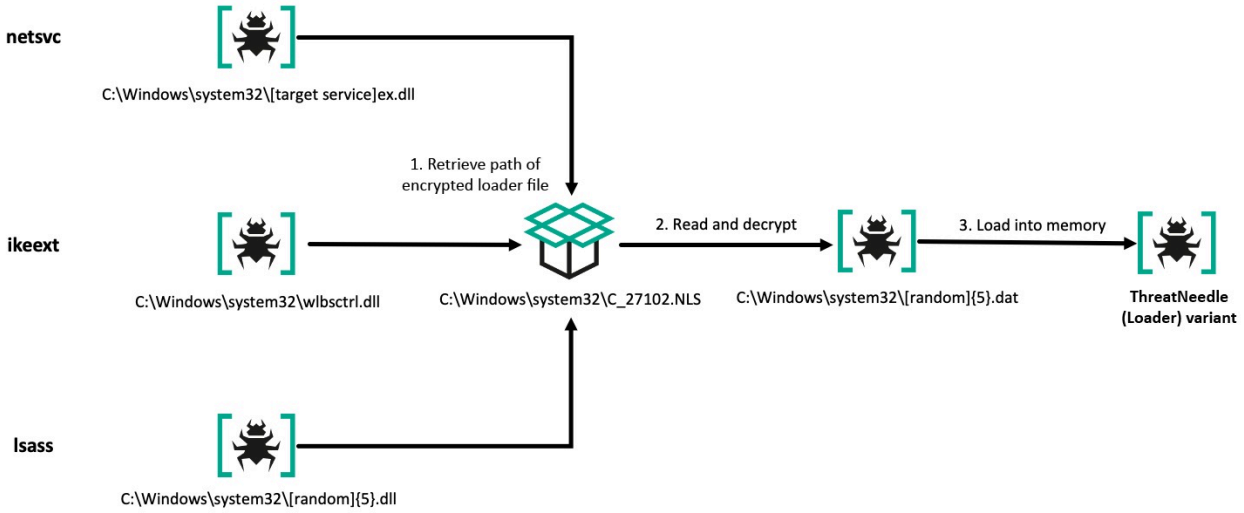
In the first infection chain, many updated versions of the malware previously used by the Lazarus group were used.

Variant of ThreatNeedle

The ThreatNeedle sample used in this campaign was also referred to as “ThreatNeedleTea” in a research paper published by ESET; we believe this is an updated version of the early ThreatNeedle. However, the ThreatNeedle seen in this attack had been modified with additional features.

This version of ThreatNeedle is divided into a Loader and Core samples. The Core version retrieves five configuration files from C_27098.NLS to C_27102.NLS, and contains a total of 37 commands. The Loader version, meanwhile, references only two configuration files and implements only four commands.

The Core component receives a specific command from the C2, resulting in an additional loader file being created for the purpose of persistence. This file can be disguised as the ServiceDLL value of a legitimate service in the netsvcs group (T1543.003), the IKEEXT service (T1574.001), or registered as a Security Service Provider (SSP) (T1547.005). It ultimately loads the ThreatNeedle Loader component.



Behavior flow to load ThreatNeedle Loader by target service

The updated ThreatNeedle generates a random key pair based on the Curve25519 algorithm (T1573.002), sends the public key to the C2 server, and then receives the attacker’s public key. Finally, the generated private key and the attacker’s public key are scalar-operated to create a shared key, which is then used as the key for the ChaCha20 algorithm to encrypt the data (T1573.001). The data is sent and received in JSON format.

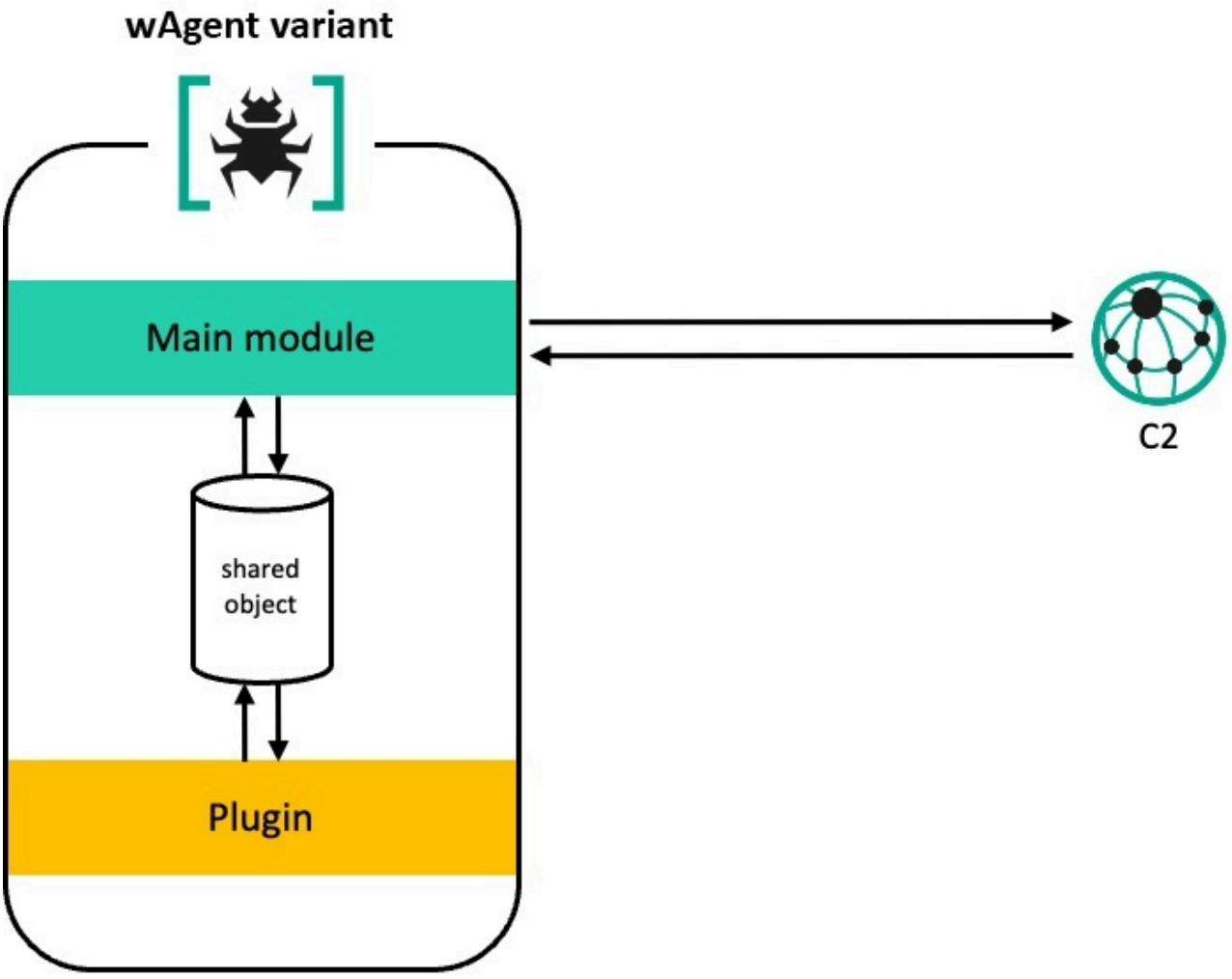
LPEClient

LPEClient is a tool known for victim profiling and payload delivery (T1105) that has previously been observed in attacks on defense contractors and the cryptocurrency industry. We disclosed that this tool had been loaded by SIGNBT when we first documented SIGNBT malware. However, we did not observe LPEClient being loaded by SIGNBT in this campaign. It was only loaded by the variant of ThreatNeedle.

Variant of wAgent

In addition to the variant of ThreatNeedle, a variant of the wAgent malware was also discovered in the first affected organization. wAgent is a malicious tool that we documented in 2020, and a similar version was mentioned in Operation GoldGoblin by KrCERT. The origin of its creation is still shrouded in mystery, but we discovered that the wAgent loader was disguised as liblzma.dll and executed via the command line rundll32.exe c:\Programdata\intel\util.dat, afunix 1W2-UUE-ZNO-B99Z (T1218.011). The export function retrieves the given filename 1W2-UUE-ZNO-B99Z in C:\ProgramData, which also serves as the decryption key. After converting this filename into wide bytes, it uses the highest 16 bytes of the resulting value as the key for the AES-128-CBC algorithm and decrypts (T1140) the contents of the file located in C:\ProgramData (T1027.013). The upper four bytes of the decrypted data subsequently represent the size of the payload (T1027.009), which we identified as an updated version of the wAgent malware.

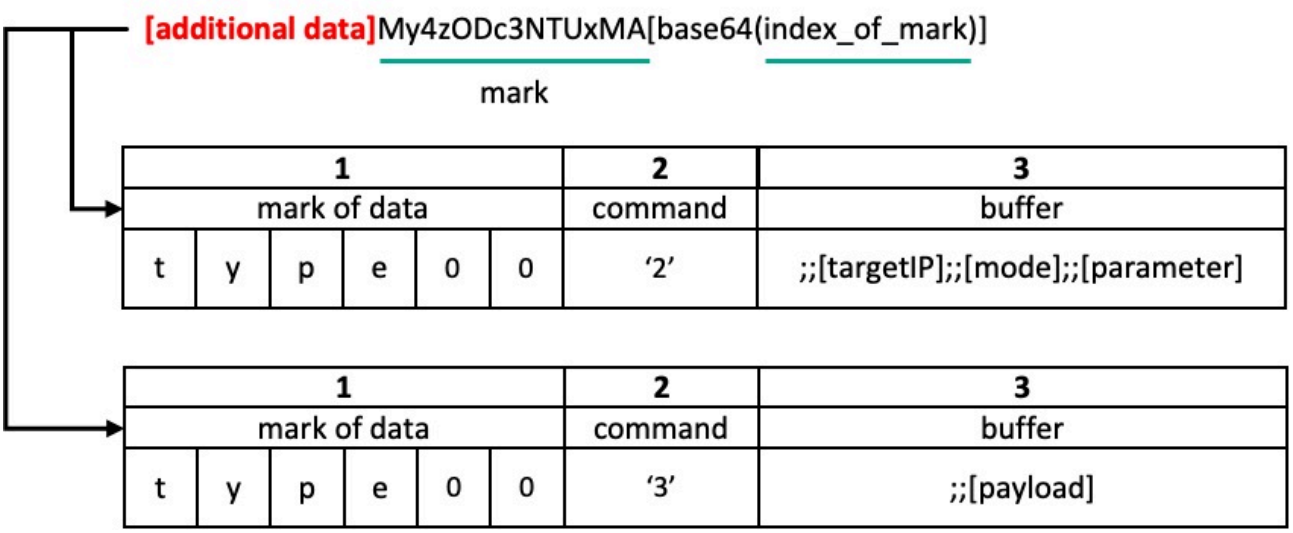
The variant of wAgent has the ability to receive data in both form-data and JSON formats, depending on the C2 server it succeeds in reaching. Notably, it includes the __Host-next-auth-token key within the Cookie field in the request header during the communication (T1071.001), carrying the sequence of communication appended by random digits. In this version, the new observed change is that an open-source GNU Multiple-Precision (GMP) library is employed to carry out RSA encryption computations, which is a previously unseen library in malware used by the Lazarus group. According to the wAgent configuration file, it is identified as the x64_2.1 version. This version manages payloads using a C++ STL map, with emphasis on receiving additional payloads from the C2 and loading them directly into memory, along with creating a shared object. With this object, the main module is able to exchange command parameters and execution results with the delivered plugins.



Operational structure of the wAgent variant

Variant of the Agamemnon downloader

The Agamemnon downloader is also responsible for downloading and executing additional payloads received from the C2 server. Although we did not obtain the configuration file of Agamemnon, it receives commands from the C2 and executes the payload by parsing the commands and parameters based on ;; characters, which serve as command and parameter delimiters. The value of the mode in response passed with a 2 command determines how to execute the additional payload, which is delivered along with a 3 command. There are two methods of execution: the first one is to load the payload reflectively (T1620), which is commonly used in malware, whereas the second one is to utilize the open-source Tartarus-TpAllocInject technique, which we have not previously seen in malware from the Lazarus group.



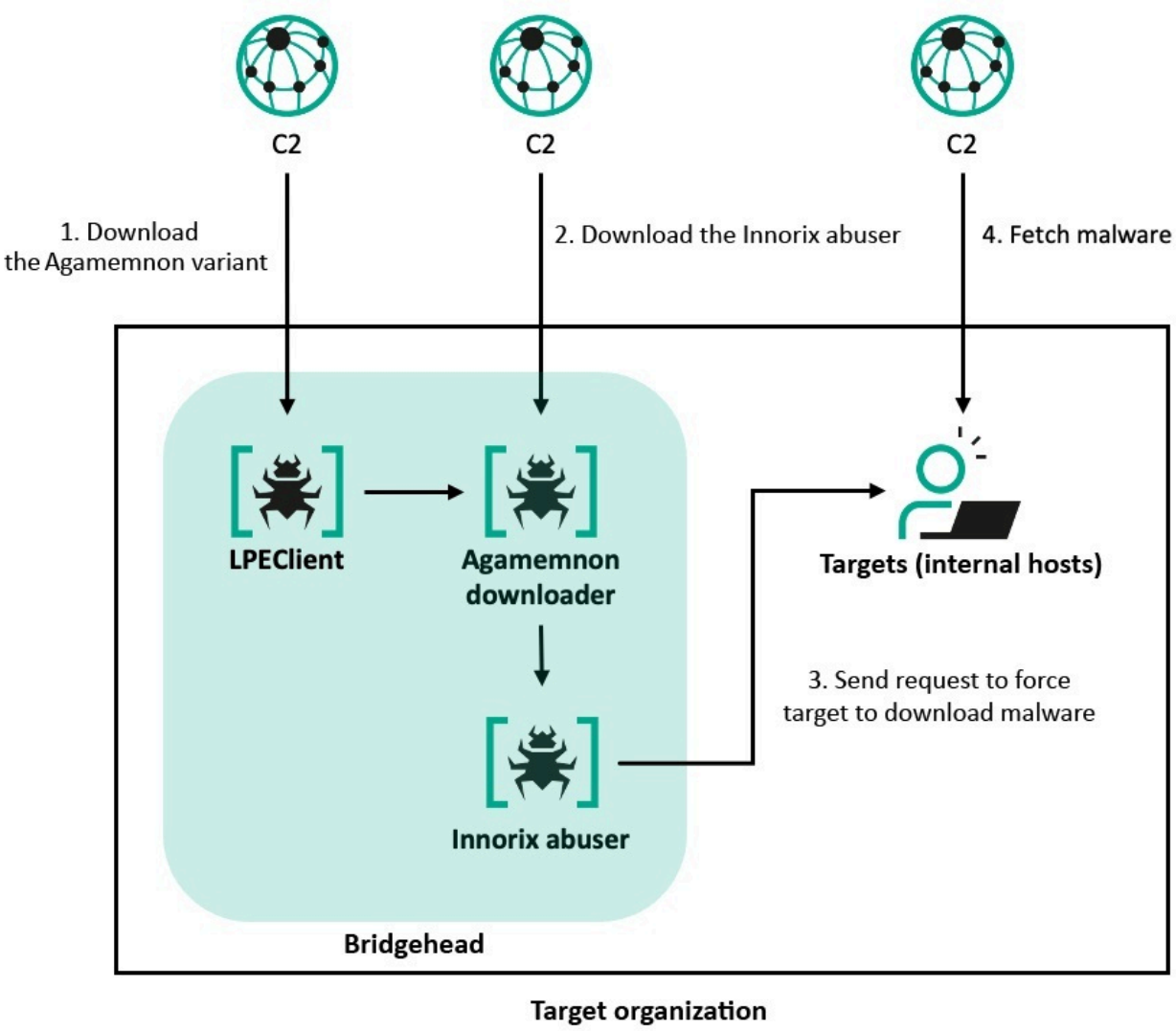
Structure of the commands where additional data is passed

The open-source loader is built on top of another open-source loader named Tartarus' Gate. Tartarus' Gate is based on Halo's Gate, which is in turn based on Hell's Gate. All of these techniques are designed to bypass security products such as antivirus and EDR solutions, but they load the payload in different ways.

Innorix Agent exploit for lateral movement

Unlike the previously mentioned tools, the Innorix abuser is used for lateral movement. It is downloaded by the Agamemnon downloader (T1105) and exploits a specific version of a file transfer software tool developed in South Korea, Innorix Agent, to fetch additional malware on internal hosts (T1570). Innorix Agent is another software product that is mandatory for some financial and administrative tasks in the South Korean internet environment, meaning that it is likely to be installed on many PCs of both corporations and individuals in South Korea, and any user with a vulnerable version is potentially a target. The malware embeds a license key allegedly bound to version 9.2.18.496, which allows it to perform lateral movement by generating malicious traffic disguised as legitimate traffic against targeted network PCs.

The Innorix abuser is given parameters from the Agamemnon downloader: the target IP, URL to download a file, and file size. It then delivers a request to that target IP to check if Innorix Agent is installed and running. If a successful response is returned, the malware assumes that the software is running properly on the targeted host and transmits traffic that allows the target to download the additional files from the given URL due to a lack of traffic validation.



Steps to deploy additional malware via the Innorix abuser

The actor created a legitimate AppVShNotify.exe and a malicious USERENV.dll file in the same path via the Innorix abuser, and then executed the former using a legitimate feature of the software. The USERENV.dll was sideloaded (T1574.002) as a result, which ultimately led to the execution of ThreatNeedle and LPEClient on the targeted hosts, thus launching the infection chain on previously unaffected machines.

We reported this vulnerability to KrCERT due to the potentially dangerous impact of the Innorix abuser, but were informed that the vulnerability has been exploited and reported in the past. We have confirmed that this malware does not work effectively in environments with Innorix Agent versions other than 9.2.18.496.

In addition, while digging into the malware's behavior, we identified another additional arbitrary file download vulnerability that applies to versions up to 9.2.18.538. It is tracked as KVE-2025-0014 and we have not yet found any evidence of its use in the wild. KVE is a vulnerability identification number issued exclusively by KrCERT. We successfully contacted Innorix to share our findings containing the vulnerabilities via KrCERT, and they managed to release a patched version in March with both vulnerabilities fixed.

Second phase malware

The second phase of the operation also introduces newer versions of malicious tools previously seen in Lazarus attacks.

SIGNBT

The SIGNBT we documented in 2023 was version 1.0, but in this attack, version 0.0.1 was used at the forefront. In addition, we identified a more recent version, SIGNBT 1.2. Unlike versions 1.0 and 0.0.1, the 1.2 version had minimal remote control capabilities and was focused on executing additional payloads. The malware developers named this version “Hijacking”.

In the second phase of this operation, SIGNBT 0.0.1 was the initial implant executed in memory in SyncHost.exe to fetch additional malware. In this version, the C2 server was hardcoded without reference to any configuration files. During this investigation, we found a credential dumping tool that was fetched by SIGNBT 0.0.1, identical to what we have seen in previous attacks.

As for version 1.2, it fetches the path to the configuration file from its resources and retrieves the file to obtain C2 server addresses. We were able to extract two configuration file paths from each identified SIGNBT 1.2 sample, which are shown below. Another change in SIGNBT 1.2 is that the number of prefixes starting with SIGN are reduced to only three: SIGNBTLG, SIGNBTRC, and SIGNBTSR. The malware receives an RSA public key from the C2 and encrypts a randomly generated AES key using the public key. All traffic is encrypted with the generated AES key.

- Configuration file path 1: C:\ProgramData\Samsung\SamsungSettings\settings.dat
- Configuration file path 2: C:\ProgramData\Microsoft\DRM\Server\drm.ver

```
{
proxylist: [{ // C2 server list
        proxy: "https%0x3A//builsf[.]com/inc/left.php"
    },
    {
        proxy: "https%0x3A//www.rsdf[.]kr/wp-content/uploads/2024/01/index.php"
    },
    {
        proxy: "http%0x3A//www.shcpump[.]com/admin/form/skin/formBasic/style.php"
    },
    {
```

```
12              proxy: "https%0x3A//htns[.]com/eng/skin/member/basic/skin.php"
13      },
14      {
15              proxy: "https%0x3A//kadsm[.]org/skin/board/basic/write_comment_skin.php"
16      },
17      {
18              proxy: "http%0x3A//bluekostec[.]com/eng/community/write.asp"
19      },
20      {
21              proxy: "http%0x3A//dream.bluit.gethompy[.]com/mobile/skin/board/gallery/index.skin.php"
22      }],
23  wake: 1739839071, // Timestamp of Tuesday, February 18, 2025 12:37:51 AM
24  status: 1 // It means the scheduled execution time is set.
25  }
```

COPPERHEDGE

COPPERHEDGE is a malicious tool that was named by US-CERT in 2020. It is a Manuscrypt variant and was primarily used in the DeathNote cluster attacks. Unlike the other malware used in this operation, COPPERHEDGE has not changed dramatically, with only several commands being slightly changed compared to the older versions. This version, however, retrieves configuration information such as the C2 server address from the ADS %appdata%\Microsoft\Internet Explorer\brndlog.txt:loginfo (T1564.004). The malware then sends HTTP traffic to C2 with three or four parameters for each request, where the parameter name is chosen randomly out of three names in any order.

- First HTTP parameter name: bih, aqs, org
- Second HTTP parameter name: wib, rlz, uid
- Third HTTP parameter name: tib, hash, lang
- Fourth HTTP parameter name: ei, ie, oq

The actor primarily used the COPPERHEDGE malware to conduct internal reconnaissance in this operation. There are a total of 30 commands from 0x2003 to 0x2032, and 11 response codes from 0x2040 to 0x2050 inside the COPPERHEDGE backdoor.

The evolution of Lazarus malware

In recent years, the malware used by the Lazarus group has been rapidly evolving to include lightweighting and modularization. This applies not only to newly added tools, but also to malware that has been used in the past. We have observed such changes for a few years, and we believe there are more on the way.

	Use of asymmetric encryption	Load plugins	Divided into core and loader version
MISTPEN	–	O	–
CookiePlus	O (RSA)	O	–
ThreatNeedle	O (Curve25519)	O	O
wAgent (downloader)	O (RSA)	O	–
Agamemnon downloader	–	–	–
SIGNBT	O (RSA)	O	O
COPPERHEDGE	O (RSA)	–	O

Discoveries

During our investigation into this campaign, we gained extensive insight into the Lazarus group's post-exploitation strategy. After installing the COPPERHEDGE malware, the actor executed numerous Windows commands to gather basic system information (T1082, T1083, T1057, T1049, T1016, T1087.001), create a malicious service (T1569.002, T1007) and attempt to find valuable hosts to perform lateral movement (T1087.002, T1135).

While analyzing the commands executed by the actor, we were able to identify the actor's mistake when using the taskkill command: the /im parameter when using taskkill means imagename, which should specify the image name of the process, not the process id. This shows that the actor is still performing internal reconnaissance by manually entering commands.

Infrastructure

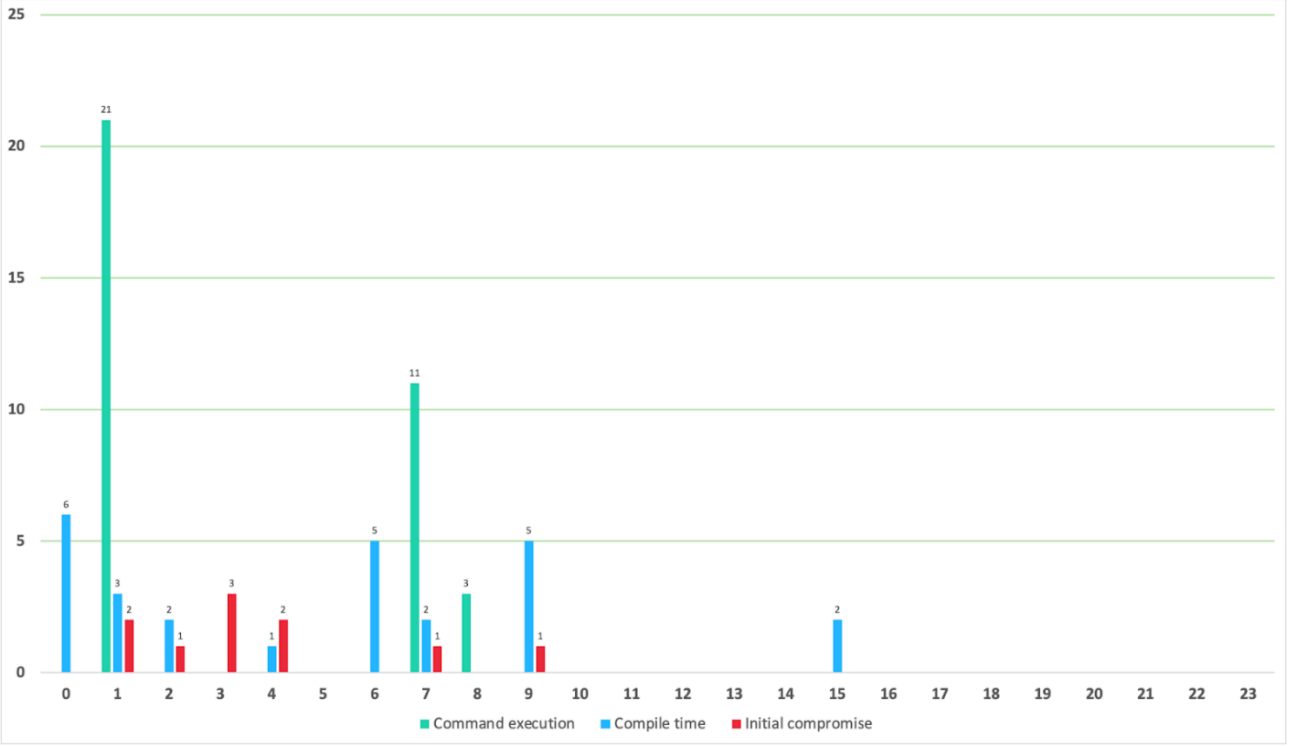
Throughout this operation, most of the C2 servers were legitimate but compromised websites in South Korea (T1584.001), further indicating that this operation was highly focused on South Korea. In the first phase, other media sites were utilized as C2 servers to avoid detection of media-initiated watering hole attacks. However, as the infection chain turned to the second phase, legitimate sites in various other industries were additionally exploited.

Unlike other cases, LPEClient's C2 server was hosted by the same hosting company as www.smartmanagerex[.]com, which was deliberately created for initial compromise. Given that LPEClient is heavily relied upon by the Lazarus group for delivering additional payloads, it is likely that the attackers deliberately rented and configured the server (T1583.003), assigning a domain under their control to maintain full operational flexibility. In addition to this, we also found that two domains that were exploited as C2 servers for SIGNBT 0.0.1 resolved to the same hosting company's IP range.

We confirmed that the domain thek-portal[.]com belonged to a South Korean ISP until 2020 and was the legitimate domain of an insurance company that was acquired by another company. Since then, the domain had been parked and its status was changed in February 2025, indicating that the Lazarus group re-registered the domain to leverage it in this operation.

Attribution

Throughout this campaign, several malware samples were used that we managed to attribute to the Lazarus group through our ongoing and dedicated research conducted for a long time. Our attribution is supported by the historical use of the malware strains, as well as their TTPs, all of which have been well documented by numerous security solutions vendors and governments. Furthermore, we have analyzed the execution time of the Windows commands delivered by the COPPERHEDGE malware, the build timestamps of all malicious samples we described above, and the time of initial compromise per host, demonstrating that the timeframes were mostly concentrated between GMT 00:00 and 09:00. Based on our knowledge of normal working hours in various time zones, we can infer that the actor is located in the GMT+09 time zone.



Timeline of malicious activity

Victims

We identified at least six software, IT, financial, semiconductor manufacturing and telecommunication organizations in South Korea that fell victim to “Operation SyncHole”. However, we are confident that there are many more affected organizations across a broader range of industries, given the popularity of the software exploited by Lazarus in this campaign.

Conclusion

This is not the first time that the Lazarus group exploited supply chains with a full understanding of the software ecosystem in South Korea. We have already described similar attacks in our analysis reports on the Bookcode cluster in 2020, the DeathNote cluster in 2022, and the SIGNBT malware in 2023. All of these cases targeted software developed by South Korean vendors that required installation for online banking and government services. Both of the software products exploited in this case are in line with past cases, meaning that the Lazarus group is endlessly adopting an effective strategy based on cascading supply chain attacks.

The Lazarus group’s specialized attacks targeting supply chains in South Korea are expected to continue in the future. Our research over the past few years provided evidence that many software development vendors in Korea have already been attacked, and if the source code of a product has been compromised, other zero-day vulnerabilities may continue to be discovered. The attackers are also making efforts to minimize detection by developing new malware or enhancing existing malware. In particular, they introduce enhancements to the communication with the C2, command structure, and the way they send and receive data.

We have proven that accurate detection and quick response can effectively deter their tactics, and in the meantime, we were able to remediate vulnerabilities and mitigate attacks to minimize damage. We will continue to monitor the activity of this group and remain agile in responding to their changes. We also recommend using reliable security solutions to stay alert and mitigate potential risks. Our product line for businesses helps identify and prevent attacks of any complexity at an early stage.

Kaspersky products detect the exploits and malware used in this attack with the following verdicts: Trojan.Win64.Lazarus.*, Trojan.Win32.Lazarus.*, MEM:Trojan.Win32.Cometer.gen, MEM:Trojan.Win32.SEPEH.gen, Trojan.Win32.Manuscrypt.*, Trojan.Win64.Manuscrypt.*, Trojan.Win32.Zenpak.*.

Indicators of Compromise

More IoCs are available to customers of the Kaspersky Intelligence Reporting Service. Contact: intelreports@kaspersky.com.

Variant of the ThreatNeedle loader
f1bcb4c5aa35220757d09fc5feea193b C:\System32\PCAuditex.dll

Variant of the wAgent loader
dc0e17879d66ea9409cdf679bfea388c C:\ProgramData\intel\util.dat

COPPERHEDGE dropper
2d47ef0089010d9b699cd1bbbc66f10a %AppData%\hnc_net.tmp

C2 servers

www[.]smartmanagerex[.]com

hxxps://thek-portal[.]com/eng/career/index.asp

hxxps://builsf[.]com/inc/left.php

hxxps://www[.]rsdf[.]kr/wp-content/uploads/2024/01/index.php

hxxp://www[.]shcpump[.]com/admin/form/skin/formBasic/style.php

hxxps://htns[.]com/eng/skin/member/basic/skin.php

hxxps://kadsm[.]org/skin/board/basic/write_comment_skin.php

hxxp://bluekostec[.]com/eng/community/write.asp

hxxp://dream.bluit.gethompy[.]com/mobile/skin/board/gallery/index.skin.php

---