PLAINBIT

2025 사이버보안전문단 프로젝트

AWS 클라우드 기반 DFIR 프레임워크 연구

2025.11

김진국, 장원희, 김서준, 안혜송

※ 해당 보고서는 사이버보안전문단 프로젝트 결과물로 작성되었습니다.

목차

1. 개 요

사이버 보안 환경은 클라우드 기술의 확산과 더불어 새로운 형태의 위협에 직면하고 있다. 클라우드의 도입은 기업의 IT 운영 효율성과 유연성을 크게 향상시킬 수 있지만, 대부분의 조직은 여전히 온프레미스 기반의 사고 대응 및 포렌식 체계를 유지하고 있다. 이로 인해 클라우드 인프라 특유의 복잡한 구조와 서비스 간 연계성을 충분히 반영하지 못하는 한계가 존재하며, 이에 따라 클라우드 환경에 특화된 디지털 포렌식 및 사고 대응(이하, DFIR) 역량이 부족한 실정이다.

최근에는 IAM 자격 증명, API, 서버리스 함수 등 클라우드 고유 자원을 악용하는 공격이 증가하고 있으며, 실제 침해 사례 또한 꾸준히 보고되고 있다. 이러한 변화 속에서 MITRE ATT&CK 프레임워크는 클라우드 환경을 별도의 매트릭스로 분리하고, 공격자가 클라우드 자원을 이용해 수행할 수 있는 전술(Tactics)과 기법(Technique)을 체계적으로 정의했다. 이는 클라우드가 단순한 서비스 인프라가 아니라, 공격자가 장악하고 조작할 수 있는 독립된 공격 표면(Attack Surface)으로 발전했음을 의미한다.

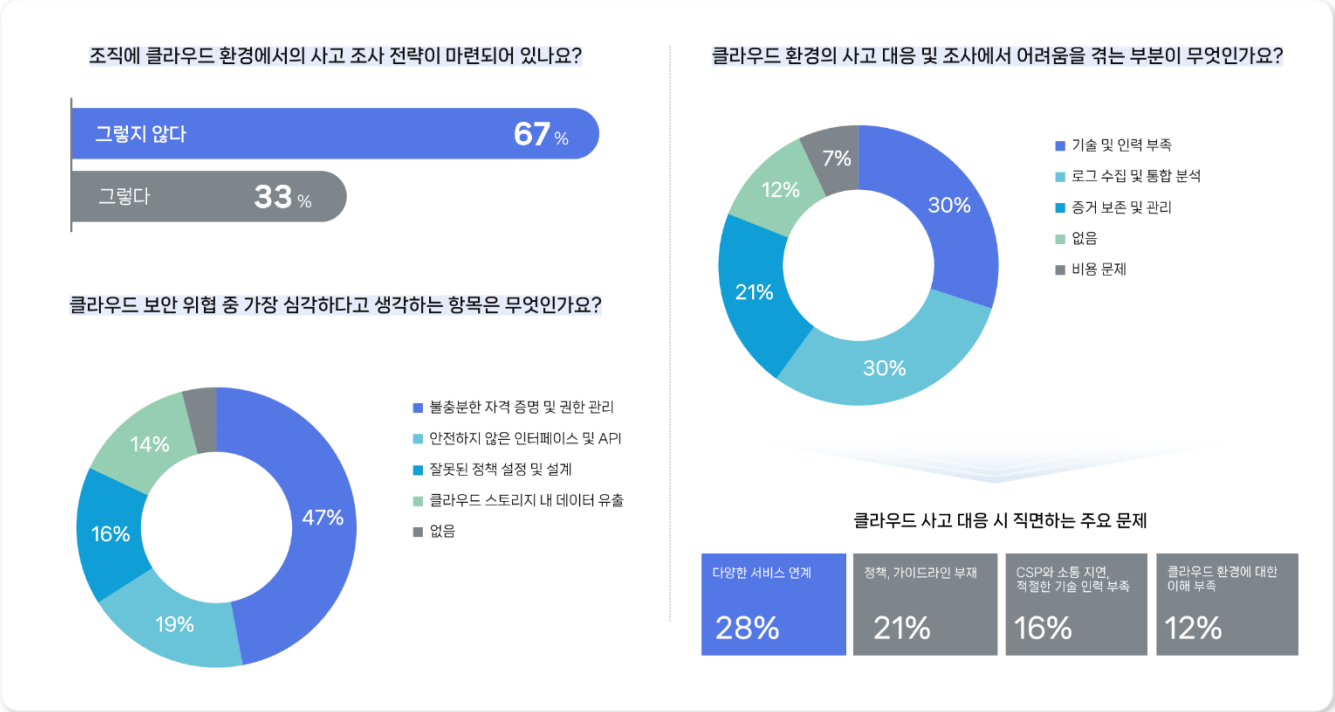
이와 같은 흐름은 글로벌 위협 인텔리전스에서도 명확히 나타난다. Mandiant의 M-Trends 2025 보고서는 클라우드 침해(Cloud Compromise)를 처음으로 독립된 세션으로 분류하고 최상위 챕터로 구성해 클라우드 환경에서의 보안 사고가 지속적으로 증가하고 있음을 강조했다.

MITRE ATT&CK과 Mandiant M-Trends 보고서 모두 클라우드 침해를 독립된 위협 영역으로 다루고 있다는 점에서, 클라우드 고유의 구조적 특성과 공격 표면을 반영한 사고 대응 체계 구축의 중요성이 그 어느 때보다 커지고 있다.

국내 상황 역시 예외가 아니다. 클라우드 설정 오류, 보안 책임에 대한 오해, SaaS 확산에 따른 Shadow IT와 같은 가시성 부족 등이 주요 보안 리스크로 지속적으로 지적되고 있다. 이러한 문제를 실증적으로 확인하기 위해 본 연구진은 국내 보안 및 사고 대응 실무자 100여 명을 대상으로 클라우드 사고 대응 현황 설문조사를 수행했다.

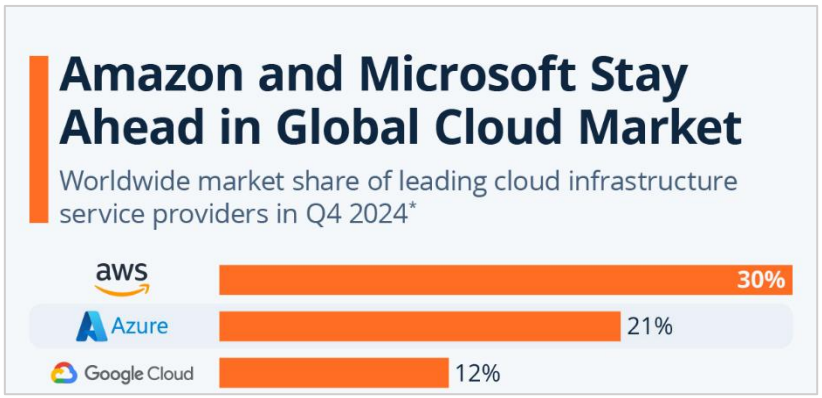
그 결과, 응답자의 67%가 “클라우드 사고 조사 전략이 마련되어 있지 않다”고 답했으며, 47%는 “불충분한 자격 증명 및 권한 관리”를 주요 위협으로 지목했다. 또한, 로그 수집 및 통합 분석의 어려움(30%)과 기술 인력 부족(30%)이 가장 큰 대응 한계로 조사되었다.

이러한 결과는 클라우드 환경에서 데이터 수집-분석-대응 절차를 통합한 표준화된 DFIR 프레임워크의 필요성을 명확히 보여준다.



[그림 1] 클라우드 사고 대응 현황 설문조사 주요 결과

본 연구는 클라우드 환경 중에서도 AWS(Amazon Web Services)를 주요 실증 분석 대상으로 선정했다. AWS는 글로벌 클라우드 시장에서 지속적 1위를 유지하며, 본 연구진이 수행한 국내 보안 실무자 대상 클라우드 사고 대응 현황 설문조사에서도 AWS를 주력 클라우드로 사용하는 비율이 가장 높게 나타났다.



[그림 2] 해외 클라우드 시장 점유율 Top 3 (Synergy Research Group)

따라서, AWS 클라우드 환경에서 발생하는 사고를 효과적으로 수집 및 분석하기 위한 DFIR 프레임워크를 실무화 하는 것을 목적으로 한다. 이를 위해 AWS 로그(CloudTrail, VPC Flow, S3 Access 등)의 상관 분석을 기반으로 사고 탐지 및 분석 절차를 체계화하고, 이를 시나리오 기반 실증 분석, CheatSheet, 자동화 도구 개발을 통해 검증한다.

연구를 통해 제안된 절차를 표준화함으로써 클라우드 DFIR 역량 강화 및 접근성 제고에 기여하고자 한다. 연구의 세부 목표는 다음과 같다.

[표 1] 연구의 세부 목표 및 연구 방법

목표 구분	연구 방법
클라우드 DFIR 프레임워크의 요구사항 도출	MITRE ATT&CK와 AWS Incident Response Playbook 등을 분석해 사고 대응에 필요한 데이터 수집 및 분석 구성요소 정의, 클라우드 사고의 특성과 기존 사고 대응 절차의 한계 도출
AWS 로그 기반 데이터 수집·분석 체계 설계	CloudTrail, VPC Flow Logs, S3 Access Log 등 AWS 주요 로그의 구조를 분석하고 DFIR 절차에 적합한 로그 수집 및 상관 분석 프로세스 설계
AWS DFIR CheatSheet 및 분석 도구 개발	서비스별 주요 로그 필드와 사고 대응 시 상관 분석이 가능한 이벤트를 표준화한 DFIR CheatSheet를 작성하고, 이를 기반으로 공격 전술별 이벤트 분석을 지원하는 도구(bitParser for AWS) 개발
시나리오 기반 사고 분석 검증 수행	IAM 권한 탈취, S3 접근권한 오설정, EC2 악성 행위 등이 악용된 공격 시나리오를 구성하고 로그 상관 분석을 통해 제안 체계의 탐지·분석 유효성 검증
프레임워크의 실무 적용성 및 확장성 평가	시나리오 실증 결과를 바탕으로 제안 프레임워크의 실무 활용성 및 자동화를 위한 확장 가능성 평가

각 단계의 연구 결과는 보고서 2~6장에서 상세히 기술하며, 최종적으로 AWS 기반 클라우드 DFIR 체계의 표준화 및 실무 적용성을 제시한다. 이로 인한 기대 효과는 다음과 같다.

첫째, 클라우드에 환경에 최적화된 DFIR 절차 확립을 통해 사고 대응 체계 고도화

기존 온프레미스 중심의 사고 대응 체계 한계를 보완하고, 클라우드 환경의 구조적 특성과 로그 생성을 반영한 표준화된 DFIR 프레임워크를 제시함으로써, 조직이 클라우드 환경에서도 체계적이고 일관된 사고 대응 절차를 수립·운영할 수 있도록 지원한다.

둘째, 로그 수집·분석 자동화를 통한 가시성 확보

클라우드 환경에서 가장 큰 문제로 지적되는 로그 수집 및 통합 분석의 어려움을 해소하기 위해, 로그 평탄화 및 상관분석 기반의 자동화된 분석 체계를 구축함으로써, 클라우드 서비스 간 로그 상호 연계성을 강화하고, 사고 원인 분석 및 이상 행위 탐지의 효율성을 향상시킨다.

셋째, DFIR CheatSheet 및 도구화를 통한 실무 접근성 향상

사고 대응 인력 및 기술 부족 문제를 해소하기 위해 서비스별 로그 필드와 상관 이벤트를 체계화한 DFIR CheatSheet와 분석 지원 도구를 제공함으로써, 보안 실무자는 클라우드 사고의 핵심 이벤트를 신속히 파악하고, 중소규모 조직도 경제적 부담 없이 효율적인 클라우드 DFIR 절차를 수행할 수 있다.

1.1. 전체 요약

클라우드 전용 사고 대응 가이드의 부재에 따라 AWS 환경의 사고 대응 역량 강화를 목표로 연구를 수행했다. 선행 연구를 통해 클라우드 환경의 증거 수집 제약과 대응 절차의 한계를 분석하고, 이에 대응하기 위해 AWS 구조에 특화된 DFIR 데이터 수집 체계와 사고 유형별 분석 절차를 정립했다.

또한, 자동화 분석 도구(bitParser)와 DFIR CheatSheet를 개발하고, 랜섬웨어 시나리오를 통해 실효성을 검증했으며, 이를 통해 신뢰성 있는 증거 확보와 신속한 행위 분석이 가능한 실무 중심의 DFIR 체계를 제시했다.

[표 2] 보고서 목차에 따른 연구 내용 및 결과 요약

번호	대제목	주요 내용
2	클라우드 인프라 및 보안 개요	클라우드의 책임 범위와 구조적 제약을 중심으로 이론적 기반 정리 - 보안 책임 공유 모델, 서비스 모델 별 보안 책임 공유 모델, AWS 클라우드 서비스 보안 아키텍처, 온프레미스와 클라우드의 사고 대응 방식, 클라우드 환경의 공통적 제약
3	선행 연구	AWS 클라우드 환경 DFIR 적용 가능성과 한계 도출 및 기존 대응 방식의 개선점 식별 - 클라우드 주요 보안 위협, 클라우드 침해 동향 사례, MITRE ATT&CK 기반 클라우드 공격 전술 및 기법 연구, AWS 사고 대응 프레임워크 조사, AWS 보안 서비스/로그 조사, AWS 사고 대응 플레이북 조사
4	사고 데이터 수집	클라우드 환경에 특화된 DFIR 수집 체계 및 방안 제시 - 명령 기반 데이터 수집, 로그 기반 데이터 수집, 포렌식 이미지 수집
5	사고 분석 기법	AWS 환경에서의 DFIR 분석 절차 체계화 및 사고 분석 방안 제시 - 로그별 주요 분석 필드 및 이벤트 분석, 공격 전술별 로그 이벤트를 매핑한 DFIR CheatSheet 개발, AWS DFIR 분석 도구 개발
6	시나리오 기반 실증 분석	랜섬웨어 시나리오를 기반으로 주요 로그 분석 방안과 bitParser 도구의 실효성 검증 - 공격 시나리오 개요, 시나리오 분석 결과

[표 3] 연구 결과물 및 성과

구분	내용
클라우드 사고 대응 현황 설문 조사 결과	국내 보안 및 사고 대응 실무자 100여 명을 대상으로 클라우드 사고 대응 현황을 조사해 연구 방향성 수립에 활용 (bit.ly/cloud-dfir-survey 참고)
AWS DFIR CheatSheet (CloudTrail, S3 Access Log)	CloudTrail Log, S3 Access Log 기반 DFIR 주요 이벤트 및 분석 포인트 정의 (bit.ly/dfir-cheatsheet-s3, bit.ly/dfir-cheatsheet-cloudtrail 참고)
bitParser for AWS Log	CloudTrail Log, S3 Access Log, VPC Flow Logs를 통합 파싱 및 분석 자동화 도구 개발 (github.com/Plainbit/bitParser 참고)

2. 클라우드 인프라 및 보안 개요

클라우드 환경에서의 사고 대응은 기존 온프레미스 환경과는 전혀 다른 구조적 특성을 지닌다. 이러한 특성은 보안 관리 및 사고 대응 절차 전반에 직접적인 영향을 미치며, 클라우드 기반 사고 대응 체계를 이해하기 위해서는 환경의 구조적 제약과 책임 범위를 명확히 이해할 필요가 있다. 이에 2장에서는 보안 책임 공유 모델, 클라우드 서비스 보안 아키텍처, 온프레미스와 클라우드 간 사고 대응 방식의 차이, 클라우드 환경의 공통적 제약 사항을 중심으로 클라우드 사고 대응의 이론적 기반을 살펴본다.

2장에서 다루는 내용은 다음과 같다.

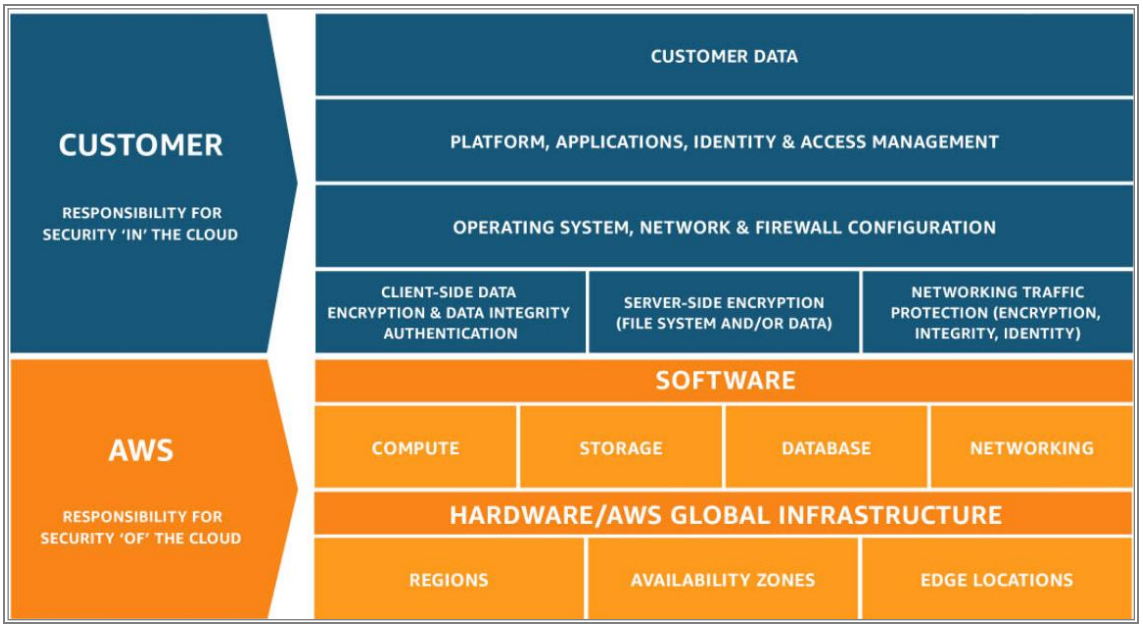
[표 4] 주요 연구 내용 – 클라우드 인프라 및 보안 개요

번호	소제목	주요 내용
1	보안 책임 공유 모델이란?	클라우드 보안을 CSP와 고객 간에 구분해 각자의 보호 범위를 명확히 정의하는 보안 책임 공유 모델 개념 설명
2	서비스 모델 별 보안 책임 공유 모델	클라우드 서비스를 IaaS, PaaS, SaaS로 구분하고 각 모델 별 고객 보안 책임 및 ISMS-P 인증 심사 범위 설명
3	AWS 클라우드 서비스 보안 아키텍처	AWS 보안 참조 아키텍처(SRA)를 기반으로 다중 계정 환경을 구조화하고, OU별 역할에 따라 보안 서비스를 통합해 체계적으로 관리하는 방안 설명
4	온프레미스와 클라우드의 사고 대응 방식	온프레미스와 클라우드 환경의 사고 대응 접근 방식 및 주요 차이점 비교
5	클라우드 환경의 공통 제약	보안 및 사고 대응 관점에서의 공통적인 제약 사항 설명

2.1. 보안 책임 공유 모델이란?

보안 책임 공유 모델(Shared Responsibility Model)은 클라우드 환경에서의 보안 책임을 클라우드 서비스 제공자(Cloud Service Provider, 이하 CSP)와 고객이 어떻게 분담하는 지를 정의하는 모델이다. 이는 '누가 무엇을 보호해야 하는가?'에 대한 명확한 지침을 제공해 보안 공백을 방지하는 데 목적이 있다. 클라우드 서비스 제공자는 클라우드 인프라 자체의 보안을 책임지며, 고객은 해당 인프라 위에서 운영하는 모든 것의 보안을 책임진다.

따라서, 클라우드 서비스 제공자마다 보안 책임 공유 모델을 정의해 운영하고 있으며, AWS의 보안 책임 공유 모델은 아래와 같다.



[그림 3] AWS 보안 책임 공유 모델

2.2. 서비스 모델 별 보안 책임 공유 모델

클라우드 서비스 모델은 크게 IaaS(Infrastructure as a Service), PaaS(Platform as a Service), SaaS(Software as a Service)로 구분되며, 각 모델에 따라 고객의 보안 책임 범위가 달라진다.

[표 5] 서비스 모델 별 보안 책임 범위

모델 구분	CSP 보안	고객 책임	대표적인 예시
IaaS	• 핵심 인프라 보호 (물리적 데이터 센터, 서버, 스토리지, 네트워크 등) • 하이퍼바이저와 같은 가상화 계층도 포함	• 가장 많은 보안 책임을 가짐 • 광범위한 영역을 직접 관리 (운영체제, 미들웨어, 데이터, 애플리케이션, 접근 관리(IAM), 네트워크 구성 등)	• Amazon EC2 • Microsoft Azure VM • Google Compute Engine
PaaS	• IaaS의 모든 책임 포함 • 운영체제, 미들웨어, 런타임까지 관리 • 개발자가 애플리케이션 개발에 집중할 수 있는 플랫폼을 안전하게 제공하는 것이 핵심	• 개발 및 배포 데이터 • 애플리케이션 • 사용자 접근 권한 관리	• AWS Elastic Beanstalk • Microsoft Azure App Service • Google App Engine
SaaS	• IaaS와 PaaS의 모든 책임 포함 • 애플리케이션까지 직접 관리 • 고객은 소프트웨어를 서비스로 구독해 사용하기만 하면 됨	• 가장 적은 책임을 가지는 모델 • 서비스 내 데이터 • 사용자 계정 및 접근 권한 관리	• Microsoft 365 • Google Workspace • Salesforce

또한, 정보보호 및 개인정보보호 관리 체계(ISMS-P) 인증에서도 클라우드 서비스 모델에 따라 심사 범위가 달라진다.

[표 6] 클라우드 서비스 모델에 따른 ISMS-P 인증 대상 서비스 및 자산 범위

구분	대상 서비스 및 자산
IaaS	신청 기관이 직접 관리하는 OS(Guest OS), 미들웨어(WAS 등), 응용 프로그램, DBMS
PaaS	신청 기관이 직접 관리하는 응용프로그램 (단, 클라우드 서비스 클라우드 서비스 제공자로부터 계정 및 권한을 할당 받아 사용하는 영역은 인증범위에 포함 – 미들웨어 계정/권한 및 비밀번호 등)
SaaS	응용 프로그램과 관련해 신청기관이 관리 가능한 영역에 한해 심사 수행 (응용 프로그램 계정/권한 관리 및 비밀번호 등)

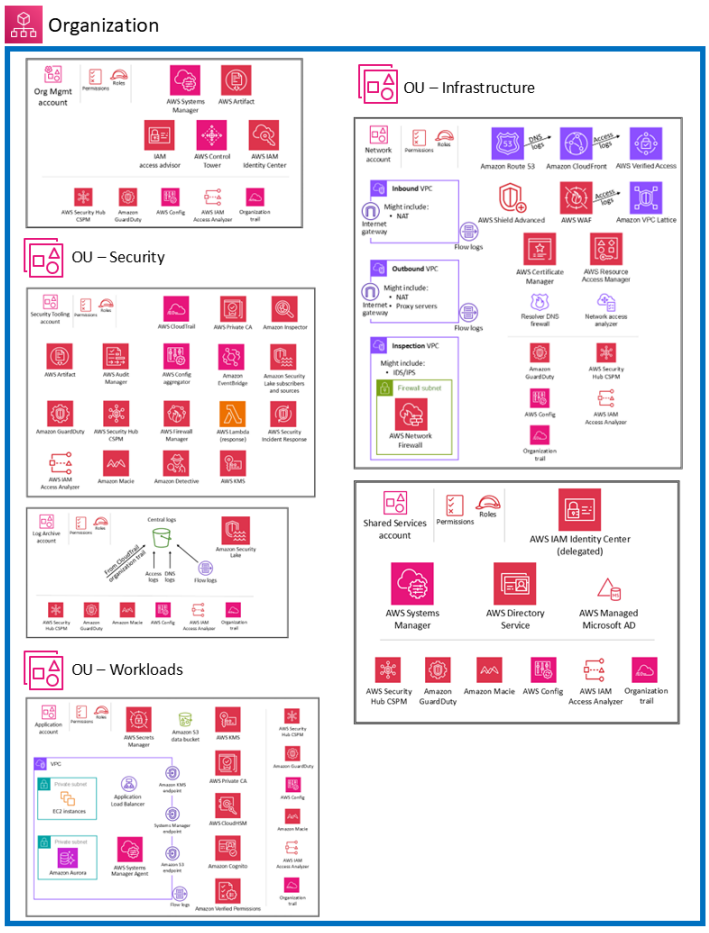
2.3. AWS 클라우드 서비스 보안 아키텍처

CSP는 고객이 클라우드 환경을 안전하게 구축하고 운영할 수 있도록 다양한 보안 서비스와 아키텍처를 제공한다.

AWS의 보안 참조 아키텍처(Security Reference Architecture, 이하 SRA)는 AWS의 보안 서비스들이 어떻게 통합되어 종합적인 보안 기능을 구현하는지를 보여주는 지침이다. 이 아키텍처는 AWS Organizations를 기반으로 다중 계정 환경에서 보안을 체계적으로 관리하는 것을 핵심으로 한다.

AWS는 계정 구조화 및 조직 단위(Organizational Unit, 이하 OU)를 통해 계정을 목적별로 분리한다. 예를 들어, 보안 서비스를 중앙에서 관리하는 보안 OU(보안 도구 계정, 로그 아카이브 계정), 네트워크 및 공통 서비스를 관리하는 인프라 OU, 실제 애플리케이션이 운영되는 워크로드 OU 등으로 구성해 각 계정의 역할과 책임에 따라 적절한 보안 정책을 적용한다. AWS 보안 아키텍처 다이어그램과 주요 보안 서비스의 원칙은 다음과 같다.

[표 7] AWS 보안 아키텍처 다이어그램과 주요 보안 서비스 원칙

AWS 보안 아키텍처 다이어그램	AWS 주요 보안 서비스의 원칙
	**AWS Control Tower** 다중 계정 환경을 안전하게 설정하고 관리하기 위한 기반 제공
	Identity and Access Management (IAM) IAM 역할을 통해 최소 권한 원칙에 따라 리소스에 대한 접근 제어
	Virtual Private Cloud (VPC) 네트워크 트래픽을 논리적으로 격리하고, 보안 그룹과 NACL을 통해 인바운드/아웃바운드 트래픽을 제어해 네트워크 경로를 안전하게 구성
	데이터 보호 AWS KMS(Key Management Service)를 사용해 암호화 키를 관리하고, Amazon S3와 같은 서비스에서 서버 측 암호화를 활성화해 저장된 데이터를 보호
	위협 탐지 및 로깅 Amazon GuardDuty, Amazon CloudWatch 등의 서비스를 활용해 위협 탐지, 모든 활동 기록, 모니터링 로그는 별도의 로그 아카이브 계정에 중앙 집중화해 무결성 보장

2.4. 온프레미스와 클라우드의 사고 대응 방식

온프레미스(On-Premise) 환경과 클라우드 환경에서의 사고 대응은 접근 방식과 고려해야 할 사항에서 명확한 차이를 보인다. 두 환경의 차이는 다음과 같다.

[표 8] 온프레미스 vs. 클라우드 간 DFIR 방식

구분	온프레미스	클라우드
인프라 특성	정적이고 통제된 인프라로 물리적 접근을 통한 모니터링, 포렌식 분석 용이 명확하게 정의된 네트워크 구조로 인해 보안 정책 적용과 침해 탐지가 비교적 쉬움	역동적이고 분산된 가상화 환경으로, CSP와 공동 관리되어 물리적 자산 접근이 불가능 마이크로서비스·컨테이너·서버리스 등 복잡한 구조로 인해 사고 대응 절차가 복잡
책임 공유 모델	모든 인프라와 데이터에 대한 보안 책임은 전적으로 해당 고객에게 있어, 사고 발생 시 고객이 모든 영역에 걸쳐 조사하고 대응할 수 있는 완전한 통제권을 가짐	인프라는 CSP가, 애플리케이션·데이터·접근 관리는 고객이 책임지며, 침해 대응을 위해 양측의 긴밀한 협업이 필수적
가시성 및 모니터링	정적 인프라로 가시성 확보와 트래픽 분석, 이상 징후 탐지가 용이하며, 물리적 접근을 통한 디스크·메모리 등 포렌식 분석이 가능	리소스의 단명성과 복잡한 구조로 가시성 실시간 로그 수집·분석이 필요하지만 CSP 도구 의존으로 심층 가시성이 제한적
데이터 접근성 및 수집	고객이 자체 데이터센터에서 인프라를 직접 구축하고 운영하기 때문에 사고 발생 시 물리적인 증거 수집 가능	가상화된 환경으로 고객이 논리적으로만 접근 가능 증거 수집 시 CSP의 스냅샷·로그 API에 의존해 범위가 제한적
도구 및 자동화	정적 인프라에 맞춘 전통적 보안 도구 (방화벽, IDS/IPS, EDR, SIEM 등)에 의존함 도구는 데이터센터 내에서 실시간 모니터링과 위협 탐지·대응 기능을 제공함	기존 온프레미스 도구 활용에 한계가 있으며, 클라우드 전용 보안 도구가 필요함 자동화를 통해 대규모·고속 클라우드 운영의 탐지·대응 효율성을 높이는 것이 중요함
공격 표면 및 위협	물리적 인프라 중심의 제한된 공격 표면을 가지며, 네트워크·엔드포인트·내부 애플리케이션이 주요 대상 악성 파일·피싱·랜섬웨어를 통한 자산 손상과 정보 유출이 주요 위협으로 작용	다중 환경에 걸친 데이터·서비스 보안 책임으로 인해 공격 표면이 확대됨 잘못된 구성, 취약한 API, 자격 증명 탈취 등 클라우드 특화 공격 기법이 주요 위협으로 작용
대응 및 복구	명확히 정의된 대응·복구 절차를 갖추고 있어 장치 격리·백업 복구·직접 패치 적용 가능	플레이북·스크립트로 신속한 격리·복구가 가능하나, 다중 리전 조정의 복잡성이 존재 중복성과 확장성을 활용하면 복구 속도를 크게 향상시킬 수 있음
기술 및 전문성	온프레미스 대응 전문가는 네트워크, 엔드포인트 보안 등 물리적 인프라 중심 기술에 특화되어 있음 기존 보안 도구 활용 역량에 주로 집중되어 있음	클라우드 아키텍처·CSP 보안 도구·자동화 기술 등 폭넓은 전문성이 요구됨 빠르게 변화하는 환경에 대응하기 위해 지속적인 학습과 최신 위협 동향 파악이 필수적

2.5. 클라우드 환경의 공통 제약

클라우드 환경은 많은 이점을 제공하지만, 보안 및 사고 대응 관점에서 다음과 같이 공통적인 제약 사항이 존재한다.

[표 9] 클라우드 환경의 공통 제약사항

구분	대상 서비스 및 자산
로그 보존 기간	CloudTrail 등 로그 서비스는 기본 보존기간이 제한적이므로, 규정 준수나 장기 분석을 위해 S3·Blob 같은 별도 스토리지로 로그를 내보내 영구 보관하도록 구성해야 함 그렇지 않은 경우, 사고 조사 시 증거를 유실할 가능성이 높음
증거 접근의 제한	고객은 자신의 가상 리소스(VM·스토리지 등) 에만 접근할 수 있으며, 물리적 인프라나 하이퍼바이저에는 접근 불가함 따라서, CSP 인프라 관련 침해 발생 시 직접 조사가 어려워 CSP의 조사 결과와 제공 정보에 의존해야 함
데이터 권한 및 위치	데이터가 저장되는 물리적인 위치(Region)는 선택할 수 있지만 국가별 법률이나 규정에 따라 정부의 데이터 접근 요청이 발생할 수 있음
멀티테넌시 환경	클라우드는 멀티테넌시(여러 고객이 물리 자원을 공유) 구조로 CSP는 논리적 격리를 제공하지만 하이퍼바이저 취약점이나 격리 실패로 인해 한 고객의 리소스가 다른 고객에게 영향을 미칠 가능성이 존재함

3. 선행 연구

3장에서는 클라우드 환경에서의 사고 대응 체계 수립을 위한 선행 연구 분석을 수행한다. 클라우드 사고의 동향과 대표 사례를 통해 클라우드 기반 위협의 특성과 기존 대응 상의 한계를 분석하고, MITRE ATT&CK 프레임워크 및 AWS 사고 대응 플레이북을 심층 분석해 사고 대응에 필요한 데이터 수집 및 분석 구성요소, 절차 개선 방향을 도출한다. 이러한 연구 결과는 4~5장에서 제시할 클라우드 DFIR 프레임워크 설계 및 구현 방안의 근거 자료로 활용된다.

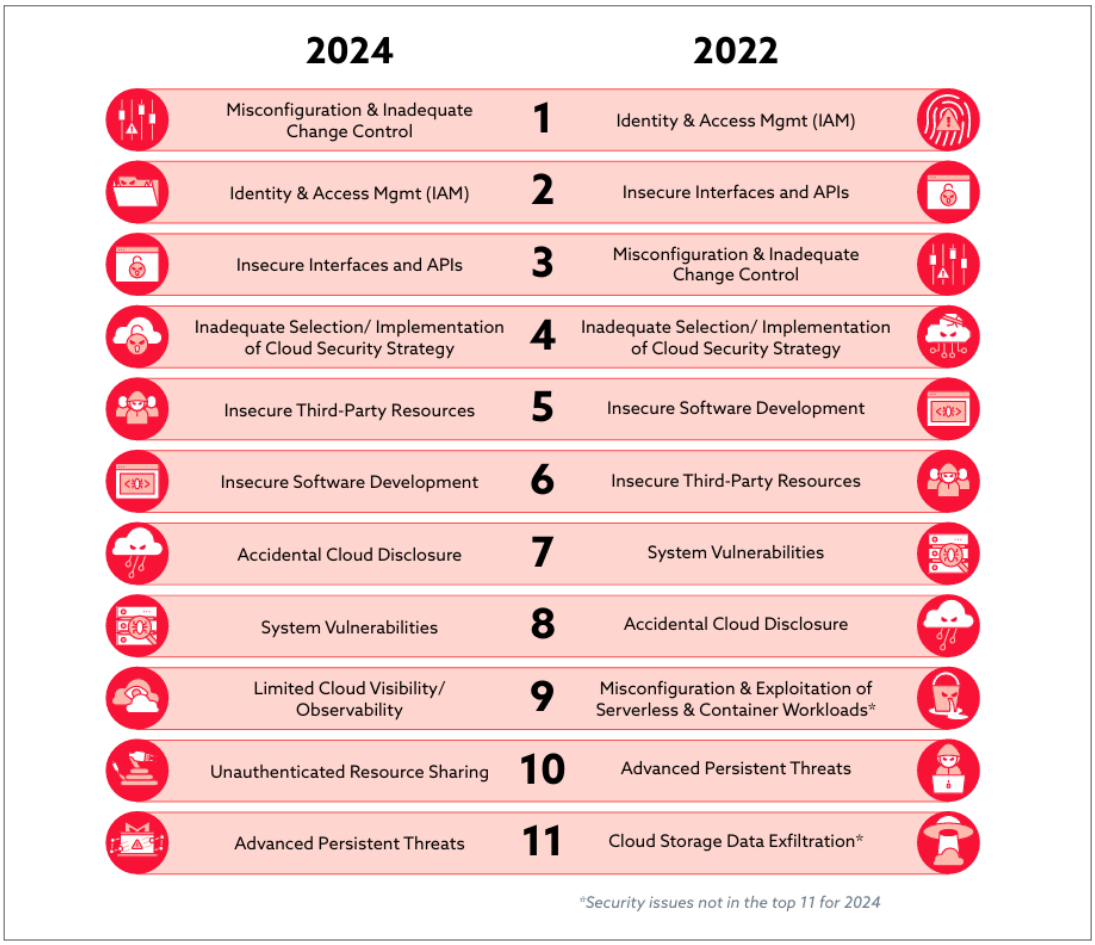
3장에서 다루는 내용은 다음과 같다.

[표 10] 주요 연구 내용 – 선행 연구

번호	소제목	주요 내용
1	클라우드 주요 보안 위협	Cloud Security Alliance(CSA)가 2024년 기준으로 선정한 클라우드 환경의 11대 주요 보안 위협 제시
2	클라우드 사고 동향 및 사례	설정 오류, 자격 증명 탈취 등 클라우드 보안 위협과 주요 사고 사례 조사
3	MITRE ATT&CK 기반 클라우드 공격 전술 및 기법 연구	MITRE ATT&CK v17.1의 전술별 주요 공격 기법 정리 (총 10개 전술, 85개 기법)
4	AWS 사고 대응 프레임워크 조사	NIST SP 800-61을 기반으로 AWS에서 제시한 5단계 사고 대응 프레임워크 조사 (각 단계별 활용 가능한 보안 서비스와 로그 정리)
5	AWS 보안 서비스 조사	로그 기반 가시성을 중심으로 한 12개의 AWS 주요 보안 서비스 활용 방안 조사
6	AWS 로그 조사	AWS가 제공하는 계층별 로그 유형과 사고 대응 활용 방안 조사 (계정 및 관리 활동 로그, 네트워크 및 트래픽 로그, 서비스별 액세스/활동 로그, 보안 서비스 로그, 시스템 및 애플리케이션 로그)
7	AWS 사고 대응 플레이북 조사	AWS 사고 대응 플레이북 중 DFIR 관점에서 유용한 16개 사고 유형 분석 (DFIR 관점 분석 포인트, 주요 로그 및 데이터, 사고 대응 절차 요약)

3.1. 클라우드 주요 보안 위협

글로벌 보안 단체인 'Cloud Security Alliance(CSA)'는 매년 클라우드 위협 보고서를 발간하고 있으며, 500명 이상의 업계 전문가를 대상으로 클라우드 산업의 보안 문제에 대해 설문조사를 실시해 클라우드 환경에서 발생하는 11가지 주요 보안 문제를 선정한다. 2024년 기준으로 발표된 클라우드 주요 보안 위협은 다음과 같다.



[그림 4] 주요 클라우드 보안 위협 - 2024년과 2022년의 비교

각 위협에 대한 상세 내용은 다음과 같다.

1) Misconfiguration & Inadequate Change Control

Misconfiguration(잘못된 설정)은 클라우드 자산을 의도치 않은 손상이나 공격에 취약하게 만드는 설정 오류로, 보안 설정 이해 부족이나 악의적 행위로 인해 발생할 수 있다. 주요 사례로는 비밀 관리 실패, 로깅 비활성화, 과도한 접근 권한, 검증 부족, 서브도메인 하이재킹, CSP 특화 설정 오류(S3 버킷 등) 등이 있다.

2) Identity & Access Management (IAM)

Identity & Access Management(IAM)는 사용자의 신원을 검증하고, 역할·권한·접근 조건을 관리해 인가된 리소스 접근만 허용하는 핵심 보안 체계이다. 주요 구성 요소로는 인증, 권한 부여, SSO, MFA, 활동 모니터링이 있으며, 잘못된 구성이나 미흡한 관리 시 무단 접근 취약점이 발생할 수 있다.

3) Insecure Interfaces & APIs

클라우드 환경에서는 CSP·고객·개발자가 제공하는 API와 UI가 주요 제어 지점이며, 부적절한 인증, 암호화 부족, 세션 관리 미흡, 입력 검증 부족, 로깅·모니터링 미비, 패치 미흡, 과도한 접근 권한, 속도 제한 부족 등으로 취약해질 수 있다. 이러한 취약점은 무단 접근, 민감 정보 유출, 서비스 중단으로 이어질 수 있다.

4) Inadequate Cloud Security Strategy

클라우드 보안 전략은 외부 요인, 기존 구현 상태, 기술 선택, 우선순위 등을 고려해 클라우드 아키텍처, 서비스 모델, CSP 선택, 서비스 지역, 청구 모델 등의 원칙을 수립하는 과정으로, 조직의 보안 목표 달성과 비즈니스 연속성 확보에 기여한다. 이러한 전략은 서비스 전반의 안전한 운영을 보장하고 위험 대응 및 의사결정을 지원한다. 부적절한 클라우드 보안 전략(Inadequate Cloud Security Strategy)은 실제 사고로 이어질 수 있다.

5) Insecure Third-Party Resources

클라우드 도입이 급증하면서 Third-Party Resources(외부 코드, 오픈소스, SaaS 등)로 인한 보안 위험이 커지고 있다. 이는 공급망 취약점으로 간주되며, 사이버 보안 공급망 위험 관리(C-CSRM)의 주요 대상이다.

6) Insecure Software Development

클라우드 기술의 복잡성으로 의도치 않은 취약점이 생기고, 취약 소프트웨어는 침해 통로가 되므로 CI/CD·자동화 환경에서는 공유 책임 모델 숙지, SDLC 적용, 최소 권한 원칙 적용과 개발자 대상 지속 교육이 필수적이다.

7) Accidental Data Disclosure

클라우드의 잘못된 구성으로 인한 데이터 유출 위험은 매년 증가하고 있으며, 공개 검색 도구로 손쉽게 노출 저장소(예: S3, Azure Blob, GCP Storage, Docker Hub, Elasticsearch, Redis, GitHub 등)를 찾을 수 있다. 이러한 유출은 주로 관리 소홀과 부적절한 접근 제어(공개 설정 실수 등)에서 발생한다.

8) System Vulnerabilities

클라우드 서비스의 결함(시스템 취약점)은 기밀성·무결성·가용성을 해치고 서비스 운영을 방해할 수 있다. 취약점 유형은 주로 잘못된 구성(Misconfiguration), 제로데이(Unknown/0-day), 미패치 소프트웨어, 약하거나 기본 인증정보로 나뉜다.

9) Limited Cloud Visibility/Observability

클라우드 가시성이 제한되면 서비스 사용이 정상인지 악의적인지 식별하기 어렵고, 이는 Shadow IT(비인가 애플리케이션 사용)과 승인된 애플리케이션의 오용을 포함한다. 클라우드 가시성 부족은 내부자나 공격자 활동을 제대로 감지하지 못해 보안 사각지대, 침해 탐지 실패, 권한 관리 문제로 이어진다.

10) Unauthenticated Resource Sharing

클라우드의 인증되지 않은 리소스 공유는 가상머신·스토리지·DB 등 민감 자산을 무단 접근 위험에 노출시킨다. 기본 비밀번호 미설정 사례가 여전히 많아 공개 검색 도구(Shodan 등)로 쉽게 찾아진다.

11) Advanced Persistent Threats (APTs)

APT(Advanced Persistent Threats)는 여전히 클라우드 보안의 주요 위협이다. 국가 지원 해커나 조직화된 범죄 그룹은 장기적이고 정교한 공격을 통해 클라우드 내 민감 데이터를 노린다.

3.2. 클라우드 사고 동향 및 사례

클라우드 보안 위협은 단순한 설정 오류에서부터 자격 증명 탈취, 공급망 침해, API 남용 등으로 진화하고 있으며, 공격자들은 클라우드 환경의 확장성과 접근성을 악용하고 있다. 최근 국내/외의 클라우드 환경에서 발생한 사고 사례는 다음과 같다.

1) [APT41] 정보 유출 사고 (2023 년 5 월)

중국 APT 그룹 APT41은 Microsoft 소프트웨어의 '폴리나(Follina)' 제로데이 취약점을 악용해 여러 정부 기관의 클라우드 시스템에 대한 무단 접근 권한을 획득하고 잠재적으로 민감한 정보를 추출했다.

2) [Toyota] 차량 데이터 유출 사고 (2023 년 5 월)

토요타 자동차가 일본 내 약 215만 명의 사용자 데이터가 10년간 공개 상태로 노출된 사고로, 원인은 클라우드 설정 실수로 밝혀졌다. 피해 대상은 T-Connect 및 G-Link 서비스 이용자이며 노출된 정보에는 차량 위치·식별 번호 등이 포함됐으나 악용 사례는 보고되지 않았다.

3) [JumpCloud] 개인정보 유출 사고 (2023 년 6 월)

신원 및 접근 관리 기업인 JumpCloud가 정교한 국가 차원의 공격자에 의해 데이터가 유출된 사고로, 원인은 스피어 피싱 캠페인과 만료되지 않은 자격 증명으로 추적됐다. 개인정보 유출 사고를 위해 공격자는 JumpCloud의 명령 프레임워크에 공격자가 데이터를 주입해 특정 고객 계정을 공격 대상으로 삼았다.

4) [DarkBeam] 개인정보 유출 사고 (2023 년 9 월)

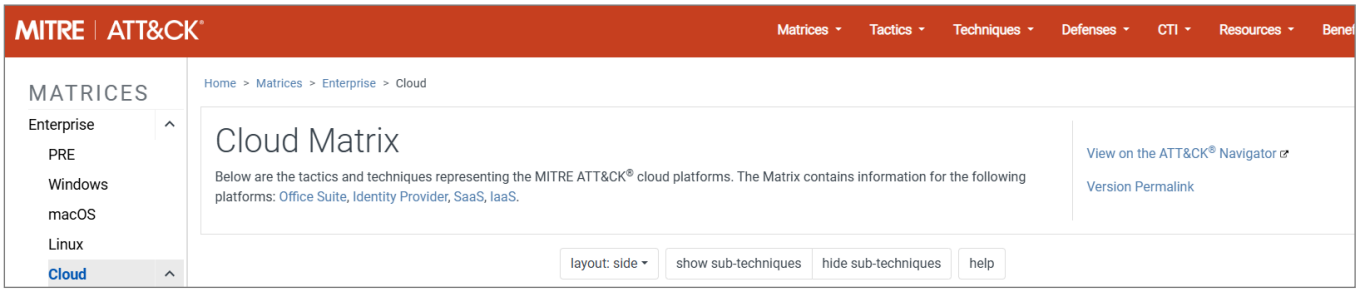
클라우드 보안 기업 DarkBeam이 보호 조치 없이 노출된 Elasticsearch·Kibana 인터페이스로 인해 38억 건 이상의 이메일과 비밀번호 데이터가 유출된 사고로, 원인은 유지보수 후 비밀번호 보호 미설정이라는 관리자의 설정 실수로 밝혀졌다. 노출된 데이터에는 "email 0–9", "email A–F" 등 16개 컬렉션이 포함되어 있었다.

5) [Mercedes Benz] 데이터 유출 사고 (2024 년 1 월)

Mercedes Benz의 API 유출 사고로 공격자들이 회사의 GitHub Enterprise에 접근해 소스 코드, 클라우드 키, 내부 문서가 유출됐다. 침해 원인은 전년도 공개 저장소에서 발견된 한 직원의 GitHub 토큰이 침투 경로로 악용된 것으로 확인됐다.

3.3. MITRE ATT&CK 기반 클라우드 공격 전술 및 기법 연구

MITRE ATT&CK v17.1을 기반으로 클라우드 매트릭스는 총 10개의 전술(tactics)과 85개의 기법(techniques)으로 구성되어 있으며, 각 전술별 공격 기법은 다음과 같다.



[그림 5] MITRE ATT&CK Framework - Cloud Matrix

1) Initial Access (최초 침투)

공격자는 인터넷에 노출된 자산의 취약점이나 보안 설정 오류를 악용하거나 피싱·자격증명 탈취 등의 기법으로 클라우드 환경에 침투한다. 클라우드 서비스의 확산으로 외부에 노출될 수 있는 애플리케이션이 증가하고 관리 대상인 사용자·서비스 계정이 늘어나면서, 이러한 침투 경로와 계정 탈취 위험이 함께 증대되어 데이터 유출 및 권한 남용 등의 사고 가능성이 높아졌다.

[표 11] MITRE ATT&CK Cloud Matrix의 Initial Access 전술에서 사용되는 기법 목록

TID	기법 구분	기법 설명
T1180	Drive-by Compromise	• 공격자는 일반적인 탐색 과정에서 웹 사이트를 방문하는 사용자를 통해 시스템에 접근할 수 있음 • 공개적으로 작성 가능한 클라우드 스토리지 버킷에서 웹 사이트로 제공되는 정상적인 스크립트 파일 수정
T1190	Exploit Public-Facing Application	• 인터넷에 공개된 호스트의 취약점을 악용해 최초 침투 시도 • 애플리케이션이 클라우드 기반 인프라에 호스팅되거나 컨테이너화된 경우, 이를 악용하면 기반 컨테이너가 손상될 수 있음
T1566.002	Phishing: Spearphishing Link	• 악성 링크가 포함된 스피어피싱 메일 활용
T1199	Trusted Relationship	• 외부 공급업체에 높은 수준의 접근 권한을 부여해 내부 시스템뿐만 아니라 클라우드 환경도 관리하는 경우, 외부 공급업체에 할당된 계정이 침해될 수 있음
T1078.004	Valid Accounts: Cloud Accounts	• 클라우드 환경에서 유효한 계정을 사용해 초기 접근을 달성하기 위한 작업을 수행할 수 있음 • 공격자는 무차별 대입 공격, 피싱 또는 기타 다양한 수단을 통해 계정에 접근할 수 있음

2) Execution (침해 실행)

클라우드 환경에 성공적으로 접근한 공격자는 악성 코드를 실행해 목표를 달성한다. 온프레미스와 달리 클라우드에서는 API를 통해 직접 명령을 실행할 수 있으며, 공격자는 AWS Systems ManagerLt Microsoft Intune 같은 중앙관리 도구를 악용해 원격으로 코드를 배포 및 실행한다. 이러한 중앙관리 도구는 네트워크 내 호스트를 광범위하게 통제할 수 있는 높은 권한을 가지므로, 관리자 권한이 탈취될 경우 연결된 다수 시스템에 대해 신속하고 광범위한 침해가 발생할 위험이 있다.

[표 12] MITRE ATT&CK Cloud Matrix의 Execution 전술에서 사용되는 기법 목록

TID	기법 구분	기법 설명
T1651	Cloud Administration Command	• 클라우드 관리 서비스를 악용해 가상 머신 내에서 명령을 실행할 수 있음 • AWS Systems Manager, Azure RunCommand, Runbooks와 같은 리소스를 사용하면, 설치된 가상 머신 에이전트를 활용해 가상 머신에서 원격으로 스크립트 실행 가능
T1059.009	Command and Scripting Interpreter: Cloud API	• 클라우드 API를 악용해 악성 명령 실행 가능 • 클라우드 API 기능을 사용하면 컴퓨팅, 스토리지, ID 및 액세스 관리(IAM), 네트워킹, 보안 정책 등 테넌트의 모든 주요 서비스에 대한 관리 액세스가 허용될 수 있음
T1648	Serverless Execution	• CSP는 서버를 관리할 필요 없이 애플리케이션을 구축할 수 있는 Serverless 리소스를 제공하며, 공격자는 해당 리소스를 악용해 임의의 명령을 실행할 수 있음 • Serverless 함수인 Lambda를 악용해 악성 코드를 실행 가능
T1072	Software Deployment Tools	• 공격자는 중앙 집중식 구성 관리 및 소프트웨어 배포 도구를 활용해 네트워크 내 다른 시스템으로 명령을 실행할 수 있음 • 해당 서비스는 클라우드 관리 명령을 지원하고, 온프레미스 호스트에서 임의의 명령 실행 가능
T1204.003	User Execution: Malicious Image	• 공격자는 악성 코드(백도어, 암호화폐 채굴 등)를 설치한 이미지를 공개 저장소(Github 등)에 업로드 할 수 있으며, 사용자는 악성 이미지를 다운로드 받아 클라우드 환경에 배포하는데 사용할 수 있음 • AWS AMI, 이미지뿐만 아니라 Docker와 같은 널리 사용되는 컨테이너 런타임에도 백도어가 설치될 수 있음 • 악성 이미지를 통해 배포된 인스턴스는 이미 악성 코드가 감염되어 있기 때문에, 공격자는 최초 침투 과정 없이 시스템에 접근 가능

3) Persistence (침해 지속)

시스템 재부팅, 자격 증명 변경 또는 기타 중단 조치 이후에도 환경에 대한 접근을 유지하는 것을 목표로 한다. 온프레미스 환경의 지속성이 주로 파일 시스템이나 레지스트리에 악성 파일을 남기는 것에 의존하는 반면, 클라우드 환경의 지속성은 상태와 구성을 조작하는 데 의존한다. 클라우드 지속성의 핵심은 IAM 객체를 조작하는 것으로 Account Manipulation과 Create Account: Cloud Account 기법은 모두 클라우드 IAM 구조를 직접적으로 공격해 지속적인 접근 경로를 확보하는 것이다. 이러한 기법은 클라우드 관리 활동과 구별하기 어렵기 때문에 탐지하기 어렵다.

[표 13] MITRE ATT&CK Cloud Matrix의 Persistence 전술에서 사용되는 기법 목록

TID	기법 구분	기법 설명
T1098.001	Account Manipulation: Additional Cloud Credentials	• 피해자의 클라우드 계정 및 인스턴스에 지속적으로 접근하기 위해 공격자가 제어하는 자격 증명(credential)을 클라우드 계정에 추가 • Azure/Entra ID 환경에서는 공격자가 기존의 정상적인 자격 증명 외에 서비스 주체(Service Principals) 및 애플리케이션에 대한 자격 증명을 추가할 수 있음 (이는 x.509 인증서 키와 비밀번호 형태일 수 있음) • 적절한 권한이 있을 경우, 이러한 자격 증명은 Azure Portal, Azure CLI, 또는 Azure/AZ PowerShell 모듈을 통해 다양한 방식으로 추가할 수 있음 • IaaS(서비스형 인프라) 환경에서는 클라우드 계정을 통해 접근한 후 공격자가 자신만의 SSH 키를 생성하거나 가져올 수 있으며, AWS API나 GCP 명령어를 활용해 계정에 액세스 키 추가 가능
T1098.003	Account Manipulation: Additional Cloud Roles	• 공격자는 자신이 제어하는 클라우드 계정에 역할이나 권한을 추가해 지속적으로 접근하는 것을 유지할 수 있음 • IAM을 조작해 지속성을 확보하고 권한을 상승하거나 대상 조직의 클라우드 인스턴스에서 생성한 계정에 글로벌 관리자 역할을 추가한 사례 존재
T1098.004	Account Manipulation: SSH Authorized Keys	• 공격자는 피해자 호스트에서 지속성을 유지하기 위해 SSH 파일(authorized_keys) 파일을 수정할 수 있음 • 클라우드 환경에서는 공격자가 명령줄 인터페이스 또는 REST API를 통해 특정 가상 머신의 SSH authorized_keys 파일을 수정할 수 있음
T1098.005	Account Manipulation: Device Registration	• 공격자는 자신이 관리하는 계정의 다중 인증(MFA) 시스템에 장치를 등록할 수 있음
T1136.003	Create Account: Cloud Account	• 공격자는 피해자 환경 내에 새로운 클라우드 사용자 계정 또는 서비스 계정(Azure 서비스 계정, GCP 서비스 계정, AWS IAM 사용자)을 생성할 수 있음
T1546	Event Triggered Execution	• 공격자는 특정 이벤트가 발생했을 때, 자신의 악성 코드가 자동으로 실행되도록 설정해 지속성을 확보할 수 있음 • Pacu 악성코드는 CloudFormation 템플릿이 버킷에 업로드 될 때 악성 Lambda 함수를 트리거하도록 알림 설정 가능

TID	기법 구분	기법 설명
T1525	Implant Internal Image	• 공격자는 환경에 접근한 후에도 지속성을 확보하기 위해 클라우드 또는 컨테이너 이미지에 악성코드를 심을 수 있음 • AWS AMI, Google Cloud Platform(GCP) 이미지, Azure 이미지뿐만 아니라 Docker와 같은 널리 사용되는 컨테이너 런타임에도 악성 코드를 심거나 백도어를 설치할 수 있음
T1556.006	Modify Authentication Process: Multi-Factor Authentication	• 공격자는 다중 인증(MFA) 시스템을 비활성화하거나 수정해 손상된 계정에 지속적으로 접근할 수 있음 • Azure AD 조건부 액세스 정책에서 계정을 제외하거나, 공격자가 제어하는 새로운 MFA 방법을 등록하는 등의 방법으로 우회할 수 있음
T1556.007	Modify Authentication Process: Hybrid Identity	• 공격자는 온프레미스 사용자 ID에 연결된 클라우드 인증 프로세스에 패치를 적용하거나, 수정하거나, 다른 방법으로 백도어를 설치해 일반적인 인증 메커니즘을 우회, 자격 증명에 액세스하고 계정에 대한 지속적인 액세스를 허용할 수 있음 • 공격자는 PTA 에이전트를 실행하는 온프레미스 서버에서 악성 DLL을 Entra ID 인증 프로세스에 삽입할 수 있음 • AD FS를 사용하는 환경에서는 악성 DLL을 로드할 수 있도록 구성 파일을 수정해 AD FS 정책을 우회할 수 있음
T1556.009	Modify Authentication Process: Conditional Access Policies	• 공격자는 조건부 액세스 정책을 비활성화하거나 수정해 침해된 계정에 대한 지속적인 접근을 허용할 수 있음

4) Privilege Escalation (권한 상승)

공격자가 침해 유입을 통해 얻은 제한된 권한을 넘어, 시스템이나 데이터에 대한 더 높은 수준의 제어권을 획득하는 단계이다. 클라우드 환경에서 권한 상승은 주로 복잡하고 세분화된 클라우드 IAM 시스템의 설정 오류나 의도된 기능을 악용하는 기법이 존재한다.

[표 14] MITRE ATT&CK Cloud Matrix의 Privilege Escalation 전술에서 사용되는 기법 목록

TID	기법 구분	기법 설명
T1548.005	Abuse Elevation Control Mechanism: Temporary Elevated Cloud Access	• 공격자는 클라우드 리소스에 대한 일시적으로 높은 접근 권한을 획득할 수 있는 권한 구성을 악용할 수 있음 • 적시 접근(Just-in-time access)은 클라우드 계정에 세분화되고 임시적인 방식으로 추가 역할을 부여하는 기능으로 계정은 매일 필요한 권한만 사용해 작업하고 필요에 따라 추가 권한을 요청 가능
T1098.001	Account Manipulation: Additional Cloud Credentials	• 공격자는 AWS의 API나 GCP 명령을 사용해 계정에 액세스 키를 추가할 수 있으며, AWS API를 사용해 관리 콘솔에 로그인하는 데 사용할 수 있는 비밀번호 추가할 수 있음 • 대상 계정의 권한이 요청 계정과 다른 경우 공격자는 클라우드 환경에서 권한을 확대할 수도 있음 • Entra ID 환경에서 애플리케이션 관리자 역할이 있는 공격자는 애플리케이션의 서비스 주체에 새 자격 증명 추가 가능
T1098.003	Account Manipulation: Additional Cloud Roles	• 공격자는 탈취한 기존 계정에 역할을 추가해 권한 상승 가능 • AWS 환경에서 공격자는 CreatePolicyVersion API를 사용해 IAM 정책의 새 버전을 정의하거나, AttachUserPolicy API를 사용해 손상된 사용자 계정에 추가적 또는 고유한 권한이 존재하는 IAM 정책 연결 가능
T1484.002	Domain or Tenant Policy Modification: Trust Modification	• 공격자는 새로운 도메인이나 기존 도메인 신뢰의 속성을 수정하거나, 신뢰 관계 구성을 변경해 권한 상승 가능 • 사용자 ID가 페더레이션되었는지 여부와 같은 신뢰 세부 정보를 공유 리소스에 액세스하기 위해 도메인 또는 테넌트 간에 인증 및 권한 부여 속성 적용 • 이러한 신뢰를 조작하면 공격자는 자신이 제어하는 객체를 추가하도록 설정을 수정해 권한을 상승시킬 수 있음
T1078.004	Valid Accounts: Cloud Accounts	• 클라우드 계정은 환경 내 다양한 수단을 통해 임시 상승된 클라우드 액세스 또는 기타 권한을 획득할 수 있음 • Azure 환경에서 공격자는 연결된 Azure 리소스가 액세스 토큰을 요청할 수 있도록 하는 Azure 관리 ID 악용 가능

5) Defense Evasion (방어 회피)

공격자가 침투 과정 전반에 걸쳐 탐지를 피하기 위해 사용하는 기술들로 구성된다. 클라우드 환경에서는 기존의 악성코드 은닉 기술 외에도 클라우드 인프라 자체의 구성과 기능을 조작해 방어 체계를 무력화하는 방식이 존재한다. 대표적으로 공격자는 시스템 내에 저장되어 있는 이벤트 로그를 삭제하는데, 클라우드 환경에서는 AWS CloudTrail, Azure Monitor, Google Cloud Audit Logs와 같은 클라우드 로깅 서비스를 비활성화하거나 삭제할 수 있다.

[표 15] MITRE ATT&CK Cloud Matrix의 Defense Evasion 전술에서 사용되는 기법 목록

TID	기법 구분	기법 설명
T1548.005	Abuse Elevation Control Mechanism: Temporary Elevated Cloud Access	• 공격자는 클라우드 리소스에 일시적으로 상승된 접근 권한을 획득할 수 있도록 허용하는 권한 설정을 악용할 수 있음 • 많은 클라우드 환경에서는 관리자들이 사용자 또는 서비스 계정에 대해 'Just-in-Time 접근 권한 요청', '다른 계정의 가장(Impersonation)', '역할(Role)을 리소스 및 서비스에 전달', '단기적인 고권한 접근 권한 획득'과 같은 권한 부여 가능 • 이런 기능들은 특정 역할로 부여되어야만 사용할 수 있으나, 클라우드 관리자에 의한 권한 설정 실수로 인해 원래 의도되지 않았던 자원에 대한 권한 상승 경로가 생성될 수 있음
T1484.002	Domain or Tenant Policy Modification: Trust Modification	• 공격자는 새로운 도메인 트러스트(Domain Trust)를 추가하거나, 기존 트러스트의 속성을 수정 및 도메인/테넌트 간의 트러스트 관계 구성을 변경함으로써 방어 체계를 우회하거나 권한을 상승시킬 수 있음 • 트러스트 관계를 조작하면 공격자는 자신이 제어하는 객체를 추가하거나 설정을 변경함으로써 권한 상승 또는 방어 우회가 가능해짐
T1672	Email Spoofing	• 공격자는 이메일 헤더 값을 조작해 발신자의 신원을 위조(Spoofing) 할 수 있으며, 이메일 본문뿐만 아니라, 발신자의 이메일 주소가 포함된 From 헤더와 같은 메일 헤더도 조작될 수 있음
T1211	Exploitation for Defense Evasion	• 공격자는 시스템 또는 애플리케이션의 취약점을 악용해 보안 기능을 우회할 수 있음 • SaaS 애플리케이션 등 공개된 인프라 취약점을 통해 은폐된 인프라 구축, 보안 로그 탐지 회피 등이 가능함
T1564.008	Hide Artifacts: Email Hiding Rules	• 공격자는 침해된 사용자의 메일함에서 수신 이메일을 숨기기 위해 이메일 규칙(Email Rules)을 악용할 수 있음 • 공격자는 침해된 계정의 메일함 내에서 이메일 규칙을 설정해 보안 경고, C2 통신, 내부 스피어피싱 이메일에 대한 응답 내용을 삭제하거나 눈에 띄지 않는 폴더로 이동시키거나 보안 사고 알림과 관련된 모든 이메일을 자동으로 수정하거나 삭제하도록 설정할 수 있음

TID	기법 구분	기법 설명
T1562.001	Impair Defenses: Disable or Modify Tools	• 공격자들은 악성코드, 도구 또는 활동이 탐지되는 것을 회피하기 위해 보안 도구를 수정하거나 비활성화할 수 있음 • 클라우드 환경에서는 AWS CloudWatch, Google Cloud Monitor 등의 모니터링 에이전트를 비활성화해 로그 수집 및 알림 기능 무력화
T1562.007	Impair Defenses: Disable or Modify Cloud Firewall	• 클라우드 환경 내의 방화벽을 비활성화하거나 수정해 클라우드 리소스 접근을 제한하는 보안 통제를 우회할 수 있음 • 공격자가 적절한 권한을 획득한 경우, '기본 보안 그룹에 신규 인바인드 규칙(Ingress Rule) 추가', '새로운 보안 그룹을 생성해 TCP/IP 접근을 허용하는 스크립트 또는 도구 실행', '암호화폐 채굴 등 악성 트래픽을 허용하기 위해 정책 설정' 등 방화벽 설정이 가능함 • 또는 클라우드 방화벽을 변경하거나 비활성화 함으로써 'C2 통신 활성화', '클라우드 제어 영역(Control Plane)에서 데이터 영역 (Data Plane)으로 내부 이동', 무차별 대입 공격 및 엔드포인트 서비스 거부를 위한 리소스 노출' 등이 가능함
T1562.008	Impair Defenses: Disable or Modify Cloud Logs	• 공격자는 자신의 활동에 대한 탐지를 회피하기 위해 클라우드 환경의 로깅 기능 및 연동 설정을 비활성화하거나 조작할 수 있음
T1656	Impersonation	• 공격자는 신뢰할 수 있는 인물이나 조직을 사칭해 대상에게 자신을 대신해 특정 행위를 하도록 속이고 설득할 수 있음
T1070.008	Indicator Removal: Clear Mailbox Data	• 공격 활동 흔적을 지우기 위해 메일 데이터 조작 가능
T1556.006	Modify Authentication Process: Multi-Factor Authentication	• 공격자는 MFA가 적용되지 않은 계정을 탈취하거나 MFA 요청 생성과 같은 우회 기법을 사용해 네트워크에 접근한 후 MFA 방어 체계를 수정하거나 비활성화할 수 있음
T1556.007	Modify Authentication Process: Hybrid Identity	• 공격자는 온프레미스 사용자 ID와 연동된 클라우드 인증 프로세스를 패치하거나 수정하거나 백도어를 심어 일반적인 인증 메커니즘을 우회하고 자격 증명을 탈취하며 계정에 대한 지속적인 접근 확보 가능
T1556.009	Modify Authentication Process: Conditional Access Policies	• 침해한 계정에 지속적으로 접근하기 위해 조건부 액세스 정책을 비활성화하거나 수정 가능
T1578.001	Modify Cloud Compute Infrastructure: Create Snapshot	• 공격자는 클라우드 계정 내에서 스냅샷 또는 데이터 백업을 생성해 방어 체계 우회 가능 • 공격자는 클라우드 인스턴스를 생성한 후 생성한 스냅샷을 인스턴스에 마운트하고 SSH 접속이 가능한 방화벽 정책 등의 공격자에게 접근 권한을 부여하는 정책을 적용함 • Revert Cloud Instance 기법과는 달리 별도의 인스턴스를 만들어 우회하는 접근 방식임 • Pacu 도구를 활용해 EBS 볼륨 및 RDS 인스턴스의 스냅샷 생성 가능

TID	기법 구분	기법 설명
T1578.002	Modify Cloud Compute Infrastructure: Create Cloud Instance	• 공격자는 클라우드 계정 내 컴퓨트 서비스에서 새로운 인스턴스 또는 가상 머신(VM)을 생성해 방어 체계를 우회할 수 있음 • 새로운 인스턴스를 생성하면, 기존 인스턴스에 적용된 방화벽 규칙이나 권한 제어를 우회할 수 있음 • 인스턴스를 새로 생성하는 방식은 현재 실행 중인 인스턴스에는 영향을 주지 않으면서 동일한 클라우드 환경 내에서 은밀하게 악성 활동을 수행할 수 있게 함
T1578.003	Modify Cloud Compute Infrastructure: Delete Cloud Instance	• 공격자는 악성 활동을 수행한 후, 자신의 흔적을 감추고 탐지 회피를 위해 클라우드 인스턴스를 삭제할 수 있음
T1578.004	Modify Cloud Compute Infrastructure: Revert Cloud Instance	• 공격자는 악성 행위를 수행한 후 탐지를 회피하고 자신의 흔적을 지우기 위해 클라우드 인스턴스의 변경 사항을 되돌릴 수 있음 • 또 다른 형태의 기법으로는 인스턴스에 연결된 임시 저장소(Temporary Storage)를 활용하는 방식 존재
T1578.005	Modify Cloud Compute Infrastructure: Modify Cloud Compute Configurations	• 공격자는 클라우드 컴퓨팅 인프라 크기, 배치 위치, 사용 가능한 리소스에 직접 영향을 주는 설정을 변경해 방어 체계 우회 • 공격자가 클라우드 환경을 장악한 경우에도 자신의 목적(예. 리소스 하이재킹)을 달성하기 위해 할당량 조정을 요청할 수 있으며, 이는 피해자의 전체 할당량을 소진하지 않고도 은밀하게 악성 작업을 수행할 수 있도록 함 • 또한, 가상 머신(VM)의 크기 제한과 같은 테넌트 수준 정책을 변경함으로써 허용된 리소스 사용량을 증가시킬 수 있으며, 지원되지 않거나 사용되지 않는 클라우드 리전을 활성화해 리소스를 특정 지역에 배포할 수 있도록 변경
T1666	Modify Cloud Resource Hierarchy	• 공격자는 방어 체계를 회피하기 위해 IaaS(서비스형 인프라) 환경의 계층 구조를 수정하려 할 수 있음
T1535	Unused/Unsupported Cloud Regions	• 공격자는 탐지를 회피하기 위해 사용되지 않는 지리적 서비스 지역(Region)에 클라우드 인스턴스를 생성할 수 있음 • 사용자는 일반적으로 사용 가능한 리전 중 일부만을 활용하고, 나머지 지역은 적극적으로 모니터링하지 않을 수 있는 점을 악용해 사용되지 않는 리전에 리소스를 생성함 • 이와 유사한 변형 방식으로는, 클라우드 리전 간 보안 기능 차이를 악용하는 행위 존재하며, 공격자는 고급 탐지 기능을 제공하지 않는 리전을 선택해 자신들의 공격 행위 은폐 가능

TID	기법 구분	기법 설명
T1550.001	Use Alternate Authentication Material: Application Access Token	• 공격자는 탈취한 애플리케이션 액세스 토큰을 이용해 일반적인 인증 절차를 우회하고, 원격 시스템의 제한된 계정, 정보 또는 서비스에 접근할 수 있음 • 이러한 토큰은 보통 사용자나 서비스로부터 탈취되며, 로그인 자격 증명 없이도 사용할 수 있음 • 탈취된 액세스 토큰은 다른 서비스로의 침투를 위한 초기 단계로 활용될 수 있음 • API를 통한 직접적인 접근은 MFA(다단계 인증)를 무력화시킬 수 있으며, 비밀번호 변경과 같은 직관적인 대응책으로도 방어하기 어려움
T1550.004	Use Alternate Authentication Material: Web Session Cookie	• 공격자는 탈취한 세션 쿠키를 사용해 웹 애플리케이션 및 서비스에 인증할 수 있으며, 이미 인증된 세션을 재사용하므로 일부 다단계 인증(MFA) 프로토콜을 우회할 수 있음
T1078.001	Valid Accounts: Default Accounts	• 공격자는 초기 접근, 지속성, 권한 상승 또는 방어 회피를 위해 기본 계정의 자격 증명을 획득하고 악용할 수 있음 • 기본 계정에는 또한 AWS의 root 사용자 계정, ESXi의 root 사용자 계정, Kubernetes의 기본 서비스 계정 등 다른 시스템, 소프트웨어 또는 장비에 공장 출하 시 또는 제공자에 의해 설정된 계정도 포함됨
T1078.004	Valid Accounts: Cloud Accounts	• 클라우드 환경의 유효한 계정은 공격자들이 초기 접근(Initial Access), 지속성(Persistence), 권한 상승(Privilege Escalation), 또는 방어 회피(Defense Evasion)를 달성하기 위해 허용 • 서비스 계정이나 사용자 계정은 공격자들이 환경에 접근하기 위해 공격 대상으로 삼을 수 있음 • 공격자는 침해된 클라우드 계정에 대해 장기간 사용할 수 있는 추가 클라우드 자격 증명을 생성함으로써 환경 내에서 지속성을 유지하고 다중 인증(MFA)와 같은 보안 통제를 우회하는데 사용

6) Credential Access (크리덴셜 획득)

공격자가 시스템 및 네트워크 내에서 상위 권한을 획득하고, 내부 이동을 위한 계정 정보, 토큰, 키와 같은 자격 증명을 탈취하는 것을 목표로 한다. 클라우드 환경에서의 크리덴셜 획득은 기존 온프레미스 환경에서 사용되는 방식과 클라우드 환경의 방식이 결합된다. 무차별 대입 공격은 외부에서 접근 가능한 클라우드 서비스를 대상으로 수행되며, AWS Secrets Manager, Azure Key Vault, GCP Secret Manager와 같은 중앙 집중식 자격 증명 관리 시스템을 직접 공격할 수도 있다.

[표 16] MITRE ATT&CK Cloud Matrix의 Credential Access 전술에서 사용되는 기법 목록

TID	기법 구분	기법 설명
T1110.001	Brute Force: Password Guessing	• 시스템이나 환경 내의 합법적인 자격 증명에 대한 사전 지식이 없는 공격자는 계정에 접근하기 위해 비밀번호 추측 • 일반적으로 공격 대상이 되는 서비스(SSH, Telnet, FTP 등)가 아닌 클라우드 기반 애플리케이션과 Office 365와 같은 외부 이메일 애플리케이션을 대상으로 공격을 수행할 수 있음
T1110.002	Brute Force: Password Cracking	• 공격자는 암호 해시와 같은 자격 증명 자료를 획득하면, 사용 가능한 자격 증명을 복구하기 위해 암호 해독 시도 • 해시 계산에 사용되는 비밀번호를 추측하는 기술을 사용하거나 미리 계산된 레인보우 테이블을 사용해 해시 해독 가능
T1110.003	Brute Force: Password Spraying	• 유효한 계정 자격 증명을 획득하기 위해 여러 계정에 대해 일반적으로 사용되는 단일 비밀번호 또는 소규모 비밀번호 목록을 사용할 수 있음
T1110.004	Brute Force: Credential Stuffing	• 공격자는 해당 환경과 관련 없는 계정에서 얻은 자격 증명을 사용해 대상 계정에 접근할 수 있음 • 사용자가 개인 계정과 비즈니스 계정에서 동일한 비밀번호를 사용하는 경향을 악용해 계정을 탈취할 수 있음
T1555.006	Credentials from Password Stores: Cloud Secrets Management Stores	• 공격자는 AWS Secrets Manager, GCP Secret Manager, Azure Key Vault, Terraform Vault와 같은 클라우드 기반 비밀 관리 솔루션에서 자격 증명을 획득할 수 있음 • 공격자가 클라우드 환경에서 충분한 권한을 획득할 수 있는 경우, AWS의 get-secret-value, GCP의 gcloud secrets describe, Azure의 az key vault secret show 등의 명령을 통해 자격 증명 요청 가능
T1212	Exploitation for Credential Access	• 공격자는 자격 증명을 수집하기 위해 소프트웨어 취약점을 악용할 수 있음 • 자격 증명 및 인증 메커니즘은 공격자가 유용한 자격 증명에 접근하거나 시스템에 대한 인증된 접근 권한을 얻는 프로세스를 우회하는 수단으로 악용될 수 있으며, 공격자는 퍼블릭 클라우드 인프라의 취약점을 악용해 의도치 않은 인증 토큰 생성 및 갱신을 허용할 수 있음

TID	기법 구분	기법 설명
T1606.001	Forge Web Credentials: Web Cookies	• 공격자는 웹 애플리케이션이나 인터넷 서비스에 접근하는 데 사용할 수 있는 웹 쿠키를 위조할 수 있음 • 웹 애플리케이션과 서비스(클라우드 SaaS 환경 또는 온프레미스 서버에 호스팅됨)는 종종 세션 쿠키를 사용해 사용자 접근을 인증하고 권한을 부여함 • 공격자는 웹 쿠키를 사용해 다중 인증 요소 및 기타 인증 보호 메커니즘을 우회할 수 있음
T1606.002	Forge Web Credentials: SAML Tokens	• 공격자는 유효한 SAML 토큰 서명 인증서를 소지하고 있는 경우, 모든 권한 클레임 및 유효 기간을 사용해 SAML 토큰을 위조할 수 있음 • 위조된 SAML 토큰을 통해 공격자는 SAML 2.0을 SSO(단일 로그인) 메커니즘으로 사용하는 서비스에서 인증 가능 • 높은 권한이 있는 계정을 나타내는 SAML 토큰이 위조될 경우, 공격자는 Entra ID 관리 권한을 획득할 수 있음
T1556.007	Modify Authentication Process: Hybrid Identity	• 공격자는 온프레미스 사용자 ID에 연결된 클라우드 인증 프로세스에 패치를 적용하거나, 수정하거나, 다른 방법으로 백도어를 설치해 일반적인 인증 메커니즘을 우회하고, 자격 증명에 접근할 수 있음 • 하이브리드 ID에 연결된 인증 프로세스를 수정함으로써 공격자는 클라우드 리소스에 대한 지속적인 권한 있는 접근을 확보 가능
T1621	Multi-Factor Authentication Request Generation	• 공격자는 다중 인증 요소(MFA) 메커니즘을 우회하고 사용자에게 MFA 요청을 전송해 계정에 액세스하려고 시도 가능 • 공격자가 피해자 계정에 대한 자격 증명이 없는 경우, 셀프 서비스 비밀번호 재설정(SSPR)을 위해 이 옵션이 구성된 경우에도 자동 푸시 알림 생성 기능을 악용할 수 있음
T1528	Steal Application Access Token	• 공격자는 원격 시스템과 리소스에 접근하기 위한 자격 증명을 획득하는 수단으로 애플리케이션 액세스 토큰을 탈취할 수 있음 • 애플리케이션 액세스 토큰은 사용자 또는 서비스를 대신해 승인된 API 요청을 하는 데 사용되며 일반적으로 클라우드 및 컨테이너 기반 애플리케이션과 SaaS 리소스에 접근하는 방법으로 사용됨 • 클라우드 및 컨테이너화된 환경에서 계정 API 토큰을 탈취한 공격자는 계정의 권한으로 데이터에 접근하고 작업을 수행 가능
T1649	Steal or Forge Authentication Certificates	• 공격자는 원격 시스템이나 리소스에 접근하기 위한 인증에 사용되는 인증서를 훔치거나 위조할 수 있음 • 인증서 관련 구성 오류는 사용자가 인증서와 연결된 ID(SAN)를 통해 권한이 있는 계정이나 권한을 획득할 수 있음
T1539	Steal Web Session Cookie	• 웹 애플리케이션이나 서비스 세션 쿠키를 탈취해 자격 증명 없이도 인증된 사용자로 웹 애플리케이션이나 인터넷 서비스에 접근할 수 있음 • 웹 애플리케이션과 서비스는 사용자가 웹사이트에 인증한 후 세션 쿠키를 인증 토큰으로 사용하는 경우가 많음 • 대상 시스템의 다른 애플리케이션(클라우드 서비스에 인증하는 앱)이 민감한 인증 쿠키를 메모리에 저장할 수도 있으며, 세션 쿠키는 일부 다중 인증 프로토콜을 우회하는 데 사용될 수 있음

TID	기법 구분	기법 설명
T1552.001	Unsecured Credentials: Credentials In Files	• 공격자는 로컬 파일 시스템과 원격 파일 공유에서 안전하지 않게 저장된 자격 증명이 포함된 파일을 검색할 수 있음 • 클라우드 및/또는 컨테이너화된 환경에서 인증된 사용자 및 서비스 계정 자격 증명은 종종 로컬 구성 및 자격 증명 파일에 저장
T1552.005	Unsecured Credentials: Cloud Instance Metadata API	• 공격자는 자격 증명 및 기타 민감한 데이터를 수집하기 위해 Cloud Instance Metadata API에 액세스하려고 시도 • 클라우드 서비스 제공업체는 실행 중인 가상 인스턴스에 제공되는 서비스인 클라우드 인스턴스 메타데이터 API 지원 • 이 API를 통해 애플리케이션은 실행 중인 가상 인스턴스에 대한 정보에 접근할 수 있음 • 공격자가 실행 중인 가상 인스턴스에 존재하는 경우, 인스턴스 메타데이터 API에 직접 쿼리해 추가 리소스에 대한 접근 권한을 부여하는 자격 증명을 식별할 수 있음
T1552.008	Unsecured Credentials: Chat Messages	• 공격자는 사용자 채팅 메시지를 통해 저장되거나 전달되는 자격 증명을 획득할 수 있음 • 사용자는 기업 내부 커뮤니케이션 채널(사설 또는 공개)을 통해 다양한 형태의 자격 증명(계정 명, 비밀번호, API 키, 인증 토큰 등)을 공유할 수 있으며, 이러한 자격 증명은 내부 이동이나 권한 상승과 같은 후속 활동을 수행하는 데 악용될 수 있음

7) Discovery (탐색)

공격자가 시스템이나 네트워크에 대한 정보를 수집해 환경을 파악하고, 다음 공격 목표를 설정하기 위해 사용하는 공격 방식들로 구성된다. 클라우드 환경에서 정보 수집 활동은 API를 통해 수행되는 경우가 많으며, 공격자는 인프라, 스토리지, 서비스, 계정, 권한 등 환경의 모든 정보를 수집할 수 있다.

[표 17] MITRE ATT&CK Cloud Matrix의 Discovery 전술에서 사용되는 기법 목록

TID	기법 구분	기법 설명
T1087.004	Account Discovery: Cloud Account	• 공격자는 클라우드 계정 목록을 수집하려고 시도할 수 있음 • Azure CLI(AZ CLI)에서는 'azad user list', AWS에서는 'aws iam list-users', 'aws iam list-roles', GCP에서는 'gcloud iam service-accounts list', 'gcloud projects get-iam-policy' 명령을 통해 획득할 수 있음
T1580	Cloud Infrastructure Discovery	• 서비스형 인프라(IaaS) 환경에서 사용 가능한 인프라와 리소스를 수집하려고 시도할 수 있음 • AWS는 DescribeInstances API를 통해, GCP의 Cloud SDK CLI는 'gcloud compute instances list' 명령을 통해 획득 가능
T1538	Cloud Service Dashboard	• 탈취한 자격 증명을 사용해 클라우드 서비스 대시보드 GUI를 통해 특정 서비스, 리소스, 기능 등 운영 클라우드 환경에서 유용한 정보 획득 가능
T1526	Cloud Service Discovery	• 공격자는 시스템에 접근한 후 시스템에서 실행 중인 클라우드 서비스를 열거하려고 시도할 수 있음
T1619	Cloud Storage Object Discovery	• 클라우드 스토리지 인프라에 있는 객체를 열거할 수 있으며, 공격자는 이 정보를 사용해 클라우드 스토리지의 모든 객체 또는 특정 객체를 요청하는 행위를 수행할 수 있음 • 클라우드 서비스 제공업체는 사용자가 클라우드 스토리지에 저장된 객체를 열거할 수 있는 API를 제공
T1654	Log Enumeration	• 유용한 정보를 획득하기 위해 시스템 및 서비스 로그 분석 가능 • 클라우드 환경에서 공격자는 Azure VM 에이전트와 같은 유틸리티(CollectGuestLogs.exe)를 활용해 클라우드 인프라에서 보안 로그를 수집할 수 있음
T1046	Network Service Discovery	• 원격 호스트 및 로컬 네트워크 인프라 장치에서 실행 중인 서비스 목록을 확보하려고 시도할 수 있음 • 클라우드 환경 내에서 공격자는 다른 클라우드 호스트에서 실행되는 서비스를 발견하려고 시도할 수 있음 • 클라우드 환경이 온프레미스 환경에 연결된 경우, 공격자는 클라우드가 아닌 시스템에서 실행되는 서비스도 식별
T1040	Network Sniffing	• 네트워크 트래픽을 수동적으로 스니핑해 네트워크를 통해 전달되는 인증 자료를 포함한 환경 정보를 탈취할 수 있음 • 클라우드 기반 환경에서 공격자는 트래픽 미러링 서비스를 사용해 가상 머신의 네트워크 트래픽을 스니핑할 수 있음

TID	기법 구분	기법 설명
T1201	Password Policy Discovery	• 공격자는 기업 네트워크 또는 클라우드 환경에서 사용되는 암호 정책에 대한 자세한 정보에 접근을 시도할 수 있음 • AWS에서는 GetAccountPasswordPolicy API를 사용해 비밀번호 정책을 획득할 수 있음
T1069.003	Permission Groups Discovery: Cloud Groups	• 클라우드 그룹과 권한 설정을 수집하려고 시도할 수 있음 • Azure CLI(AZ CLI)와 Google Cloud Identity Provider API는 권한 그룹을 얻기 위한 인터페이스를 제공 • 공격자는 이 정보를 이용해 특정 객체에 대한 권한이 있는 계정을 표적으로 삼거나, 이미 침해된 계정을 활용해 객체에 접근할 수 있음
T1518.001	Software Discovery: Security Software Discovery	• 시스템이나 클라우드 환경에 설치된 보안 소프트웨어, 구성, 방어 도구 및 센서 목록을 수집하려고 시도할 수 있음 • 공격자는 AWS CloudWatch 에이전트, Azure VM 에이전트, Google Cloud Monitor 에이전트와 같이 컴퓨팅 인프라에 설치된 클라우드 네이티브 보안 소프트웨어를 발견할 수 있음
T1082	System Information Discovery	• 버전, 패치, 핫픽스, 서비스 팩, 아키텍처 등 운영 체제 및 하드웨어에 대한 자세한 정보를 얻으려 할 수 있음
T1614	System Location Discovery	• 피해자 호스트의 지리적 위치를 파악하기 위해 정보 수집 가능 • 공격자는 시간대, 키보드 레이아웃 및/또는 언어 설정과 같은 다양한 시스템 검사를 사용해 시스템의 위치를 유추하려고 시도할 수 있음 • 클라우드 환경에서 인스턴스의 가용성 영역은 인스턴스에서 인스턴스 메타데이터 서비스에 접근해 수집할 수도 있음
T1049	System Network Connections Discovery	• 네트워크를 통해 정보를 쿼리해 현재 액세스 중인 손상된 시스템이나 원격 시스템에서 네트워크 연결 목록을 얻으려고 시도 • 클라우드 기반 환경의 일부인 시스템에 접근하는 공격자는 연결된 시스템과 서비스를 확인하기 위해 가상 사설 클라우드 또는 가상 네트워크를 매핑할 수 있음 • 검색 결과 정보에는 공격자의 목표와 관련된 네트워크 클라우드 환경에 대한 세부 정보가 포함될 수 있음

8) Lateral Movement (침해 전파)

공격자가 침입한 시스템에서 네트워크 내의 다른 시스템으로 이동해 접근 범위를 확장하는 데 사용하는 기술로 구성된다. 클라우드 환경에서의 침해 전파는 클라우드 서비스 간의 신뢰 관계와 공유 리소스를 악용해 내부 다른 시스템에서 악성 파일이 실행되게 한다. SaaS 플랫폼이 보편화되면서, 클라우드 기반 공유 드라이브나 코드 저장소를 통해 악성 파일을 유포할 수 있다.

[표 18] MITRE ATT&CK Cloud Matrix의 Lateral Movement 전술에서 사용되는 기법 목록

TID	기법 구분	기법 설명
T1534	Internal Spearphishing	• 공격자는 환경 내 계정이나 시스템에 이미 접근 권한을 획득한 후, 내부 스피어피싱을 사용해 추가 정보에 접근하거나 동일 환경 내 다른 계정을 침해할 수 있음 • 공격자는 내부 스피어피싱의 일부로 첨부 파일이나 링크를 활용해 페이로드를 전달하거나, 외부 사이트로 리디렉션해 피싱 사이트에서 자격 증명을 탈취할 수 있음
T1021.007	Remote Services: Cloud Services	• 온프레미스 사용자 계정과 동기화되거나, 공유된 계정을 사용해 침해된 환경 내에서 접근 가능한 클라우드 서비스에 로그인 가능 • 공격자는 웹 콘솔이나, 클라우드 명령줄 인터페이스를 통해 Cloud API, Azure PowerShell, Google Cloud CLI 명령을 사용해 사용 가능한 클라우드 서비스에 연결할 수 있음
T1021.008	Remote Services: Direct Cloud VM Connections	• 유효한 계정을 활용해 클라우드 네이티브 방식을 통해 접근 가능한 클라우드 호스팅 컴퓨팅 인프라에 직접 로그인할 수 있음 • 클라우드 제공업체는 Azure Serial Console, AWS EC2 Instance Connect, AWS System Manager와 같은 클라우드 API를 통해 접근할 수 있는 가상 인프라에 대한 대화형 연결을 제공함 • 공격자는 클라우드 기반 방식을 사용해 가상 인프라에 직접 액세스하고 환경을 전환할 수 있음
T1072	Software Deployment Tools	• 기업 내에 설치된 중앙 집중식 소프트웨어 제품군에 접근해 명령을 실행하고 내부 이동을 수행할 수 있음 • SaaS 기반 구성 관리 서비스는 클라우드 호스팅 인스턴스에서 광범위한 클라우드 관리 명령을 지원하고 온프레미스 엔드포인트에 임의의 명령을 실행할 수 있도록 함 • Microsoft Configuration Manager를 사용하면 글로벌 관리자 또는 Intune 관리자가 Entra ID에 연결된 온프레미스 장치에서 SYSTEM 권한으로 스크립트를 실행할 수 있음
T1080	Taint Shared Content	• 공격자는 네트워크 드라이브나 내부 코드 저장소와 같은 공유 저장 위치에 콘텐츠를 추가해 원격 시스템에 페이로드 전송 • 네트워크 드라이브나 기타 공유 위치에 저장된 콘텐츠는 악성 프로그램, 스크립트 또는 악성 코드를 정상적인 파일에 추가할 수 있으며, 공격자는 감염된 공유 콘텐츠를 이용해 내부 이동을 수행할 수 있음

TID	기법 구분	기법 설명
T1550.001	Use Alternate Authentication Material: Application Access Token	• 공격자는 애플리케이션 액세스 토큰을 탈취해 일반적인 인증 절차를 우회하고 원격 시스템의 제한된 계정, 정보 또는 서비스에 접근할 수 있음 • 애플리케이션 액세스 토큰은 사용자 또는 서비스를 대신해 승인된 API 요청을 하는 데 사용되며 일반적으로 클라우드, 컨테이너 기반 애플리케이션 및 SaaS의 리소스에 액세스하는 데 사용 • AWS 및 GCP 환경에서 공격자는 다른 사용자 계정의 권한을 가진 단기 액세스 토큰 요청을 트리거할 수 있으며, 해당 토큰을 사용해 데이터를 요청하거나 기존 계정으로는 수행할 수 없었던 작업 수행
T1550.004	Use Alternate Authentication Material: Web Session Cookie	• 탈취한 세션 쿠키를 사용해 웹 애플리케이션 및 서비스에 인증 가능 • 공격자는 쿠키를 획득한 후, 민감한 정보에 액세스하거나 이메일을 읽거나 피해자 계정에 권한이 있는 작업을 수행할 수 있음

9) Collection (수집)

공격자가 자신의 목표를 달성하기 위해 데이터를 식별하고 수집하는 단계이다. 클라우드 환경은 많은 데이터가 중앙 집중화되어 저장되는 특성 때문에 공격자에게 중요한 수집 대상이 된다. 공격자는 클라우드 환경의 자동화 기능과 API를 악용해 데이터를 수집할 수 있다.

[표 19] MITRE ATT&CK Cloud Matrix의 Collection 전술에서 사용되는 기법 목록

TID	기법 구분	기법 설명
T1119	Automated Collection	• 시스템이나 네트워크 내에 침투한 공격자는 자동화된 기술을 사용해 내부 데이터를 수집할 수 있음 • 클라우드 기반 환경에서 공격자는 클라우드 API, 데이터 파이프라인, 명령줄 인터페이스 또는 ETL(추출, 변환 및 로드) 서비스를 사용해 자동으로 데이터를 수집할 수도 있음
T1530	Data from Cloud Storage	• 클라우드 스토리지 솔루션에서 민감한 데이터 수집 가능 • 인증되지 않은 사용자에게 의도치 않게 공개 접근을 허용하거나, 모든 사용자에게 지나치게 광범위한 접근 권한을 부여하거나, 심지어 기본 사용자 권한 없이도 ID 액세스 관리 시스템의 통제 밖에 있는 익명의 사용자에게 접근하는 경우가 있음
T1213	Data from Information Repositories	• 공격자는 정보 저장소를 활용해 정보를 획득할 수 있음 • 정보 저장소는 일반적으로 사용자 간 협업이나 정보 공유를 용이하게 하기 위해 정보를 저장할 수 있는 도구이며, 신원 정보 접근, 측면 이동, 방어 회피 등 공격자의 추가 목표 달성에 도움이 될 수 있는 다양한 데이터를 저장하거나 대상 정보에 직접 접근할 수 있음
T1074.002	Data Staged: Remote Data Staging	• 공격자는 유출 전에 여러 시스템에서 수집한 데이터를 중앙 또는 단일 시스템의 디렉터리에 저장할 수 있음 • 클라우드 환경에서 공격자는 유출되기 전에 특정 인스턴스 또는 가상 머신 내에 데이터를 저장하거나 클라우드 인스턴스를 생성하고 해당 인스턴스에 데이터를 저장할 수 있음
T1114.002	Email Collection: Remote Email Collection	• 공격자는 Office 365 또는 Google Workspace를 공격 대상으로 선정해 민감한 정보를 수집할 수 있음
T1114.003	Email Collection: Email Forwarding Rule	• 공격자는 민감한 정보를 수집하기 위해 이메일 전달 규칙을 설정할 수 있음 • 공격자는 이메일 전달 규칙을 악용해 피해자의 활동을 모니터링하고, 정보를 탈취하고, 피해자 또는 피해자 조직에 대한 정보를 추가로 확보할 수 있음

10) Exfiltration (유출)

공격자가 수집한 데이터를 외부 시스템으로 유출하는 단계이다. 공격자는 데이터를 수집한 후, 이를 제거하는 과정에서 탐지를 피하기 위해 데이터를 압축 및 암호화 등을 통해 패키징하는 경우가 존재한다. 클라우드 환경에서의 데이터 유출은 대역폭 제한이 거의 없고, 암호화된 채널(HTTPS) 사용이 보편적이기 때문에 합법적인 클라우드 서비스 트래픽과 악의적인 데이터 유출 트래픽을 구분하기 어렵다.

[표 20] MITRE ATT&CK Cloud Matrix의 Exfiltration 전술에서 사용되는 기법 목록

TID	기법 구분	기법 설명
T1048	Exfiltration Over Alternative Protocol	• 기존 C2 채널과 다른 프로토콜을 통해 데이터 유출 • IaaS 및 SaaS 플랫폼(AWS S3, Microsoft Exchange, Microsoft SharePoint 등)은 웹 콘솔이나 클라우드 API를 통해 파일, 소스 코드 및 기타 중요한 정보에 대해 다운로드 지원
T1567.004	Exfiltration Over Web Service: Exfiltration Over Webhook	• 기존 C2 채널이 아닌 Webhook 엔드포인트로 데이터 유출 • Discord, Slack과 같은 서비스는 Github, Jira, Trello와 같은 다른 서비스에서 사용할 수 있는 Webhook 엔드포인트 생성을 지원함
T1537	Transfer Data to Cloud Account	• 공격자는 클라우드 환경의 공유/동기화 및 백업 생성 등을 통해 데이터를 다른 클라우드 계정으로 전송해 데이터를 유출할 수 있음 • 공격자는 익명의 파일 공유 링크나 Azure의 SAS(공유 액세스 서명) URI를 생성하는 등 클라우드 기반 파일 공유 서비스를 악용해 공격자 클라우드 계정으로 데이터를 공유할 수 있음

11) Impact (영향)

공격자가 시스템 및 데이터의 가용성이나 무결성을 조작, 중단, 파괴해 자신의 목표를 달성하는 단계이다. 클라우드 환경에서 영향 단계의 공격은 막대한 금전적 손실과 서비스 중단, 신뢰도 하락을 발생시킬 수 있다. 공격자는 클라우드 자원을 무단으로 사용해 암호 화폐를 채굴할 수 있으며, 데이터 유출 이후 클라우드 스토리지 객체, 머신 이미지, 데이터베이스 인스턴스 등 운영에 필수적인 데이터를 영구적으로 삭제하거나 복구 불가능하도록 덮어쓸 수 있다.

[표 21] MITRE ATT&CK Cloud Matrix의 Impact 전술에서 사용되는 기법 목록

TID	기법 구분	기법 설명
T1531	Account Access Removal	• 사용자가 사용하는 계정에 대한 접근을 차단해 시스템 및 네트워크 자원에 접근하지 못하도록 할 수 있음 • 계정 접근 권한을 제거하기 위해 계정 삭제, 잠금 또는 조작(자격 증명 변경 등) 할 수 있음
T1485.001	Data Destruction: Lifecycle-Triggered Deletion	• 클라우드 스토리지 버킷의 수명 주기 정책을 수정해 저장된 모든 객체를 삭제할 수 있음 • 클라우드 스토리지 버킷을 사용하면 사용자가 특정 기간 후에 개체의 마이그레이션, 보관 또는 삭제를 자동화하는 수명 주기 정책을 설정할 수 있음 • 공격자가 해당 정책을 수정할 수 있는 권한이 존재하는 경우, 모든 개체를 한 번에 삭제할 수 있음
T1486	Data Encrypted for Impact	• 공격자는 대상 시스템이나 네트워크 내 데이터를 암호화해 시스템 및 네트워크 자원에 접근하지 못하도록 할 수 있음 • 클라우드 환경에서는 손상된 계정 내의 스토리지 객체도 암호화될 수 있음 • AWS 환경에서 공격자는 AWS의 고객 제공 키를 사용한 서버 측 암호화(SSE-C)와 같은 서비스를 활용해 데이터를 암호화할 수 있음
T1491.002	Defacement: External Defacement	• 공격자는 조직의 시스템을 통해 사용자에게 메시지를 전달하거나 위협을 줄 수 있음 • 사용자가 시스템을 신뢰하지 못하도록 만들고 정치적 메시지 또는 선전을 유포하기 위해 발생할 수 있음
T1667	Email Bombing	• 공격자는 특정 이메일 주소에 대량의 메시지를 발송할 수 있음 • 정상적인 이메일을 정확히 수신 받지 못해 비즈니스 운영이 중단될 수 있음
T1499	Endpoint Denial of Service	• 공격자는 서비스에 접근하지 못하도록 엔드포인트 서비스 거부(DoS) 공격을 수행할 수 있음
T1657	Financial Theft	• 공격자는 랜섬웨어, 비즈니스 이메일 침해(BEC) 및 사기, 암호화폐 네트워크 악용 등 여러 공격 유형을 통해 금전적 이득을 목표로 함
T1490	Inhibit System Recovery	• 손상된 시스템 복구를 위한 서비스를 종료하고, 내장 데이터를 삭제할 수 있음 • ESXi 서버에서 공격자는 가상 머신의 스냅샷을 삭제하거나 암호화해 백업으로 활용하지 못하도록 할 수 있고, 클라우드 서비스와 동기화되는 폴더를 삭제해 온라인 백업 삭제

TID	기법 구분	기법 설명
T1498	Network Denial of Service	• 공격자는 서비스가 사용하는 네트워크 대역폭을 고갈시키는 네트워크 서비스 거부(DoS) 공격을 수행할 수 있음
T1496.001	Resource Hijacking: Compute Hijacking	• 공격자는 컴퓨터 자원을 활용해 암호화폐를 채굴하는데 사용 가능
T1496.002	Resource Hijacking: Bandwidth Hijacking	• 공격자는 네트워크 서비스 거부에 활용하거나 악성 토렌트를 배포하기 위해 시스템의 네트워크 대역폭을 활용할 수 있음
T1496.003	Resource Hijacking: SMS Pumping	• 공격자가 통신 사업자로부터 전화번호를 확보한 후, 피해자의 메시징 인프라를 활용해 해당 전화번호로 대량의 SMS 메시지를 전송
T1496.004	Resource Hijacking: Cloud Service Hijacking	• 공격자는 손상된 SaaS 애플리케이션을 이용해 자원이 많이 필요한 작업을 수행할 수 있음 • 공격자는 AWS Simple Email Service(SES), AWS Simple Notification Service(SNS), SendGrid, Twilio와 같은 이메일 및 메시징 서비스를 활용해 대량의 스팸 및 피싱 메일을 보낼 수 있음

3.4. AWS 사고 대응 프레임워크 조사

AWS는 NIST SP 800-61 사고 대응 표준을 기반으로 클라우드 환경의 동적 특성과 로그 중심 조사 방식을 반영해 클라우드 이용 조직이 AWS 리소스 전반에서 통합적 대응을 수행할 수 있도록 5단계 보안 사고 대응 프레임워크를 제시했다. 이 프레임워크는 각 단계별로 활용 가능한 AWS 보안 서비스와 로그를 구체적으로 제안하고 있다. 프레임워크에서 제시하고 있는 절차에 대한 주요 내용은 다음과 같다.

[표 22] AWS 사고 대응 프레임워크 절차

단계	설명
준비 (Preparation)	사고 발생 전 대응 체계를 마련하고, 조직과 기술적 준비를 완료하는 단계 # 주요 수행 항목 • IAM 최소 권한 원칙 적용, Root 계정 보호 • CloudTrail, AWS Config, VPC Flow Logs, Route 53 Resolver Log 활성화 • S3 기반 로그 저장소(Evidence Bucket) 구성 • Systems Manager, Lambda, EventBridge 등 자동화 도구 준비 • Gameday, Runbook 기반의 대응 훈련 # 주요 AWS 서비스 • IAM, CloudTrail, AWS Config, S3, Systems Manager, Lambda
탐지 및 분석 (Detection and Analysis)	이상 징후를 탐지하고, 로그 및 이벤트를 분석해 사고의 원인과 영향을 판단하는 단계 # 주요 수행 항목 • GuardDuty를 통한 비정상 행위 탐지 (IAM 남용, 악성 IP 통신 등) • Security Hub로 여러 보안 서비스의 결과 집계 • Detective를 통한 이벤트 상관관계 분석 • CloudTrail, VPC Flow Logs, DNS Log 등 로그 분석 • 알림과 탐지 결과를 기반으로 사고 분류 및 우선순위 지정 # 주요 AWS 서비스 • GuardDuty, Security Hub, Detective, CloudTrail, CloudWatch
격리 (Containment)	확산을 방지하기 위해 공격이 감지된 리소스를 신속히 분리하고 접근을 차단하는 단계 # 주요 수행 항목 • 감염 인스턴스 네트워크 격리 (보안 그룹, NACL 수정) • IAM 키 및 세션 토큰 비활성화 • EventBridge + Lambda 기반 자동 격리 트리거 • System Manager Session Manager를 통한 원격 조치 # 주요 AWS 서비스 • System Manager, Lambda, EventBridge, Network Firewall, IAM

단계	설명
근절 및 복구 (Eradication and Recovery)	침해 흔적을 제거하고 시스템을 정상 상태로 복원하는 단계 # 주요 수행 항목 • 악성 코드, 비정상 계정, 백도어 제거 • 스냅샷(EBS, RDS) 또는 백업을 통한 시스템 복원 • CloudFormation Stack 재배포 및 Config 규칙 검증 • 복구 후 보안 모니터링 재활성화 # 주요 AWS 서비스 • Backup, CloudFormation, AWS Config, EBS, Systems Manager
사후 개선 (Post-IR)	사고 종료 후 원인 분석, 프로세스 개선, 탐지 규칙 보완을 통해 지속적 학습을 수행 # 주요 수행 항목 • 사고 보고서 및 Lessons Learned 문서 작성 • 대응 과정의 병목, 실패 원인 분석 • GuardDuty 및 Security Hub 탐지 규칙 업데이트 • Lambda, EventBridge 자동화 로직 개선 • 정기적 재훈련 및 GameDay 반복 # 주요 AWS 서비스 • Security Hub, GuardDuty, SSM Runbook

3.5. AWS 보안 서비스 조사

클라우드 환경에서의 사고 대응은 로그 중심의 데이터 가시성 확보에 기반한다. AWS는 이를 위해 다양한 네이티브 보안 서비스를 제공하며, 각 서비스는 '탐지-분석-대응' 단계별로 상호 연계되어 운영된다. AWS 기반의 사고 대응에서 활용되는 주요 보안 서비스는 다음과 같다.

[표 23] AWS 기반의 사고 대응에서 활용되는 주요 보안 서비스 목록

번호	서비스명	주요 역할	사고 대응 단계
1	Amazon GuardDuty	이상행위 탐지 (행위 기반)	Detection
2	Amazon CloudWatch	지표·로그·이벤트 통합 모니터링, 알람·이상탐지, 이벤트 기반 자동화	Detection, Analysis, Containment, Post-IR
3	Amazon Detective	로그 상관분석, 원인 조사	Analysis
4	Amazon Athena	S3 로그에 SQL로 포렌식/상관분석	Analysis, Post-IR
5	Amazon Security Hub	보안 결과 통합, 규정 준수 관리	Detection, Post-IR
6	AWS Systems Manager	대응 및 복구 자동화(런북/원격명령/세션)	Containment, Eradication & Recovery
7	Amazon Macie	S3 민감정보 식별·위험 버킷 탐지	Preparation, Detection, Post-IR
8	AWS Config	리소스 구성 변경 추적, 정책 위반 평가, 시정 조치 자동화	Preparation, Detection, Post-IR
9	Amazon Inspector	취약점 스캔(CVE·네트워크 노출도)	Preparation, Detection
10	Prowler	보안 구성 점검, 취약 설정 탐색, 규정 준수 평가	Preparation, Post-IR
11	Self-Service Security Assessment (SSSA)	자가 보안 점검·준비도/컴플라이언스 평가	Preparation, Post-IR

각 주요 보안 서비스에 대한 상세 내용(서비스 설명, 주요 기능, 주요 특징, 사고 대응 연계 방안)은 다음과 같다.

1) AWS GuardDuty

AWS GuardDuty는 AWS 환경 전반에서 발생하는 로그를 분석해 비정상 행위, 계정 탈취, 네트워크 침입, 데이터 유출 등을 자동으로 탐지하는 관리형/지능형 위협 탐지 서비스이다. 주요 기능으로는 CloudTrail/VPC Flow Logs/DNS Log/EKS Audit Log 분석, IAM Anomaly Detection, S3 Data Access Monitoring, Findings 생성 및 분류, 자동 연계 조치가 있다.

[표 24] AWS GuardDuty 주요 특징

구분	내용
관리형 위협 탐지 서비스	별도의 에이전트 설치나 인프라 운영 필요 없이, AWS 내부 로그를 자동 분석
로그 기반 이상 행위 탐지	CloudTrail, VPC Flow Logs, DNS Query Log, EKS Audit Log 등 주요 로그를 분석해 인증 남용, 데이터 유출, 비정상 API 호출, 외부 통신 시도 등 식별
머신러닝 및 위협 인텔리전스 기반 탐지	AWS 자체 ML 모델 + AWS Threat Intelligence + Partner Feed(MISP, Abuse.ch 등)을 활용해 C2, 피싱, 악성 IP 등 탐지
실시간 지속 모니터링	모든 리전에서 24/7 모니터링 수행, 탐지 결과를 거의 실시간으로 생성
자동 대응 연계성	Security Hub, EventBridge, Lambda, Systems Manager와 연계해 자동 격리, 알림
다계정 및 조직 단위 통합 관리	AWS Organizations와 통합되어 다계정 환경 탐지 관리 가능
비용 효율성 및 무중단 운용	로그 샘플링, 메타데이터 분석 기반으로 저비용/무중단 모니터링 제공

[표 25] AWS GuardDuty 사고 대응 연계 방안

단계	AWS GuardDuty의 역할
탐지 (Detection)	로그 기반 이상 행위 탐지 및 Findings 생성
분석 (Analysis)	Security Hub로 Findings 통합, Detective로 상관분석
격리 (Containment)	EventBridge 트리거로 Lambda 격리 워크플로우 실행
사후 개선 (Post-IR)	탐지 규칙 보강, Custom Threat List 업데이트

2) Amazon CloudWatch

Amazon CloudWatch는 AWS 리소스, 애플리케이션, 온프레미스 서버의 로그 파일을 중앙에서 수집, 모니터링, 저장 및 분석할 수 있게 해주는 관리형 서비스이다. 시스템의 성능 이상이나 장애 징후를 조기에 탐지해 경고를 발송하고, 로그 기반 분석을 통해 운영 효율성 및 보안 가시성을 향상시킨다. CloudTrail, GuardDuty, Config 등 다른 보안 서비스와의 연동을 통해 보안 이벤트를 트리거로 활용할 수 있어, AWS 환경에서의 사고 탐지(Detection) 및 대응 자동화(Auto-Remediation)의 기반 역할을 수행한다.

주요 기능으로는 AWS 리소스의 주요 매트릭스 수집, 로그 중앙 집중화 관리, 매트릭스와 로그 기반으로 알림, 대시보드 생성, 이벤트 기반 자동화 트리거, 로그 패턴 분석, 비정상적인 지표 변화 자동 탐지가 있다.

[표 26] Amazon CloudWatch 주요 특징

구분	내용
통합 모니터링 허브	모든 AWS 리소스의 지표와 로그를 단일 서비스에서 통합 관리 가능
자동화 및 확장성	EventBridge, Lambda, SNS와 연계해 자동 대응 워크플로 구축 가능
실시간 이상 탐지	Metrics 기반 이상 감지(Anomaly Detection) 기능으로 비정상 동작 실시간 식별
장기 보관 및 분석 지원	Log 데이터를 S3, OpenSearch로 내보내 장기 저장·검색·시각화 가능
보안 서비스 연계성	GuardDuty, Config, CloudTrail 등과 결합 시 완전한 가시성 확보 및 대응 자동화
운영, 보안 통합 분석 가능	성능 문제와 보안 이벤트를 동일 환경에서 상관분석 가능 (예: CPU 급등 + 의심 API 호출 연계 탐지)

[표 27] Amazon CloudWatch 사고 대응 연계 방안

단계	Amazon CloudWatch의 역할
탐지 (Detection)	CloudTrail 로그를 CloudWatch Logs로 전송 후 특정 API 이벤트 (CreateUser, DeleteBucket 등)에 대한 실시간 경보 설정 VPC Flow Logs 수집으로 비인가 IP 접근, 포트 스캔 등 이상 트래픽 탐지 GuardDuty/Config Findings 감지 시 CloudWatch Alarms를 통해 보안 담당자 즉시 통보
분석 (Analysis)	CloudWatch Logs Insights를 통해 시점별 로그 분석 및 공격자 활동 시퀀스 추적 Contributor Insights를 활용해 공격자 IP, 사용자 ID, 호출 API 등 주요 원인 분석
격리 (Containment)	CloudWatch Event → Lambda 자동 실행 구조를 활용해 보안 이벤트 발생 시 자동 격리
사후 개선 (Post-IR)	CloudWatch Logs 및 Metrics 데이터를 장기 보관(S3 Export)해 포렌식 증거로 활용 사고 시점의 시스템 상태를 재현하고, 대응 효율성 검증을 위한 리뷰 수행

3) AWS Detective

AWS Detective는 AWS 계정, 리소스, IAM 활동, 네트워크 로그 간의 관계를 자동 모델링해 보안 사고의 원인 분석과 상관관계 조사를 지원하는 도구이다. Detective는 최대 1년간의 이벤트 기록을 기반으로 과거 활동까지 추적 가능한 서비스로, AWS의 다른 서비스(GuardDuty, Security Hub, Security Lake 등)와도 통합되어 원클릭 조사가 가능하다.

주요 기능으로는 Findings 조사, 행위 프로필 생성, 그래프 기반 자동 관계 모델링, 타임라인 분석, 특정 리소스에 대한 최근 활동/연결 이벤트 요약, IR Playbook과 연계가 있다.

[표 28] AWS Detective 주요 특징

구분	내용
관리형 보안 조사 서비스	별도의 인프라 운영, 로그 저장, 인덱싱이 불필요함 GuardDuty와 Security Hub로부터 Findings를 자동 수집 및 분석 환경 구성
관계 기반 데이터 모델링	AWS 리소스(계정, EC2, IAM, IP, S3 등)간 관계를 그래프 형태로 시각화해 사건 간 연관성 탐색 가능
로그 상관 분석 통합	CloudTrail, VPC Flow Logs, GuardDuty Findings, EKS Audit Log 등을 통합 분석해 "누가 무엇을, 언제, 어디서 수행했는가" 추적
자동 데이터 수집 및 보존	선택된 로그 소스를 자동으로 수집, 요약해 최대 1년간 저장하며, 포렌식 분석 시 효율적임
시각적 원인 추적	콘솔에서 특정 이벤트(예. 악성 IP, IAM 권한 오남용 등)에 대해 관련 사용자, 리소스, 행위 타임라인을 시각적으로 표시
GuardDuty, Security Hub, IAM Access Analyzer와 통합	GuardDuty 탐지 이벤트를 클릭 한 번으로 Detective 콘솔에서 상세 조사 가능
비용 효율성 및 무중단 운용	분석용 로그를 AWS 내부에서 자동 요약, 인덱싱하기 때문에 SIEM 대비 저비용, 고속 분석 가능

[표 29] AWS Detective 사고 대응 연계 방안

단계	AWS Detective의 역할
탐지 (Detection)	GuardDuty Findings 수신 및 분석 대상 확보
분석 (Analysis)	CloudTrail, VPC Flow Logs 기반 상관관계 분석, Root Cause 파악
격리 (Containment)	공격자 접근 경로 차단 근거 확보, IAM 재구성 지원
사후 개선 (Post-IR)	재발 방지용 탐지 규칙 개선 인사이트 제공

4) Amazon Athena

Amazon Athena는 Amazon S3의 데이터를 표준 SQL(Presto/Trino 기반)으로 바로 조회 및 분석할 수 있는 서버리스 대화형 쿼리 서비스이다.

주요 기능으로는 데이터 카탈로그 연계, 경로 기반 파티션을 정의해 스캔량 최소화 지원, 열 지향 포맷으로 머티리얼라이즈(Materialize) 지원, UDF(user Defined Function) 및 JSON 함수 활용이 있다.

[표 30] Amazon Athena 주요 특징

구분	내용
서버리스·즉시 쿼리	클러스터 관리 불필요, 콘솔/CLI/JDBC에서 바로 실행
S3 네이티브	CSV/JSON/Parquet/ORC/Avro, 압축 형식(Gzip, Snappy 등) 지원
스키마 온 리드	파일을 옮기지 않고, Glue Data Catalog로 테이블 정의 후 즉시 분석
분산 SQL 엔진	Presto/Trino 기반으로 대용량 데이터 병렬 처리
통합 생태계	CloudTrail, ALB/ELB 로그, VPC Flow Logs, WAF/CloudFront 로그 등 S3 로그에 바로 SQL 질의
보안/거버넌스	IAM, Lake Formation, 데이터 마스킹/열 단위 권한(컬럼 권한), S3/KMS 암호화, 쿼리 결과 암호화
연결성	JDBC/ODBC, QuickSight 시각화, Pandas/BI 툴 연계

[표 31] Amazon Athena 사고 대응 연계 방안

단계	Amazon Athena의 역할
탐지 (Detection), 분석 (Analysis)	S3에 로깅된 CloudTrail/VPC Flow/WAF/ALB/CloudFront 로그를 SQL로 상관 분석 공격 타임라인 재구성, 의심 IAM 활동/출발지 IP/리소스 추적
사후 개선 (Post-IR)	재발 방지 규칙 발굴(이상 패턴 기준), 탐지 룰 후보 도출

5) AWS Security Hub

AWS Security Hub는 AWS 환경의 다양한 보안 서비스와 타사 솔루션에서 발생하는 보안 결과를 단일 콘솔에서 통합, 분석, 관리하도록 지원하는 서비스다.

주요 기능으로는 Findings 결과 통합, 표준 준수 여부 자동 검사, Findings 필터링 및 우선순위화, 자동 격리·알림 워크플로우 구성, AWS Organization 환경에서 계정 정체 Findings 집계 관리, 보안 이벤트 발생 시 자동 업데이트 및 실시간 대시보드 반영이 있다.

[표 32] AWS Security Hub 주요 특징

구분	내용
보안 결과 통합	GuardDuty, Inspector, Macie, Detective 등 AWS 보안 서비스 및 Third-Pary 툴의 탐지 결과(Findings)를 통합 관리
표준화된 결과 형식	모든 보안 결과를 공통 JSON 형식(ASFF)으로 표준화해 분석 자동화 및 연계 용이
보안 상태 평가	AWS CIS Benchmark, PCI DSS, NIST 800-53 등 표준 규정 기반으로 계정의 보안 준수 상태를 자동 평가
자동화된 조치 연계	EventBridge 규칙과 Lambda 자동화 함수를 활용해 Findings에 대해 자동 조치 트리거 가능
다계정 관리 및 조직 통합	여러 AWS 계정의 보안 결과를 조직 단위로 중앙 집계 및 모니터링
대시보다 시각화	보안 결과, CIS 규정 준수 현황, 탐지 추이 등을 그래픽 형태로 제공
타 보안 도구 연동	Splunk, CrowdStrike, Palo Alto Prisma Cloud 등 보안 솔루션과 API 연동 가능

[표 33] AWS Security Hub 사고 대응 연계 방안

단계	AWS Security Hub의 역할
탐지 (Detection)	GuardDuty, Inspector 등의 결과를 집계 및 표준화 형식으로 수집
분석 (Analysis)	Findings 우선순위화, 중복 제거, Compliance 상태 확인
격리 (Containment)	EventBridge와 Lambda를 활용해 자동 대응 시나리오 트리거
사후 개선 (Post-IR)	보안 상태 점수 (Secure Score) 및 Lessons Learned 기반 정책 보완

6) AWS Systems Manager (SSM)

AWS Systems Manager는 AWS 인스턴스, 애플리케이션, 계정, 구성요소를 중앙에서 제어하고 보안·운영 작업을 자동화해 운영 효율성과 사고 대응 속도를 높이기 위한 자동화 서비스이다.

주요 기능으로는 EC2 등에 원격 명령 실행, SSM Document로 반복 작업 자동화, OS 및 애플리케이션 패치 관리 자동화, 인스턴스 구성 상태 지속 관리, SSH키 없이 터미널 접근 제공(Session Manager), 암호·API Key·설정 값을 암호화해 안전하게 저장 및 참조, 패치/구성 상태의 규정 준수 여부 평가가 있다.

[표 34] AWS Systems Manager 주요 특징

구분	내용
중앙 통합 관리	EC2, 온프레미스 서버, 하이브리드 환경을 단일 콘솔에서 통제 가능
자동화된 운영 및 보안 조치	Lambda, Run Command, Automation Document(SSM Doc)를 통해 패치, 격리, 복구 절차 자동 수행
에이전트 기반 관리	각 EC2, 온프레미스 인스턴스에 설치된 Agent가 명령을 안전하게 수행
세분화된 권한 제어	IAM 정책을 통해 각 명령·Runbook 실행 권한을 세밀하게 관리
보안 이벤트 대응 연계	GuardDuty FindingsL나 EventBridge 이벤트를 트리거로 자동 조치 가능
패치, 설정, 규정 준수 관리	Patch Manager, State Manager, Compliance 기능으로 보안 패치·설정 기준 유지
하이브리드 환경 지원	온프레미스 시스템도 AWS Systems Manager Fleet에 등록해 동일한 정책·패치 적용 가능

[표 35] AWS Systems Manager 사고 대응 연계 방안

단계	AWS Systems Manager의 역할
탐지 (Detection)	GuardDuty Findings 발생 시 자동 트리거
격리 (Containment)	Run Command 또는 Automation으로 EC2 네트워크 인터페이스 분리, 보안 그룹 변경
근절 (Eradication)	자동화된 스크립트로 악성 프로세스 종료, 백도어 파일 삭제
복구 (Recovery)	SSM Document로 스냅샷 복원, 패치 적용, 서비스 재시작
사후 개선 (Post-IR)	Compliance 기능으로 설정 이탈 여부 검증

7) Amazon Macie

Amazon Macie는 Amazon S3에 저장된 데이터에서 민감정보(PII 등)를 머신러닝 및 패턴 매칭으로 자동 식별하고, 위험 버킷/객체를 찾아 경고(Findings)로 제공하는 서버리스 데이터 보안·프라이버시 서비스이다.

주요 기능으로는 스캔 대상·샘플링·식별자 세트·스케줄 설정, 버킷 평가, Findings 관리, 커스텀 조정이 있다.

[표 36] Amazon Macie 주요 특징

구분	내용
민감정보 식별	이름/주민등록번호/여권/신용카드/계좌 등 Managed Data Identifier + Custom Identifier(정규식/키워드) 지원
S3 전수·선별 스캔	S3 내 데이터 스캔 주기 설정 및 원하는 데이터 범위(버킷, 프리픽스)만 선별 스캔 지원
리스크 가시성	Public/Shared 버킷, 암호화 미적용, 과도한 정책 등 버킷 수준 위험과 객체 수준 민감도를 분리 보고
다계정 통합	AWS Organizations로 관리 계정에서 일괄 관리
통합·자동화	Findings를 Security Hub로 통합, EventBridge → Lambda/SSM으로 자동 조치 연계
서버리스·관리형	에이전트/클러스터 불필요, 사용한 만큼 과금(스캔 바이트·평가 건수 기준)

[표 37] Amazon Macie 사고 대응 연계 방안

단계	Amazon Macie의 역할
사전 준비 (Preparation)	민감데이터 보유 현황 식별, 고위험 버킷 범위 산정
탐지 (Detection), 분석 (Analysis)	데이터 유출 가능지점 파악, 버킷 정책/접근 경로와 결합 분석
격리 (Containment)	EventBridge 트리거로 Public Access Block, 정책 수정, 오브젝트 암호화 자동화
사후 개선 (Post-IR)	유출 범위(객체·필드) 정량화, 재발 방지 기준/식별자 보완

8) AWS Config

AWS Config는 클라우드 리소스의 구성 상태를 지속적으로 기록, 평가, 시정함으로써 사고 발생 전에는 보안정책 준수 상태를 감시하고, 사고 발생 후에는 변경 이력과 시점별 구성 복원을 통해 포렌식 분석 및 원인 규명을 지원하는 서비스이다. 즉, AWS 계정 내의 리소스가 "어떻게 설정되어 있었는가", "언제, 누가, 어떤 변경을 했는가", "현재 정책 기준을 준수하고 있는가"를 자동으로 기록하고 분석하는 역할을 수행한다. CloudTrail과 달리 행위가 아니라 상태 중심의 데이터를 다루며 시간별 스냅샷 형태로 리소스 구성을 추적한다.

주요 기능으로는 구성 이력 추적, 구성 스냅샷, 규칙 기반 평가, 규칙 위반 발생 시 SSM Runbook을 호출해 자동으로 조치, 여러 계정과 리전의 Config 데이터를 중앙에서 통합 조회 및 평가, 리소스 연관 자원을 맵핑한 의존성 추적 지원이 있다.

[표 38] AWS Config 주요 특징

구분	내용
상태 중심 기록	CloudTrail이 '무엇을 했다'를 기록한다면, Config는 '무엇이 어떻게 설정되어 있었는가'를 기록
지속적 모니터링	리소스 변경 시마다 자동으로 평가 수행, Drift(정책 이탈) 실시간 탐지
정책 준수·컴플라이언스 지원	CIS, NIST, PCI-DSS 등 주요 보안 표준의 정책을 Config Rule로 구현 가능
통합 관리	Security Hub, CloudTrail, SNS, S3, SSM 등과 통합되어 '탐지–분석–조치' 자동화 가능
포렌식 지원	특정 시점의 리소스 상태를 JSON Snapshot으로 복원할 수 있어 사고 당시 환경 재현 가능

[표 39] AWS Config 사고 대응 연계 방안

단계	AWS Config의 역할
사전 준비 (Preparation)	주요 리소스(S3, IAM, Security Group 등)에 대한 Config Rule 기반 준수 모니터링 활성화
탐지 (Detection)	구성 변경 중 정책 위반 감지 시 비정상 설정 탐지
분석 (Analysis)	사고 발생 시점의 리소스 구성 상태 및 변경자(Actor) 확인
격리 (Containment), 복구 (Recovery)	위반 항목 자동 수정 또는 이전 상태로 복원
사후 개선 (Post-IR)	사고 이후 Drift 발생 여부 재검증, 개선된 규칙 재배포

9) Amazon Inspector

Amazon Inspector는 AWS 인스턴스, 컨테이너, 소프트웨어 패키지의 취약점을 자동으로 스캔하고 CVSS 점수, 영향도, 패치 상태 등을 기반으로 보안 위험을 평가하는 서비스이다.

주요 기능으로는 EC2 인스턴스 취약점 평가, 컨테이너 이미지 스캔, Lambda 함수 스캔, 퍼블릭 노출 가능 경로를 분석해 공격 노출도 평가, Findings 관리 및 우선순위 부여, 자동화된 알림 및 조치 연계가 있다.

[표 40] Amazon Inspector 주요 특징

구분	내용
자동화된 취약점 스캐닝	EC2, ECR(컨테이너 이미지), Lambda 함수의 코드, 패키지, 구성 요소를 자동 분석해 CVE 취약점 탐지
에이전트 기반 실시간 평가	AWS Systems Manager(SSM) Agent를 통해 EC2의 OS, 패키지, 네트워크 구성을 실시간 평가
CVE/CVSS 기반 위험도 산정	NVD(National Vulnerability Database) 및 AWS 자체 평가지표를 기반으로 취약점 위험도 계산
보안 표준 연계	CIS Benchmark, NIST 800-53 등과 매핑된 보안 기준 평가 가능
보안 허용 정책 기반 예외 관리	특정 취약점을 일시적으로 무시하거나 예외 처리 가능
Findings 자동 전송 및 통합 관리	탐지 결과를 Security Hub로 자동 전송해 중앙 관리 가능
지속적 모니터링	새 인스턴스나 새 컨테이너 이미지가 생성되면 자동으로 스캔 트리거

[표 41] Amazon Inspector 사고 대응 연계 방안

단계	Amazon Inspector의 역할
사전 준비 (Preparation)	취약점 및 잘못된 설정 사전 식별
탐지 (Detection)	CVE 기반 시스템·컨테이너·Lambda 취약점 탐지
격리 (Containment), 복구 (Recovery)	패치 적용 및 보안 설정 수정 조치 지원
사후 개선 (Post-IR)	재발 방지용 취약점 개선 가이드 제공

10) Self-Service Security Assessment (SSSA)

Self-Service Security Assessment는 AWS 보안 구성 및 운영 절차를 체크리스트/가이드에 따라 점검해 규정 준수 상태를 평가하는 자가 보안 점검 툴킷이다. 이 툴킷은 AWS CloudFormation 템플릿으로 배포되며, 보안 모범사례와 비교해 현재 AWS 설정을 평가한다.

주요 기능으로는 도메인 별 점검, 증빙 관리, 리포트 생성이다.

[표 42] Self-Service Security Assessment 주요 특징

구분	내용
자가 점검(셀프 체크) 중심	정책·절차·구성 상태를 문항별로 검토(증빙 링크·담당자·기한 기록)
보안 표준 및 사례 매핑	CIS, NIST 800-53, ISO 27001 등과 AWS 모범사례 매핑
자동화/수동 혼합	일부 항목은 Config, Security Hub 데이터로 자동 검증, 절차·조직 항목은 인터뷰·문서 확인
점수·우선순위 산출	위험도/성숙도 스코어, 개선 백로그 도출

[표 43] Self-Service Security Assessment 사고 대응 연계 방안

단계	Self-Service Security Assessment의 역할
사전 준비 (Preparation)	IR 준비도 점검(CloudTrail/GuardDuty/Backup/Access Control/연락망/훈련)
탐지 (Detection) 보조	탐지 실패 가능 요인 사전 제거
사후 개선 (Post-IR)	재평가로 보안 수준 상향

11) Prowler

Prowler는 AWS 보안 점검, 감사, 하드닝 및 사고 대응을 지원하는 CLI 도구이다.

주요 기능으로는 보안 표준(CIS AWS Foundation, NIST SP 800-53, ISO/IEC 27001, PCI DSS 등) 기반 보안 점검 엔진 제공, 조직 단위 일괄 점검, 핵심 서비스 및 영역 커버리지, 결과 포맷과 통합, 조직 정책에 맞춘 커스터마이즈, CI/CD 및 스케줄링 운용이 있다.

[표 44] AWS Prowler 주요 특징

구분	내용
AWS 환경 종합 보안 점검	IAM, CloudTrail, Config, S3, Security Group, KMS, GuardDuty 등 AWS 전반의 설정을 자동 검사
표준 기반 평가	CIS AWS Foundations Benchmark, NIST SP 800-53, ISO 27001, PCI DSS, GDPR 등 주요 규정과 매핑된 점검 제공
자동화 및 스케줄링 지원	CI/CD 파이프라인(GitHub Actions, GitLab CI, Jenkins 등) 또는 Lambda/CloudWatch Events를 통한 정기 점검 가능
다계정·멀티리전 지원	AWS Organizations 또는 AssumeRole을 통해 여러 계정·리전을 동시에 점검
리포트 및 통합 관리 기능	CSV, JSON, HTML 형태의 리포트 출력, 결과를 Security Hub로 전송 가능
커스터마이즈 가능한 점검 항목 구성지원	필요에 따라 특정 규정(예: CIS Level 1/2) 또는 자체 정책 기반으로 점검 항목 구성 가능
Cloud-Native 연계성	AWS CLI·API 기반으로 작동하며, 외부 에이전트 설치 불필요

[표 45] AWS Prowler 사고 대응 연계 방안

단계	Prowler의 역할
사전 준비 (Preparation)	사고 발생 전 보안 환경을 점검해 취약 구성을 사전에 식별하고 개선
탐지 (Detection)	GuardDuty와 달리 실시간 탐지는 아니지만, 잘못된 로그·권한 설정 등 잠재적 사고 요인 발견
분석 (Analysis)	사고 후 환경 재점검을 통해 공격자가 악용했을 가능성이 있는 설정 취약점을 파악
격리 (Containment), 복구 (Recovery)	SSM, Config 등과 함께 리포트 결과를 근거로 자동화된 재설정·패치 절차 수행 가능
사후 개선 (Post-IR)	사고 후 CIS 기준에 따른 보안성 재평가 및 재발 방지 점검

12) AWS Shield

AWS Shield는 네트워크/전송/애플리케이션 계층에서 발생하는 DDoS 공격을 탐지 및 완화하는 관리형 서비스이다.

주요 기능으로는 실시간 DDoS 탐지 및 완화, 다계층 보호, 전담 대응팀 연계, 비용 보호, 정책 및 그룹 기반 보호 운영, 자동 대응 연계가 있다.

[표 46] AWS Shield 주요 특징

구분	내용
상시 모니터링 & 자동 완화	글로벌 에지/백본에서 트래픽 패턴을 학습해 공격을 실시간(Inline) 완화
다계층 보호	L3/L4(UDP/TCP 대역폭 소진, SYN/ACK 플러딩 등) + L7(HTTP/S 플러딩)
리소스 단위 보호	CloudFront, Elastic Load Balancing(ALB/NLB/CLB), Elastic IP(EC2), Amazon Route 53, AWS Global Accelerator 등 지정 보호
DRT(전담 대응팀) 24/7 지원	대형/지속 공격 시 전문가 실시간 지원 및 룰 튜닝
비용 보호	대규모 DDoS로 인한 스케일링/데이터 전송 비용 폭증을 크레딧으로 상쇄
WAF·Firewall Manager 통합	자동 룰 추천/적용, 조직 단위 정책 배포, Protection Group 구성
가시성 & 알림	공격 벡터/규모/지속시간 메트릭, CloudWatch/SNS, AWS Health 이벤트로 알림

[표 47] AWS Shield 사고 대응 연계 방안

단계	AWS Shield의 역할
사전 준비 (Preparation)	중요 엔드포인트(CloudFront/ALB/EIP/Route 53/Global Accelerator)에 Shield Advanced 활성화, 임계치 알람·연락망·DRT 접근 권한 사전 구성
탐지 (Detection)	공격 발생 시 Shield 이벤트 + AWS Health 알림 수신, 메트릭으로 공격 규모 파악
격리 (Containment)	자동 완화(L3/L4) + WAF 연동 룰로 L7 차단, 필요 시 DRT 즉시 참여
근절 (Eradication), 복구 (Recovery)	공격 종료 확인 후 정상 정책으로 점진 복귀, 트래픽 엔지니어링/리다이렉션
사후 개선 (Post-IR)	공격 리포트/메트릭 리뷰, WAF 룰 튜닝, 비용 보호(claim) 요청, 플레이북 업데이트

3.6. AWS 로그 조사

AWS는 계정, 네트워크, 데이터, 애플리케이션, 보안 이벤트 전 계층의 로그를 제공하며, 해당 로그들은 사고 대응 절차에서 상호 보완적으로 활용된다. AWS에서 제공하고 있는 유형별 로그는 다음과 같다.

1) 계정 및 관리 활동 로그

AWS 계정의 전반적인 관리 활동과 리소스의 구성 변경 사항을 추적하며, 주로 보안 감사, 거버넌스, 규정 준수 목적으로 사용된다.

[표 48] 계정 및 관리 활동 로그 목록

로그 구분	설명
AWS CloudTrail	• AWS 계정에서 발생하는 모든 API 호출 및 관리 이벤트를 기록 • 누가, 언제, 어디서 어떤 작업을 수행했는지 상세하게 기록해 계정 활동에 대한 완전한 가시성을 제공 • 계정 별로 최근 90일 간의 관리 이벤트 로그를 무료로 조회 가능 • 90일 이상의 로그 기록을 저장하거나, 데이터 이벤트를 기록하려면 비용 발생
AWS Config	• AWS 리소스의 구성 변경 이력을 기록하고 모니터링 • 리소스가 특정 규칙을 준수하는지 평가하고, 변경 이력을 추적해 규정 준수 상태를 유지하는 데 도움을 줌 • 보안 이벤트 중에 수행된 활동을 추적하는 데 도움이 되는 리소스에 대한 구성 기록 제공 • 해당 로그는 유료 서비스로 리소스의 구성 항목을 기록하고, 규칙을 평가하는 데 비용 부과
AWS Identity and IAM Access Analyzer	• 리소스에 대한 외부 접근 권한을 분석해 의도하지 않은 공개 접근 식별 • S3 버킷이나 IAM 역할 같은 리소스가 외부 계정이나 사용자와 공유될 때 잠재적인 보안 위험을 알려줌 • 해당 서비스는 무료 서비스로, 별도 요금은 부과되지 않음

2) 네트워크 및 트래픽 로그

AWS 네트워크 내/외부의 트래픽 흐름을 기록하고 분석하는 데 사용된다. AWS 클라우드에서는 네트워크 트래픽을 기록하는 프록시를 생성하거나 트래픽 미러링을 사용해 네트워크 트래픽의 사본을 로깅 서버로 전송함으로써 네트워크 활동을 기록할 수 있다. 네트워크 성능 문제 해결, 보안 감사, 트래픽 패턴 분석에 필수적이다.

[표 49] 네트워크 및 트래픽 로그

로그 구분	설명
VPC Flow Logs	• VPC(Virtual Private Cloud)의 네트워크 인터페이스를 통과하는 IP 트래픽에 대한 상세 정보 기록 • 트래픽의 소스, 목적지, 포트, 프로토콜, 허용 또는 거부 여부를 파악할 수 있어 네트워크 보안 그룹과 NACL(Network ACL) 규칙을 검증하는 데 유용 • 해당 로그는 유료 서비스로 플로우 로그를 생성하고 S3나 CloudWatch Logs에 저장하는 비용 발생
ELB Access Log	• Elastic Load Balancer(ELB)가 처리하는 모든 HTTP/HTTPS 요청에 대한 정보 기록 • 클라이언트의 IP 주소, 요청 시간, ELB의 응답 시간, HTTP 상태 코드 등 상세한 정보를 제공해 애플리케이션 성능을 분석하고 디버깅하는 데 사용 • 해당 로그는 무료로 활성화할 수 있으나, 생성된 로그를 S3 버킷에 저장해 S3 저장 비용이 발생
AWS Global Accelerator Flow Log	• Global Accelerator를 통과하는 네트워크 트래픽 기록 • 사용자의 지리적 위치, 소스 및 목적지 IP 주소, 프로토콜 등 트래픽 관련 정보를 제공해 전역 네트워크 성능을 분석하는 데 도움을 줌 • 해당 로그는 유료 서비스로 로그를 생성하고 S3에 저장하는 비용 발생

3) 서비스별 액세스/활동 로그

각 AWS 서비스는 자체적인 활동 로그를 생성해 해당 서비스의 사용 패턴과 접근 기록을 상세하게 로깅한다.

[표 50] 서비스별 액세스/활동 로그

로그 구분	설명
Amazon S3 Server Access Log	• Amazon S3 버킷에 대한 모든 요청 기록 • 누가, 언제, 어떤 방식으로 버킷의 객체에 접근했는지 파악해 버킷의 사용량을 분석하고 보안 문제를 해결하는 데 활용 • 해당 로그는 무료로 활성화할 수 있으나, 로그가 저장되는 S3 버킷의 저장 비용 발생
Amazon RDS Logs	• RDS(Relational Database Service) 인스턴스에서 생성되는 다양한 데이터베이스 로그 포함 • MySQL의 일반 쿼리 로그, 에러 로그, PostgreSQL의 로그 등 데이터베이스 성능 분석 및 문제 해결에 필요한 정보를 담고 있음 • 해당 로그는 무료로 활성화할 수 있으나, 로그가 CloudWatch Logs로 전송되어 CloudWatch Logs의 저장 및 분석 비용 발생
Amazon CloudFront Access Log	• CloudFront 엣지 로케이션으로 들어오는 모든 사용자 요청을 기록 • 요청 시간, 클라이언트 IP, 요청된 파일, HTTP 상태 코드 등 CDN 사용 패턴을 분석하고 캐시 히트율을 최적화하는 데 도움을 줌 • 해당 로그는 무료로 활성화할 수 있으나, 로그가 저장되는 S3 버킷의 저장 비용 발생
Amazon API Gateway Access Log	• API Gateway로 들어오는 API 요청에 대한 상세 정보 기록 • 요청자, 응답 코드, 지연 시간 등 API 호출 관련 데이터를 제공해 API 사용량을 모니터링하고 문제점을 진단하는 데 유용 • 해당 로그는 유료 서비스로, 로그가 CloudWatch Logs로 전송되어 CloudWatch Logs의 저장 및 분석 비용 발생

4) 보안 서비스 로그

잠재적인 위협과 보안 위반을 식별하고 기록하는 로그를 생성한다. 이 로그들은 보안 상태를 지속적으로 모니터링하고 위협에 대응하는 데 필수적이다.

[표 51] 보안 서비스 로그

로그 구분	설명
AWS GuardDuty Findings	• AWS 환경에서 잠재적인 위협을 탐지하고 보고 • EC2 인스턴스에 대한 비정상적인 접근, 비활성화된 포트 스캔 등 악의적인 활동을 식별해 경고(findings) 형태로 기록 • 해당 서비스는 유료로 처리되는 데이터의 양과 생성되는 탐지 결과에 따라 비용 부과
AWS Security Hub	• AWS 계정 전반의 보안 경고 및 준수 상태를 통합적으로 보여줌 • GuardDuty, Inspector 등 여러 보안 서비스의 탐지 결과를 한 곳에 모아 중앙 집중식으로 관리할 수 있음 • 해당 서비스는 유료로 탐지 결과를 수집하고 분석하는 데 비용 발생
AWS WAF Log	• 웹 애플리케이션 방화벽(WAF)이 웹 애플리케이션이나 API에 대한 웹 요청을 모니터링하고 기록 • SQL 인젝션, 크로스 사이트 스크립팅(XSS) 등 웹 공격을 차단하고 상세한 로그를 로깅함 • 해당 로그는 유료 서비스로, WAF 로그를 CloudWatch Logs 또는 S3 버킷으로 전송하는 데 비용 발생
Amazon Inspector Findings	• EC2 인스턴스와 같은 리소스의 취약성을 스캔해 보안 취약점과 모범 사례 위반 사항을 식별 • 이러한 분석 결과를 'findings' 형태로 기록해 보안 강화 조치를 취하는 데 도움을 줌 • 해당 서비스는 유료로 Inspector가 리소스를 스캔해 취약점을 분석하는 데 비용 부과

5) 시스템 및 애플리케이션 로그

EC2 인스턴스, 컨테이너, Lambda 함수 등 컴퓨팅 리소스에서 직접 생성되는 로그를 의미한다. 애플리케이션의 동작과 시스템의 상태를 확인하는 데 사용된다.

[표 52] 시스템 및 애플리케이션 로그

로그 구분	설명
Amazon CloudWatch Logs	• 다양한 AWS 서비스와 사용자 정의 애플리케이션에서 발생하는 로그를 수집, 모니터링, 저장 • 로그 스트림과 로그 그룹을 통해 체계적으로 로그를 관리하며, 이를 기반으로 알람을 설정하거나 대시보드를 구축할 수 있음 • 로그를 수집, 저장, 분석하는 데 비용이 발생하지만, 무료 티어를 제공해 일정 용량(5GB)까지는 무료로 사용할 수 있음
Amazon EC2 System Log	• EC2 인스턴스의 운영체제(OS) 로그를 포함하며, 부팅 시 발생하는 이벤트나 시스템 메시지 기록 • 인스턴스의 부팅 문제나 시스템 오류를 진단하는 데 유용 • EC2 인스턴스의 시스템 로그 자체는 무료로 제공되며, 콘솔에서 바로 확인 가능
EKS/ECS Container Log	• EKS(Elastic Kubernetes Service) 및 ECS(Elastic Container Service)에서 실행되는 컨테이너의 로그 기록 • 애플리케이션 로그와 시스템 로그를 포함하며, 컨테이너화된 애플리케이션의 동작 상태를 모니터링하는 데 필수적 • 해당 로그는 유료로 컨테이너 로그가 CloudWatch Logs로 전송되어 저장 및 분석 비용 발생

3.7. AWS 사고 대응 플레이북 조사

AWS에서는 고객이 직면할 수 있는 대표적인 사고 시나리오를 다루고 있으며, 이는 NIST 컴퓨터 보안 사고 처리 가이드(SP 800-61 Rev. 2)의 절차를 기반으로 작성되었다. 플레이북은 증거 수집, 사고의 격리 및 제거, 사고로부터의 복구, 사후 활동 수행 등 각 단계에서 참고할 수 있도록 구성되어 있다.

이 중 본 연구에서는 사고 대응에 참고하기 유용한 16개의 플레이북을 선정했다. 각 플레이북은 DFIR 관점 분석 포인트, 주요 로그 및 데이터, 사고 대응 절차 요약으로 내용을 요약할 수 있으며, DFIR 관점 분석 포인트는 플레이북을 조사한 후 본 연구진이 재구성 및 요약한 내용이다. 각 사고 유형별 플레이북은 다음과 같다.

1) Unintended access to an Amazon Simple Storage Service (Amazon S3) bucket

Amazon Simple Storage Service(S3) 버킷의 접근 정책이 잘못 구성되어, 인증되지 않은 외부 사용자에게 데이터 접근이 허용된 사고 사례이다.

S3는 기본적으로 비공개 상태이지만, 운영 편의나 서비스 요구사항 충족을 위해 버킷 정책이나 ACL을 변경하는 과정에서 IAM 주체 또는 익명 사용자("Principal": "*")에게 과도한 권한이 부여되는 경우가 있다.

[표 53] DFIR 관점 분석 포인트 - S3 버킷의 의도치 않은 접근 사고

구분	핵심 분석 항목
원인 분석	잘못된 S3 버킷 정책 변경 (`PutBucketPolicy`)에 따른 공개 설정
공격 흔적	CloudTrail 이벤트 내 `GetObject`, `ListBucket`, 외부 IP 호출
영향 범위	공개된 객체 목록, 데이터 유형, 접근 로그 IP 식별
대응 강화	IAM Access Analyzer 상시 모니터링, Config Rule 기반 자동 탐지, S3 정책 변경 알림 설정

[표 54] 주요 로그 및 데이터 - S3 버킷의 의도치 않은 접근 사고

구분	데이터 소스	확보 목적
CloudTrail	이벤트 유형: `PutBucketPolicy`, `PutBucketAcl`, `PutObjectAcl` 등	버킷 정책 및 접근 권한 변경 시점 식별
AWS Config	리소스 상태 변경 이력	정책 수정 이력 및 접근 제어 변경 추적
GuardDuty, Security Hub	탐지 알림	퍼블릭 접근 또는 비정상 접근 탐지
IAM Access Analyzer	IAM 정책 외부 노출 여부 식별	잘못된 신뢰 정책 및 비인가 접근 탐지

[표 55] 사고 대응 절차 요약 - S3 버킷의 의도치 않은 접근 사고

절차 구분	수행 행위
증거 수집 및 보존 (Acquire, Preserve, Document)	• CloudTrail 로그에서 버킷 정책, ACL, IAM 역할 변경 이벤트 확인 (`PutBucketPolicy`, `PutBucketAcl`) • AWS Config 및 Detective를 통해 변경 시점과 관련 주체(Principal) 분석 • GuardDuty 결과를 기반으로 비정상 접근, API 호출 이력 확보 • 로그 및 설정 스냅샷(S3 정책, IAM Role 등)을 별도 증거 버킷에 보존

사고 격리
(Containment)
	• S3 콘솔 또는 CLI를 통해 퍼블릭 접근 즉시 차단
	- Block All Public Access 활성화
	- ACL의 Everyone, AuthenticateUsers 권한 제거
	- 버킷 정책에서 Principal: "*" 및 Get*/Put* 액션 제거
	• IAM 콘솔에서 유출 가능성이 있는 자격 증명(Key, Role, STS 세션)을 모두 차단 및 재발급
	• 공격 확산 차단을 위해 EC2 인스턴스의 IMDSv2 적용
	`aws ec2 modify-instance-metadata-options \`
	`--instance-id <ID> --http-tokens required --http-endpoint enabled`

사고 제거 (Eradicate)	• IAM 역할 세션 무효화: 신뢰 정책(Trust Policy) 수정 및 Role Detach • S3 버킷 정책 최소 권한 원칙으로 복원 • S3 버전 관리(Versioning) 및 MFA Delete 활성화 • S3 서버사이드 암호화(SSE-KMS) 적용 및 Object Lock 설정으로 삭제/수정 방지
복구 및 후속 조치 (Post-IR)	• 손상 또는 삭제된 객체는 백업/버전에서 복구 • 백업 정책 및 수명 주기 설정 재검토 • 공개 접근이 필요한 경우 사전 서명된 URL(Pre-signed URL) 사용으로 대체 • 모든 IAM 사용자 및 키 재검토 및 최소 권한 재설정

2) Personal Data Breach

AWS Config 규칙, CloudTrail, GuardDuty 등에서 탐지 또는 외부 제보를 통해 인지될 수 있다. AWS 환경에서 개인정보(Personally Identifiable Information, 이하 PII)가 잘못된 접근 제어, 자격 증명 탈취, 내부자의 부적절한 행위 등으로 인해 외부에 노출된 사고 사례이다.

PII는 S3, DynamoDB, Lambda 로그, CloudWatch 등 다양한 경로에서 노출될 수 있으며, 보안 경계 침투나 잘못된 버킷 정책, 애플리케이션 로그 기록 실수 등 인적 오류 또는 설정 부주의가 주요 원인으로 작용한다.

[표 56] DFIR 관점 분석 포인트 – 개인정보 유출 사고

구분	핵심 분석 항목
원인 분석	IAM 권한 과다 및 S3/DynamoDB 설정 오류로 인한 외부 접근 허용
공격 흔적	CloudTrail 이벤트 내 `GetSecretValue`, `GetObject`, 외부 IP 탐지
영향 범위	유출된 데이터 항목(PII 필드), 접근 주체, API 호출 시점
대응 강화	IAM 최소 권한 재정비, DLP·Macie 기반 자동 탐지, Secrets Manager 자동 갱신 활성화

[표 57] 주요 로그 및 데이터 – 개인정보 유출 사고

구분	데이터 소스	확보 목적
CloudTrail	이벤트 유형: `GetObject`, `PutObject`, `ListBucket`, `GetParameter`, `GetSecretValue` 등	PII 접근 및 무단 다운로드 행위 확인
VPC Flow Logs	비정상 외부 트래픽 탐지	PII 유출 경로 식별
AWS Config	리소스 정책·권한 변경 내역	IAM, 리소스 설정 변경 시점 추적
GuardDuty, Security Hub	탐지 알림	비인가 데이터 접근 및 유출 의심 탐지
DynamoDB, CloudWatch Logs	데이터 내 PII 포함 여부	PII 기록 또는 로그 유출 확인
EBS Snapshot, 메모리 덤프	포렌식 분석용 원본 증거	메모리 내 PII 잔존 데이터 추출
IAM Access Analyzer	잘못된 권한 위임 및 외부 주체 접근 탐지	사고 원인 규명 및 피해 범위 분석

[표 58] 사고 대응 절차 요약 – 개인정보 유출 사고

절차 구분	수행 행위
증거 수집 및 보존 (Acquire, Preserve, Document)	• CloudTrail 로그에서 PII 관련 API 호출이벤트 확인 (`GetObject`, `GetParameter`, `GetSecretValue`) • IAM Access Analyzer로 외부 주체 접근 권한 및 버킷 정책 노출 여부 점검 • AWS Config, VPC Flow Logs, GuardDuty 결과를 함께 교차 분석해 PII 접근 주체·시점·경로 식별 • DynamoDB, CloudWatch 로그 등에 포함된 PII를 식별 후 로그 그룹 스냅샷 보존

사고 격리
(Containment)
	• IAM 콘솔 또는 CLI를 통해 유출 계정·역할의 접근 즉시 차단
	`aws iam detach-user-policy --user-name CompromisedUser --policy-arn arn:aws:iam::aws:policy/AmazonS3FullAccess`
	• S3 콘솔 공개 접근 차단
	- Block All Public Access 활성화
	- 버킷 정책에서 Principal: "*" 및 과도한 권한 삭제
	• 벤더/외부 계정 접근 해제 (DynamoDB/SQS)
	`aws iam detach-role-policy --role-name VendorRole --policy-arn arn:aws:iam::aws:policy/AmazonDynamoDBFullAccess`
	`aws sqs remove-permission --queue-url <QueueURL> --label VendorAccess`
	• CloudWatch 로그 그룹 내 PII가 포함된 로그는 수동 삭제 또는
	로그 만료 기간 단축 설정

사고 제거
(Eradicate)
	• IAM 역할 세션 무효화 및 신뢰 정책 수정으로 STS 세션 차단
	• EC2 인스턴스에 연결된 역할 및 키페어 교체,
	IMDSv2(Instance Metadata Service v2) 적용으로 메타데이터 노출 방지
	`aws ec2 modify-instance-metadata-options \`
	`--instance-id i-0123456789abcdef0 \`
	`--http-tokens required --http-endpoint enabled`
	• DynamoDB, Lambda 등 데이터 처리 경로의 민감정보 필드 암호화 또는 삭제
	`aws dynamodb update-item \`
	`--table-name MyPIITable \`
	`--key '{"UserId": {"S": "123"}}' \`
	`--update-expression "SET pii_field = :val" \`
	`--expression-attribute-values '{":val":{"S":"[REDACTED]"}}'`
	• Secrets Manager 및 Parameter Store 재점검 및 키·비밀번호 재발급

복구 및 후속 조치 (Post-IR)	• 영향받은 데이터 복원 지점 식별 후 백업 또는 버전 관리 이력 기반 복구 수행 • 유출된 PII 범위에 대한 데이터 주체 통지 및 규제기관 보고 수행 (GDPR/개인정보보호법 준수) • 재발 방지를 위한 IAM Role 최소 권한 정책 재설계, PII 암호화 적용, 자동 탐지(Macie/DLP) 강화

3) Credential Leakage / Compromise

Access Key, STS 토큰, 콘솔 자격 등 AWS 자격 증명이 유출 및 탈취되어 계정 내 리소스 생성, 변경, 접근에 악용된 사고 사례이다. 본 사고는 GuardDuty 또는 Security Hub 알림, CloudTrail 이상 징후, 운영 리전 외 리소스 생성, CMDB 미등재 리소스 발견, 외부 제보 등에 의해 인지될 수 있다. 자격 증명 특성상 은밀히 장기간 사용되는 경향이 있기 때문에 사고 인지 시 초동적으로 차단하고 타임라인을 재구성해 조치하는 게 핵심이다.

[표 59] DFIR 관점 분석 포인트 – 자격 증명 유출 및 탈취 사고

구분	핵심 분석 항목
원인 분석	Access Key 또는 STS 토큰 유출, 과도한 IAM 권한 설정, Secret 관리 미흡 등으로 인한 자격 증명 노출
공격 흔적	CloudTrail에서 `AssumeRole`, `AttachPolicy`, `PassRole`, 권한 상승 및 신규 리소스 생성 이벤트 탐지
영향 범위	유출된 자격 증명으로 접근 가능한 리소스 및 데이터 식별 (S3, EC2, IAM, RDS 등)
대응 강화	Access Key 갱신 정책 적용, IMDSv2 강제, IAM 최소 권한 재정비, Secrets Manager 자동 갱신 및 탐지 모니터링 강화

[표 60] 주요 로그 및 데이터 – 자격 증명 유출 및 탈취 사고

구분	데이터 소스	확보 목적
CloudTrail	이벤트 유형: `AssumeRole*`, `CreateUser/Role`, `Attach*Policy`, `Get*Token`, `RunInstances(PassRole)`, `GetObject` 등	유출 자격으로 수행된 모든 API 타임라인 복원
GuardDuty, Security Hub	`CredentialAccess`, `UnauthorizedAccess` 등	유출·오남용 정황의 초동 이벤트 인덱스
AWS Config	리소스/정책 변경 이력	권한, 구성 변경 시점 및 주체 추적
VPC Flow Logs, WAF, ELB Log	외부 통신/이상 트래픽	C2·대량 통신·데이터 유출 경로 식별
Detective, Athena	관계 그래프·대용량 쿼리	행위 상관분석, 대량 로그 탐색
EBS 스냅샷, 메모리(필요시)	포렌식 원본	휘발/디스크 증거 보존

[표 61] 사고 대응 절차 요약 – 자격 증명 유출 및 탈취 사고

절차 구분	수행 행위
증거 수집 및 보존 (Acquire, Preserve, Document)	• CloudTrail 전 기간(최소 의심 시점 전후)에서 해당 Access Key, Principal ID 기준으로 API 전수 검색(Athena/검색 도구) • AWS Config로 정책, 역할, 보안그룹, S3 정책 등 변경 시점과 변경 주체 상호 검증 • GuardDuty, Security Hub Findings로 사고 인지 시점과 이벤트 인덱싱 후 타임라인 키로 활용 • 로그와 설정 스냅샷은 별도 증거 버킷에 보존(해시 포함)
사고 격리 (Containment)	• 자격 증명 즉시 차단 - 장기 키(IAM User Access Key) 비활성화/삭제, 콘솔 비밀번호 초기화·MFA 강제 - STS 세션은 TTL 만료 전 유효 → Role 권한 Detach/정책 전면 차단, Trust Policy 수정으로 실사용 차단 • 고위험 경로 차단 - S3 공개/광역 권한 정책 제거, 보안그룹 과도한 인바운드 폐쇄, IMDSv2 강제 • 비인가 리전, CMDB 외 리소스 즉시 태깅 및 정리 계획 수립 (즉시 삭제 전 증거 보존 우선)
사고 제거 (Eradicate)	• CloudTrail 재분석으로 신규 생성된 사용자, 역할, 액세스 키, 프로필 전수 식별 후 비활성 및 삭제 • 권한 상승 흔적(AttachPolicy, PassRole 등) 제거, 최소 권한 재정의 • 취약점 및 오류 원인 제거 및 파이프라인 Secret 갱신 - 원인 예시: 키 노출, CI/CD Secret 유출, 공개 리포지토리, 잘못된 정책 등 • 로그/구성/네트워크 측면의 지속성(발판) 제거 - 스케줄러, Lambda 트리거, EventBridge Rule, Access Key 재생성 여부 점검
복구 및 후속 조치 (Post-IR)	• 정상 사용자, 워크로드에 필수 권한만 복원 (PBAC/ABAC 또는 Permission Boundary 적용) • 전체 시크릿(Secrets Manager/Parameter Store/KMS 키 정책) 갱신 및 참조 애플리케이션 재배포 • 키 발급·갱신·만료 정책, MFA·조건부 액세스, 비공개 리포지토리/Secret 스캐너 강제

4) Web Application Dos/DDoS Attack

웹 애플리케이션을 대상으로 한 대량 요청(HTTP Flood) 또는 네트워크 트래픽 유발 공격으로 서비스 가용성이 저하되거나 중단되는 사고 사례이다. AWS 환경에서는 CloudFront, ALB, WAF, Shield 등 내장 방어 메커니즘을 활용해 대응 가능하며, 로그 기반으로 공격 패턴을 분석하고 Auto Scaling으로 복구가 가능하다.

[표 62] DFIR 관점 분석 포인트 – 웹 애플리케이션 서비스 거부 공격

구분	핵심 분석 항목
원인 분석	비정상적인 HTTP 요청 급증, 외부 봇/스크립트 트래픽, 취약한 WAF 설정 또는 Auto Scaling 미구성
공격 흔적	CloudFront, ALB 로그의 과도한 요청, 동일 IP 및 User-Agent 반복, 특정 리전 집중 요청
영향 범위	웹 애플리케이션 가용성 저하, 서비스 응답 지연 및 장애 발생
대응 강화	AWS WAF/Web ACL 정책 강화, Shield Advanced 적용, Auto Scaling 및 트래픽 분산 구조 고도화

[표 63] 주요 로그 및 데이터 – 웹 애플리케이션 서비스 거부 공격

구분	데이터 소스	확보 목적
CloudTrail, ALB Log	요청 패턴, 4xx·5xx 비율, 요청량 급증 등	공격 트래픽 유형 및 집중 구간 식별
Web Server Access Log	비정상 요청, 공격자 IP, 요청 파라미터 분석	애플리케이션 단 공격 확인
CloudWatch Metrics	RequestCount, TargetResponseTime, ActiveConnection 등	부하 지표 추적 및 자동 확장 판단
AWS WAF, Shield Log	차단된 요청, 규칙 적용 내역	필터링 정책 검증 및 탐지 우회 여부 분석

[표 64] 사고 대응 절차 요약 – 웹 애플리케이션 서비스 거부 공격

절차 구분	수행 행위
증거 수집 및 보존 (Acquire, Preserve, Document)	• CloudFront, ALB, 웹 서버 로그를 수집해 비정상 요청 패턴 식별 (다중 IP, 요청 폭증, 4xx·5xx 비율 급증 등) • CloudWatch 지표 분석으로 성능 저하 원인 및 공격 시점 도출
사고 격리 (Containment)	• Auto Scaling, 로드밸런서, CloudFront를 통한 트래픽 분산 • 보안 그룹 재구성(퍼블릭 접근 제한, 로드밸런서만 허용) • CloudFront 및 ALB에 WAF Web ACL 연결
사고 제거 (Eradicate)	• AWS WAF 규칙(AWS-Managed Rules) 또는 커스텀 룰 적용 • 공격 IP, User-Agent, URL 패턴 기반 필터링 정책 강화 • 불필요한 리소스 및 임시 설정 제거, CloudFront IP 자동 업데이트 설정
복구 및 후속 조치 (Post-IR)	• 정상 트래픽 복귀 모니터링 • Shield Advanced 활성화 및 DDoS 대응 룰 자동화 검토 • CloudWatch 경보 및 AWS Config 규칙 재정비

5) Dos/DDoS Attack

AWS 환경에서 웹 애플리케이션, 로드 밸런서, 또는 네트워크 인프라가 과도한 요청이나 트래픽으로 인해 정상적인 서비스 제공이 불가능해지는 공격 사례이다. 단일 인스턴스에 대한 트래픽 집중(DoS) 또는 다수의 분산된 소스에서 동시 발생(DDoS) 형태로 나타날 수 있다. AWS는 Shield, WAF, CloudFront, Auto Scaling 등을 통해 기본적인 완화 기능을 제공하지만, 침해 조치 과정에서는 트래픽 원인 분석과 격리, 그리고 방어 정책 검증이 핵심이다.

[표 65] DFIR 관점 분석 포인트 – 서비스 거부 공격

구분	핵심 분석 항목
원인 분석	공격자가 공개된 서비스 엔드포인트에 대량의 요청·패킷을 발생시켜 리소스 고갈 유도
공격 흔적	GuardDuty DoS Findings, Flow Log 내 동일 IP 폭주, CloudWatch 트래픽 급증
영향 범위	서비스 지연, 응답 실패(5xx), CPU·네트워크 사용률 과다, 자동 확장 비용 증가
대응 강화	WAF Rate-Based Rule, Shield Advanced 구독, CloudFront 캐싱 강화, CloudWatch 기반 실시간 경보 구성

[표 66] 주요 로그 및 데이터 – 서비스 거부 공격

구분	데이터 소스	확보 목적
AWS WAF, Shield Log	공격 요약, 이벤트 세부 정보	공격 발생 시점, 트래픽 유형, 주요 타깃 리소스 식별
CloudWatch Metrics	서비스별 지표 (ALB, NLB, CloudFront 등)	요청량, 연결 수, 오류율, 트래픽 패턴 이상 감지
CloudTrail Log	API 호출 기록	리소스 변경·정책 수정, 보안 설정 비활성화 등 조작 여부 파악
VPC Flow Logs	네트워크 트래픽 기록	외부 공격 IP, 포트, 프로토콜 분석 및 차단 근거 확보
GuardDuty Findings	자동 탐지 결과	DoS, 스팸봇, 비정상 포트 사용 등 네트워크 기반 이상 행위 탐지
AWS Config	리소스 변경 추적	보안 그룹, WAF 규칙, 로드 밸런서 설정 변경 여부 확인

[표 67] 사고 대응 절차 요약 – 서비스 거부 공격

절차 구분	수행 행위
증거 수집 및 보존 (Acquire, Preserve, Document)	• 공격 징후 식별 - CloudWatch 지표에서 급증 확인 (RequestCount, ActiveConnectionCount, TargetResponseTime, HTTPCode_ELB_4XX_Count, RejectedConnectionCount 등) - CloudFront의 지표로 L7(애플리케이션 계층) 공격 탐지 (Requests, TotalErrorRate) - GuardDuty Findings 확인 (예. Backdoor:EC2/DenialOfService.*, Behavior:EC2/TrafficVolumeUnusual) • 트래픽 출처 분석 - VPC Flow Logs에서 동일 IP·ASN 반복 요청, 해외 IP 대역 확인 (Flow Log → Athena 쿼리로 대규모 트래픽 소스 파악) • WAF & Shield 이벤트 검토 - AWS Management Console → WAF & Shield → Global Threat Dashboard - 공격 유형, 대상 리소스, 패킷·요청량 확인 • 증거 보존 - WAF & Shield 로그 다운로드 - CloudWatch Metrics 스냅샷 - Flow Log 백업 (S3 Evidence Bucket)
사고 격리 (Containment)	• 공격 트래픽 차단 - AWS WAF 규칙 생성 (공격자 IP, User-Agent, URI 기반) - 초기에 Count 모드로 설정 후 CloudWatch 모니터링, 정상 요청 영향 없을 경우 Block 모드로 전환 • 보안 그룹 및 NACL 조정 - 공격 소스 IP 또는 포트 범위를 차단 - NACL 조정 시 서브넷 전체에 영향 주의 • Auto Scaling 그룹 확장 - Auto Scaling 정책을 수동 조정해 일시적으로 용량 확보 - 서비스 중단 방지를 위한 트래픽 분산 유도 • Shield Advanced 활성화(해당 시) - 콘솔 → WAF & Shield → Events에서 공격 메트릭 모니터링 (DDoSDetected, DDoSAttackRequestsPerSecond) - AWS DDoS Response Team(DRT) 지원 요청 가능

절차 구분	수행 행위
사고 제거 (Eradicate)	• 서비스 복구 - 공격 트래픽 종료 후 CloudWatch 지표 정상화 여부 확인 - CloudFront 캐시 정리(Invalidations) 및 API Gateway Rate Limiting 설정 • 불필요한 차단 해제 - 과도한 WAF 차단 규칙 또는 잘못된 보안 그룹 수정 • 정상 트래픽 테스트 - Route53 Health Check 상태(HealthCheckStatus) 및 Application Load Balancer 응답 (5xx 비율) 검증 • 정책 검토 - Shield 및 WAF의 탐지 설정, CloudFront 원본 접근 정책 재검증
복구 및 후속 조치 (Post-IR)	• 정상 인프라 상태 복원 - CloudFormation 또는 Terraform 기반으로 표준 보안 정책 재배포 - CloudFront + WAF + API Gateway 조합으로 공격 표면 축소 • 지속적 방어 강화 - Rate-Based Rule 구성, 지역 기반 필터링, IP Reputation 리스트 활용 - CloudWatch 알람 트리거 설정 (DDoSDetected, TargetResponseTime 기준) • 운영 대응 개선 - 공격 탐지 → 완화 → 보고까지의 경로 문서화 - DDoS 대응 훈련 및 비상 연락 체계 점검

6) Public Resources Exposure - RDS

퍼블릭 접근이 허용된 RDS 인스턴스나 스냅샷으로 인해 데이터베이스가 인터넷 환경에 노출된 사고 사례이다.

[표 68] DFIR 관점 분석 포인트 – 데이터베이스 노출

구분	핵심 분석 항목
원인 분석	퍼블릭 접근 설정 변경, 과도한 Security Groups 규칙(0.0.0.0/0), 스냅샷 공개, IaC/스크립트 오구성
공격 흔적	외부 IP의 DB 포트 스캔·연결 시도, 비정상 로그인 실패 증가, 대량 Select/Export 시도
영향 범위	데이터 노출 위험, 자격 증명 탈취 가능성, 운영 중단·성능 저하, 규정 위반 리스크
대응 강화	Public Access 금지 가드레일, 최소 권한 IAM, 비밀번호/토큰 주기적 순환, Config/WAF·GuardDuty 연계 경보, 주기적 Prowler 점검

[표 69] 주요 로그 및 데이터 – 데이터베이스 노출

구분	데이터 소스	확보 목적
CloudTrail	AWS API 호출 로그 (`ModifyDBInstance`, `ModifyDBSnapshotAttribute`, `ModifyDBClusterSnapshotAttribute`)	RDS 인스턴스 또는 스냅샷이 퍼블릭으로 변경된 시점, 수행자, 사용된 자격 증명(IAM 사용자·역할) 확인
VPC Flow Logs	VPC 내 ENI 트래픽 기록	인터넷에서 DB 포트 접속 시도 식별 (3306/5432/1433 등), 퍼블릭 엔드포인트로의 외부 연결, 비인가 IP 접근, 포트 스캔 및 반복 접속 행위 탐지
RDS 엔진 로그	Error/General/Audit 로그	로그인 실패, 권한 거부, 대량 쿼리·Export 시도 등 침입 후 흔적 확인
CloudWatch Metrics	DB 연결 수, CPU·네트워크 I/O	트래픽 급증·부하 증가 등 서비스 이상 징후 감지
AWS Config	`rds-instance-public-access-check`, `rds-snapshots-public-prohibited` 등	구성 변경 이력 및 퍼블릭 접근 위반 여부 추적, 변경 전후 상태 비교

[표 70] 사고 대응 절차 요약 – 데이터베이스 노출

절차 구분	수행 행위
증거 수집 및 보존 (Acquire, Preserve, Document)	• 영향 범위 식별 - 퍼블릭 RDS 인스턴스/클러스터, 퍼블릭 스냅샷 목록 작성 • CloudTrail 조회 - 퍼블릭 설정 변경 시점·주체, 관련 IAM 권한 사용 내역 확보 • VPC Flow Logs 분석 - 외부 IP·ASN·국가, 접속 포트/빈도 통계화 • RDS 엔진 로그 수집 - 실패한 인증, 권한 오류, 대량 덤프/스캔 패턴 확인 • 증거 보존 - 원본 로그 및 리포트를 증거 S3 버킷에 스냅샷(버전 관리·SSE-KMS 적용)
사고 격리 (Containment)	• 퍼블릭 접근 즉시 차단 - RDS Public access 비활성화, 엔드포인트를 프라이빗 서브넷로 한정 • 네트워크 경계 축소 - 보안 그룹 인바운드 최소화(VPN/Bastion 전용 CIDR), NACL 재검토 • 스냅샷 공개 해제 - 퍼블릭/교차계정 공유 속성 제거 • 자격 증명 위험 완화 - DB 사용자 비밀번호 강제 변경, Secrets Manager/DB 인증 토큰 순환
사고 제거 (Eradicate)	• 원인 제거 - 오남용된 IAM 정책·Role 정리(권한 축소, 인라인 정책 점검), 변경 경로(콘솔·CLI·IaC) 교정 • 불필요 리소스 정리 - 인가되지 않은 퍼블릭 스냅샷 삭제, 테스트용 공개 인스턴스 종료 • 접근 경로 폐쇄 - 퍼블릭 라우팅/IGW 경유 경로 점검, 엔드포인트 정책·Route 테이블 재정비
복구 및 후속 조치 (Post-IR)	• 아키텍처 정비 - RDS 프라이빗 서브넷 고정, 필요 시 RDS Proxy·프라이빗 링크 사용 • 모니터링 강화 - CloudTrail → S3/Athena 상시 분석, Config 규칙 경보, GuardDuty 연계 • 베이스라인 적용 - 보안 그룹 표준 템플릿, Terraform/CloudFormation 가드레일, 변경 승인(SoD) • 정기 점검 - Prowler extra78(퍼블릭 RDS), extra723(퍼블릭 스냅샷), group13(RDS 보안) 주기적 실행 실행

7) Public Resources Exposure – S3

퍼블릭 접근이 허용된 S3 버킷 또는 객체로 인해 내부 데이터가 외부에 노출된 사고 사례이다. 잘못된 버킷 정책, ACL 설정, Public Access Block 해제, 사용자 실수로 인한 구성 오류가 주요 원인이다.

[표 71] DFIR 관점 분석 포인트 – S3 버킷 노출

구분	핵심 분석 항목
원인 분석	S3 버킷 정책, ACL 설정 오류, Public Access Block 비활성화, 잘못된 IaC 배포로 인한 오구성
공격 흔적	CloudTrail의 `PutBucketAcl`, `PutObjectAcl`, `DeletePublicAccessBlock`, 외부 IP의 GET 요청, GuardDuty `S3/ObjectRead.Unusual` 탐지
영향 범위	민감 데이터 노출, 외부 다운로드·복제 가능, 권한 남용 및 후속 침투 가능성
대응 강화	Block Public Access 기본 적용, Config 규칙 자동화, CloudWatch/EventBridge 기반 실시간 알림, Prowler 정기 점검, IAM 최소 권한 설계

[표 72] 주요 로그 및 데이터 – S3 버킷 노출

구분	데이터 소스	확보 목적
CloudTrail	`PutBucketAcl`, `PutBucketPolicy`, `DeletePublicAccessBlock`, `PutObjectAcl`, `GetObjectAcl` 이벤트	S3 버킷 또는 객체가 퍼블릭으로 전환된 시점, 수행자, 변경 원인 파악 (IAM 사용자, 콘솔/CLI 여부 등)
S3 Server Access Log	Access Log, Server Log	외부 IP의 객체 접근 기록, HTTP 요청 메서드(Get, Put, Delete) 및 응답 코드 분석
VPC Flow Logs	ENI 트래픽 기록	S3 엔드포인트로 향하는 비정상적 외부 요청 탐지 (S3 게이트웨이 엔드포인트 사용 환경에서)
CloudWatch Metrics	GetRequests, 4xxErrors, 5xxErrors, FirstByteLatency	과도한 객체 요청, 에러율 증가, 비정상 트래픽 패턴 식별
AWS Config	`s3-bucket-level-public-access-prohibited`, `s3-bucket-public-read-prohibited` 등	버킷/객체 단위 Public Access 위반 탐지, 변경 이력 및 상태 추적

[표 73] 사고 대응 절차 요약 – S3 버킷 노출

절차 구분	수행 행위
증거 수집 및 보존 (Acquire, Preserve, Document)	• CloudTrail에서 공개 설정 변경 이벤트 추출 - 수행자, 접근 경로(IP, UserAgent), 콘솔/CLI 구분 - `DeletePublicAccessBlock`, `PutBucketAcl`, `PutBucketPolicy`, `PutObjectAcl` • S3 서버 액세스 로그 분석 - 퍼블릭 노출 시점 동안 외부 IP의 GET 요청 내역 및 접근 성공 응답 기록 식별 • VPC Flow Logs 및 GuardDuty Findings 확인 - 외부 C2 IP 또는 Tor 노드 접근 탐지 - 예: `S3/MaliciousIPCaller`, `S3/ObjectRead.Unusual` • 모든 로그·분석 결과를 S3 Evidence Bucket에 보존 (버전 관리·SSE-KMS 적용)
사고 격리 (Containment)	• 계정 단위 및 버킷 단위로 Block Public Access 즉시 적용 `aws s3api put-public-access-block --bucket <Bucket_Name> --public-access-block-configuration "BlockPublicAcls=true,IgnorePublicAcls=true, BlockPublicPolicy=true,RestrictPublicBuckets=true"` • 퍼블릭 정책/ACL 제거 - 콘솔 → S3 → 버킷 → Permissions → Bucket Policy / Access Control List 검토 후 수정 • IAM 정책 검토 - s3:* 전체 권한 부여, Principal: * 포함 정책 식별 시 즉시 수정 • 필요한 경우, 버킷을 프라이빗 서브넷의 VPC 엔드포인트로 한정해 접근 제어 강화
사고 제거 (Eradicate)	• 비인가 객체 및 변경 내용 정리 - 공격자 업로드 객체, 스크립트, 무단 파일 식별 후 삭제 - 콘솔 → S3 → 버킷 → 객체 목록 → 버전별 확인 후 불필요 버전 삭제 • 퍼블릭 스냅샷 및 공유 해제 - AWS CLI 또는 콘솔에서 AccessControlList 확인 → PublicRead 또는 PublicReadWrite 설정 제거 • IAM Access Key·Role 교체 - 노출된 키 또는 역할이 있는 경우 즉시 비활성화 및 재발급
복구 및 후속 조치 (Post-IR)	• 삭제·손상된 객체 복원 - 버전 관리(Versioning) 및 백업 객체에서 최신 정상 데이터 복원 • 접근 제어 재정립 - 최소 권한 정책(Least Privilege), VPC Endpoint, S3 Prefix 기반 권한 제한 적용 • 자동화된 규칙 적용 - AWS Config 규칙 활성화 (`s3-bucket-level-public-access-prohibited`, `s3-public-read-prohibited`) • 지속 모니터링 - CloudWatch EventBridge로 Public Access 변경 이벤트 실시간 감지 - Prowler 기반 정기 점검 (extra73 – 공개 S3 버킷 탐지, extra769 – 외부 공유 리소스 탐지, group17 – 인터넷 노출 리소스 그룹 탐색)

8) Code Exposure

내부 저장소의 코드나 구성 파일이 외부로 복제되거나 공개된 정황이 확인된 소스코드 유출 사고 사례이다. 이 사고는 대체로 DLP 경보 발생, 외부 플랫폼(GitHub, Pastebin 등)에 게시된 코드, 또는 CloudTrail 상에서 비정상적인 대량 다운로드 요청으로 탐지된다.

[표 74] DFIR 관점 분석 포인트 – 소스코드 유출

구분	핵심 분석 항목
원인 분석	자격 증명 노출, 퍼블릭 정책 설정 오류, 과도한 권한, 외부 공유 링크 오남용
공격 흔적	CloudTrail의 대량 `GetObject`, `BatchGetCommits`, 비정상 IP 또는 시간대의 접근
영향 범위	코드, 구성 정보, 인증 정보, 환경 변수 등의 노출로 인한 후속 공격 가능성
대응 강화	MFA 전면화, S3 객체 이벤트 로깅, Secrets 스캐닝, DLP 룰 정교화, Permission Boundary 강화

[표 75] 주요 로그 및 데이터 – 소스코드 유출

구분	데이터 소스	확보 목적
CloudTrail	CodeCommit, S3, ECR, CodeBuild 등 API 호출	코드 조회, 다운로드, 정책 변경, 자격 증명 사용 내역 확인
VPC Flow Logs	리소스와 외부 IP 간 트래픽 정보	대량 데이터 전송 여부, 비정상 포트나 국가 식별
DNS Log	내부 DNS 질의 기록	Pastebin, GitHub 등 데이터 반출 채널 탐지
CloudWatch	서비스별 로그와 지표	비정상 트래픽, 에러, 스로틀링 증가 확인
CodeCommit 또는 S3 Access Log	저장소 접근 기록	개별 커밋, 오브젝트 단위 접근 타임라인 분석

[표 76] 사고 대응 절차 요약 – 소스코드 유출

절차 구분	수행 행위
증거 수집 및 보존 (Acquire, Preserve, Document)	• CloudTrail 로그 확인 - `GetObject`, `BatchGetCommits`, `Download*` 등 대량 코드 조회 또는 다운로드 이벤트 식별 - API 호출 주체, 시간, IP, 위치 정보 검증 • CodeCommit 또는 S3 접근 내역 확인 - 최근 24~72시간 내 대량 접근 내역 검토 - S3 퍼블릭 접근, 외부 IP에서의 `GetObject` 요청 여부 확인 • VPC Flow Logs 분석 - 코드 저장소 또는 S3와 외부 IP 간 대규모 전송 패턴 식별 • DNS Log 검토 - GitHub, Pastebin, Discord 등 외부 도메인 요청 확인 • CloudWatch Metrics 검토 - 네트워크 송신 트래픽 급증, 에러율 상승, 비정상적인 요청 패턴 탐지 • 관련 IAM 사용자 및 자격 증명 식별 - Access Key, Role Session, Federation Token 사용 여부 확인
사고 격리 (Containment)	• 비인가된 IAM 사용자 또는 역할의 접근 차단 • 유출에 연관된 Access Key 비활성화 및 신규 키 갱신 • CodeCommit 저장소 접근 권한 회수 • S3 퍼블릭 접근 차단 정책 적용 • 영향을 받는 리소스(저장소, 버킷, 코드 빌드 환경) 태깅 및 "Quarantine" 표시 • 외부 게시된 코드가 실제 내부 자산과 동일한지 식별 후 증거 확보
사고 제거 (Eradicate)	• 비인가 커밋, 브랜치, 또는 오브젝트 제거 • 노출된 자격 증명 및 시크릿 키 모두 갱신 • 관련된 IAM Role, 정책, Access Key, Secrets Manager 항목 점검 및 정리 • 자동화 파이프라인이나 배포 스크립트 내 유출 경로 제거 • CloudTrail을 활용해 유출 직전/직후 API 호출 이력 검토
복구 및 후속 조치 (Post-IR)	• 정상 저장소 기준으로 코드 재배포 • 영향받은 키 및 환경 변수 수정 • 보안 설정 재점검 (S3, CodeCommit, CloudTrail, VPC Flow Logs 활성화 상태 확인) • 개발자 계정 MFA 활성화 및 비밀번호 재설정 • 관련 로그 및 증거 아카이브 저장 (S3 Evidence Bucket 등)

9) Ransom Response for EC2

AWS EC2 인스턴스가 랜섬웨어에 감염되어 데이터 암호화, 서비스 중단 또는 금전 요구 메시지가 탐지된 사고 사례이다. 공격자는 주로 주로 자격 증명(IAM Keys) 탈취나 취약한 원격 접속(SSH/RDP)을 통해 침투하며, EBS 볼륨 암호화, S3 데이터 삭제, 또는 추가 내부 이동으로 이어질 수 있다. 신속한 격리, 증거 보존, 복구 시점 식별이 핵심 대응 포인트다.

[표 77] DFIR 관점 분석 포인트 – EC2 랜섬웨어 감염

구분	핵심 분석 항목
원인 분석	탈취된 IAM Key 사용, 취약한 SSH 접근, 미적용 패치로 인한 초기 감염
공격 흔적	CloudTrail 상의 EBS 암호화·스냅샷 삭제, EC2 내부 암호화 프로세스, 외부 C2 통신
영향 범위	EC2 서비스 마비, EBS 데이터 암호화·손실, IAM 자격 증명 노출
대응 강화	백업 자동화 및 오프라인 보관, IAM Key 주기적 갱신, IMDSv2 강제화, VPC Flow Logs 지속 수집, EDR 기반 실시간 감염 탐지

[표 78] 주요 로그 및 데이터 – EC2 랜섬웨어 감염

구분	데이터 소스	확보 목적
CloudTrail	EC2, EBS, IAM, KMS 관련 API 호출 (`RunInstances`, `CreateVolume`, `EncryptVolume`, `DeleteSnapshot`, `PutBucketLifecycle`)	공격자에 의한 인스턴스 생성, 스냅샷 삭제, 볼륨 암호화 등 행위 추적
CloudWatch Metrics	CPUUtilization, NetworkPacketsOut, DiskWriteOps	데이터 유출, 암호화 프로세스 실행, 비정상적 리소스 부하 식별
VPC Flow Logs	VPC 내 네트워크 트래픽 흐름	외부 C2 서버 또는 공격자 IP와의 통신 여부 확인
AWS Config	`ec2-instance-no-public-ip`, `ec2-volume-inuse-check`, `ebs-snapshot-public-restorable-check` 등	인스턴스 및 볼륨 보안 구성 상태, 변경 이력 추적
EBS Snapshot	포렌식 증거 이미지	감염 파일, 암호화 프로세스, 랜섬 노트 등 내부 증거 확보

[표 79] 사고 대응 절차 요약 – EC2 랜섬웨어 감염

절차 구분	수행 행위

증거 수집 및 보존
(Acquire, Preserve, Document)
	• CloudTrail을 통해 이벤트 발생 시점, 행위자, API 경로 식별
	- `RunInstances`, `CreateTags`, `EncryptVolume`, `DeleteSnapshot` 등 호출 여부 확인
	• EBS 볼륨을 분리해 스냅샷 생성
	`aws ec2 create-snapshot --volume-id <EBS_ID>`
	`--description "Forensic Snapshot before isolation"`
	• AWS Config에서 비정상 구성 변경(공개 IP 부여, IMDSv1 사용 등) 이력 확인
	• S3 Evidence Bucket에 로그 및 스냅샷 보존 (버전관리 + KMS 암호화 적용)

사고 격리
(Containment)
	• 감염 인스턴스 네트워크 차단
	- 모든 트래픽 차단 보안 그룹 생성
	- 기본 egress 규칙 제거 후 인스턴스에 연결
	`aws ec2 modify-instance-attribute --instance-id <INSTANCE_ID>`
	`--groups <ISOLATION_SG_ID>`
	• Auto Scaling 그룹 또는 ELB에 연결된 경우 분리
	`aws autoscaling detach-instances --instance-ids <INSTANCE_ID>`
	`--auto-scaling-group-name <ASG_NAME>`
	`aws elb deregister-instances-from-load-balancer --instances <INSTANCE_ID>`
	`--load-balancer-name <ELB_NAME>`
	• EC2 Systems Manager(SSM) 또는 EDR을 통해 메모리/프로세스 목록 등 휘발성 증거 확보 후 종료

사고 제거
(Eradicate)
	• 감염 인스턴스 및 네트워크 정리
	- 공격자 C2 IP 기반 NACL 차단 규칙 적용, 악성 스크립트, 계정, 스케줄러 제거
	• NACL 생성 및 적용 절차
	- Amazon VPC 콘솔 → Network ACLs → Create → Inbound/Outbound Rules 편집
	- IoC 기반 IP CIDR 입력 → "DENY" 설정 후 Subnet에 연결
	• 악성 IAM 사용자·역할·액세스 키 식별 및 삭제
	`aws iam list-access-keys --user-name <User>`
	`aws iam update-access-key --user-name <User> --access-key-id <KeyID> --status Inactive`

복구 및 후속 조치 (Post-IR)	• 데이터 복구 - CloudEndure Disaster Recovery 활용 → 랜섬 감염 이전 복구 시점 선택 후 워크로드 복원 - 백업 데이터가 암호화되지 않았는지 검증 후 복원 수행 • 인스턴스 재배포 - 신뢰할 수 있는 AMI 또는 백업 EBS로 새 인스턴스 생성 - 기존 인스턴스 종료 전 IAM·KMS 키 갱신 • IAM 정책 재검토 및 정리 - 루트·운영자 권한 최소화, 임시 자격 증명 폐기 • 장기 모니터링 및 예방 - CloudWatch 이상 징후(네트워크 출력 급증, CPU 스파이크) 기반 경보 구성 - Prowler·Security Hub를 활용한 보안 규정 준수 점검 자동화

10) Ransom Response for RDS

AWS RDS가 랜섬웨어 공격에 의해 데이터 암호화, 삭제, 무단 스냅샷 유출 시도가 발생한 사고 사례이다. 공격자는 IAM 키 탈취, 공개된 RDS 엔드포인트, 취약한 접근제어(Security Group, Public Access) 등을 악용해 DB 인스턴스에 접근하고, 스냅샷 생성·암호화·삭제 등의 API 호출을 통해 데이터를 암호화해 금전적 요구를 한다. RDS는 EC2보다 자동화된 백업·복원 기능이 존재하지만, 초기 감염 단계에서의 탐지와 백업 무결성 검증이 핵심이다.

[표 80] DFIR 관점 분석 포인트 – RDS 랜섬웨어 감염

구분	핵심 분석 항목
원인 분석	공개된 RDS 엔드포인트, 약한 인증, IAM Key 탈취 또는 취약한 Security Group 설정
공격 흔적	CloudTrail의 `CreateDBSnapshot`, `StartExportTask`, `DeleteDBSnapshot` 이벤트, RDS 로그 내 대량 쿼리·삭제 명령, 외부 IP 접근
영향 범위	데이터 암호화·삭제, 스냅샷 무단 유출, 서비스 불가 상태 발생
대응 강화	Deletion Protection 활성화, RDS 암호화 및 다중 AZ 백업, IAM 최소 권한 정책, AWS Config 및 Security Hub 규칙 상시 모니터링, 백업 무결성 주기적 검증

[표 81] 주요 로그 및 데이터 – RDS 랜섬웨어 감염

구분	데이터 소스	확보 목적
CloudTrail	`CreateDBSnapshot`, `ModifyDBInstance`, `DeleteDBSnapshot`, `StartExportTask`, `ModifyDBClusterSnapshotAttribute`	비정상적인 스냅샷 생성·삭제·내보내기 행위, 공격자 IAM 식별
CloudWatch Metrics	CPUUtilization, FreeStorageSpace, NetworkPacketsOut	암호화 프로세스 실행, 데이터 유출 징후, 스토리지 급감 탐지
VPC Flow Logs	VPC 내 네트워크 트래픽 흐름	외부 IP에서 RDS 엔드포인트로 접근 시도, 반복 로그인 실패 등 이상 트래픽 식별
RDS Database Log	Error / General / Audit Log	무단 사용자 로그인, 대량 쿼리, Export 명령 실행 여부 확인
AWS Config	`rds-logging-enabled`, `rds-storage-encrypted`, `rds-snapshots-public-prohibited` 등	백업·암호화·공개 설정 등 보안 구성 상태 및 변경 이력 검증

[표 82] 사고 대응 절차 요약 – RDS 랜섬웨어 감염

절차 구분	수행 행위
증거 수집 및 보존 (Acquire, Preserve, Document)	• CloudTrail에서 비정상 스냅샷 생성/삭제 이벤트 추출 - `CreateDBSnapshot`, `DeleteDBSnapshot`, `StartExportTask` 호출 여부 확인 - 수행자, IP, 접근 채널(Console/API/CLI) 및 호출 시간 기록 • AWS Config의 리소스 타임라인에서 준수 상태 점검 - `rds-snapshots-public-prohibited`, `rds-storage-encrypted` 등 • RDS Error/Audit Log를 통해 무단 로그인 시도 및 SQL 명령 실행 여부 분석 (ALTER/DROP/EXPORT) • VPC Flow Logs로 외부 접근 IP, C2 서버 가능성 탐지 • 모든 증거를 S3 Evidence Bucket에 백업 및 암호화(KMS 적용)
사고 격리 (Containment)	• RDS 퍼블릭 접근 차단 - 콘솔 → RDS → DB 인스턴스 선택 → Connectivity → Public access "No" 설정 • 보안 그룹 격리 - 허용된 IP 외의 인바운드/아웃바운드 규칙 제거 • IAM 정책 검토 - RDS 관련 과도한 권한(rds:*, *:*)을 보유한 역할·사용자 권한 제거 • 스냅샷 Export 중단 - 진행 중인 StartExportTask 작업 중지 `aws rds cancel-export-task --export-task-identifier <task_id>` • AWS Config 규칙 활성화 확인 - `rds-snapshots-public-prohibited`, `rds-instance-public-access-check`

사고 제거
(Eradicate)
	• 비인가 스냅샷 및 인식되지 않은 DB 리소스 제거
	- 콘솔 → RDS → Snapshots → Manual snapshots → 삭제 대상 선택 → Delete Snapshot
	• 네트워크 IoC 기반 차단
	- VPC NACL 수정 → 공격자 IP CIDR에 대해 Inbound/Outbound DENY 규칙 추가
	• IAM 자격 증명 정리
	- 노출 가능성이 있는 IAM Key 갱신
	- 무단 생성된 IAM 사용자·역할·정책 삭제
	`aws iam update-access-key --user-name <user> --access-key-id <key> --status Inactive`
	•
	• 데이터베이스 삭제 보호(Deletion Protection) 재활성화
	- 중요 DB 인스턴스에 대해 삭제 방지 속성 설정
	`aws rds modify-db-instance --db-instance-identifier <db_id> --deletion-protection`

절차 구분	수행 행위
복구 및 후속 조치 (Post-IR)	• 백업 데이터 검증 후 복원 - CloudEndure Disaster Recovery 또는 자동 백업(AWS Backup, RDS Point-in-Time Restore) 사용 - 복구 시 감염된 데이터 재반입 방지를 위해 스냅샷 무결성 확인 • 신규 RDS 인스턴스 배포 - 신뢰할 수 있는 백업에서 복원, 보안 그룹·IAM 역할 재검토 후 배포 • IAM 키·루트 키 갱신 및 관리 강화 - 전체 액세스 키 로테이션 수행 • 보안 구성 자동 점검 - AWS Config 및 Security Hub 규칙 활성화 - Prowler를 통한 정기 점검 (extra723 – 공개 스냅샷 탐지, extra735 – RDS 암호화 확인, group13 – 전체 RDS 보안 점검

11) Ransom Response for S3

AWS S3가 랜섬웨어 공격을 통해 데이터가 삭제 및 암호화되거나 외부로 유출된 사고 사례이다. 공격자는 탈취된 IAM 키, 공개된 S3 버킷, 퍼블릭 접근 차단 설정 해제, S3 API 호출 권한 남용 등을 통해 접근하며, 주로 API를 활용해 데이터를 암호화하거나 삭제한다.

[표 83] DFIR 관점 분석 포인트 – S3 랜섬웨어 감염

구분	핵심 분석 항목
원인 분석	탈취된 IAM 키 또는 과도한 권한 정책, MFA 미적용, S3 퍼블릭 접근 차단 해제
공격 흔적	CloudTrail의 `DeleteObject`, `DeleteBucket`, `PutBucketEncryption`, S3 Access Log의 대량 DELETE/COPY 요청
영향 범위	객체 삭제·암호화, 데이터 유출, 백업 무력화 및 스냅샷 손상
대응 강화	S3 Object Lock·Versioning 활성화, MFA Delete 적용, Config·GuardDuty 모니터링, IAM 최소 권한 정책 준수, CloudTrail Data Event Logging 활성화

[표 84] 주요 로그 및 데이터 – S3 랜섬웨어 감염

구분	데이터 소스	확보 목적
CloudTrail	`DeleteBucket`, `DeleteObject`, `PutBucketEncryption`, `PutObjectAcl`, `PutBucketPolicy`, `PutBucketReplication`	데이터 삭제, 암호화 정책 변경, 버킷 정책 조작 등 행위자 및 시점 확인
S3 Server Access Log	요청자(Requester), 원격 IP(Remote IP), 요청 유형(`REST.COPY.OBJECT`, `GET`, `DELETE`)	대량 객체 삭제·복사 등 유출 시도 또는 파괴 행위 식별
CloudWatch Metrics	NumberOfObjects, BucketSizeBytes, 4xxErrors, 5xxErrors	갑작스러운 객체 수 감소, 스토리지 급감, 에러율 급증 탐지
AWS Config	`s3-bucket-versioning-enabled`, `s3-bucket-default-lock-enabled`, `s3-bucket-level-public-access-prohibited`, `s3-bucket-server-side-encryption-enabled`	버전 관리, 암호화, 퍼블릭 접근 차단 등 보안 설정 준수 상태 확인
IAM Access Analyzer	s3:* 또는 외부 공유 정책(Principal: *) 포함 여부	권한 남용 및 외부 계정 공유 여부 파악

[표 85] 사고 대응 절차 요약 – S3 랜섬웨어 감염

절차 구분	수행 행위
증거 수집 및 보존 (Acquire, Preserve, Document)	• CloudTrail 로그에서 비인가 API 호출 식별 - `DeleteBucket`, `DeleteObject`, `PutBucketEncryption`호출 여부 확인 - 호출자, IP, UserAgent, 접근 경로(API/Console) 추출 • S3 Access Log에서 동일한 원격 IP·요청자에 의한 연속적 REST.COPY.OBJECT, DELETE 요청 확인 - 대량 삭제 또는 데이터 유출 여부 분석 • AWS Config 규칙 위반 내역 확인 - `s3-bucket-versioning-enabled`, `s3-bucket-public-write-prohibited` • 로그 및 복구용 백업을 S3 Evidence Bucket에 보존 (SSE-KMS 암호화 및 버전 관리 적용)
사고 격리 (Containment)	• 퍼블릭 접근 차단 및 정책 제한 - 콘솔 또는 CLI에서 즉시 Block Public Access 설정 `aws s3api put-public-access-block --bucket <Bucket_Name> --public-access-block-configuration "BlockPublicAcls=true,IgnorePublicAcls=true, BlockPublicPolicy=true, RestrictPublicBuckets=true"` • IAM 권한 제한 - 과도한 s3:* 권한을 가진 사용자·역할을 식별해 비활성화 - 루트 사용자 접근 제한 및 MFA 적용 • 버킷 정책 검토 - Principal: *, Effect: Allow 조합 확인 후 제거 • 로그 백업 - S3 서버 액세스 로그 및 CloudTrail 로그를 별도 버킷으로 복사
사고 제거 (Eradicate)	• 비인가 객체 및 변경 사항 제거 - 콘솔 → S3 → 버킷 → Objects → 삭제 마커(Delete Marker) 확인 및 제거 • S3 암호화 정책 복원 - 공격자에 의해 변경된 암호화 설정(`PutBucketEncryption`) 복구 - 정상적인 SSE-KMS 또는 SSE-S3 설정 재적용 • 비인가 IAM 자격 증명 및 Role 삭제 - `aws iam delete-user` 또는 `delete-access-key`를 사용해 공격자 생성 계정 정리 • 네트워크 기반 차단 - CloudFront, API Gateway 등과 연계된 외부 유출 경로 식별 후 폐쇄

절차 구분	수행 행위
복구 및 후속 조치 (Post-IR)	• 버전 관리 객체 복원 - Versioning 활성화된 버킷의 이전 버전에서 정상 데이터 복원 `aws s3api delete-object --bucket <Bucket_Name> --key <Object_Key> --version-id <DeleteMarkerID>` • 삭제된 버킷 복구 - 동일 이름으로 재생성 후 교차 리전 복제(Cross-Region Replication) 버킷 또는 백업 버킷에서 데이터 복원 • CloudEndure Disaster Recovery 및 백업 복원 - 랜섬웨어 발생 이전 시점으로 복원 지점 선택 - 외부 백업 솔루션(Veritas, CommVault 등) 사용 시 백업 무결성 확인 후 복원 • • 보안 정책 강화 - MFA Delete 및 Object Lock(Compliance Mode) 활성화 - IAM Access Analyzer를 통해 외부 공유 리소스 점검 - 신규 리전 자동 로그 활성화(SCP/Control Tower 설정)

12) Unauthorized Network Changes

AWS 계정 내 보안 그룹, NACL, 라우팅, 게이트웨이 ENI 등 네트워크 자산에 무단 또는 의심 변경이 발생한 사고 사례이다. 알 수 없는 리소스 생성(EC2, LB, NAT 등), 보안 그룹 개방, 라우팅 변경, 비용 급증 등을 통해 사고를 인지할 수 있다.

[표 86] DFIR 관점 분석 포인트 – 네트워크 자산 무단 변경 사고

구분	핵심 분석 항목
원인 분석	탈취/오용된 IAM 자격 증명, 과도한 권한 정책, 변경 승인 미준수, IaC 파이프라인 오동작
공격 흔적	CloudTrail의 Security Group/NACL/Route/IGW 변경 연속 호출, `RunInstances`로 공격 발판 생성, Flow Log의 신규 외부 통신
영향 범위	외부 노출 확대, 트래픽 우회/가로채기, 데이터 유출 경로 생성, 비용 급증
대응 강화	최소 권한·권한경계, 전 리전 CloudTrail/Config/FlowLog 상시 활성화, EventBridge 실시간 탐지, SSM 자동 롤백, 변경관리(CAB)·태깅·CMDB 정합성 점검

[표 87] 주요 로그 및 데이터 – 네트워크 자산 무단 변경 사고

구분	데이터 소스	확보 목적
CloudTrail	`Authorize/RevokeSecurityGroup*`, `Create/ModifyRoute*`, `Create/AttachInternetGateway`, `CreateNatGateway`, `RunInstances`	누가/언제/어디서 네트워크 Control Plane 변경을 했는지 추적
VPC Flow Logs	ENI/VPC/Subnet 레벨 트래픽 기록	변경 전후 허용/거부 트래픽 흐름 비교, 외부 통신 여부 확인
AWS Config	Security Group/NACL/RouteTable/IGW/ENI 구성 히스토리	변경 타임라인, 이전 상태 스냅샷, 비준수 리소스 식별
CloudWatch	NAT 게이트웨이·ALB·EC2 네트워크 지표	트래픽 급증/이상 포트 사용 등 운영 영향 탐지
Network Manager, VPC 콘솔	토폴로지·경로 분석	경로 변동, 신규 경유지, 비정상 연결 지점 확인

[표 88] 사고 대응 절차 요약 – 네트워크 자산 무단 변경 사고

절차 구분	수행 행위
증거 수집 및 보존 (Acquire, Preserve, Document)	• CloudTrail 로그에서 비인가 API 호출 식별 - AuthorizeSecurityGroupIngress/Egress, Revoke*, Create/ModifyRoute*, AttachInternetGateway, RunInstances 호출 여부 확인 - 호출자, sourceIPAddress, userAgent, 리전 추출 • AWS Config 리소스 타임라인 - 보안 그룹 규칙, 라우트 테이블, NACL 변경 전/후 상태 캡처 • VPC Flow Logs 스냅샷 보존 - 변경 시점 ±(전/후) 2시간 파티션 보관, S3 증거 버킷에 복사 • 운영 영향 확인 - NAT/ALB/EC2 네트워크 지표(BytesOut, ActiveFlow, 4xx/5xx) 점검
사고 격리 (Containment)	• 과다 개방 보안 그룹 임시 교체(격리 Security Group로 바인딩), 필요 시 NACL DENY로 추가 차단 • 의심 리소스 생성자 식별(CloudTrail RunInstances 등) → 해당 IAM 사용자/역할 권한 제한(정책 Detach 또는 신뢰 정책 차단) • 경로 오염 의심 시 - 문제 라우트 엔트리 비활성화(우회 경로로 일시 전환)
사고 제거 (Eradicate)	• 비인가 변경 원복 - AWS Config 이전 정상 스냅샷 기준으로 Security Group/NACL/RouteTable/IGW 상태 복구 • 비인가 리소스 정리 - 알 수 없는 EC2/NAT/ENI/LB 제거, 잔여 Security Group/Key/Role/태그 점검 • CloudTrail 재검토 - 추가 권한 상승(AssumeRole, Attach*Policy) 및 확장 흔적 유무 확인
복구 및 후속 조치 (Post-IR)	• 서비스 경로 및 헬스체크 정상화 확인(ALB/NLB/Route53) • CloudWatch Alarm 구성 - NAT BytesOutToDestination, ALB RejectedConnectionCount, EC2 NetworkIn/Out, Flow Log 기반 탐지 • 자동 탐지/차단 강화 - EventBridge → Authorize/RevokeSecurityGroup*, ModifyRoute* 이벤트 즉시 알림 - AWS Config 규칙(보안 그룹 과다 개방, 퍼블릭 경로) + 자동 수정(SSM Automation) - Security Hub FSBP 활성화

13) GuardDuty: PrivilegeEscalation-Kubernetes:PrivilegedContainer

Amazon GuardDuty가 탐지한 EKS 클러스터 내 권한 상승 관련 사고를 탐지한 사례이다. 공격자는 관리 권한이 과도하게 부여된 IAM Role 또는 Kubernetes 계정을 통해 Root 권한의 컨테이너(Privileged Container)를 실행함으로써 클러스터 제어, 데이터 유출, 내부 네트워크 확장 공격 등으로 이어질 가능성이 있다.

[표 89] DFIR 관점 분석 포인트 – EKS 클러스터 내 권한 상승

구분	핵심 분석 항목
원인 분석	과도한 ServiceAccount/IAM Role 권한, 오용된 OIDC 인증 정보, Privileged 컨테이너 허용 설정
공격 흔적	Privileged Pod 실행, 비정상 RoleBinding·ClusterRole 변경, GuardDuty 탐지 이벤트
영향 범위	EKS 클러스터 제어권 탈취, 내부 네트워크 확장, 데이터 유출 가능성
대응 강화	RBAC 최소 권한 원칙 적용, Privileged 모드 차단, Secrets 갱신 및 모니터링 강화, GuardDuty 및 Config 규칙 지속 검증

[표 90] 주요 로그 및 데이터 – EKS 클러스터 내 권한 상승

구분	데이터 소스	확보 목적
GuardDuty Findings	GuardDuty 탐지 이벤트	권한 상승(PrivilegedContainer) 탐지 근거 및 실행 주체 식별
CloudTrail Log	AWS 계정 내 API 호출 로그	EKS, IAM, STS 관련 API 호출 및 자격 증명 사용 내역 추적
CloudWatch (EKS Audit Log)	EKS Control Plane 감사 로그	RoleBinding/ClusterRole 변경, API 호출, 서비스 계정 활동 식별
AWS Config	리소스 변경 이력 및 규칙 평가 결과	클러스터 및 IAM 리소스의 설정 변경 여부 확인
Security Hub, Detective	GuardDuty 연계 탐지 결과 분석	계정, 리전 단위 위협 상관 분석
포렌식 아티팩트 (EBS Snapshot, 컨테이너 로그 등)	워커 노드 디스크, 런타임 로그	침해된 노드/Pod의 파일, 프로세스, 포트, 네트워크 상태 등 증거 보존

[표 91] 사고 대응 절차 요약 – EKS 클러스터 내 권한 상승

절차 구분	수행 행위
증거 수집 및 보존 (Acquire, Preserve, Document)	• GuardDuty Finding 기반으로 EKS 클러스터, Pod, User, Node 식별 • CloudTrail 및 EKS Audit Log로 API 호출 및 권한 변경 내역 분석 • Pod 컨테이너 로그, 프로세스 목록, 변경 파일 등 휘발성 데이터 수집 • 증거 보존을 위해 EBS 스냅샷 생성
사고 격리 (Containment)	• 관련 Kubernetes 사용자 차단, IAM Role 자격 증명 갱신 • 문제된 Pod 격리 또는 일시 중지 • 워커 노드 Cordon 적용(신규 스케줄링 차단) • ConfigMap 및 ServiceAccount 매핑 관계 검토
사고 제거 (Eradicate)	• Privileged Container 생성 원인 분석 (잘못된 RoleBinding, ServiceAccount 권한 등) • 비정상 IAM Role 및 ServiceAccount 삭제 또는 권한 축소 • CloudTrail 기반으로 신규 리소스 생성 및 변경 내역 확인 및 정리
복구 및 후속 조치 (Post-IR)	• Kubernetes RBAC 정책 재정비 • OIDC 인증 정보, Secrets 갱신 • GuardDuty, Security Hub, Config 규칙 재점검 • Kubernetes Audit 로그 장기 보존 설정

14) GuardDuty: Discovery – Kubernetes/SuccessfulAnonymousAccess

Amazon GuardDuty가 Kubernetes 클러스터에서 인증되지 않은 사용자에 의해 API 요청이 성공적으로 수행된 활동을 탐지한 사례이다. 이 탐지는 Kubernetes API 서버의 익명 접근 허용(Anonymous Access) 설정이 잘못되어 있음을 시사하며, 이는 구성 오류(Misconfiguration) 또는 자격 증명 탈취(Compromise) 로 이어질 수 있다.

[표 92] DFIR 관점 분석 포인트 – Kubernetes 클러스터에서 인증되지 않은 사용자에 의한 API 요청

구분	핵심 분석 항목
원인 분석	Kubernetes RBAC 오구성, `system:anonymous` 허용 설정 유지, API 서버 인증 정책 미적용
공격 흔적	익명 사용자(`system:anonymous`)의 API 호출, ClusterRoleBinding 변경 로그
영향 범위	클러스터 메타데이터 유출, Pod/서비스 구조 노출, 추가 침입으로의 확산 가능성
대응 강화	익명 접근 차단(`anonymous-auth=false`), RBAC 검증 자동화, GuardDuty 및 Audit Log 모니터링 강화

[표 93] 주요 로그 및 데이터 – Kubernetes 클러스터에서 인증되지 않은 사용자에 의한 API 요청

구분	데이터 소스	확보 목적
GuardDuty Findings	GuardDuty 탐지 이벤트	익명 사용자(`system:anonymous`)에 의해 호출된 API 이벤트 식별 및 호출 주체 파악 (IP·ASN·Region 등)
CloudTrail Log	AWS 계정 내 API 호출 로그	EKS 관련 API 호출 이력 및 IAM·STS 관련 활동 추적
CloudWatch (EKS Audit Log)	Kubernetes Control Plane 감사 로그	비정상 API 호출, RoleBinding/ClusterRoleBinding 변경, 익명 사용자 접근 내역 확인
AWS Config	리소스 설정 변경 이력	EKS Cluster, IAM Role, Security Group 등의 변경 여부 및 규칙 위반 탐지
Security Hub, Detective	GuardDuty 연계 결과 상관 분석	탐지된 익명 API 호출과 다른 의심 활동의 연관성 평가
포렌식 아티팩트 (EBS Snapshot)	EKS 워커 노드 증거 이미지	향후 분석을 위한 시스템 레벨 증거 보존 및 변경 이력 검증

[표 94] 사고 대응 절차 요약 – Kubernetes 클러스터에서 인증되지 않은 사용자에 의한 API 요청

절차 구분	수행 행위
증거 수집 및 보존 (Acquire, Preserve, Document)	• GuardDuty Finding 기반으로 EKS 클러스터, Pod, User, Node 식별 • CloudWatch(EKS Audit Log)로 `system:anonymous` API 호출 시점 및 리소스 종류 파악 • CloudTrail 로그에서 EKS 및 IAM 관련 API 호출 내역 검토 • EBS 스냅샷 등 관련 증거 데이터 보존
사고 격리 (Containment)	• system:anonymous 사용자 사용 필요성을 검토 - 운영 요구사항이 없다면 익명 접근을 비활성화 - 프로덕션 워크로드 영향이 예상되면 변경 전 검증 필수 • RBAC 구성 점검 - rbac-lookup 도구 등을 활용해 system:unauthenticated 또는 system:anonymous 그룹에 부여된 ClusterRoleBinding 식별 - 불필요한 바인딩 제거 (system:discovery, system:basic-user 등) • 클러스터 관리자에게 접근 현황 공유 및 승인 절차 검증
사고 제거 (Eradicate)	• CloudTrail을 활용해 최근 90일간 EKS 관련 API 호출 내역 분석 - `CreateUser`, `CreateRole`, `AssumeRole*`, `Attach*Policy`, `Get*Token` - RunInstances (PassRole 포함) 및 신규 리소스 생성 API - 기존 리소스의 수정·삭제 흔적 • 익명 접근이 활성화된 RootCause 확인 (API Server 설정, RoleBinding 등) • 불필요한 ClusterRole 및 RoleBinding 제거 • IAM 및 ServiceAccount 자격 증명 갱신
복구 및 후속 조치 (Post-IR)	• RBAC 최소 권한 원칙 재적용 및 system:anonymous 사용 제한 • API Server의 --anonymous-auth 설정 비활성화 검토 • CloudWatch 기반 EKS Audit 로그 장기 보존 활성화 • GuardDuty, Config, Security Hub의 탐지 규칙 상시 검증

15) GuardDuty: GuardDuty: Impact – IAMUser/AnomalousBehavior

Amazon GuardDuty가 IAM 사용자 또는 역할(IAMUser/Role)의 비정상적인 API 호출을 탐지한 사례이다. 이는 평소와 데이터 변조, 삭제, 운영, 방해시도와 관련된 API 패턴이 탐지되었음을 시사한다.

일반적으로 DeleteSecurityGroup, PutBucketPolicy, UpdateUser 등의 API가 반복 호출되는 경우 탐지된다.

[표 95] DFIR 관점 분석 포인트 – IAM 사용자 또는 역할의 비정상적인 API 호출

구분	핵심 분석 항목
원인 분석	IAM 사용자 또는 Role의 자격 증명 탈취, 과도한 권한 부여, 자동화 스크립트 오남용
공격 흔적	비정상 API 호출(`DeleteSecurityGroup`, `PutBucketPolicy` 등), IAM 정책 수정, Access Key 재발급 흔적
영향 범위	계정 내 리소스 조작, 서비스 중단, 데이터 무결성 훼손
대응 강화	IAM 접근 제어 재점검, Access Key 주기적 갱신, GuardDuty/CloudTrail 실시간 알림 통합

[표 96] 주요 로그 및 데이터 – IAM 사용자 또는 역할의 비정상적인 API 호출

구분	데이터 소스	확보 목적
GuardDuty Findings	GuardDuty 탐지 이벤트	비정상 API 호출 탐지 근거 및 호출 주체(Principal) 확인
CloudTrail Log	AWS 계정 내 API 호출 로그	해당 IAM 주체의 전체 API 호출 내역 및 이상 활동 시점 분석
VPC Flow Logs	네트워크 플로우 기록	외부 접근 IP, 포트, 트래픽 패턴 분석
S3 Server Access Log	S3 객체 접근 로그	데이터 변조 및 삭제 요청 여부 확인
AWS Config	리소스 설정 변경 내역	IAM 정책, S3 정책, 보안 그룹 등 설정 변경 추적
Security Hub, Detective	GuardDuty 연계 탐지 상관분석	동일한 사용자·리전에서 발생한 다른 보안 이벤트와의 연계성 평가

[표 97] 사고 대응 절차 요약 – IAM 사용자 또는 역할의 비정상적인 API 호출

절차 구분	수행 행위
증거 수집 및 보존 (Acquire, Preserve, Document)	• GuardDuty Findings를 통해 비정상 API 호출 유형 및 호출 주체(Principal ID, User Name, ARN) 식별 • CloudTrail을 통해 해당 주체의 최근 90일간 모든 API 활동 내역 확보 • VPC Flow Logs, S3 Access Log로 IP·트래픽 및 데이터 접근 흔적 분석 • GuardDuty Finding이 발생한 시점 전후의 로그를 S3에 백업해 증거 보존

사고 격리
(Containment)
	• 운영 영향 평가
	- IAM 사용자 또는 Role이 프로덕션 워크로드에 사용 중인지 확인
	- 즉시 비활성화 시 서비스 중단이 발생할 수 있으므로 단계적 격리 수행
	• 권한 기록 및 백업
	- 현재 권한 백업
	`aws iam list-attached-user-policies` 및 `get-user-policy` 명령
	- CloudTrail 로그를 로컬 또는 S3로 백업
	• 권한 제거 및 차단
	- Access Key 비활성화
	`aws iam update-access-key --status Inactive`
	- IAM 정책 Detach
	`aws iam detach-user-policy --user-name <user> --policy-arn <arn>`
	- IAM Role의 경우, lookup-events로 AssumeRole 사용자 추적 후 조건부 차단 적용

사고 제거 (Eradicate)	• CloudTrail 로그 분석 (Athena 기반) - 지난 90일간 비정상 IAM 주체의 모든 API 활동 쿼리 수행 - 민감 데이터 접근: `S3 GetObject`, `DescribeInstances` 등 - 신규 리소스 생성: `EC2`, `Lambda`, `RDS`, `CloudFormation`, `Beanstalk` - IAM 관련 리소스 조작: `CreateUser`, `AssumeRole`, `Attach*Policy`, `Get*Token` - 기존 리소스 삭제 및 수정: `DeleteBucket`, `UpdatePolicy`, `UntagResource` • 추가 자격 증명 생성 여부 확인 - CloudTrail 이벤트 내 아래 이미지 호출 탐지 시 즉시 차단 `CreateAccessKey`, `CreateRole`, `GetFederationToken` 등 • 비정상 리소스 정리 - 공격자가 생성한 IAM User, Role, EC2 인스턴스 등을 비활성화 또는 삭제 - 변경된 정책 및 리소스 원상복구
복구 및 후속 조치 (Post-IR)	• IAM 사용자·역할의 최소 권한 원칙(Least Privilege) 재적용 • Access Key 갱신 정책 강화 • CloudTrail, Config, GuardDuty 로그 장기 보존 및 상관 탐지 자동화 • IAM 정책 변경 이벤트에 대한 SNS 알림 규칙 추가

16) GuardDuty: Execution – EC2/MaliciousFile

Amazon GuardDuty의 Malware Protection 스캔 기능이 EC2 인스턴스 내에서 악성 파일을 탐지했을 때 발생하는 사례이다. 이 탐지는 해당 인스턴스가 이미 침해되었을 가능성이 높음을 의미하며, 악성 파일의 경로, 볼륨 ID, 감염 트리거 원인 등은 GuardDuty Finding 세부정보에서 확인 가능하다.

[표 98] DFIR 관점 분석 포인트 – EC2 인스턴스 내에서 악성 파일 탐지

구분	핵심 분석 항목
원인 분석	EC2 인스턴스에 업로드된 악성 파일, 취약한 서비스 노출, 감염된 AMI 또는 S3 오브젝트 실행
공격 흔적	GuardDuty MalwareFinding, CloudTrail 상의 의심 API 호출 (`RunInstances`, `GetObject`, `PutUserData` 등)
영향 범위	인스턴스 내부 악성코드 실행, 시스템 리소스 악용(CPU/네트워크), lateral movement 가능성
대응 강화	EBS Snapshot 자동화 백업, Malware Protection 실시간 스캔 활성화, IAM Instance Profile 최소 권한화, CloudWatch 기반 CPU/네트워크 이상 탐지 경보 구성

[표 99] 주요 로그 및 데이터 – EC2 인스턴스 내에서 악성 파일 탐지

구분	데이터 소스	확보 목적
GuardDuty Findings	Malware Protection 결과	악성 파일이 발견된 인스턴스, 볼륨 ARN, 파일 경로 식별
CloudTrail Log	EC2 관련 API 기록	감염된 인스턴스의 생성/변경/스냅샷 이벤트 추적
VPC Flow Logs	네트워크 트래픽 로그	C2(Command & Control) 또는 외부 데이터 전송 행위 탐지
EBS Snapshot (포렌식 이미지)	EBS 증거 복제본	악성 파일, 프로세스, 로그 포렌식 분석용 보존
Security Group, NACL 설정 로그	보안 규칙 기록	인스턴스 외부 노출 여부 및 격리 가능성 평가
CloudWatch Metrics	CPU, 네트워크, Disk I/O	비정상 프로세스 활동 탐지 및 리소스 부하 패턴 확인

[표 100] 사고 대응 절차 요약 – EC2 인스턴스 내에서 악성 파일 탐지

절차 구분	수행 행위

증거 수집 및 보존
(Acquire, Preserve, Document)
	• GuardDuty Findings에서 다음 항목 식별
	- Instance ID, Volume ARN, 악성 파일 경로/이름, Trigger Finding ID
	• EC2 메타데이터 및 보안 그룹 구성 수집
	`aws ec2 describe-instances`
	• EBS 데이터 볼륨 스냅샷 생성
	`aws ec2 create-snapshot --volume-id <Volume_ID>`
	`--description "Forensic Snapshot of Infected Instance"`
	• 인스턴스 종료 보호 활성화
	`aws ec2 modify-instance-attribute --instance-id <Instance_ID>`
	`--attribute disableApiTermination --value true`
	• DeleteOnTermination 비활성화 및 종료 시 동작을 Stop으로 변경
	• "Quarantine" 태그를 부여해 격리 상태 표시

사고 격리
(Containment)
	• Auto Scaling 그룹 및 ELB에서 인스턴스 제거
	`aws autoscaling detach-instances --instance-ids <Instance_ID>`
	`--auto-scaling-group-name <ASG_NAME>`
	`aws elb deregister-instances-from-load-balancer --instances <Instance_ID>`
	`--load-balancer-name <ELB_NAME>`
	• IAM 인스턴스 프로파일 분리
	`aws ec2 disassociate-iam-instance-profile --association-id <Association_ID>`
	• 보안 그룹 교체로 네트워크 격리
	`aws ec2 modify-instance-attribute --instance-id <Instance_ID> --groups <Isolation_SG_ID>`
	• 필요 시 인스턴스 강제 종료(Shutdown) 전 휘발성 데이터 확보
	- 메모리 덤프, 네트워크 세션, 프로세스 목록
	- Host-based EDR 에이전트 또는 margaritashotgun 활용

사고 제거 (Eradicate)	• 악성코드 제거 - 감염된 인스턴스 내부에서 검증된 AV/EDR 에이전트를 통해 정리 - AWS Marketplace 보안 솔루션(TrendMicro, SentinelOne 등) 활용 가능 • 감염된 인스턴스 유지가 위험한 경우 인스턴스 중지 (신규 인스턴스로 교체 후 서비스 복구) `aws ec2 stop-instances --instance-ids <Instance_ID>` • 감염된 AMI 여부 점검 - 동일 AMI로 배포된 인스턴스 존재 시, 전수 검사 및 재생성 수행
복구 및 후속 조치 (Post-IR)	• 정상 인스턴스 재배포 후 서비스 복원 • CloudTrail, GuardDuty 연계 알림 규칙 검증 • Malware Protection의 스캔 주기 및 감염 대응 정책 개선 • 동일 리전 내 다른 인스턴스, AMI의 무결성 검사 병행

4. 사고 데이터 수집

클라우드 환경에서의 사고 대응은 온프레미스 환경과 달리 물리적 장비나 스토리지에 직접 접근할 수 없기 때문에, 디지털 증거 확보 절차가 핵심으로 작용한다. 다시 말해, 클라우드 서비스가 생성 및 보유하는 데이터를 신속하고 체계적으로 수집하는 과정이 DFIR 수행의 핵심 요소로 작용하며, 이는 온프레미스 환경의 전통적 수집 절차와 뚜렷한 차이를 가진다.

[표 101] 온프레미스 vs. 클라우드 환경에서의 DFIR 데이터 수집 구조 체계

구분	온프레미스 환경	클라우드 환경
활성 데이터	물리적 접근을 통해 직접 수집 가능 (메모리, 세션, 프로세스, 네트워크 연결 등)	게스트 OS 수준에서는 수집 가능 (SSM, Agent, 콘솔 기반 원격 명령)
비활성 데이터	디스크 이미지, 이벤트 로그, 설정 파일 등 로컬 저장 매체에 보존	CloudTrail, Config, S3, EBS Snapshot 등 관리형 저장소에 자동 보존
접근 방식	현장 물리 장비 연결 또는 로컬 접근	가상 인스턴스 내부 진입(EC2) 또는 SSM을 통한 원격 명령 실행
제한 요소	로컬 관리자 또는 물리 장비 접근 권한 필요	하이퍼바이저 및 호스트 레벨 접근 불가 게스트 OS 및 서비스 API 계층만 접근 가능
수집 도구	**휘발성** 메모리 덤프 도구, 네트워크 캡처 도구, 선별 수집 자동화 스크립트 등 **비휘발성** 디스크 이미지 도구, 선별 로그 수집 자동화 스크립트 등	AWS CLI, Systems Manager(SSM), SDK 명령, CloudTrail, AWS Config, CloudWatch, Athena, S3 Export 등
데이터 보존 특성	수동 보존(분석가에 의해 추출·보관)	자동 보존(서비스에 의해 지속 관리)
수집 한계	장비 손상·전원 차단 시 휘발성 데이터 손실 로그 삭제·위·변조 가능성 현장 접근이 제한될 경우 수집 불가	하이퍼바이저 메모리 접근 불가 서비스 비활성·로그 미연동 시 데이터 누락 일부 관리형 서비스 영역은 수집 불가

클라우드 환경에서의 DFIR에서 데이터 수집 절차의 가장 중요한 목적은 사건 재현성을 확보하는 데 있다. 재현성이 확보되어야 공격자가 어떤 권한으로 접근했는지, 어떤 리소스를 수정했는지, 어떤 로그가 생성되었는지 등의 공격 과정을 타임라인 기반으로 재구성할 수 있다.

따라서, 본 연구에서는 AWS 클라우드 환경에서의 사고 대응 시 재현 가능한 데이터 확보를 목표로 수집 대상 분류를 명령 기반 데이터(Command-based Data)와 로그 기반 데이터(Log-based Data), 포렌식 이미지(Forensic Image)로 구분하고 각각의 수집 절차를 설명한다.

4장에서 다루는 내용은 다음과 같다.

[표 102] 주요 연구 내용 – 사고 데이터 수집

번호	소제목	주요 내용
1	사고 분석에서 자주 활용되는 주요 로그	AWS 환경에서 DFIR 수행을 위해 구성요소별 로그 소스와 수집 및 활용 절차를 정의해 7개의 핵심 로그 선별
2	수집 데이터 유형 분류 및 수집 절차	AWS 사고 대응에서 데이터 유형을 명령 기반, 로그 기반, 포렌식 이미지로 구분해 단계별 활용 목적과 재현성 확보 방안 제시

4.1. 사고 분석에서 자주 활용되는 주요 로그

클라우드 환경에서 효과적으로 DFIR을 수행하려면 AWS 환경에서 어떤 로그를 수집할지 우선 결정해야 한다. 이를 위해 먼저 배포된 애플리케이션의 구성 요소(EC2, RDS, S3 등)와 클라우드 애플리케이션 스택의 계층(네트워크, 애플리케이션, 데이터 계층)을 파악해야 한다. 각 구성요소별로 로그 소스의 종류와 활용 목적, 수집 및 저장 절차를 명확히 정의함으로써, 사고 분석에 활용할 핵심 로그를 선별할 수 있다. 주요 활용 로그는 다음과 같다.

1) AWS CloudTrail Log

AWS CloudTrail은 AWS 계정의 거버넌스, 규정 준수, 운영 및 위험 감사를 지원하는 서비스이다. 사용자가 관리 콘솔, SDK, 명령줄 도구 및 기타 AWS 서비스를 통해 수행되는 모든 작업은 CloudTrail 이벤트로 로깅된다. 이렇게 로깅된 로그는 보안 분석, 리소스 변경 추적, 규정 준수 감사 등에 필수적인 데이터로 활용된다.

CloudTrail은 기본적으로 활성화되어 있으며, 지난 90일 간의 관리 이벤트만 확인 가능하다. 따라서, 로그 보관을 위해 Trail을 생성해 로그 파일을 Amazon S3 버킷에 저장해야 한다. CloudTrail 로그는 JSON 형식으로 기록되며, 각 이벤트는 여러 개의 키-값 쌍으로 구성된 필드를 포함한다.

2) AWS VPC Flow Logs

AWS VPC Flow Logs는 사용자의 Amazon Virtual Private Cloud(VPC) 내 네트워크 인터페이스를 통과하는 IP 트래픽 정보를 캡처하는 기능이다. 이 로그는 누가, 어디서, 언제, 어떤 포트를 통해 통신했는지에 대한 상세한 기록을 제공하며, 네트워크 트래픽 패턴 분석, 잠재적 위협 탐지, 보안 및 규정 준수 요구사항 충족에 필수적인 데이터를 제공한다. Flow Log는 생성 후 Amazon S3 또는 CloudWatch Logs에 저장할 수 있으며, 로그 저장 및 전송에는 추가 비용이 발생한다.

VPC Flow Logs를 S3에 저장할 경우 일반 텍스트 또는 Parquet 형식(Gzip 압축을 사용하는 열 기반 데이터 형식)으로 기록되며, CloudWatch에 저장할 경우 CloudWatch 서비스 콘솔에 기록된다.

3) Amazon S3 Server Access Log

Amazon S3 Server Access Log는 Amazon S3 버킷에 대한 모든 요청(Request) 정보를 기록하는 기능이다. 이 로그는 누가, 언제, 어디서, 어떻게 S3 버킷의 객체(파일)에 접근했는지에 대한 상세한 정보를 제공한다. 이를 통해 S3 버킷에 대한 접근 패턴을 분석하거나, 보안 및 규정 준수 감사, 사고 발생 시 분석의 핵심 자료로 활용할 수 있다.

AWS에서 S3 버킷의 객체 단위 접근 로그를 S3 Server Access Log와 CloudTrail S3 Data Event에서 확인 가능하지만, 목적과 활용 방안에서 차이점이 존재한다. S3 Server Access Log는 공백으로 구분된 필드들의 목록으로 구성된 텍스트 파일 형식으로 기록되며, 각 로그 레코드는 단일 S3 요청에 대한 정보를 담고 있다.

[표 103] S3 Server Access Log vs. CloudTrail S3 Data Event

구분	S3 Server Access Log	CloudTrail S3 Data Event
기본 상태	• 비활성화 (별도 설정 필요)	• 비활성화 (Trail에서 Data Event 활성화 필요)
로그 저장 위치	• 지정한 S3 버킷	• S3 버킷, CloudWatch Log, CloudTrail Lake
포맷	• Text (Apache access log와 유사)	• JSON
기록 단위	• 요청(Request) 기반	• API 호출 이벤트 기반
포함 정보	• 요청자, 요청 일시(UTC), 요청 IP, 요청 유형(GET/PUT 등), HTTP 상태 코드, 전송 바이트 수, User-Agent 등	• 이벤트 시간, 이벤트 소스, 이벤트 이름 (GetObject, PutObject 등), 요청자, 요청 파라미터, 응답 요소, 리전 등
로깅 시간	• 로깅에 지연 발생 가능	• 실시간 로깅
활용 목적	• 어떤 IP에서 객체에 접근했는지 추적 • 다운로드/업로드 발생 여부 추적 • DDoS/대량 접근 행위 확인	• API 호출 행위 확인 • IAM 권한 오남용 탐지 • 특정 사용자/역할 기반 접근 추적
비용	• 로그 저장용 S3 버킷 비용 발생	• Data Event 설정 시 과금 발생 (API 호출 100,000건 단위+로그 저장 비용)
장점	• 실제 요청 흐름, HTTP 정보(User-Agent, Bytes 전송 등) 확인 가능	• CloudTrail 표준 이벤트 포맷으로 다른 AWS 로그와 연계 분석 용이
한계	• JSON 구조가 아니기 때문에 추가적인 파싱 과정 필요 • CloudTrail처럼 IAM/STS 세션 정보 확인 불가	• HTTP 정보 확인 불가 (User-Agent, Bytes 전송 등) • 과금 부담 존재

4) Amazon CloudWatch Logs

CloudWatch Logs는 Amazon CloudWatch 서비스가 AWS 리소스와 애플리케이션으로부터 수집한 로그 데이터를 저장·관리하는 기능이다. 이 기능은 다양한 로그 소스(EC2 인스턴스의 시스템 로그, Lambda 실행 로그, VPC Flow Logs, CloudTrail 이벤트 로그 등)를 중앙에서 통합 관리하도록 설계되어 있다. CloudWatch Logs를 통해 운영 및 보안 이벤트 중 비정상 활동을 탐지할 수 있다.

로그는 JSON 기반의 구조화된 형식으로 기록되며, 발생 시각, 로그 그룹, 로그 스트림, 메시지 등의 정보를 포함한다. 공격 행위 식별, 로그 상관분석을 수행할 수 있어 DFIR의 핵심 데이터로 활용된다.

5) Amazon RDS Logs

Amazon RDS Logs는 Amazon RDS DB 인스턴스에서 발생하는 다양한 활동과 이벤트에 대한 기록을 담고 있는 파일이다. 이 로그들은 데이터베이스의 운영 상태를 모니터링하고 성능 문제를 해결하며, 보안 감사를 수행하고 잠재적인 보안 위협을 탐지하는 데 결정적인 역할을 한다. Amazon RDS DB 인스턴스에서는 데이터베이스 엔진으로 MariaDB, Microsoft SQL Server, MySQL, Oracle, PostgreSQL을 지원한다.

Amazon RDS Logs는 DB 엔진 내부에서 발생하는 이벤트(접속, 쿼리 실행, 오류 등)에 대해 기록하는 데이터베이스 로그와 AWS DB 인스턴스와 관련된 변경 사항에 대해 기록하는 AWS DB 인스턴스 이벤트로 구분된다. 데이터베이스 로그의 구조는 데이터베이스 엔진(MySQL, PostgreSQL, MariaDB 등)과 로그 유형에 따라 다르다. 로그는 데이터베이스 엔진 종류에 따라 다양한 유형의 로그를 제공하며, 대표적인 로그는 다음과 같다.

[표 104] 대표적인 Amazon RDS 로그

로그 구분	설명
오류 로그 (Error Log)	• 데이터베이스 시작 및 종료 시간, 오류, 경고, 참고 사항 등 진단 메시지를 기록
Slow 쿼리 로그 (Slow Query Log)	• 실행 시간이 오래 걸리는 SQL 쿼리를 기록해 데이터베이스 성능 저하의 원인을 파악하는 데 도움을 줌
일반 쿼리 로그 (General Query Log)	• 클라이언트의 연결 및 연결 해제, 실행된 모든 SQL 쿼리를 기록
감사 로그 (Audit Log)	• 데이터베이스에 대한 접근 및 활동을 추적하기 위한 로그로, 로그인 성공 및 실패, 특정 데이터에 대한 접근, 데이터 변경 등의 활동을 기록

6) AWS GuardDuty Findings

AWS GuardDuty Findings는 GuardDuty가 AWS 환경 내에서 잠재적인 보안 위협을 탐지했을 때 생성하는 상세한 보안 경고이다. GuardDuty는 다양한 데이터 소스(VPC Flow Logs, AWS CloudTrail 이벤트 로그, DNS 로그 등)를 머신러닝, 이상 행위 탐지, 통합 위협 인텔리전스를 활용해 분석하고, 위협으로 판단될 경우 Finding을 생성해 사용자에게 알린다.

각 Finding은 탐지된 보안 문제에 대한 풍부한 정보를 담고 있어, 보안 담당자가 상황을 신속하게 파악하고 적절한 조치를 취할 수 있도록 돕는다. 예를 들어, 특정 EC2 인스턴스가 악성 IP 주소와 통신하거나 평소와 다른 지역에서 IAM 사용자의 API 호출이 발생하는 등의 활동이 감지되면 관련 Finding이 생성된다.

GuardDuty Finding은 표준화된 JSON 형식을 가지며, 침해 사고 분석에 필수적인 다양한 세부 정보를 포함하고 있다.

7) AWS WAF Log

AWS WAF Log는 Amazon Web Services에서 제공하는 웹 애플리케이션 방화벽인 AWS WAF에 의해 처리된 모든 웹 요청에 대한 자세한 정보를 저장하고 있는 로그이다. 웹사이트나 웹 애플리케이션으로 들어오는 HTTP/HTTPS 요청이 AWS WAF의 규칙에 의해 허용(ALLOW), 차단(BLOCK) 되었는지 등의 처리 결과를 포함해 각 요청의 상세 내역을 확인할 수 있다. 이 로그를 통해 보안 담당자는 잠재적인 위협을 식별하고, 비정상적인 접근 시도를 분석하며, 설정된 보안 규칙이 효과적으로 작동하는지 모니터링할 수 있다.

AWS WAF Log는 JSON 형식으로 기록되며, 각 요청에 대한 다양한 필드 정보를 포함하고 있다. 로그 구조는 WAF 버전에 따라 상이할 수 있다.

4.2. 수집 데이터 유형 분류 및 수집 절차

AWS 환경의 사고 대응에서 수집 데이터는 수집 방식과 특성에 따라 명령 기반 데이터(Command-based Data), 로그 기반 데이터(Log-based Data), 포렌식 이미지(Forensic Image)로 구분된다.

데이터의 유형은 각각 실시간 대응(Detection, Containment)과 사후 분석(Analysis, Post-IR) 단계에서 상호 보완적으로 활용된다. 이러한 절차적 구분을 통해 사고 대응의 단계별 목표를 명확히 하고 수집 데이터의 신뢰성과 재현성을 확보할 수 있다.

[표 105] 데이터 수집 절차에 따른 유형

수집 순서	구분	목적 및 특징
1	명령 기반 데이터 수집	• **목적** - 사고 시점의 인스턴스 상태, 구성 정보 등 실시간 데이터를 확보해 휘발성 정보 보존 • **주요 특징** - 시간이 지나면서 자동 변경 및 삭제될 수 있는 휘발성 데이터 - AWS CLI, AWS SSM 등을 통해 API 명령 호출로 확보 - 로그보다 먼저 수집해야 함 (변경 가능성이 높기 때문)
2	로그 기반 데이터 수집	• **목적** - 사고 전후의 행위 이력 분석 • **주요 특징** - CloudTrail, VPC Flow Logs, S3 Access Log, CloudWatch 등 활성화된 로그 - 명령 기반 데이터보다 변동성이 낮아 조사 단계에 수집해도 무방
3	포렌식 이미지 수집	• **목적** - 사고 심층 분석용 데이터를 확보해 사고 원인 규명과 재현성 보장 • **주요 특징** - 디스크, 메모리, 스냅샷 등 비교적 대용량 증거로 구성 - 사후 복구 및 법적 증거 보존용

유형별 데이터 수집 방법은 다음과 같다.

1) 명령 기반 데이터 수집 (Command-based Data)

명령 기반 데이터는 운영 중인 인스턴스나 서비스의 현재 상태를 실시간으로 명령과 API를 통해 조회한 데이터를 의미한다. 온프레미스 환경에서는 관리자 권한으로 직접 명령을 실행해 확보하는 시스템 상태 데이터를 뜻하나, AWS 클라우드 환경에서는 로컬 명령 실행 대신 API, SSM, 콘솔 명령을 통한 원격 수집 구조로 바뀐다.

[표 106] AWS 환경에서의 명령 기반 데이터 개요

구분	설명
정의	• 사고 대응 중 관리자 또는 자동화 도구가 API와 명령을 통해 실시간 조회 또는 수집하는 데이터 • 인스턴스 내부 상태·프로세스·네트워크 연결 등 운영 상태(State) 중심 정보
주요 수집 방식	• AWS Systems Manager (SSM) Run Command / Session Manager • EC2 Instance Connect CLI • AWS CLI / SDK • Lambda를 통한 원격 진단 명령 실행
데이터 특성	• 수집 시점에 따라 내용이 바뀜 - 휘발성(Volatile) • 실시간성 높음 • 수집 권한 필요 • 일시적 데이터(Instance State) 중심
DFIR 활용 목적	• 공격 발생 시 인스턴스 내부 행위 파악 및 이상 프로세스 확인 • 실시간 침해 범위 파악 및 격리 전 조사
예시	• AWS SSM Run Command: 프로세스 목록(ps), 네트워크 연결(netstat) • EC2 describe-instances, describe-network-interfaces: 인스턴스 구성 상태 확인 • AWS CLI get-console-screenshot: 인스턴스 화면 상태 확보

명령 기반 데이터의 대표적인 수집 방식은 AWS CLI, SSM, Prowler를 통한 수집이 있다. 각 서비스를 통한 수집 방법과 수집 시 권장 사항은 다음과 같다.

AWS CLI

AWS CLI를 통해서는 계정 및 인증 관리, 인스턴스 및 서버 상태, 네트워크 구성 및 인터페이스, 스토리지 관련, 로그 및 모니터링 설정, 애플리케이션/서버리스 구성, 키 관리/암호화, 기타 운영 구성으로 구분해 수집할 수 있다.

[표 107] AWS CLI를 활용한 명령 기반 데이터 수집 방법

구분 (주요 수집 대상)	수집 목적 및 실행 명령 예시
계정 및 인증 관련 (IAM 사용자, 역할, 정책)	• 계정 생성/삭제, 권한 변경, 의심 계정 확인 - `aws iam list-users` - `aws iam list-roles` - `aws iam list-policies` - `aws iam get-account-authorization-details`
계정 및 인증 관련 (활성화된 Access Key)	• 오래된 키 또는 무단 생성 키 확인 - `aws iam list-access-keys --user-name <user>`
계정 및 인증 관련 (MFA 설정 여부)	• MFA 미적용 계정 탐지 - `aws iam list-virtual-mfa-devices`
계정 및 인증 관련 (현재 인증된 사용자)	• 현재 세션의 IAM 엔터티(사용자/Role) 확인 - `aws sts get-caller-identity`
인스턴스 및 서버 상태 (EC2 인스턴스 목록)	• 현재 실행 중인 인스턴스, 공격자 생성 인스턴스 탐지 - `aws ec2 describe-instances`
인스턴스 및 서버 상태 (보안 그룹)	• 포트 개방 여부, 인바운드/아웃바운드 규칙 확인 - `aws ec2 describe-security-groups`
인스턴스 및 서버 상태 (네트워크 ACL, 라우팅 테이블)	• 네트워크 경로 변조 여부 분석 - `aws ec2 describe-network-acls` - `aws ec2 describe-route-tables`
인스턴스 및 서버 상태 (인스턴스 메타데이터)	• IAM Role, AMI, IP, region 정보 확보 - `curl http://169.254.169.254/latest/meta-data/`
네트워크 구성 및 인터페이스 (VPC 설정)	• VPC 구조 및 서브넷 관계 확인 - `aws ec2 describe-vpcs`
네트워크 구성 및 인터페이스 (ENI(Elastic Network Interface))	• 연결된 IP, 보안그룹, 트래픽 경로 확인 - `aws ec2 describe-network-interfaces`
네트워크 구성 및 인터페이스 (Elastic IP)	• 공격자가 사용한 외부 IP 여부 검증 - `aws ec2 describe-addresses`
스토리지 관련 (EBS 볼륨)	• 디스크 구성, 크기, 연결된 인스턴스 식별 - `aws ec2 describe-volumes`
스토리지 관련 (EBS 스냅샷)	• 무단 생성/삭제 스냅샷 확인 - `aws ec2 describe-snapshots --owner-ids self`
스토리지 관련 (S3 버킷 목록)	• 민감 데이터 버킷 존재 여부 확인 - `aws s3 ls`

구분 (주요 수집 대상)	수집 목적 및 실행 명령 예시
스토리지 관련 (S3 버킷 정책)	• 공개 접근 설정, 버킷 권한 확인 - `aws s3api get-bucket-policy --bucket <bucket>`
스토리지 관련 (S3 ACL 설정)	• 비인가자 여부 확인 - `aws s3api get-bucket-acl --bucket <bucket>`
로그 및 모니터링 설정 (CloudTrail 설정)	• 로깅 활성화 상태, 로그 대상 버킷 확인 - `aws cloudtrail describe-trails`
로그 및 모니터링 설정 (VPC Flow Logs)	• 네트워크 트래픽 로깅 여부 - `aws ec2 describe-flow-logs`
로그 및 모니터링 설정 (CloudWatch 로그 그룹)	• 로그 저장 위치 및 수집 상태 - `aws logs describe-log-groups`
로그 및 모니터링 설정 (Config 설정)	• 리소스 변경 기록 여부 확인 - `aws config describe-configuration-recorders`
애플리케이션/서버리스 구성 (Lambda 함수)	• 공격자 코드 삽입 또는 악성 Lambda 탐지 - `aws lambda list-functions`
애플리케이션/서버리스 구성 (Lambda 환경 변수)	• 공격자 코드 삽입 또는 악성 Lambda 탐지 - `aws lambda get-function-configuration --function-name <fn>`
애플리케이션/서버리스 구성 (API Gateway)	• 비정상적인 API 엔드포인트 생성 - `aws apigateway get-rest-apis`
키 관리/암호화 (KSM 키)	• 암호화 키 관리 및 접근 권한 점검 - `aws kms list-keys`
키 관리/암호화 (키 정책)	• 키 접근 정책 위조 여부 확인 - `aws kms get-key-policy --key-id <id>`
기타 운영 구성 (CloudFormation 스택)	• 공격자가 자동 배포 스택을 사용했는지 확인 - `aws cloudformation describe-stacks`
기타 운영 구성 (ECS/EKS 상태)	• 컨테이너 기반 침해 여부 탐지 - `aws ecs list-clusters` - `aws eks list-clusters`
기타 운영 구성 (Elastic Beanstalk/RDS)	• 애플리케이션·DB 환경 구성 추적 - `aws elasticbeanstalk describe-environments` - `aws rds describe-db-instances`

AWS CLI로 수집 시 권장되는 사항은 다음과 같다.

[표 108] AWS CLI를 통해 수집 시 권장되는 사항

순서	내용
1	• AWS CLI 출력은 반드시 JSON 포맷으로 저장 - 실행 명령 예시: `aws ec2 describe-instances --output json > ec2_status.json`
2	• 무결성 확보 - 실행 명령 예시: `sha256sum ec2_status.json >> evidence_hash.log`
3	• 증거 저장 버킷으로 업로드 (Read-Only 설정) - 실행 명령 예시: `aws s3 cp ec2_status.json s3://dfir-evidence-bucket/`

SSM (AWS Systems Manager)

SSM(AWS Systems Manager) 명령을 활용해 수집하려면 사전에 구성해야 하는 사항들이 있다.

[표 109] AWS SSM 사전 구성 사항

번호	내용
1	대상 인스턴스에 SSM Agent 설치 및 실행 필수
2	인스턴스에 연결된 IAM Role(Instance Profile)에 `AmazonSSMManagedInstanceCore` 권한 필요
3	S3 업로드를 할 경우 인스턴스 프로파일에 `s3:Putobject` 권한 필요
4	명령 실행자는 `ssm:SendCommand`, `ssm:GetCommandInvocation` 권한 필요

SSM 명령 실행 구조는 3가지 형태로 구분할 수 있다.

[표 110] AWS SSM 명령 실행 구조

구분	내용
실제 명령 (스크립트)	• 인스턴스에서 실행되어 결과를 생성하는 로컬 명령 - 예) Linux: ps awx, Windows: tasklist
SSM 문서 (Document)	• AWS가 제공하거나 사용자가 만든 실행 템플릿 - 예) AWS-RunshellScript (Linux Bash 명령 실행), AWS-RunPowerShellScript (Windows PowerShell 실행)
AWS CLI 호출	• 실제 명령을 인스턴스에서 실행 - 예) aws ssm send-command --document-name "AWS-RunShellScript" --parameters commands=["..."] --instance-ids ... 형태로 호출

SSM 명령을 활용한 데이터 수집은 AWS CLI에서 SSM 문서를 호출해 인스턴스 내에서 로컬 명령을 원격으로 실행하는 방식으로 수행할 수 있다.

SSM 명령 실행 예시는 다음과 같으며, 인스턴스에서 실행되는 실제 명령은 빨간색으로 별도 표기되어 있다. 운영체제 종류에 맞게 SSM 문서 유형(*--document-name*)과 실제 명령(*--parameters commands*)을 파라미터로 주면 실행이 가능하다.

[표 111] SSM 명령 실행 예시

구분	실행 명령 예시
인스턴스 메타데이터 수집 (Linux)	aws ssm send-command \ --instance-ids i-0123456789abcdef0 \ --document-name "AWS-RunShellScript" \ --parameters commands=["curl -s http://169.254.169.254/latest/meta-data/ > /tmp/metadata.txt","sha256sum /tmp/metadata.txt > /tmp/metadata.txt.sha256"] \ --output-s3-bucket-name my-evidence-bucket \ --output-s3-key-prefix "incidents/IR-2025-10-12" \ --comment "collect instance metadata"
이벤트 로그(파워셸) 수집 (Windows)	aws ssm send-command \ --instance-ids i-0123456789abcdef0 \ --document-name "AWS-RunPowerShellScript" \ --parameters commands=["Get-WinEvent -LogName Security -MaxEvents 200 ¦ Export-Clixml -Path C:\\temp\\SecurityEvents.xml","Get-FileHash C:\\temp\\SecurityEvents.xml ¦ Out-File C:\\temp\\SecurityEvents.hash"] \ --output-s3-bucket-name my-evidence-bucket \ --output-s3-key-prefix "incidents/IR-2025-10-12"

결과(파일)은 SSM이 아니라 명령 내부에 생성되며, S3에 자동으로 저장하게 설정할 수도 있다.

[표 112] SSM 명령을 통해 실행한 결과 확인 및 추출 방안

구분	설명
SSM이 stdout/stderr를 S3에 자동 저장 ('--output-s3-bucket-name' 옵션 사용)	send-command 실행 시 SSM이 생성한 stdout/stderr 파일을 지정한 S3 버킷에 저장 (권장: 증거 버킷에 저장 후 무결성 검증)
get-command-invocation으로 결과 조회	S3 자동 저장을 안 썼다면 다음으로 개별 명령의 결과를 조회 **# 명령 실행 예시** `aws ssm get-command-invocation --command-id <command-id> --instance-id i-0123456789abcdef0` * command-id는 send-command 실행 시 반환된 결과의 Command.CommandId 값

또한, '--targets "Key=tag:name,Values=webserver-*'와 같은 대상을 지정해 여러 인스턴스에 한 번에 실행도 가능하다.

[표 113] SSM 명령 실행 예시

구분	실행 명령 예시
여러 인스턴스에 한 번에 실행	aws ssm send-command \ --targets "Key=tag:Role,Values=web" \ --document-name "AWS-RunShellScript" \ --parameters commands=["ps aux > /tmp/ps.txt","sha256sum /tmp/ps.txt"] \ --output-s3-bucket-name my-evidence-bucket

SSM 명령을 통해서는 인스턴스 메타데이터, 프로세스 목록, 네트워크 연결 이력, 인증 및 접속 로그 등을 수집할 수 있다. 또한, 아래 수집 방법 형태를 응용해 사고 대응에 필요한 추가 데이터를 수집할 수 있다.)

[표 114] SSM 명령을 활용한 명령 기반 데이터 수집 방법

구분	수집 목적 및 실행 명령 예시
인스턴스 메타데이터 (Linux)	• 인스턴스 ID, AMI, IAM Role, Local/Public IP, AZ 등 인스턴스 식별용 메타데이터 확보 # Linux --parameters commands=["curl -s http://169.254.169.254/latest/meta-data/ > /tmp/metadata.txt","sha256sum /tmp/metadata.txt"] # Windows --parameters commands=["Invoke-RestMethod -Uri http://169.254.169.254/latest/meta-data/"]
프로세스 목록	• 실행 중인 프로세스 목록 수집 (악성 프로세스, 자동 실행 프로그램 식별) # Linux --parameters commands=["ps aux > /tmp/ps.txt","sha256sum /tmp/ps.txt"] # Windows --parameters commands=["tasklist > C:\\temp\\tasklist.txt", "Get-FileHash C:\\temp\\tasklist.txt"]
사용자/세션 로그인 이력	• 로그인 사용자, 원격 세션 기록, 침입 경로 추적 # Linux --parameters commands=["w; who; last -n 10 > /tmp/login.txt"] # Windows --parameters commands=["Get-EventLog -LogName Security -InstanceId 4624 -Newest 20 > C:\temp\logon.txt"]
시스템/보안 로그 (최근)	• 시스템 및 보안 관련 로그 확보 (로그온 실패, 권한 상승 등) # Linux --parameters commands=["tail -n 500 /var/log/auth.log > /tmp/auth_tail.log"] # Windows --parameters commands=["Get-WinEvent -LogName Security -MaxEvents 100 > C:\temp\Security.evtx"]

구분	수집 목적 및 실행 명령 예시

디스크 마운트 정보	• 디스크 사용 현황, 마운트 상태, 외부 볼륨 연결 확인
	# Linux
	`--parameters commands=["df -h > /tmp/df.txt; lsblk > /tmp/lsblk.txt"]`
	# Windows
	`--parameters commands=["Get-Volume > C:\temp\volumes.txt”]`

컨테이너 상태 (ECS/Docker)	• 컨테이너 기반 공격 여부 식별 # Linux `--parameters commands=["docker ps -a > /tmp/docker_ps.txt; docker images > /tmp/docker_images.txt"]`

명령어 이력
(최근 활동)
	• 최근 실행 명령을 통한 공격자 활동 흔적 확인
	# Linux
	`--parameters commands=["~/.bash_history > /tmp/history.txt"]`
	# Windows
	`--parameters commands=["Get-Content (Get-PSReadlineOption).HistorySavePath > C:\temp\history.txt”]`

작업 스케줄 목록	• 자동 실행 및 지속성 유지 스케줄 탐지
	# Linux
	`--parameters commands=["crontab -l > /tmp/cron.txt; ls /etc/cron* > /tmp/cron_dir.txt"]`
	#Windows
	`--parameters commands=["schtasks /query /fo LIST /v > C:\temp\schtasks.txt”]`

서비스,
드라이브 목록
	• 악성 서비스 및 드라이브 존재 여부 탐지
	# Linux
	`--parameters commands=["systemctl list-units --type=service > /tmp/services.txt"]`
	#Windows
	`--parameters commands=["Get-Service Select Name,Status,DisplayName > C:\temp\services.txt”]`

네트워크 설정 정보	• 네트워크 인터페이스, 라우팅 테이블, DNS 구성
	# Linux
	`--parameters commands=["systemctl list-units --type=service > /tmp/services.txt"]`
	#Windows
	`--parameters commands=["Get-Service Select Name,Status,DisplayName > C:\temp\services.txt”]`

네트워크 연결 정보	• 외부 연결, 리스닝 포트, C2 연결 확인
	# Linux
	`--parameters commands=["ss -tunap > /tmp/netstat.txt”]`
	# Windows
	`--parameters commands=["netstat -ano > C:\temp\netstat.txt”]`

결과 압축 및 업로드	• 수집 데이터 일괄 압축 및 S3 증거 저장
	# Linux
	`--parameters commands=["tar zcvf /tmp/evidence.tar.gz /tmp/*.txt; aws s3 cp /tmp/evidence.tar.gz s3://my-evidence-bucket/"]`
	#Windows
	`--parameters commands=["Compress-Archive -Path C:\temp\* -DestinationPath C:\temp\evidence.zip; aws s3 cp C:\temp\evidence.zip s3://my-evidence-bucket/”]`

SSM 명령으로 수집 시 주의 사항은 다음과 같다.

[표 115] SSM 명령을 통해 수집 시 권장되는 사항

번호	내용
1	문서 타입(`--document-name`) 주의 (Linux → "AWS-RunShellScript", Windows → "AWS-RunPowerShellScript")
2	S3 자동 저장 옵션 활용 (`--output-s3-bucket-name my-evidence-bucket --output-s3-key-prefix "incident/IR-2025-10-12"`)
3	세션 로깅 활성화 (SSM Manager 사용 시, CloudWatch/S3에 세션 로그 로깅)
4	파라미터 형태의 커맨드 문자열 배열 작성 주의 `--parameters commands=["cmd1","cmd2"]`에서 커맨드는 문자열 배열로 작성 (큰따옴표/셸 이스케이프에 주의해야하며, Windows Powershell 명령은 PowerShell 문법으로 작성)

Prowler

Prowler는 SSM과 같이 인스턴스 내부에서 셸 명령을 실행하지는 못하고, AWS 계정 전반의 구성·설정 상태(IAM 정책, MFA 적용, CloudTrail 설정, S3 공개 여부 등)를 API 호출로 빠르게 스캔해 보안 상태 스냅샷 생성하는 데 강점이 있다.

Prowler는 주로 설정 기준을 검사하기 때문에 "누가 지금 세션에 로그인해 무엇을 실행 중인지"와 같은 런타임 증거를 제공하지는 않는다. 또한, Prowler를 통한 명령 기반 데이터는 AWS API 응답 결과를 수집하는 형태이기 때문에 SSM으로 추가 수집해 데이터 보완이 필요하다.

Prowler는 AWS의 구성 및 설정의 현재 상태를 빠르게 수집할 수 있기 때문에 DFIR 초동 단계에서 매우 유용하게 활용할 수 있다. 수집 가능한 명령 기반의 데이터 예시 항목은 다음과 같다.

[표 116] Prowler로 수집 가능한 명령 기반의 데이터 항목

구분	내용
IAM	사용자/역할/정책(광범위 권한, 루트 사용, 오래된 액세스키 등)
CloudTrail	Trail 활성화 여부, 로그 파일 배치 위치, 관리 이벤트 기록 설정
S3	버킷 공개 여부, 버전관리/암호화 설정, ACL 정책 등
VPC 및 네트워크	Flow 로그 활성화 여부, 퍼블릭 서브넷 구성 등
KMS	키 정책 및 외부 접근 가능성
Config	AWS Config 활성화 상태(리소스 변경 캡처 여부)
기타	EBS 암호화, RDS 보안 설정, Lambda 권한 과다 등

Prowler의 각 항목은 내부적으로 체크 ID로 정의되어 있어, 명령 실행 시 check ID를 명시하는 것을 권장하며, check ID와 해당 체크 동작은 Prowler 버전에 따라 달라질 수 있다.

[표 117] Prowler 버전 및 체크 목록 확인 방법

구분	명령
현재 Prowler 버전 확인	`prowler -v` 또는 `prowler --version`
사용 가능한 체크 목록 확인	`prowler aws --list-checks`

Prowler로 명령 기반 데이터를 수집할 때 권장되는 워크플로우와 구성은 다음과 같다.

[표 118] Prowler 실행 시 권장 워크플로우

순서	구분	내용
1	권한 준비	조사용 역할(또는 프로파일)에 읽기 전용(최소한의) 권한 부여 다계정 조사면 AssumeRole 설정
2	Prowler 실행 (스냅샷 생성)	대상 계정/리전으로 Prowler 실행 → JSON/CSV 출력 생성
3	결과 데이터 저장	출력 파일(예: prowler-results.json)을 증거 S3 버킷에 업로드하고 해시값 생성 및 보관
4	확인 및 필터링	Prowler 결과에서 'High/Fail' 항목을 확인해 우선 대응 항목 선정 (예: 공개 S3, CloudTrail 비활성 등)
5	SSM 연계 수집	Prowler가 찾은 의심 지점을 기반으로 SSM 명령을 통해 해당 인스턴스의 실시간 상태를 수집
6	보고 및 타임라인 복원	Prowler와 SSM 결과를 연계 분석해 원인 규명 및 침해지표 식별

[표 119] Prowler 실행 시 권장 구성

번호	내용
1	Prowler 실행 시 `-M json` 또는 `-M csv` 형식으로 출력
2	결과 파일에 대해 SHA256 해시 생성 및 별도 안전저장
3	Prowler 실행 로그(누가 언제 실행했는지)를 CloudTrail/CI 로그에 로깅
4	Prowler로 탐지된 High/Fail 항목을 자동 태스크로 변환해 SSM 조치 템플릿 호출

Prowler를 통해서는 전체 계정 보안 상태, IAM 보안 구성 점검, CloudTrail 활성화 상태, S3 버킷 접근 통제 점검 등이 가능하다.

[표 120] Prowler를 활용한 명령 기반 데이터 수집 방법 (v5.15.0)

구분	수집 목적 및 실행 명령 예시
전체 계정 보안 상태 스냅샷	• AWS 계정 전반의 구성 및 보안 설정 상태 점검 `./prowler aws --compliance cis_1.5_aws -M csv`

IAM 계정 및
권한 구성 점검
	• 루트 계정 사용, MFA 미적용, 약한 비밀번호 정책 등 계정 접근 취약점 식별
	`./prowler aws --checks iam_root_mfa_enabled iam_avoid_root_usage \`
	`iam_user_mfa_enabled_console_access \`
	`iam_password_policy_minimum_length_14 \`
	`iam_password_policy_symbol \`
	`iam_password_policy_lowercase \`
	`iam_password_policy_uppercase \`
	`iam_password_policy_number \`
	`iam_password_policy_expires_passwords_within_90_days_or_less -M csv`

Access Key 및
자격 증명 관리
	• 장기 사용·미사용 키 식별 및 잠재적 키 유출 위험 파악
	`./prowler aws --checks \`
	`iam_rotate_access_key_90_days \`
	`iam_user_accesskey_unused \`
	`iam_user_two_active_access_key iam_no_root_access_key -M csv`

구분	수집 목적 및 실행 명령 예시

CloudTrail 로그
수집 구성 점검
	• 로그 수집 상태 및 무결성 검증 활성화 여부 확인
	(CloudTrail 활성 여부, 로그 무결성 검증, CloudTrail 로그의 Cloudwatch 연동 여부)
	`./prowler aws --checks \`
	`cloudtrail_multi_region_enabled \`
	`cloudtrail_log_file_validation_enabled \`
	`cloudtrail_cloudwatch_logging_enabled -M csv`

CloudWatch
탐지 구성 점검
	• 보안 이벤트 탐지 및 알람 설정 활성화 여부 확인
	`./prowler aws –checks \`
	`cloudwatch_log_group_not_publicly_accessible \`
	`cloudwatch_log_group_kms_encryption_enabled \`
	`cloudwatch_alarm_actions_enabled -M csv`

S3 버킷
접근 통제 점검
	• 데이터 노출 여부 및 로깅 활성화 상태 점검
	`./prowler aws --checks s3_bucket_public_access s3_bucket_object_versioning \`
	`s3_bucket_default_encryption -M csv`
	(S3 공개 여부 확인, S3 버전 관리 활성 여부 확인, S3 기본 암호화 여부 확인)

GuardDuty 설정 확인	• GuardDuty 서비스 활성화 여부 확인 `./prowler aws --checks guardduty_s3_protection_enabled -M csv`

VPC Flow Logs,
Security Group 점검
	• Flow Log 활성화, 보안그룹 과도한 인바운드 규칙 확인
	`./prowler aws --checks \`
	`vpc_flow_logs_enabled \`
	`ec2_securitygroup_allow_ingress_from_internet_to_any_port \`
	`ec2_securitygroup_allow_ingress_from_internet_to_all_ports -M csv`
	(VPC Flow Log 활성화 여부, 과도한 인바운드 포트(0.0.0.0/0) 노출 여부, 특정 포트 노출 여부)

EC2 및 EBS
접근 구성 점검
	• 인스턴스 암호화, 메타데이터 보호, 외부 노출 확인
	`./prowler aws --checks \`
	`ec2_instance_imdsv2_enabled ec2_launch_template_no_public_ip -M csv`
	(IMDSv2 적용 여부, 퍼블릭 IP 노출 여부)

KMS 키 관리 점검	• KMS 사용 여부 확인 `./ prowler aws --checks kms_cmk_are_used -M csv`
RDS 보안 구성 확인	• DB 접근제어, 저장 암호화, 백업 기능 점검 `./ prowler aws --checks rds_instance_no_public_access rds_instance_storage_encrypted -M csv` (RDS 퍼블릭 접근 설정 여부, RDS 저장 암호화 설정 여부)

Lambda 구성 점검	• 컨테이너 실행 환경의 격리·암호화 여부 확인
	`./prowler aws --checks \`
	`awslambda_function_inside_vpc awslambda_function_not_publicly_accessible -M csv`
	(Lambda VPC 내 배치, Lambda 퍼블릭 접근 차단 여부)

CloudFront, Route53 구성 점검	• HTTPS 전용 설정 확인 `./prowler aws --checks cloudfront_distributions_custom_ssl_certificate -M csv`

결과 저장 및 보존	• 결과 JSON 파일을 증거 버킷(S3)에 저장하고 해시 검증 수행 (무결성 보존)
	`./prowler -M json -r ap-northeast-2 -g all; aws s3 cp prowler-output.json \`
	`s3://dfir-evidence-bucket/IR-2025/prowler-output.json`

2) 로그 기반 데이터 수집 (Log-based Data)

로그 기반 데이터는 AWS가 자동으로 생성 또는 저장한 이벤트 데이터를 의미한다. API 호출, 트래픽 흐름, 접근 이력, 구성 변경 등을 로깅해 공격 행위를 재현하거나 상관 분석할 때 활용된다.

[표 121] AWS 환경에서의 로그 기반 데이터

구분	설명
정의	• AWS 서비스가 자동으로 생성·보존하는 행위(Behavior) 기반 기록 데이터 • 계정 활동, 네트워크 트래픽, 접근 이력, 구성 변경 등 장기적 분석용
주요 수집 방식	• CloudTrail, Config, VPC Flow Logs, S3 Access Log, WAF Log 등 자동 로그 수집 • Security Lake, S3를 통한 중앙화 • 관리 콘솔 또는 명령 기반으로도 수집이 가능
데이터 특성	• 보존 가능 - 비휘발성(Non-Volatile) • 자동성과 지속성 높음 • 서비스별 포맷 상이 • 장기적 타임라인 분석에 적합 • 일부 로그의 경우, 관리자가 활성화해야 로깅 됨
DFIR 활용 목적	• 공격 경로 및 공격 행위 타임라인 재구성 • 침입 흔적 상관분석 및 대응 자동화 • 사후 감사·컴플라이언스 검증
예시	• AWS CloudTrail: API 호출 이력 • AWS Config: 리소스 변경 추적 • VPC Flow Logs: 네트워크 흐름 • S3 Access Log: 객체 접근 로그 • GuardDuty Findings: 위협 탐지 이벤트

로그 기반 데이터의 대표적인 수집 방식은 각 콘솔 또는 서비스나 버킷을 통해 수집할 수 있다. 다만, 일부 로그의 경우 사전에 활성화가 필요하기 때문에 활성화가 되어있는지 확인할 필요가 있다.

각 로그의 수집 방법과 수집 시 권장 사항은 다음과 같다.

[표 122] 로그 기반 데이터 수집 방법

로그 구분	수집 방법
CloudTrail	• **활성화 여부 확인** - 경로: AWS 관리 콘솔 → CloudTrail Console → Trails - S3 버킷과 CloudWatch Logs에 저장 가능 - Event history는 자동 활성화되나, 기본적으로 90일에 해당하는 로그만 보관 (장기 보존하려면 Trail을 명시적으로 생성해야 함) • **Trail이 존재하지 않는 경우** - 경로: AWS 관리 콘솔 → CloudTrail Console → Event history • **S3 버킷에 저장** - 날짜별 디렉터리 구조에 JSON 형태의 로그 파일 압축해 아래와 같은 경로에 저장 (s3://<bucket-name>/AWSLogs/<account-id>/CloudTrail/<region>/<YYYY>/<MM>/<DD>/<filename>.json.gz) • **CloudWatch Logs에 저장** - 경로: CloudWatch Console → Logs → Log groups에서 해당 그룹 찾기 - Actions → Export data to Amazon S3를 통해 로그를 S3 버킷으로 내보낼 수 있음 - S3로 CloudWatch 로그를 Export 후 다운로드
VPC Flow Logs	• **활성화 여부 확인** - 경로: AWS 관리 콘솔 → VPC Console → VPC 선택 → Flow Log - 사전 활성화가 필요하며, S3 버킷과 CloudWatch Logs에 저장 가능 • **S3 버킷에 저장** - 날짜별 디렉터리 구조에 JSON 형태의 로그 파일 압축해 아래와 같은 경로에 저장 (s3://<bucket-name>/AWSLogs/<account-id>/CloudTrail/<region>/<YYYY>/<MM>/<DD>/<filename>.json.gz) • **CloudWatch Logs에 저장** - 경로: CloudWatch Console → Logs → Log groups에서 해당 그룹 찾기 - Actions → Export data to Amazon S3를 통해 로그를 S3 버킷으로 내보낼 수 있음 - S3로 VPC Flow Logs를 Export 후 다운로드
S3 Server Access Log	• **활성화 여부 확인** - 경로: AWS 관리 콘솔 → S3 Console → Bucket 선택 → 속성(Properties) - 사전 활성화가 필요하며, S3 버킷에 저장 가능 - Server access logging 항목의 상태(Enabled, Disabled) 확인 • **S3 버킷에 저장** - 공백으로 구분된 평문 텍스트 형태로 아래와 같은 경로에 저장 (s3://<target-bucket>/<target-prefix>/<source-bucket-name>/YYYY-MM-DD-HH-MM-SS-<UniqueString>)
CloudWatch Logs	• **활성화 여부 확인** - 경로: AWS 관리 콘솔 → CloudWatch Console → Logs → Log groups - 사전 활성화가 필요하며, S3 버킷에 저장 가능 • **S3 버킷에 저장** - Actions → Export data to Amazon S3를 통해 로그를 S3 버킷으로 내보낼 수 있음

로그 구분	수집 방법
RDS Logs	• **활성화 여부 확인** - 경로: Amazon RDS Console → Databases → DB 인스턴스 선택 → Logs & events - 사전 활성화가 필요하며, RDS 콘솔, S3 버킷, CloudWatch Logs에서 수집 가능 - 비활성 상태면 CloudWatch 그룹이나 로그 파일 자체가 존재하지 않음 • **RDS 콘솔에서 직접 수집** - 경로: Amazon RDS Console → Databases → DB 인스턴스 선택 → Logs & events → 로그 파일 선택 → 다운로드 • **S3 버킷에 저장** - 아래와 같은 경로에 저장 (s3://<bucket-name>/AWSLogs/<account-id>/rds/<db-instance>/...) • **CloudWatch Logs에 저장** - 경로: CloudWatch Console → Logs → Log groups에서 해당 그룹 찾기 - Actions → Export data to Amazon S3를 통해 로그를 S3 버킷으로 내보낼 수 있음 - S3로 RDS Logs를 Export 후 다운로드
GuardDuty Findings	• **활성화 여부 확인** - 경로: AWS 관리 콘솔 → AWS GuardDuty Console → Detectors - 사전 활성화가 필요하며, GuardDuty 콘솔과 S3 버킷에서 수집 가능 - Detector를 만들지 않으면 Findings 자체가 존재하지 않음 • **GuardDuty 콘솔에서 직접 수집** - 경로: AWS 관리 콘솔 → AWS GuardDuty Console → Findings (탐지 이벤트 확인 후 다운로드) • **S3 버킷에 저장** - 경로: AWS GuardDuty Console → Findings → Export → S3 설정 - 지정한 버킷에 JSON으로 아래와 같은 경로에 저장 (s3://my-guardduty-findings/AWSLogs/<account-id>/GuardDuty/<region>/YYYY/MM/DD/findings.json)
WAF Log	• **활성화 여부 확인** - 경로: AWS WAF → Web ACLs → Web ACL 선택 → Enable logging • **S3 버킷에 저장** - 날짜별 디렉터리 구조에 로그 파일이 압축(Gzip)되어 저장 (s3://<bucket-name>/<prefix>/AWSLogs/<account-id>/AWSWAFLogs/<web-acl-name>/region/YYYY/MM/DD/HH/<file-name>.gz) • **CloudWatch Logs에 저장** - 경로: CloudWatch Console → Logs → Log groups에서 해당 그룹 찾기 - Actions → Export data to Amazon S3를 통해 로그를 S3 버킷으로 내보낼 수 있음 - S3로 WAF Log를 Export 후 다운로드

EC2 포렌식 데이터 수집 절차

[표 123] EC2 포렌식 데이터 수집 절차 목록

순서	구분	내용

1	침해 의심 인스턴스 격리	침해가 의심되는 EC2 인스턴스는 추가 피해 확산 방지를 위해 즉시 네트워크 격리 수행
		
		격리를 통해 공격자의 세션을 차단하고, 포렌식 대상 인스턴스의 상태를 안정적으로 유지 (격리는 인스턴스의 삭제나 종료 없이 보안 그룹 재구성)
		• **보안 그룹 차단 정책을 아래와 같이 적용**
		(Ingress: 분석가 IP 1개만 SSH 또는 RDP 허용,
		Egress: 모든 트래픽 차단 (기본 `allow all` 제거)
		• **AWS CLI 명령 예시**
		`aws ec2 create-security-group --group-name Quarantine-SG --description "Forensic quarantine"`
		`aws ec2 authorize-security-group-ingress --group-name Quarantine-SG --protocol tcp --port 22 --cidr <분석가 IP>/32`
		`aws ec2 modify-instance-attribute --instance-id i-xxxx --groups <Quarantine-SG-ID>`

2	인스턴스 메타데이터 수집	격리된 이스턴스에 대해 기본 정보와 환경 구성 요소를 수집 • **수집 항목** - 인스턴스 ID, 타입, AMI ID, Private/Public IP, VPC/Subnet/Security Group, Attached EBS Volume ID, 시작 시간 및 리전 정보 • **AWS CLI 명령** `aws ec2 describe-instances --instance-ids i-xxxx \ --query 'Reservations[].Instances[].{InstanceId:InstanceId, ImageId:ImageId,PrivateIP:PrivateIpAddress,PublicIP:PublicIpAddress, SecurityGroups:SecurityGroups[*].GroupId, VpcId:VpcId,SubnetId:SubnetId,LaunchTime:LaunchTime}'`

3	인스턴스 보호 설정	증거 훼손 방지를 위한 보호 설정 적용
		• **AWS CLI 명령 - 종료 방지(Disable API Termination)**
		`aws ec2 modify-instance-attribute --instance-id i-xxxx --disable-api-termination`
		• **AWS CLI 명령 - EBS 볼륨 삭제 방지**
		(DeleteOnTermination 비활성화)
		`aws ec2 modify-instance-attribute --instance-id i-xxxx \ --block-device-mappings \ "[{\"DeviceName\":\"/dev/sda1\",\"Ebs\":{\"DeleteOnTermination\":false}}]"`

순서	구분	내용

4	EBS 스냅샷 생성	디스크 상태를 보존하기 위해 EBS 스냅샷 생성
		(스냅샷은 온프레미스 환경의 디스크 이미징에 해당하며, 데이터 손실 없이 증거 확보 가능)
		• **생성 절차**
		1. 인스턴스 중지 (볼륨 상태 안정화)
		2. 연결된 볼륨 ID 확인
		- 손상된 인스턴스 선택 후 Storage 탭으로 이동해 볼륨 ID 확인 또는 아래 AWS CLI 명령 실행
		`aws ec2 describe-instances --instance-ids i-xxxx \`
		`--query 'Reservations[].Instances[].BlockDeviceMappings[].Ebs.VolumeId'`
		3. 스냅샷 생성
		`aws ec2 create-snapshot --volume-id vol-xxxx \`
		`--description "Forensic snapshot - i-xxxx" \`
		`--tag-specifications 'ResourceType=snapshot,`
		`Tags=[{Key=Forensic,Value=True}]'`

5	포렌식 워크스테이션 준비	증거를 분석할 전용 워크스테이션(EC2)을 별도로 포렌식 환경에 구축
		(조사 전용 VPC, 별도 계정, Golden AMI 기반으로 구성 필요)
		• **설정 권장 사항**
		- AMI: 사전 구축된 Forensics Golden AMI 사용
		- 네트워크: 외부 인터넷 차단, S3 Evidence Bucket 접근 허용
		- 보안 그룹: 조사자 IP만 SSH/RDP 허용
		- IAM Role: 읽기 전용 S3 접근 및 Snapshot 복제 권한만 부여
		• **AWS CLI 명령 예시**
		`aws ec2 run-instances --image-id ami-xxxx \`
		`--instance-type m5.large \`
		`--subnet-id subnet-forensic \`
		`--security-group-ids sg-forensic \`
		`--iam-instance-profile Name=ForensicRole \`
		`--tag-specifications`
		`'ResourceType=instance,Tags=[{Key=Purpose,Value=Forensic}]'`

순서	구분	내용

6	증거 볼륨 생성 및 연결	스냅샷으로부터 새 EBS 볼륨을 생성하고, 포렌식 워크스테이션에 연결
		• **생성 절차 (AWS 명령 예시 포함)**
		1. 침해 인스턴스 EBS 스냅샷을 기반으로 새로운 EBS 볼륨 생성
		`aws ec2 create-volume \`
		`--availability-zone ap-northeast-2a \`
		`--snapshot-id snap-0abcd1234efgh5678 \`
		`--tag-specifications 'ResourceType=volume,Tags=[{Key=Source,Value=ForensicSnapshot}]'`
		2. 증거 분석용 포렌식 워크스테이션에 볼륨 연결
		`aws ec2 attach-volume \`
		`--volume-id vol-0abc1234def5678gh \`
		`--instance-id i-0123456789abcdef0 \`
		`--device /dev/sdfaws ec2 attach-volume \`
		3-a. 볼륨 인식 및 읽기 전용 마운트 (Linux) – 명령 예시 포함
		- 인스턴스 접속 (SSH)
		`ssh -i key.pem ec2-user@<Forensic-EC2-IP>`
		- 디바이스 인식 확인
		`lsblk`
		- 파일시스템 타입 식별
		`sudo file -s /dev/xvdf`
		- 읽기 전용(Read-Only) 마운트
		`sudo mkdir /mnt/evidence`
		`sudo mount -o ro /dev/xvdf1 /mnt/evidence`
		- 마운트 확인
		`df -h ¦ grep evidence`
		3-b. 볼륨 인식 및 읽기 전용 설정 (Windows) – 명령 예시 포함
		- RDP로 포렌식 워크스테이션 접속
		- 디스크 관리자 실행 (`diskmgmt.msc`)
		- 새 디스크 인식 확인 (일반적으로 Offline 상태로 표시)
		- 디스크를 Online으로 변경하되, 쓰기 방지를 위해 "Read-Only" 속성 부여 (아래 명령은 파워셸 명령 예시)
		`Get-Disk ¦ Where-Object IsOffline -Eq $true ¦`
		`Set-Disk -IsOffline $false`
		`Set-Disk -Number <DiskNumber> -IsReadOnly $true`
		- 디스크 드라이브가 인식되면, 포렌식 도구로 접근 가능
		(디스크 관리자에서 "Initialize Disk"를 절대 누르지 않아야 함)

7	포렌식 분석 수행	마운트된 볼륨을 대상으로 포렌식 툴을 활용해 분석 수행 (분석은 원본 볼륨이 아닌 복제본에서 수행)
8	후속 조치	증거 확보 완료 시, 인스턴스를 종료하고 필요 시에는 AMI로 보존

EKS 포렌식 데이터 수집 절차

[표 124] EKS 포렌식 데이터 수집 절차 목록

순서	구분	내용

1	침해 의심 Pod, 노드 격리	EKS 환경에서는 Pod, Deployment 또는 노드 단위로 침해가 발생할 수 있다. EC2처럼 인스턴스를 바로 차단하지 않고 Kubernetes 제어 명령으로 논리적 격리를 수행한다.
		• kubectl 명령 - Pod 격리
		- 침해 의심 Pod 라벨 부여
		`kubectl label pods -n <namespace> <pod-name> status=quarantine`
		- Pod 로그 접근 차단을 위해 네트워크 정책 적용
		`kubectl apply -f quarantine-networkpolicy.yaml`
		• kubectl 명령 - Node 격리 (Cordon + Drain)
		- 노드에 신규 Pod 스케줄링 금지
		`kubectl cordon <node-name>`
		- (선택) 실행 중인 Pod를 다른 노드로 이동
		`kubectl drain <node-name> --ignore-daemonsets --delete-emptydir-data`
		• AWS CLI 명령 - EC2 노드 차단 (필요 시)
		- 해당 노드의 EC2 인스턴스 ID 식별 후, AWS CLI로 보안 그룹을 격리 전용 그룹으로 변경
		`aws ec2 modify-instance-attribute --instance-id i-xxxx --groups sg-quarantine`

순서	구분	내용

2	클러스터 및
	노드 메타데이터 수집
		Kubernetes 및 EKS 계층의 구조적 정보 확보
		(추후 어떤 워크로드가 어디서 실행됐는지 분석하는 핵심 데이터가 됨)
		• AWS CLI 명령 - 클러스터 메타데이터
		`aws eks describe-cluster --name <cluster-name> --region ap-northeast-2`
		• kubectl 명령 - 노드 및 인스턴스 매핑
		`kubectl get nodes <node-name> -n <namespace> --show-labels \ -o custom-columns=NAME:.metadata.name,INSTANCEID:.spec.providerID`
		- EC2 인스턴스 ID만 추출
		• `kubectl get nodes <node-name> -n <namespace> --show-labels \ -o custom-columns=NAME:.metadata.name,INSTANCEID:.spec.providerID ¦ sed -e 's/aws:.*\///g'`
		- Node 이름과 EC2 인스턴스 ID를 매핑해 EC2 포렌식 절차와 연계 가능
		• kubectl 명령 - Pod / Deployment / Service 매핑
		- 특정 Deployment와 연결된 Pod 식별
		`kubectl get pods -l app=<deployment-name>`
		- 특정 컨테이너 이미지로 실행 중인 Pod 식별
		`kubectl get pods --all-namespaces -o json ¦ jq -r --arg image "<image_name>" \`
		`'.items[] ¦ select(.spec.containers[] ¦ .image == $image) ¦ "\(.metadata.namespace) \(.metadata.name)"'`
		- 특정 서비스 계정으로 실행 중인 Pod 식별
		`kubectl get pods -A -o json ¦ jq -r \`
		`'.items[] ¦ select(.spec.serviceAccount == "<service_account>") ¦ "\(.metadata.namespace) \(.metadata.name)"'`
		- 서비스 IP 식별
		`kubectl get service [--all-namespaces, -n <namespace>]`
		- 특정 네임스페이스 내 Pod와 Cluster IP, Worker Node 확인
		`kubectl get pods -n <namespace> --show-labels -o wide`
		`kubectl get pods -n <namespace> --show-labels -o json`
		- 특정 Pod 상세 정보 확인
		`kubectl get pods <pod-name> -n <namespace> --show-labels -o wide`
		`kubectl get pods <pod-name> -n <namespace> -o=jsonpath='{.spec.nodeName}{"\n"}'`
		• kubectl 명령 – 영향받은 리소스 라벨링
		`kubectl label pod -n <namespace> <pod-name> status=compromised`
		`kubectl label node <node-name> status=quarantine`

3	워커 노드 종료 방지 설정	EC2 콘솔에서 해당 인스턴스 → Termination Protection 활성화

순서	구분	내용

4	메모리 및 실행 상태 수집	Pod나 Node의 실행 중인 프로세스 및 컨테이너 상태를 보존
		(컨테이너 단위 명령으로 휘발성 데이터를 확보 가능)
		• **docker 명령 - 컨테이너 상태 보존**
		- 컨테이너 프로세스 확인
		`docker top <container_id>`
		- 컨테이너 로그 수집
		`docker logs <container_id> > /tmp/<container_id>_logs.txt`
		- 컨테이너 설정 및 환경 정보 수집
		`docker inspect <container_id> > /tmp/<container_id>_inspect.json`
		• **로컬 명령 – 노드 메모리 덤프(AVML)**
		- SSM을 활용하거나 수동으로 노드에 접근해 덤프 수행
		(Amazon EKS Addon 중 SSM Agent Addon을 배포하면 원격 명령을 안전하게 수행할 수 있음)
		- AVML 다운로드 및 실행
		`sudo curl \`
		`-LO https://github.com/microsoft/avml/releases/download/v0.3.0/avml`
		`sudo chmod +x avml`
		`sudo ./avml /mnt/forensic/memory.dmp`

5	EBS 스냅샷 생성	EKS Worker Node는 EC2 인스턴스이므로, EC2 포렌식과 동일하게 EBS 스냅샷 기반으로 디스크 증거 확보
		• **AWS CLI 명령 – 노드(EC2) EBS 스냅샷 생성**
		- EC2 인스턴스 ID 확보
		`NODE_INSTANCE=$(kubectl get node <node-name> \`
		`-o jsonpath='{.spec.providerID}' | sed 's|.*instance/||')`
		- 연결된 EBS 볼륨 ID 추출
		`aws ec2 describe-instances --instance-ids $NODE_INSTANCE \`
		`--query 'Reservations[].Instances[].BlockDeviceMappings[].Ebs.VolumeId' --output text`
		- 스냅샷 생성
		`aws ec2 create-snapshot --volume-id vol-xxxx --description "EKS node forensic snapshot"`

6	Kubernetes Audit 및 Pod 로그 수집	Kubernetes API 호출, Pod 생성/삭제, Role 변경 등은 Audit 로그에서 확인 가능 (감사 로그가 CloudWatch Logs에 활성화되어 있어야 하며, CLI로 직접 수집 가능)
		• **AWS CLI 명령 – Kubernetes Audit 로그 수집**
		`aws logs filter-log-events \`
		`--log-group-name "/aws/eks/<ClusterName>/cluster" \`
		`--start-time "$START_MS" --end-time "$END_MS" \`
		`--output json > eks_audit_logs.json`
		• **kubetcl 명령 – Pod 및 컨테이너 로그 수집**
		- 일반 워크로드
		`kubectl logs -n <namespace> <pod-name> --all-containers > pod_logs.txt`
		- 시스템 네임스페이스(kube-system)
		`kubectl logs -n kube-system <pod-name> > kube_system_logs.txt`

순서	구분	내용

7	증거 보존 및 보안 태깅	수집된 아티팩트(메모리, 스냅샷, 로그)는 즉시 S3 Evidence 버킷에 업로드하고, 무결성을 검증해 조사 기록에 포함
		• **AWS CLI 명령 – 해시 생성 및 S3 업로드**
		- 해시 생성 (로컬)
		`sha256sum memory.dmp > memory.hash`
		- S3 업로드 예시
		`aws s3 cp memory.dmp s3://forensic-evidence-bucket/EKS/memory.dmp \`
		`--sse aws:kms`
		`aws s3 cp eks_audit_logs.json s3://forensic-evidence-bucket/EKS/audit/`

5. 사고 분석 기법

클라우드 환경에서의 DFIR 분석 단계의 핵심 목적은 수집된 데이터를 기반으로 공격 행위를 규명하고 그 과정을 타임라인 형태로 재구성하는 것에 있다. 정확한 로그 해석과 서비스별 행위 상관분석을 통해 공격자의 침입 경로, 권한 상승, 데이터 조작 및 유출 여부 등을 규명할 수 있으며, 이는 보안 강화 방안 마련과 재발 방지 조치의 근거로 활용된다.

본 연구에서는 AWS 환경에서의 사고 분석을 체계화하기 위해 로그 및 서비스별 주요 분석 필드와 공격에서 빈번하게 확인되는 이벤트를 정의하고, 공격 전술별로 주요 로그 이벤트를 매핑해 DFIR CheatSheet를 개발했다.

또한, 분석가가 AWS 환경에서 사고 분석을 수행할 때 효율성을 향상시키기 위해 CloudTrail, VPC Flow Logs, S3 Access Log에서 공격 징후 기반 로그 이벤트 분석 및 전술 기반 이벤트 가시화가 가능하도록 분석 도구를 구현했다. 해당 도구는 실제 공격에서 자주 확인되는 로그 패턴을 탐지해 분석가가 우선적으로 집중해야 할 분석 포인트를 제시한다.

5장에서 다루는 내용은 다음과 같다.

[표 125] 주요 연구 내용 – 사고 분석 기법

번호	소제목	주요 내용
1	로그별 주요 분석 필드 및 이벤트 분석	CloudTrail, VPC Flow Logs, S3 Access Log, CloudWatch Logs 등 AWS 주요 로그 유형별로 DFIR 관점에서 핵심적으로 활용되는 필드와 이벤트 분석
2	공격 전술별 로그 이벤트 매핑 DFIR CheatSheet 개발	MITRE ATT&CK 전술 기반으로 AWS 로그를 매핑 및 분류해 전술별 핵심 이벤트/오퍼레이션과 DFIR 분석 기준을 정리한 CheatSheet 제시
3	AWS DFIR 로그 분석 도구 개발	CloudTrail, VPC Flow, S3 로그를 자동 분석하는 AWS DFIR 도구(bitParser)를 개발해 전술 기반 로그 분석과 핵심 탐지 포인트 제공

5.1. 로그별 주요 분석 필드 및 이벤트 분석

CloudTrail, VPC Flow Logs, S3 Access Log, CloudWatch Logs 등 AWS 주요 로그 유형별로 DFIR 관점에서 핵심적으로 활용되는 필드와 이벤트를 분석했다.

1) CloudTrail

CloudTrail 로그는 JSON 형식으로 로깅되며, 각 이벤트는 여러 개의 '키-값' 쌍으로 구성된 필드를 포함한다. 사고 분석에 활용되는 주요 필드는 다음과 같다.

[표 126] CloudTrail Log 예시

로그 예시 일부
{"Records": [{ "eventVersion": "1.08", "userIdentity": { "type": "IAMUser", "principalId": "AIDA6ON6E4XEGITEXAMPLE", "arn": "arn:aws:iam::888888888888:user/Mary", "accountId": "888888888888", "accessKeyId": "AKIAIOSFODNN7EXAMPLE", "userName": "Mary", "sessionContext": { "sessionIssuer": {}, "webIdFederationData": {}, "attributes": { "creationDate": "2023-07-19T21:11:57Z", "mfaAuthenticated": "false" } } }, - 이하 생략 -

[표 127] 사고 분석 시 주로 활용되는 CloudTrail 로그 필드

필드 구분	설명
eventVersion	CloudTrail 이벤트 구조의 버전
userIdentity	요청을 수행한 주체에 대한 정보 - 대상 IAM 사용자, 역할(Role), AWS 서비스 등을 식별 가능 - type(Root, IAMUser, AssumedRole 등)과 arn(Amazon Resource Name)을 통해 행위자 명확하게 특정 가능
eventTime	이벤트 발생 일시 (UTC 기준)
eventName	수행된 이벤트 명
awsRegion	이벤트가 발생한 AWS 리전
sourceIPAddress	API 호출이 시작된 IP 주소 - 공격자 IP나 내부 IP를 판단하는 중요한 단서가 됨
userAgent	요청을 보낸 클라이언트(AWS CLI, SDK, 웹 콘솔 등) 정보
requestParameters	API 호출에 사용된 파라미터 - 어떤 리소스를 대상으로 어떤 값을 사용해 요청했는지 확인 가능
responseElements	API 호출의 응답 값 - 성공 여부와 생성된 리소스 정보 등 포함
errorMessage	API 호출이 실패한 경우 에러 메시지 표시 - 요청 실패 원인 파악 가능

CloudTrail가 로깅하는 주요 이벤트는 관리 이벤트, 데이터 이벤트, 인사이트 이벤트로 구분할 수 있다. MITRE ATT&CK 전술에 따른 CloudTrail 이벤트는 다음과 같다.

[표 128] MITRE ATT&CK 전술에 따른 주요 CloudTrail 이벤트

이벤트 유형	내용
Initial Access (최초 침투)	공격자가 시스템에 최초로 침투하는 행위에 대한 가시성 제공 (AWS 계정에 접근하거나 사용자 권한을 획득하는 행위 탐지) • `ConsoleLogin`: AWS 관리 콘솔에 로그인 • `PasswordRecoveryRequested`: 비밀번호 복구 요청 • `AssumeRoleWithWebIdentity`: 웹 자격 증명을 사용해 임시 보안 자격 증명으로 역할 수행 • `GetSessionToken`: AWS API에 임시 세션 토큰 요청
Execution (침해 실행)	공격자가 접근한 환경에서 악의적인 코드를 실행하는 행위에 대한 가시성 제공 (AWS 환경 내에서 컴퓨팅 리소스를 시작하거나 명령 실행 행위 탐지) • `StartInstance`: 중지된 EC2 인스턴스 시작 • `StartInstances`: 중지된 다수의 EC2 인스턴스 시작 • `Invoke`: AWS Lambda 함수 호출 • `SendCommand`: EC2 인스턴스에 명령 전송
Persistence (침해 지속)	공격자가 자격 증명 변경 이후에도 접근을 유지하려는 지속 행위에 대한 가시성 제공 (AWS 계정 내에서 백도어를 생성하거나 영구적인 접근 권한을 확보하려는 행위 탐지) • `CreateAccessKey`: AWS 사용자 또는 역할에 대한 액세스 키 생성 • `CreateUser`: 새로운 IAM 사용자 생성 • `CreateNetworkAclEntry`: NACL 항목을 추가해 네트워크 접근 경로 생성 • `CreateRoute`: 라우팅 테이블에 항목을 추가해 네트워크 접근 경로 생성 • `CreateLoginProfile`: IAM 사용자의 로그인 프로필 생성 • `AuthorizeSecurityGroupEgress`: 보안 그룹의 아웃바운드 규칙 변경 • `AuthorizeSecurityGroupIngress`: 보안 그룹의 인바운드 규칙 변경 • `CreateVirtualMFADevice`: 가상 MFA 디바이스 생성 • `CreateConnection`: Direct Connect 연결 생성 • `ApplySecurityGroupsToLoadBalancer`: 로드 밸런서에 보안 그룹 적용 • `SetSecurityGroups`: 로드 밸런서에 보안 그룹 설정 • `AuthorizeDBSecurityGroupIngress`: RDS 데이터베이스 보안 그룹의 인바운드 규칙 허용 • `CreateDBSecurityGroup`: RDS 데이터베이스 보안 그룹 생성 • `ChangePassword`: 사용자 비밀번호 변경

이벤트 유형	내용
Privilege Escalation (권한 상승)	공격자가 낮은 권한에서 더 높은 권한으로 상승하려는 행위에 대한 가시성 제공 (IAM 권한을 변경해 더 많은 AWS 리소스에 접근할 수 있도록 행위 탐지) • `CreateGroup`: IAM 그룹 생성 • `CreateRole`: IAM 역할 생성 • `UpdateAccessKey`: 기존 액세스 키 업데이트 • `PutGroupPolicy`: 그룹에 대한 인라인 정책 추가 및 변경 • `PutRolePolicy`: 역할(Role)에 대한 인라인 정책 추가 및 변경 • `PutUserPolicy`: 사용자에 대한 인라인 정책 추가 및 변경 • `AddRoleToInstanceProfile`: 역할(Role)을 프로필이나 그룹에 추가 • `AddUserToGroup`: 사용자를 프로필이나 그룹에 추가 • `AttachUserPolicy`: 사용자에 IAM 관리형 정책 연결 • `AttachRolePolicy`: 역할(Role)에 IAM 관리형 정책 연결
Defense Evasion (방어 회피)	공격자가 탐지 및 방어 체계를 무력화하려는 행위에 대한 가시성 제공 (CloudTrail 로깅을 중단하거나 보안 솔루션 구성을 삭제 및 변경하는 행위 등 탐지) • `StopLogging`: CloudTrail 로깅 중단 • `DeleteTrail`: CloudTrail 트레일(Trail) 삭제 • `UpdateTrail`: CloudTrail 트레일(Trail) 구성 업데이트 • `PutEventSelectors`: 트레일(Trail)의 이벤트 선택기 수정 • `DeleteFlowLogs`: VPC Flow Log 삭제 • `DeleteDetector`: GuardDuty 탐지기 삭제 • `DeleteMembers`: GuardDuty 멤버 계정 삭제 • `DeleteSnapshot`: EBS 또는 RDS 스냅샷 삭제 • `DeactivateMFADevice`: 사용자 계정의 MFA 장치 비활성화 • `DeleteCertificate`: SSL/TLS 인증서 삭제 • `DeleteConfigRule`: AWS Config 규칙 삭제 • `DeleteAccessKey`: 액세스 키 삭제 • `LeaveOrganization`: AWS Organization에서 계정 탈퇴 • `DisassociateFromMasterAccount`: GuardDuty 마스터 계정에서 계정 연결 해제 • `DisassociateMembers`: GuardDuty 멤버에서 계정 연결 해제 • `StopMonitoringMembers`: GuardDuty 멤버 계정 모니터링 중단
Credential Access (자격증명 접근)	공격자가 자격 증명을 탈취하려는 행위에 대한 가시성 제공 (비밀번호 또는 보안 인증 정보 조회, 생성, 변경 행위 탐지) • `GetSecretValue`: AWS Secrets Manager에 저장된 시크릿 값 확인 • `PutSecretValue`: AWS Secrets Manager에 저장된 시크릿 값 변경 • `GetPasswordData`: EC2 인스턴스에 대한 관리자 비밀번호 확인 • `RequestCertificate`: AWS Certificate Manager에서 인증서 요청 • `UpdateAssumeRolePolicy`: 역할(Role)의 신뢰 정책 업데이트 • `CreateSecret`: Secrets Manager에서 시크릿 생성 • `DeleteSecret`: Secrets Manager에서 시크릿 삭제

이벤트 유형	내용
Discovery (탐색)	공격자가 시스템과 네트워크 환경에 대해 탐색하려는 행위에 대한 가시성 제공 (AWS 환경 내 리소스, 사용자, 권한 등을 나열하고 정보를 수집하려는 행위 탐지) • `ListUsers`: IAM 사용자 목록 나열 • `ListRoles`: IAM 역할(Role) 목록 나열 • `ListIdentities`: IAM 자격 증명 목록 나열 • `ListAccessKeys`: IAM 사용자 액세스 키 나열 • `ListServiceQuotas`: AWS 서비스 할당량 나열 • `ListInstanceProfiles`: EC2 인스턴스 프로필 나열 • `ListBuckets`: S3 버킷 나열 • `ListGroups`: IAM 그룹 나열 • `GetSendQuota`: SES(Simple Email Service) 전송 할당량 확인 • `GetCallerIdentity`: 현재 사용자의 자격 증명 정보 확인 • `DescribeInstances`: EC2 인스턴스에 대한 세부 정보 확인 • `GetBucketAcl`: S3 버킷의 ACL 확인 • `GetBucketVersioning`: S3 버킷의 버전 관리 상태 확인 • `GetAccountAuthorizationDetails`: AWS 계정의 IAM Entity(사용자, 그룹, 역할 등)에 대한 세부 권한 정보 확인
Lateral Movement (내부 이동)	공격자가 네트워크 내의 다른 시스템으로 이동하려는 행위에 대한 가시성 제공 (하나의 역할에서 다른 역할로 전환해 AWS 환경 내에서 이동하려는 행위 탐지) • `AssumeRole`: 현재 역할에서 다른 역할의 권한을 일시적으로 부여 • `SwitchRole`: 현재 역할에서 다른 역할의 권한을 일시적으로 부여
Exfiltration (유출)	공격자가 현재 환경에서 외부로 데이터를 유출하려는 행위에 대한 가시성 제공 (S3 버킷에서 데이터를 다운로드, 스냅샷을 공유해 데이터 외부로 유출하려는 행위 탐지) • `GetObject`: S3 버킷에서 객체 확인 • `CopyObject`: S3 버킷에서 객체 복사 • `CreateSnapShot`: EBS 스냅샷을 생성해 외부와 공유 • `ModifySnapshotAttributes`: EBS 스냅샷의 속성을 수정해 외부와 공유 • `ModifyImageAttribute`: AMI(Amazon Machine Image) 속성을 수정해 외부와 공유 • `SharedSnapshotCopyInitiated`: 공유된 스냅샷 복사 • `SharedSnapshotVolumeCreated`: 공유된 스냅샷의 볼륨 생성 • `ModifyDBSnapshotAttribute`: RDS 데이터 스냅샷의 속성 수정 • `CreateDBSnapshot`: RDS 스냅샷 생성 • `PutBucketPolicy`: S3 버킷의 정책을 변경해 공개적으로 접근 가능하도록 수정 • `PutBucketAcl`: S3 버킷의 ACL을 변경해 공개적으로 접근 가능하도록 수정

이벤트 유형	내용
Impact (영향)	공격자가 데이터, 시스템, 네트워크에 영향을 미치려는 행위에 대한 가시성 제공 (데이터 삭제, 시스템 중단 등의 행위 탐지) • `PutBucketVersioning`: S3 버킷의 버전 관리를 활성화하거나 비활성화 • `RunInstances`: 새로운 EC2 인스턴스 시작 (이벤트 실행 비용을 발생시켜 서비스 거부 공격에 악용 가능) • `DeleteAccountPublicAccessBlock`: S3 퍼블릭 액세스 차단 설정 삭제 • `DeleteObject`: S3 버킷 객체 삭제 • `DeleteDBInstance`: RDS 데이터베이스 인스턴스 삭제 • `ModifyDBInstance`: RDS 데이터베이스 인스턴스 수정

2) VPC Flow Logs

VPC Flow Logs를 S3에 저장할 경우, 일반 텍스트나 Parquet 형식(Gzip 압축을 사용하는 열 기반 데이터 형식)으로 로깅되며, CloudWatch에 저장할 경우 CloudWatch 서비스 콘솔이 로깅된다. 사고 분석에 활용되는 주요 필드는 다음과 같다.

[표 129] VPC Flow Logs 예시

로그 예시
123456789012 eni-1a2b3c4d 203.0.113.10 172.31.5.10 54321 22 6 1 40 1678886400 1678886401 REJECT OK

[표 130] 사고 분석 시 주로 활용되는 VPC Flow Logs 로그 필드

필드 구분	설명
account-id	트래픽이 로깅되는 소스 네트워크 인터페이스 소유자의 AWS 계정 ID
interface-id	트래픽이 로깅되는 네트워크 인터페이스 ID
srcaddr	수신 트래픽인 경우: 트래픽 Soure IP, 송신 트래픽인 경우: 트래픽을 전송하는 네트워크 인터페이스의 IP
dstaddr	송신 트래픽인 경우: 트래픽 대상 IP, 수신 트래픽인 경우 트래픽이 들어오는 네트워크 인터페이스의 IP
srcport	트래픽의 srcaddr에서 사용된 포트
dstport	트래픽의 dstaddr에서 사용된 포트
protocol	트래픽의 IANA 프로토콜 번호(TCP 6, UDP 17 등)
packets	네트워크 트래픽에서 전송된 패킷 수
bytes	네트워크 트래픽에서 전송된 바이트 수
start	집계 간격 내에서 네트워크 트래픽의 첫 번째 패킷이 수신된 시간 (Unix Timestamp)
end	집계 간격 내에서 네트워크 트래픽의 마지막 패킷을 수신한 시간 (Unix Timestamp)
action	트래픽과 연결된 작업(ACCEPT, REJECT)

MITRE ATT&CK 전술에 따른 VPC Flow Logs 이벤트는 다음과 같다.

[표 131] MITRE ATT&CK 전술에 따른 주요 VPC Flow Logs 이벤트

이벤트 유형	내용

Reconnaissance
(정찰)
	공격자가 대상 네트워크 정보를 수집하는 행위 탐지
	(외부 IP로부터 비정상적인 포트 스캐닝 행위 탐지)
	
	• 외부 IP(203.0.113.10)에서 내부 서버들로 SSH 포트 스캐닝 시도했으나, 보안 그룹 및 네트워크 ACL을 통해 차단(REJECT)된 로그 예시
	`123456789012 eni-1a2b3c4d 203.0.113.10 172.31.5.10 54321 22 6 1 40 1678886400 1678886401 REJECT OK`
	`123456789012 eni-1a2b3c4d 203.0.113.10 172.31.5.11 54321 22 6 1 40 1678886401 1678886402 REJECT OK`
	`123456789012 eni-1a2b3c4d 203.0.113.10 172.31.5.12 54321 22 6 1 40 1678886402 1678886402 REJECT OK`

Initial Access (최초 침투)	공격자가 시스템에 최초로 침투하는 행위 탐지 (외부 IP로부터 접속한 행위 탐지, 알려지지 않은 외부 IP 또는 악성 IP 접근 행위 탐지) • 외부 IP(203.0.113.10)에서 내부 서버(172.31.5.10)로 RDP 접근 성공한 로그 예시 `123456789012 eni-1a2b3c4d 203.0.113.10 172.31.5.10 54321 3389 6 1 40 1678886400 1678886430 ACCEPT OK`
Lateral Movement (내부 이동)	공격자가 최초로 침투한 시스템에서 다른 내부 시스템으로 이동하려는 행위 탐지 (평소 통신하지 않던 시스템 간의 비정상적인 통신 분석) • 내부에서 다른 내부 IP로 SMB 포트를 통해 비정상적인 통신이 발생한 로그 예시 `123456789012 eni-1a2b3c4d 172.31.5.10 172.31.5.20 445 55555 6 100 10000 1678888000 1678888010 ACCEPT OK`
Command and Control (명령 제어)	공격자가 침투한 시스템과 통신해 명령을 실행하거나 악성코드를 제어하려는 행위 탐지 (악성코드가 외부 C2 서버와의 통신 가능성 분석) • 내부 서버에서 외부 IP로 5GB 데이터 전송 `123456789012 eni-1a2b3c4d 172.31.5.10 198.51.100.1 54321 443 6 200 20000 1678889000 1678889030 ACCEPT OK`
Exfiltration (유출)	공격자가 침투한 시스템에서 데이터를 외부로 유출하려는 행위 탐지 (평소와 다른 시간대에 대량의 아웃바운드 트래픽이 발생 시 데이터 유출 의심) • 내부 서버에서 외부 IP로 5GB 데이터 전송 로그 예시 `123456789012 eni-1a2b3c4d 172.31.5.10 203.0.113.1 54321 80 6 100000 5368709120 1678890000 1678910020 ACCEPT OK`

3) S3 Server Access Log

S3 Server Access Log는 공백으로 구분된 필드들의 목록으로 구성되어 있으며 텍스트 파일 형식으로 로깅된다. 각 레코드는 단일 S3 요청에 대한 정보를 담고 있고 있다. 사고 분석에 활용되는 주요 필드는 다음과 같다.

[표 132] S3 Server Access Log 예시

로그 예시
123456789012 eni-1a2b3c4d 203.0.113.10 172.31.5.10 54321 22 6 1 40 1678886400 1678886401 REJECT OK

[표 133] 사고 분석 시 주로 활용되는 S3 Server Access Log 로그 필드

필드 구분	설명
Bucket Owner	S3 버킷 소유자의 AWS ID
Bucket	요청이 발생한 S3 버킷 명
Time	요청이 완료된 시간 (UTC)
Remote IP	요청을 보낸 클라이언트 IP 주소
Requester	요청자가 IAM 사용자의 경우, IAM 사용자 이름과 사용자가 속한 AWS 계정
Request ID	Amazon S3에서 각 요청을 고유하게 식별하기 위해 생성한 ID
Operation	요청을 통해 S3 버킷에서 수행된 작업
Key	요청의 객체 명
Request-URI	HTTP 요청 메시지 내 Request-URI 내용
HTTP Status	요청에 대한 HTTP 응답 코드(200, 403, 404 등)
Error Code	오류가 발생했을 경우 S3 오류 코드
Bytes Sent	응답을 통해 전송된 바이트 수
Object Size	객체의 크기
Total Time	S3 버킷이 요청을 처리하는데 걸린 시간
User-Agent	요청을 보낸 클라이언트 애플리케이션 정보

MITRE ATT&CK 전술에 따른 S3 Server Access Log 오퍼레이션은 다음과 같다.

[표 134] MITRE ATT&CK 전술에 따른 주요 S3 Server Access Log 오퍼레이션

이벤트 유형	내용
Privilege Escalation (권한 상승)	공격자가 낮은 권한에서 더 높은 권한으로 상승하려는 행위에 대한 가시성 제공 (객체 또는 S3 버킷의 권한 변경 행위 탐지) • `REST.PUT.ACL`: 객체 또는 S3 버킷의 ACL 수정
Discovery (탐색)	공격자가 S3 버킷의 환경에 대해 탐색하려는 행위에 대한 가시성 제공 (S3 목록과 권한, 설정 정보에 대한 정보 수집 행위 탐지) • `REST.GET.BUCKET`: S3 버킷 목록 조회 • `REST.GET.ACL`: 객체 또는 S3 버킷의 ACL 조회 • `REST.GET.BUCKET.LOCATION`: S3 버킷 리전 정보 조회 • `REST.GET.ENCRYPTION`: 암호화 설정 정보 조회
Exfiltration (유출)	공격자가 S3 버킷 외부로 데이터를 유출하려는 행위에 대한 가시성 제공 (S3 버킷에서 데이터 다운로드, 공격자 S3 버킷으로 복사해 외부 유출하는 행위 탐지) • `REST.GET.OBJECT`: 객체 다운로드 요청 • `REST.COPY.OBJECT`: 객체 복사 요청
Impact (영향)	공격자가 데이터에 영향을 미치려는 행위에 대한 가시성 제공 (S3 버킷 내 객체와 S3 버킷을 삭제하려는 행위 탐지) • `REST.DELETE.OBJECT`: 객체 삭제 요청 • `REST.DELETE.BUCKET`: S3 버킷 삭제 요청

4) CloudWatch Logs

CloudWatch Logs Insights를 활용하면 AWS에서 수집되는 로그를 대상으로 이상 징후 기반 탐지 쿼리를 수행해 공격 전술에 대응할 수 있다. 대표적인 로그 유형별 탐지 목적과 예시 쿼리는 다음과 같다.

[표 135] CloudWatch를 활용한 이상 행위 탐지 예시

행위 탐지 구분	내용

비정상
로그인 시도 탐지
(CloudTrail Log)
	최근 24시간동안 실패한 콘솔 로그인(`ConsoleLogin`) 이벤트를 IP 주소별로 집계해
	비정상적인 로그인 실패 패턴을 분석함으로써 무차별 대입 공격 시도 탐지
	• 탐지 쿼리 예시
	`fields @timestamp, @message`
	`| filter eventName = 'ConsoleLogin' and errorMessage = 'Failed authentication'`
	`| stats count(*) as login_failures by sourceIPAddress`
	`| sort login_failures desc`

신규 액세스 키
악용 탐지
(CloudTrail Log)
	생성 직후 신규 액세스 키를 사용해 짧은 시간 내 다수의 API 호출을 식별해
	키 탈취 또는 남용 행위 탐지
	• 탐지 쿼리 예시
	`fields @timestamp, eventName, userIdentity.arn, requestParameters.userName, sourceIPAddress`
	`| filter eventName = "CreateAccessKey"`
	`| sort @timestamp desc`
	`| limit 50`

포트 스캐닝 탐지
(VPC Flow Logs)
	네트워크 트래픽 중 거부(REJECT) 응답이 비정상적으로 높은 출발지 IP를 식별해
	포트 스캐닝 등 탐색 전술에 해당하는 의심스러운 트래픽 탐지
	• 탐지 쿼리 예시
	`fields @timestamp, @message`
	`| filter action = 'REJECT'`
	`| stats count(*) as rejected_packets by srcAddr`
	`| sort rejected_packets desc`
	`| limit 10`

비인가자 접근 탐지
(S3 Access Log)
	버킷에 허용되지 않은 IP 또는 IAM 사용자의 접근 시도를 식별해
	데이터 유출 전술에 해당하는 의심 행위 탐지
	• 탐지 쿼리 예시
	`fields @timestamp, requester, bucket, requestUri, status`
	`| filter bucket = 'important-data-bucket' and status = 'AccessDenied'`
	`| stats count(*) as denied_access by requester, remoteIP`
	`| sort denied_access desc`

5) RDS Logs

데이터베이스 로그의 구조는 데이터베이스 엔진(MySQL, PostgreSQL, MariaDB 등)과 로그 유형에 따라 상이하다. 각 로그는 일반적으로 시간 정보, 연결 정보, 이벤트 내용 등 공통된 필드를 포함하지만 세부적인 형식에는 차이가 있다. AWS DB 인스턴스 이벤트(AWS DB Instance Events)의 구조는 JSON 형식으로 로깅되며 이벤트 기본 구조는 다음과 같다.

[표 136] RDS Logs 예시

로그 예시
{ "version": "0", "id": "68f6e973-1a0c-d37b-f2f2-94a7f62ffd4e", "detail-type": "RDS DB Instance Event", "source": "aws.rds", "account": "123456789012", "time": "2018-09-27T22:36:43Z", "region": "us-east-1", "resources": ["arn:aws:rds:us-east-1:123456789012:db:my-db-instance"], "detail": { "EventCategories": ["failover"], "SourceType": "DB_INSTANCE", "SourceArn": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance", "Date": "2018-09-27T22:36:43.292Z", "Message": "A Multi-AZ failover has completed.", "SourceIdentifier": "my-db-instance", "EventID": "RDS-EVENT-0049" } }

[표 137] AWS DB Instance Events 기본 구조

필드 구분	설명
ID	이벤트 ID (고유 식별)
Detail-type	이벤트의 구체적인 유형 (EventBridge 규칙의 필터 키로 사용)
Account	이벤트를 발생시킨 AWS 계정 ID
Time	이벤트가 발생시킨 시간 (UTC)
Region	이벤트를 발생시킨 AWS 리전
EventCategories	이벤트의 분류(availability, security, configuration change 등)
Data	이벤트가 발생한 시간 (UTC)
Message	이벤트에 대한 설명
SourceIdentifier	이벤트가 발생한 리소스의 이름
EventID	발생한 이벤트에 대한 고유한 ID

MITRE ATT&CK 전술에 따른 RDS Log 이벤트는 다음과 같다.

[표 138] MITRE ATT&CK 전술에 따른 주요 RDS Logs 이벤트

이벤트 유형	내용
Defense Evasion (권한 상승)	공격자가 탐지 및 방어 체계를 무력화하려는 행위에 대한 가시성 제공 (AWS DB 인스턴스 이벤트 로깅 중단 행위 탐지) • `RDS-EVENT-0332`: 전용 로그 볼륨 비활성화
Exfiltration (유출)	공격자가 외부로 데이터를 유출하려는 행위에 대한 가시성 제공 (DB 인스턴스 클래스에 대한 공개 범위 설정을 변경해 외부로 노출시키려는 행위 탐지) • `RDS-EVENT-0014`: DB 인스턴스 클래스에 대한 수정 사항 적용 완료
Impact (영향)	공격자가 데이터에 영향을 미치려는 행위에 대한 가시성 제공 (DB 인스턴스 삭제, 암호화 전 백업 비활성, 스냅샷 삭제 등 행위 탐지) • `RDS-EVENT-0003`: DB 인스턴스 삭제 • `RDS-EVENT-0041`: 사용자 스냅샷 삭제 • `RDS-EVENT-0028`: 자동 백업 비활성화

6) GuardDuty Findings

GuardDuty Findings는 JSON 형식으로 로깅되며, 사고 분석에 필수적인 다양한 세부 정보를 포함하고 있다. 가장 핵심적인 요소는 Finding Type으로 이는 탐지된 위협의 유형을 나타낸다. Finding Type 구조는 다음과 같다.

[표 139] GuardDuty Findings Type 구조

Finding Type 구조 형식	
ThreatPurpose:ResourceTypeAffected/ThreatFamilyName.DetectionMechanism!Artifact	
필드 구분	설명
ThreatPurpose	위협의 주요 목적(Backdoor, DefenseEvasion, Discovery, Recon 등)
ResourceTypeAffected	공격 대상 AWS 리소스
ThreatFamilyName	위협 또는 악의적인 활동명
DetectionMechanism	위협을 탐지한 방법(TCP, UDP 등)
Artifact	위협과 관련된 아티팩트(부가 정보)

이 외에도 Findings에는 다음과 같은 주요 정보들이 포함된다.

[표 140] GuardDuty Findings 주요 정보

구분	설명
심각도	위협의 위험 수준을 나타내며, 높음(High), 중간(Medium), 낮음(Low)으로 구분
계정 ID	위협이 탐지된 AWS 계정의 ID
리전	위협이 발생한 AWS 리전
리소스 정보	영향을 받은 리소스에 대한 구체적인 정보(EC2 인스턴스 ID, S3 버킷 이름 등
행위자 정보	공격을 시도한 주체에 대한 정보(IP, 위치, 공격 그룹 등)
이벤트 발생 시간	위협 행위가 처음 발생한 시간과 마지막으로 탐지된 시간을 기록

GuardDuty Findings 활용 방안은 다음과 같다.

[표 141] GuardDuty Findings 활용 방안

구분	설명
자동화된 알림 및 대응	Amazon EventBridge와 AWS Lambda를 연동해 특정 유형이나 심각도 수준의 Findings가 발생했을 때 자동으로 알림 받거나 대응 조치 가능
중앙 집중식 로그 분석 및 시각화	생성된 Findings는 Amazon S3 버킷으로 내보내 장기간 보관하고, Amazon Athena를 활용해 SQL 쿼리로 분석 가능 Amazon OpenSeacrh 서비스나 QuickSight와 같은 시각화 도구와 연동해 대시보드를 구축하면 시간에 따른 위협 트랜드 파악 및 잠재적인 보안 취약점 식별에 도움이 된다.
사고 초동 분석	사고가 발생했을 때, Findings에 포함된 공격자 IP, 영향을 받은 리소스, 이벤트 발생 일시 등의 정보를 기반으로 CloudTrail Log, VPC Flow Logs 등 다른 로그 데이터와 연관 분석을 수행해 공격의 전반적인 흐름과 영향 범위 파악 가능
신뢰할 수 있는 IP 및 위협 인텔리전스 목록 활용	신뢰 IP 목록과 위협 IP 목록을 직접 업로드해 탐지 정확도를 높일 수 있음 (공유되는 악성 IP 목록을 위협 목록에 추가해 알려진 위협에 대해 신속하게 탐지하고 대응 가능)

7) WAF Log

WAF Log는 JSON 형식으로 로깅되며, 각 요청에 대한 다양한 필드 정보를 포함하고 있다. 로그 구조는 WAF 버전에 따라 차이가 있을 수 있으며, 일반적으로 다음과 같은 주요 필드들이 포함된다.

[표 142] AWS Logs 예시

로그 예시 일부
"timestamp": 1758865233531, "formatVersion": 1, "webaclId": "arn:aws:wafv2:ap-southeast-2:111122223333:regional/webacl/STMTest/1EXAMPLE-2ARN-3ARN-4ARN-123456EXAMPLE", "terminatingRuleId": "Test_SQLi_XSS", "terminatingRuleType": "REGULAR", "action": "BLOCK", "terminatingRuleMatchDetails": [{ "conditionType": "SQL_INJECTION", "sensitivityLevel": "HIGH", "location": "HEADER", "matchedData": ["10", "AND", "1"] }], - 이하 생략 -

[표 143] WAF Log 주요 필드

필드 구분	설명
timestamp	로그가 발생된 시간
formatVersion	로그 형식의 버전
webaclId	요청을 처리한 Web ACL의 ID
terminatingRuleId	요청을 최종적으로 허용 또는 차단한 규칙 ID
action	규칙에 의해 수행된 조치(ALLOW, BLOCK, COUNT 등)
terminatingRuleMatchDetails	요청이 규칙과 일치한 구체적인 조건
httpRequest	HTTP 요청에 대한 상세 정보 - clientIp: 요청을 보낸 클라이언트의 IP 주소 - country: 클라이언트 IP의 국가 코드(KR, US 등) - headers: 요청 헤더 정보 - uri: 요청된 URI 경로 - args: 요청에 포함된 쿼리 문자열 - httpVersion: HTTP 버전 - httpMethod: 요청에서 사용된 HTTP 메서드(GET, POST 등)
rateBasedRuleList	속도 기반 규칙에 의해 관리되는 IP 목록 (해당 시)

WAF Log 분석을 통해 다양한 보안 위협 및 비정상적인 활동을 탐지할 수 있으며, 주요 활용 방안은 다음과 같다.

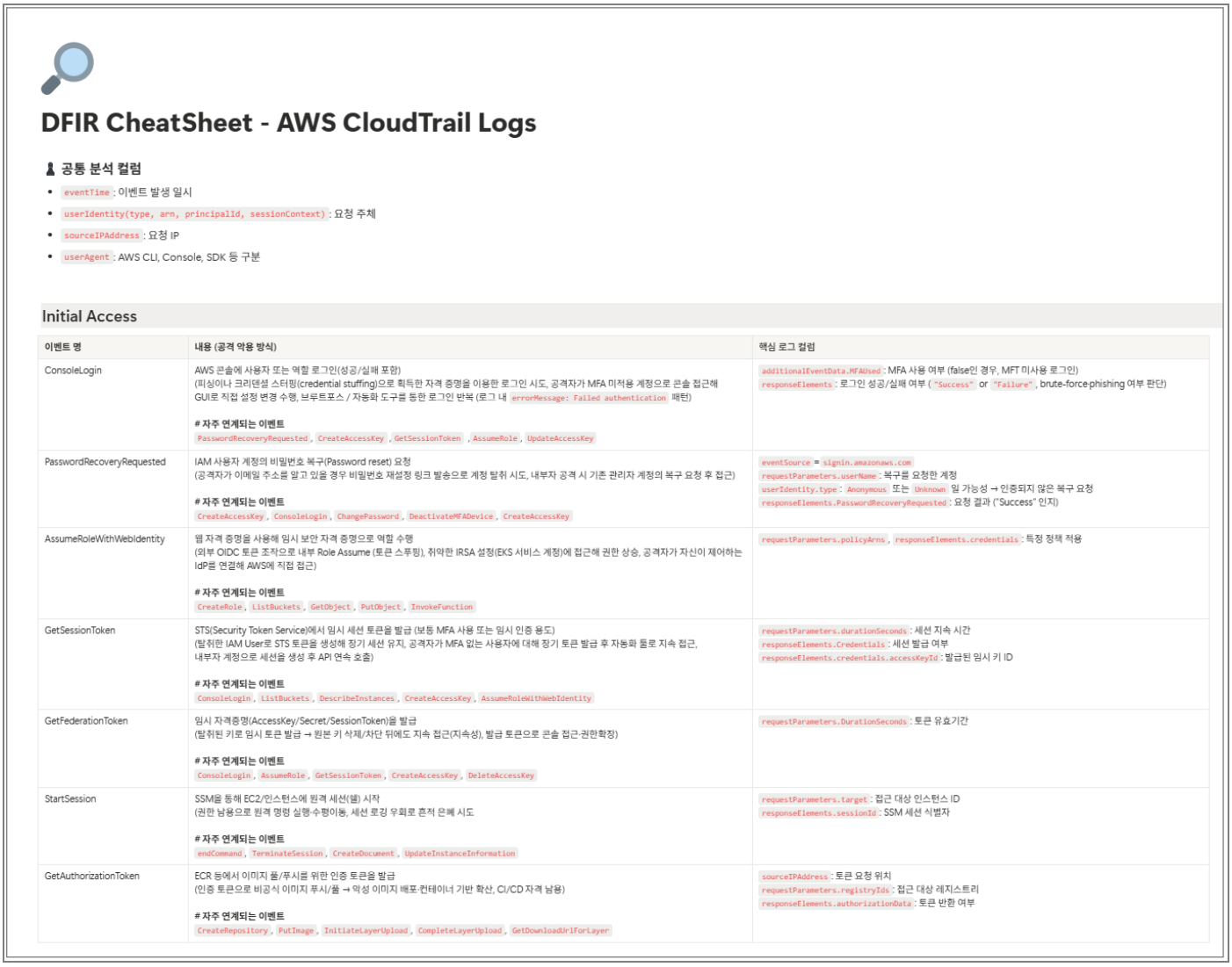
[표 144] WAF Log 활용 방안

구분	설명
SQL 인젝션 (SQL Injection)	'terminatingRuleMatchDetails' 필드에서 SQL 쿼리와 관련된 패턴이 확인된 경우
크로스 사이트 스크립팅 (XSS)	요청의 URI, 쿼리 문자열 또는 본문에서 스크립트 태그(<script>, </script>)와 같은 악성 스크립트가 발견되는 경우
디렉터리 트래버설 (Directory Traversal)	상위 디렉터리(../)로 이동하려는 시도가 URI나 파라미터에서 발견되는 경우
서비스 거부 공격 (DoS/DDoS)	특정 클라이언트 IP에서 비정상적으로 많은 요청이 단시간 내에 발생하는 경우

5.2. 공격 전술별 로그 이벤트 매핑 DFIR CheatSheet 개발

MITRE ATT&CK 전술을 기준으로 AWS 로그(CloudTrail Log, S3 Access Log)를 매핑 및 분류해 사고 대응에서 활용 가능한 DFIR CheatSheet를 제시한다. 제시된 자료는 전술별 핵심 이벤트, 자주 연계되는 이벤트, 공격자 악용 패턴 및 각 이벤트에 대한 DFIR 핵심 컬럼을 정형화해 탐지 및 분석의 일관성과 효율성을 향상시키는 것을 목표로 한다.

Cheatsheet에는 CloudTrail Log의 175개 이벤트와 S3 Access Log의 47개 오퍼레이션이 포함되어 있다.



DFIR CheatSheet - AWS CloudTrail Logs

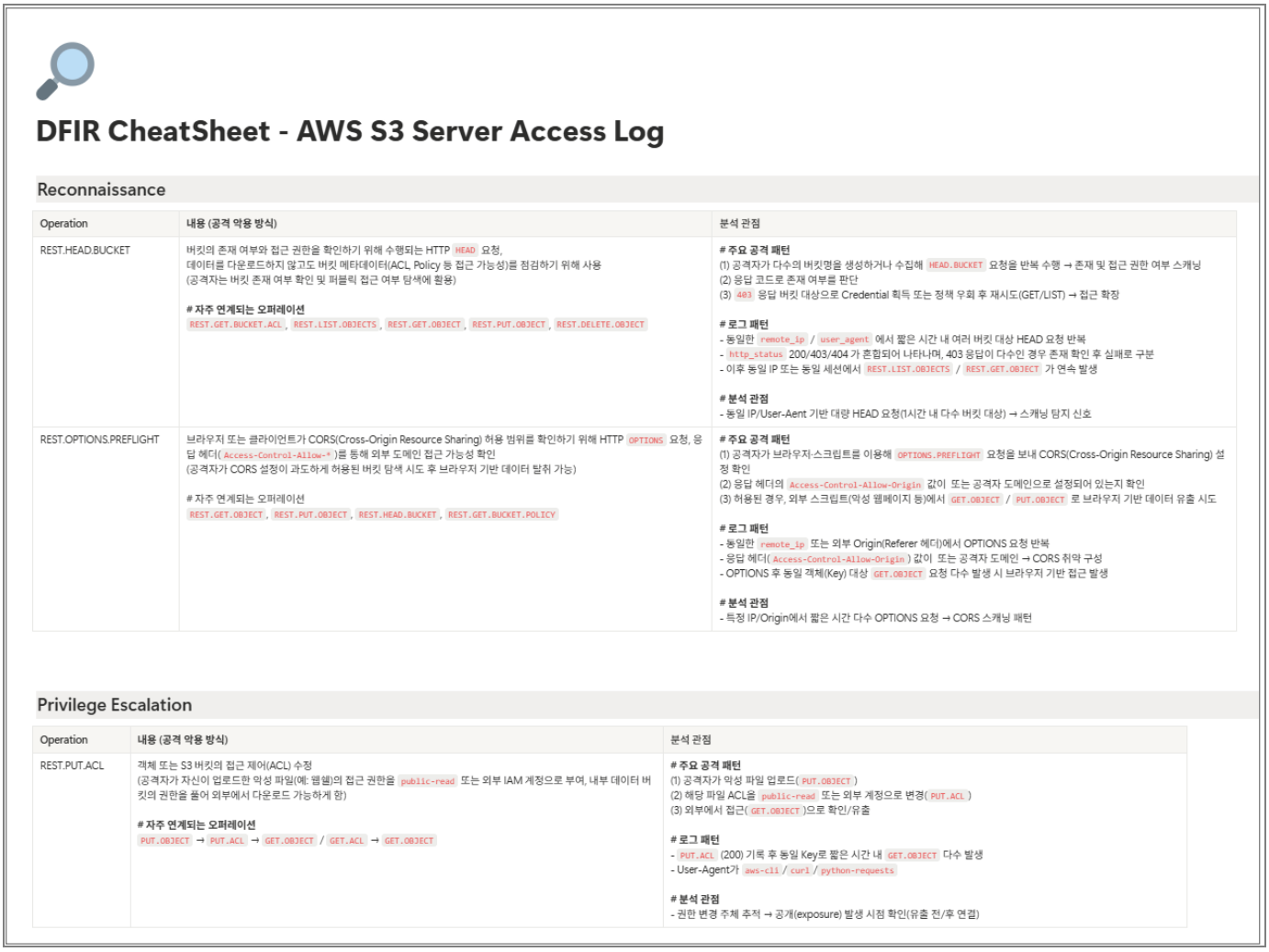
공통 분석 컬럼

- `eventTime` : 이벤트 발생 일시
- `userIdentity(type, arn, principalId, sessionContext)` : 요청 주체
- `sourceIPAddress` : 요청 IP
- `userAgent` : AWS CLI, Console, SDK 등 구분

Initial Access

이벤트 명	내용 (공격 악용 방식)	핵심 로그 컬럼
ConsoleLogin	AWS 콘솔에 사용자 또는 역할 로그인(성공/실패 포함) (피싱이나 크리덴셜 스터핑(credential stuffing)으로 획득한 자격 증명을 이용한 로그인 시도, 공격자가 MFA 미적용 계정으로 콘솔 접근해 GUI로 직접 설정 변경 수행, 브루트포스 / 자동화 도구를 통한 로그인 반복 (로그 내 `errorMessage: Failed authentication` 패턴) # 자주 연계되는 이벤트 `PasswordRecoveryRequested` , `CreateAccessKey` , `GetSessionToken` , `AssumeRole` , `UpdateAccessKey`	`additionalEventData.MFAUsed` : MFA 사용 여부 (false인 경우, MFT 미사용 로그인) `responseElements` : 로그인 성공/실패 여부 (`"Success"` or `"Failure"` , brute-force·phishing 여부 판단)
PasswordRecoveryRequested	IAM 사용자 계정의 비밀번호 복구(Password reset) 요청 (공격자가 이메일 주소를 알고 있을 경우 비밀번호 재설정 링크 발송으로 계정 탈취 시도, 내부자 공격 시 기존 관리자 계정의 복구 요청 후 접근) # 자주 연계되는 이벤트 `CreateAccessKey` , `ConsoleLogin` , `ChangePassword` , `DeactivateMFADevice` , `CreateAccessKey`	`eventSource` = `signin.amazonaws.com` `requestParameters.userName` : 복구를 요청한 계정 `userIdentity.type` : `Anonymous` 또는 `Unknown` 일 가능성 → 인증되지 않은 복구 요청 `responseElements.PasswordRecoveryRequested` : 요청 결과 ("Success" 인지)
AssumeRoleWithWebIdentity	웹 자격 증명을 사용해 임시 보안 자격 증명으로 역할 수행 (외부 OIDC 토큰 조작으로 내부 Role Assume (토큰 스푸핑), 취약한 IRSA 설정(EKS 서비스 계정)에 접근해 권한 상승, 공격자가 자신이 제어하는 IdP를 연결해 AWS에 직접 접근) # 자주 연계되는 이벤트 `CreateRole` , `ListBuckets` , `GetObject` , `PutObject` , `InvokeFunction`	`requestParameters.policyArns` , `responseElements.credentials` : 특정 정책 적용
GetSessionToken	STS(Security Token Service)에서 임시 세션 토큰을 발급 (보통 MFA 사용 또는 임시 인증 용도) (탈취한 IAM User로 STS 토큰을 생성해 장기 세션 유지, 공격자가 MFA 없는 사용자에 대해 장기 토큰 발급 후 자동화 툴로 지속 접근, 내부자 계정으로 세션을 생성 후 API 연속 호출) # 자주 연계되는 이벤트 `ConsoleLogin` , `ListBuckets` , `DescribeInstances` , `CreateAccessKey` , `AssumeRoleWithWebIdentity`	`requestParameters.durationSeconds` : 세션 지속 시간 `responseElements.Credentials` : 세션 발급 여부 `responseElements.credentials.accessKeyId` : 발급된 임시 키 ID
GetFederationToken	임시 자격증명(AccessKey/Secret/SessionToken)을 발급 (탈취된 키로 임시 토큰 발급 → 원본 키 삭제/차단 뒤에도 지속 접근(지속성), 발급 토큰으로 콘솔 접근·권한확장) # 자주 연계되는 이벤트 `ConsoleLogin` , `AssumeRole` , `GetSessionToken` , `CreateAccessKey` , `DeleteAccessKey`	`requestParameters.DurationSeconds` : 토큰 유효기간
StartSession	SSM을 통해 EC2/인스턴스에 원격 세션(쉘) 시작 (권한 남용으로 원격 명령 실행·수평이동, 세션 로깅 우회로 흔적 은폐 시도 # 자주 연계되는 이벤트 `endCommand` , `TerminateSession` , `CreateDocument` , `UpdateInstanceInformation`	`requestParameters.target` : 접근 대상 인스턴스 ID `responseElements.sessionId` : SSM 세션 식별자
GetAuthorizationToken	ECR 등에서 이미지 풀/푸시를 위한 인증 토큰을 발급 (인증 토큰으로 비공식 이미지 푸시/풀 → 악성 이미지 배포·컨테이너 기반 확산, CI/CD 자격 남용) # 자주 연계되는 이벤트 `CreateRepository` , `PutImage` , `InitiateLayerUpload` , `CompleteLayerUpload` , `GetDownloadUrlForLayer`	`sourceIPAddress` : 토큰 요청 위치 `requestParameters.registryIds` : 접근 대상 레지스트리 `responseElements.authorizationData` : 토큰 반환 여부

[그림 6] DFIR Cheatsheet – AWS CloudTrail Log 내용 일부



DFIR CheatSheet - AWS S3 Server Access Log

Reconnaissance

Operation	내용 (공격 악용 방식)	분석 관점
REST.HEAD.BUCKET	버킷의 존재 여부와 접근 권한을 확인하기 위해 수행되는 HTTP `HEAD` 요청, 데이터를 다운로드하지 않고도 버킷 메타데이터(ACL, Policy 등 접근 가능성)를 점검하기 위해 사용 (공격자는 버킷 존재 여부 확인 및 퍼블릭 접근 여부 탐색에 활용) **# 자주 연계되는 오퍼레이션** `REST.GET.BUCKET.ACL`, `REST.LIST.OBJECTS`, `REST.GET.OBJECT`, `REST.PUT.OBJECT`, `REST.DELETE.OBJECT`	**# 주요 공격 패턴** (1) 공격자가 다수의 버킷명을 생성하거나 수집해 `HEAD.BUCKET` 요청을 반복 수행 → 존재 및 접근 권한 여부 스캐닝 (2) 응답 코드로 존재 여부를 판단 (3) `403` 응답 버킷 대상으로 Credential 획득 또는 정책 우회 후 재시도(GET/LIST) → 접근 확장 **# 로그 패턴** - 동일한 `remote_ip` / `user_agent` 에서 짧은 시간 내 여러 버킷 대상 HEAD 요청 반복 - `http_status` 200/403/404 가 혼합되어 나타나며, 403 응답이 다수인 경우 존재 확인 후 실패로 구분 - 이후 동일 IP 또는 동일 세션에서 `REST.LIST.OBJECTS` / `REST.GET.OBJECT` 가 연속 발생 **# 분석 관점** - 동일 IP/User-Aent 기반 대량 HEAD 요청(1시간 내 다수 버킷 대상) → 스캐닝 탐지 신호
REST.OPTIONS.PREFLIGHT	브라우저 또는 클라이언트가 CORS(Cross-Origin Resource Sharing) 허용 범위를 확인하기 위해 HTTP `OPTIONS` 요청, 응답 헤더(`Access-Control-Allow-*`)를 통해 외부 도메인 접근 가능성 확인 (공격자가 CORS 설정이 과도하게 허용된 버킷 탐색 시도 후 브라우저 기반 데이터 탈취 가능) # 자주 연계되는 오퍼레이션 `REST.GET.OBJECT`, `REST.PUT.OBJECT`, `REST.HEAD.BUCKET`, `REST.GET.BUCKET.POLICY`	**# 주요 공격 패턴** (1) 공격자가 브라우저 스크립트를 이용해 `OPTIONS.PREFLIGHT` 요청을 보내 CORS(Cross-Origin Resource Sharing) 설정 확인 (2) 응답 헤더의 `Access-Control-Allow-Origin` 값이 또는 공격자 도메인으로 설정되어 있는지 확인 (3) 허용된 경우, 외부 스크립트(악성 웹페이지 등)에서 `GET.OBJECT` / `PUT.OBJECT` 로 브라우저 기반 데이터 유출 시도 **# 로그 패턴** - 동일한 `remote_ip` 또는 외부 Origin(Referer 헤더)에서 OPTIONS 요청 반복 - 응답 헤더(`Access-Control-Allow-Origin`) 값이 또는 공격자 도메인 → CORS 취약 구성 - OPTIONS 후 동일 객체(Key) 대상 `GET.OBJECT` 요청 다수 발생 시 브라우저 기반 접근 발생 **# 분석 관점** - 특정 IP/Origin에서 짧은 시간 다수 OPTIONS 요청 → CORS 스캐닝 패턴

Privilege Escalation

Operation	내용 (공격 악용 방식)	분석 관점
REST.PUT.ACL	객체 또는 S3 버킷의 접근 제어(ACL) 수정 (공격자가 자신이 업로드한 악성 파일(예: 웹쉘)의 접근 권한을 `public-read` 또는 외부 IAM 계정으로 부여, 내부 데이터 버킷의 권한을 풀어 외부에서 다운로드 가능하게 함) **# 자주 연계되는 오퍼레이션** `PUT.OBJECT` → `PUT.ACL` → `GET.OBJECT` / `GET.ACL` → `GET.OBJECT`	**# 주요 공격 패턴** (1) 공격자가 악성 파일 업로드(`PUT.OBJECT`) (2) 해당 파일 ACL을 `public-read` 또는 외부 계정으로 변경(`PUT.ACL`) (3) 외부에서 접근(`GET.OBJECT`)으로 확인/유출 **# 로그 패턴** - `PUT.ACL` (200) 기록 후 동일 Key로 짧은 시간 내 `GET.OBJECT` 다수 발생 - User-Agent가 `aws-cli` / `curl` / `python-requests` **# 분석 관점** - 권한 변경 주체 추적 → 공개(exposure) 발생 시점 확인(유출 전/후 연결)

[그림 7] DFIR Cheatsheet – AWS S3 Server Access Logs 내용 일부

CloudTrail Log의 각 전술 별 핵심 이벤트는 다음과 같다.

[표 145] DFIR CheatSheet에 작성된 전술별 핵심 이벤트 - CloudTrail

기능 구분	이벤트 명	이벤트 설명
Initial Access (최초 침투)	ConsoleLogin	AWS 콘솔에 사용자 또는 역할 로그인
	PasswordRecoveryRequested	IAM 사용자 계정의 비밀번호 복구(Password reset) 요청
	AssumeRoleWithWebIdentity	웹 자격 증명을 사용해 임시 보안 자격 증명으로 역할 수행
	GetSessionToken	STS(Security Token Service)에서 임시 세션 토큰을 발급
	GetFederationToken	임시 자격증명(AccessKey/Secret/SessionToken)을 발급
	StartSession	SSM을 통해 EC2/인스턴스에 원격 세션(쉘) 시작
	GetAuthorizationToken	ECR 등에서 이미지 풀/푸시를 위한 인증 토큰을 발급
Execution (침해 실행)	StartInstance	중지된 EC2 인스턴스 시작
	StartInstances	중지된 다수의 EC2 인스턴스 시작
	Invoke	AWS Lambda 함수 호출
	SendCommand	EC2 인스턴스에 명령 전송
Persistence (침해 지속)	CreateAccessKey	AWS 사용자 또는 역할에 대한 액세스 키 생성
	CreateUser	새 IAM 사용자 생성
	CreateNetworkAclEntry	VPC 네트워크 ACL에 인바운드/아웃바운드 규칙 추가
	CreateRoute	라우팅 테이블에 새 경로(Route) 추가
	CreateLoginProfile	IAM 사용자 콘솔 로그인용 비밀번호 생성
	AuthorizeSecurityGroupEgress	보안 그룹의 송신 규칙(인/아웃바운드) 변경해 네트워크 통신 허용
	AuthorizeSecurityGroupIngress	보안 그룹의 수신 규칙(인/아웃바운드)을 변경해 네트워크 통신 허용
	CreateVirtualMFADevice	가상 MFA 디바이스 생성
	CreateConnection	Direct Connect 연결 또는 VPN Connection 생성
	ApplySecurityGroupsToLoadBalancer	로드 밸런서(ELB)에 보안 그룹 적용
	SetSecurityGroups	EC2 Network Interface, Lambda, ENI 등 자원에 로드 밸런서의 보안 그룹을 직접 적용
	AuthorizeDBSecurityGroupIngress	RDS용 DB 보안 그룹에 인바운드 허용 규칙 추가
	CreateDBSecurityGroup	RDS 보안그룹 생성
	ChangePassword	IAM 사용자 비밀번호 변경
	CreateFunction	새로운 Lambda 함수(코드 + 구성) 생성
	CreateTags	AWS 리소스에 메타데이터 태그 추가
	DeleteBucketCors	버킷 CORS(Cross-Origin Resource Sharing) 설정 제거
	DeleteBucketPolicy	S3 버킷 정책 삭제
	CreateImage	EC2 인스턴스의 AMI(시스템 이미지) 생성

기능 구분	이벤트 명	이벤트 설명
Persistence (침해 지속)	CreateInstance	EC2 인스턴스 생성
	CreateKeyPair	SSH 접속용 키페어(공개키/개인키)를 생성하고 개인키(키 재사용 불가)을 반환
	CreateRepository	컨테이너 레지스트리(Repository)를 생성
	PutImage	ECR 레포지토리에 컨테이너 이미지 업로드
	PutUserData	EC2 인스턴스의 User Data(부팅 시 실행되는 스크립트)를 설정 또는 수정해 인스턴스 시작 시 자동 실행 명령 구성
	EnableSerialConsoleAccess	EC2 Serial Console 기능 활성화
Privilege Escalation (권한 상승)	CreateGroup	조직(그룹) 단위의 IAM 그룹 생성
	UpdateAccessKey	특정 IAM 사용자의 Access Key 상태 변경(활성/비활성) 또는 AccessKey 값 업데이트
	PutGroupPolicy	특정 IAM 그룹에 인라인 정책을 추가하거나 변경
	PutRolePolicy	특정 역할(Role)에 인라인 정책을 추가하거나 변경
	PutUserPolicy	특정 IAM 사용자에 인라인 정책 부여
	AddRoleToInstanceProfile	EC2 인스턴스 프로파일에 역할(Role) 추가
	AddUserToGroup	특정 IAM 사용자를 특정 그룹에 추가
	AttachUserPolicy	AWS-managed 또는 custom 관리형 정책을 IAM 사용자에 연결
	AttachRolePolicy	역할에 IAM 관리형 정책 연결
	AddPermission	리소스에 대해 특정 Principal(주체)이 호출할 수 있도록 정책 추가
	UpdateFunctionCode	기존 Lambda 함수의 코드 패키지 갱신
	CreatePolicy	새 IAM 정책 생성
	UpdateFunctionConfiguration	Lambda 함수의 설정 변경
	CreatePolicyVersion	기존 IAM 정책에 대해 새 버전 생성
	CreateInstanceProfile	EC2 인스턴스에 연결할 수 있는 IAM Instance 프로파일 생성
	CreateRole	새 IAM 역할 생성
	PassRole	특정 서비스(Lambda, EC2 등)에 IAM 역할을 위임해 해당 서비스가 해당 권한을 사용하도록 설정
Defense Evasion (방어 회피)	StopLogging	CloudTrail의 특정 Trail 로깅 중단
	DeleteTrail	CloudTrail의 트레일 완전 삭제
	UpdateTrail	CloudTrail의 트레일 설정 변경
	PutEventSelectors	CloudTrail의 트레일 이벤트(Data/Management) 로깅 설정
	DeleteFlowLogs	VPC Flow Logs 삭제
	DeleteDetector	GuardDuty 탐지기를 삭제해 탐지 기능 정지
	DeleteMembers	GuardDuty 멤버 계정 삭제

기능 구분	이벤트 명	이벤트 설명
Defense Evasion (방어 회피)	DeleteSnapshot	EBS 또는 RDS 스냅샷 삭제
	DeactivateMFADevice	사용자 계정의 MFA 장치 비활성화
	DeleteCertificate	IAM Server/Client 인증서(SSL/TLS) 삭제
	DeleteConfigRule	AWS Config 규칙 삭제
	DeleteAccessKey	IAM 사용자의 Access Key 삭제
	LeaveOrganization	계정이 AWS Organization (조직 관리 계정)에서 탈퇴
	DisassociateFromMasterAccount	AWS GuardDutyL나 SecurityHub에서 마스터 계정과 연결 해제
	DisassociateMembers	GuardDuty 멤버 계정과의 관계 연결 해제
	StopMonitoringMembers	GuardDuty 마스터 계정이 멤버 감시 중단
	DeleteLogGroup	CloudWatch Logs의 로그 그룹 삭제
	DetachUserPolicy	IAM 사용자에서 관리형 정책(Policy ARN) 분리
	DeletePolicy	IAM 관리형 정책 삭제
	DisableKey	KMS 키 비활성화
	ScheduleKeyDeletion	KMS 키 삭제 예약
	DeleteDBCluster	Amazon RDS 환경에서 DB 클러스터 전체 삭제
	DeleteDBClusterSnapshot	DB 클러스터의 백업 스냅샷(DB Cluster Snapshot) 삭제
	DeletePublicAccessBlock	S3 Public Access 차단 설정 제거
	RevokeSecurityGroupIngress	EC2 서비스 보안 그룹의 인바운드 규칙 삭제
	RevokeSecurityGroupEgress	EC2 서비스 보안 그룹의 아웃바운드 규칙 삭제
	PutMetricAlarm	CloudWatch 알람 생성
	DeleteAlarms	CloudWatch 알람 삭제
	StopConfigurationRecorder	AWS Config의 리소스 구성 변경 감시 중단
	PutDeliveryChannel	AWS Config 데이터 전송 채널 변경
	PutKeyPolicy	특정 KMS 키 정책 변경
	DeleteAlias	KMS, Lambda 등에서 별칭(Alias) 삭제
	CreateAlias	KMS, Lambda 등에서 별칭(Alias) 생성
	DeleteBucketTagging	버킷 태깅(식별 메타데이터) 삭제
	PutBucketLifecycle	S3 버킷의 Lifecycle 규칙을 설정(오브젝트 만료 등)해 자동 삭제, 보관 정책 변경
	ModifyNetworkInterfaceAttribute	ENI 속성 변경
Credential Access (크리덴셜 획득)	GetSecretValue	AWS Secrets Manager에 저장된 시크릿 값 조회
	PutSecretValue	AWS Secrets Manager에 저장된 시크릿 값 추가/업데이트
	GetPasswordData	EC2 인스턴스의 Windows 관리자 비밀번호를 암호화된 형태로 조회
	RequestCertificate	AWS Certificate Manager에서 새 SSL/TLS 인증서 요청

기능 구분	이벤트 명	이벤트 설명
Credential Access (크리덴셜 획득)	CreateSecret	AWS Secrets Manager에 시크릿 생성
	DeleteSecret	AWS Secrets Manager에 시크릿 삭제
	UpdateAssumeRolePolicy	IAM 역할의 신뢰 정책 변경
	ListSecrets	AWS Secrets Manager에서 저장된 시크릿의 메타데이터(이름, 설명, ARN 등) 조회
Discovery (탐색)	ListUsers	IAM 사용자 목록 조회
	ListRoles	IAM 역할 목록 조회
	ListIdentities	Cognito/AWS Identity Pool 내 사용자 목록 조회
	ListAccessKeys	IAM 사용자의 Access Key 목록 조회
	ListServiceQuotas	AWS 서비스별 한도 조회
	ListInstanceProfiles	인스턴스 프로파일(EC2 Role 연결용) 목록 조회
	ListBucket	특정 버킷 내부의 객체 목록 조회
	ListBuckets	S3 버킷 목록 조회
	ListGroups	IAM 그룹 목록 조회
	GetSendQuota	SES 메일 발송 한도 조회
	GetCallerIdentity	현재 STS 세션/계정 정보 조회
	DescribeInstances	EC2 인스턴스 세부정보 조회
	GetBucketAcl	S3 버킷의 접근제어(ACL) 조회
	GetBucketVersioning	S3 버킷의 버전관리 설정 확인
	GetAccountAuthorizationDetails	IAM 정책, 사용자, 역할 전체 세부 정보 조회
	ListObjects	S3 버킷 내 객체 목록 조회
	HeadObject	S3 객체의 데이터를 내려받지 않고 메타데이터 조회
	GetBucketPolicy	S3 버킷의 버킷 정책 조회
	DescribeDBClusters	RDS 클러스터 구성 및 엔드포인트 조회
	DescribeDBClusterSnapshots	RDS 클러스터 백업 스냅샷 목록 조회
	GetPublicAccessBlock	계정 또는 버킷의 퍼블릭 접근 차단 설정 조회
	GetObjectAcl	S3 객체 접근 제어(ACL) 조회
	GetConsoleScreenshot	EC2 인스턴스의 콘솔 스크린샷(가상 화면)을 요청해 이미지(Base64)로 받음
	BatchGetCommits	CodeCommit 저장소의 여러 커밋 정보를 조회해 코드 변경 내역 확인
	DescribeTrails	CloudTrail 트레일 설정 정보 조회
	DescribeSnapshots	EBS 스냅샷(디스크 백업)의 목록 및 메타데이터 조회
Lateral Movement (내부 이동)	AssumeRole	STS(Security Token Service)를 통해 다른 IAM 역할로 임시 자격 증명 발급

기능 구분	이벤트 명	이벤트 설명
Lateral Movement (내부 이동)	SwitchRole	AWS Management Console 내에서 사용자가 역할을 전환해 다른 Role의 권한으로 세션 시작
	CreateVpcPeeringConnection	VPC 피어링 연결 생성
	AuthorizeSecurityGroupIngress	보안 그룹에 인바운드 규칙 추가
	ReplaceRoute	VPC의 Route Table 내 경로를 수정 트래픽의 목적지 변경
	CreateGrant	KMS Key에 대한 권한을 다른 주체에 위임해 암/복호화 허용
	CreateNatGateway	프라이빗 서브넷이 외부 네트워크로 통신할 수 있도록 NAT 게이트웨이 생성
Exfiltration (유출)	GetObject	S3 객체의 실제 내용을 다운로드 또는 읽기
	CopyObject	S3 객체를 동일 버킷 내 또는 다른 버킷으로 복사
	CreateSnapShot	EBS 볼륨의 상태를 스냅샷으로 백업
	CopySnapshot	기존 스냅샷을 다른 리전/계정으로 복사
	ModifySnapshotAttributes	EBS 스냅샷의 공유 권한 수정
	ModifyImageAttribute	EC2 AMI 이미지의 공유 권한 변경
	SharedSnapshotCopyInitiated	공유된 스냅샷의 복사 작업이 시작됨을 의미
	SharedSnapshotVolumeCreated	공유된 스냅샷으로 새 볼륨 생성
	ModifyDBSnapshotAttribute	RDS DB 스냅샷 공유 권한 변경
	CreateDBSnapshot	RDS DB의 현재 상태를 스냅샷으로 백업
	PutBucketPolicy	S3 버킷 정책 수정
	PutBucketAcl	S3 버킷 접근 제어(ACL) 수정
	ModifyDBClusterSnapshotAttribute	RDS 클러스터 스냅샷의 공유 권한 수정
	RestoreDBClusterFromSnapshot	클러스터 스냅샷을 이용해 새 클러스터 복원
	PutObjectAcl	S3 객체에 대한 접근 제어 목록(ACL)을 수정
	PutPublicAccessBlock	S3 퍼블릭 접근 차단 설정(Public Access Block) 수정
	CopyDBSnapshot	DB 스냅샷을 리전 간/계정 간 복사
	RestoreDBInstanceFromDBSnapshot	기존 RDS 스냅샷으로부터 새로운 데이터베이스 인스턴스 복원
	InvokeFunction	Lambda 함수를 수동 혹은 자동(트리거)에 의해 호출
	DeleteBucketPublicAccessBlock	S3 버킷 단위 퍼블릭 접근 차단 설정 삭제
	CreateKey	KMS 마스터키 생성
	DeleteBucketEncryption	S3 버킷의 암호화 설정 삭제
	StartExportTask	CloudWatch Logs, AWS Config 등 로그 데이터를 외부 대상(S3 등)으로 내보내는 Export 작업 시작
영향 (Impact)	PutBucketVersioning	S3 버킷에 객체 버전 관리 기능을 활성화 또는 중지

기능 구분	이벤트 명	이벤트 설명
영향 (Impact)	RunInstances	새 EC2 인스턴스 실행
	DeleteAccountPublicAccessBlock	AWS 계정 전체의 S3 퍼블릭 접근 차단 정책 삭제
	DeleteObject	S3 버킷 내 단일 객체 삭제
	DeleteObjects	S3 버킷 내 다수의 객체(최대 1000개) 일괄 삭제
	DeleteDBInstance	RDS 인스턴스 삭제
	ModifyDBInstance	RDS 인스턴스의 설정 변경
	PutObject	S3 객체 업로드(생성 또는 덮어쓰기)
	DeleteBucket	S3 버킷 자체 삭제
	DeleteBucketLifecycle	S3 버킷의 수명 주기(Lifecycle rule) 제거
	DeleteDBSnapshot	RDS 스냅샷 삭제
	DeleteBucketReplication	버킷 간 데이터 복제 설정 삭제
	DisableKey	KMS 키 비활성화
	TerminateInstances	AWS EC2 인스턴스 종료
	DeleteVolume	AWS EBS 볼륨 삭제
	DeleteRecoveryPoint	AWS Backup의 복원 지점 삭제
	EncryptVolume	EBS 볼륨을 암호화하거나 암호화 구성 변경
	PutBucketEncryption	S3 버킷의 서버 측 암호화 설정을 추가 또는 변경
	PutBucketReplication	S3 버킷 간 복제 규칙을 설정해 데이터를 자동 전송하도록 구성
	AttachInternetGateway	Internet Gateway를 VPC에 연결해 외부 인터넷 통신 설정
	DeleteSecurityGroup	지정된 보안 그룹을 삭제해 네트워크 접근 제어 구성 제거

S3 Access Logs의 각 전술 별 핵심 오퍼레이션은 다음과 같다.

[표 146] DFIR CheatSheet에 작성된 전술별 핵심 오퍼레이션 – S3 Access Log

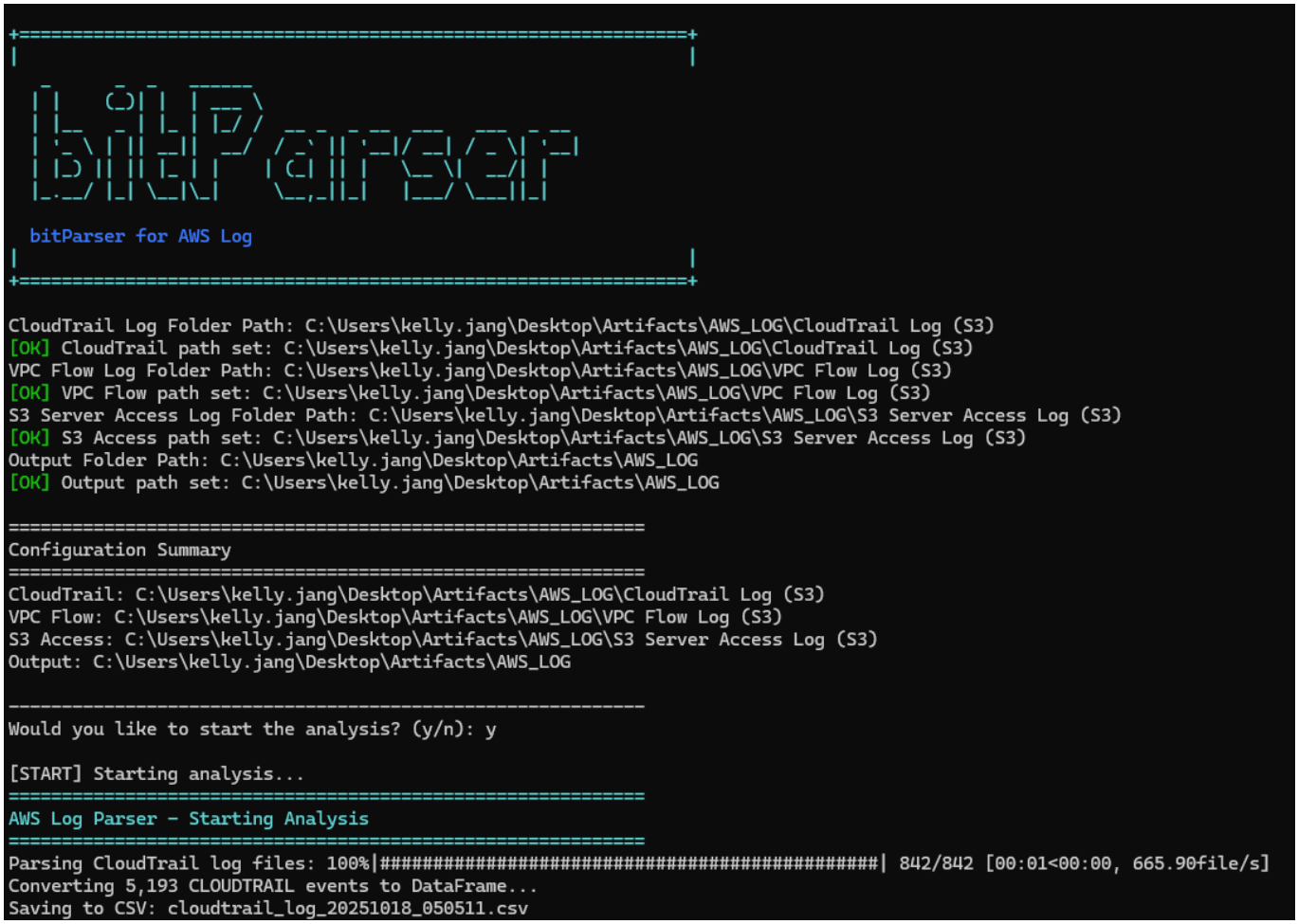
기능 구분	오퍼레이션 명	이벤트 설명
Reconnaissance (정찰)	REST.HEAD.BUCKET	버킷의 존재 여부와 접근 권한을 확인하기 위해 HTTP HEAD 요청
	REST.OPTIONS.PREFLIGHT	브라우저 또는 클라이언트가 CORS 허용 범위를 확인하기 위해 HTTP OPTIONS 요청
Privilege Escalation (권한 상승)	REST.PUT.ACL	객체 또는 S3 버킷의 접근 제어(ACL) 수정
Persistence (지속)	REST.PUT.OBJECT	S3 객체(파일)를 업로드하거나 기존 객체를 덮어씀
	REST.PUT.BUCKETNOTIFICATION	S3 버킷의 이벤트 알림 설정을 생성하거나 갱신
	REST.GET.BUCKETNOTIFICATION	버킷에 설정된 이벤트 알림 구성 조회
Discovery (탐색)	REST.GET.BUCKET	S3 버킷 목록 조회
	REST.GET.ACL	객체 또는 S3 버킷의 ACL 조회
	REST.GET.BUCKET.LOCATION	S3 버킷 리전 정보 조회
	REST.GET.ENCRYPTION	버킷의 기본 암호화(SSE) 설정 조회
	REST.GET.BUCKETACL	S3 버킷 ACL 조회
	REST.GET.BUCKETPOLICY	S3 버킷 정책 내용 조회
	REST.GET.SERVICE	계정(서비스) 수준에서 존재하는 모든 버킷 목록 조회
	REST.LIST.MULTIPART.UPLOADS	특정 버킷에서 현재 진행 중인 멀티파트 업로드 목록 조회
	REST.HEAD.OBJECT	객체의 존재/메타(크기, ETag, Content-Type 등) 확인
	REST.GET.OBJECT.VERSION	버전 관리가 활성화된 오브젝트의 특정 버전 조회
	REST.LIST.OBJECT.VERSIONS	버전 관리가 활성화된 버킷의 오브젝트 별 모든 버전 목록 조회
Defense Evasion (방어 회피)	REST.DELETE.BUCKETPUBLICACCESSBLOCK	조직/계정 수준의 S3 PublicAccessBlock 설정 제거
	DELETE.OBJECT.VERSION	버전 관리가 활성화된 객체의 특정 버전 삭제
	REST.GET.OBJECT.TAGGING	객체의 태그(메타데이터) 조회
	REST.PUT.OBJECT.TAGGING	객체의 태그(메타데이터) 수정
	REST.PUT.BUCKETVERSIONING	버킷의 버전관리(Versioning)를 활성화하거나 비활성
	REST.GET.BUCKETVERSIONING	버킷의 버전관리 상태(활성화 여부) 조회
	REST.GET.BUCKETLIFECYCLE	버킷에 설정된 lifecycle 규칙 조회
	REST.PUT.OBJECT.RETENTION	특정 객체에 대해 보존기간(Retention) 설정
	REST.PUT.OBJECT.LEGALHOLD	객체에 법적 보류(Legal Hold)를 설정/해제
	REST.PUT.BUCKETLOGGING	버킷의 서버 액세스 로깅(로그 전송 대상 지정)을 활성화/변경
	REST.GET.BUCKETLOGGING	버킷에 설정된 액세스 로깅 구성 조회

기능 구분	오퍼레이션 명	이벤트 설명
Defense Evasion (방어 회피)	REST.DELETE.BUCKETLOGGING	버킷의 액세스 로깅 비활성화
	REST.DELETE.BUCKETNOTIFICATION	버킷에 설정된 이벤트 알림 삭제
	REST.DELETE.BUCKETREPLICATION	버킷의 복제 설정 제거
Exfiltration (유출)	REST.GET.OBJECT	S3 객체(파일) 다운로드
	REST.COPY.OBJECT	S3 객체를 같은/다른 버킷(또는 계정)으로 복사
	REST.PUT.BUCKETACL	버킷 전체의 ACL 변경
	REST.PUT.BUCKETPOLICY	버킷 정책을 생성 및 수정해 접근 범위 제어
	REST.PUT.BUCKET	새로운 S3 버킷 생성
	REST.INITIATE.MULTIPART.UPLOAD	멀티파트 업로드를 시작하고 uploadId(세션) 발급
	REST.UPLOAD.PART	멀티파트 업로드의 개별 파트 업로드
	REST.COMPLETE.MULTIPART.UPLOAD	업로드된 파트들을 하나의 객체로 합쳐 업로드 완료
	REST.ABORT.MULTIPART.UPLOAD	진행 중인 멀티파트 업로드를 중단하고 관련 파트 정리
	REST.PUT.BUCKETREPLICATION	버킷 복제 규칙을 설정해 객체를 자동으로 다른 버킷으로 복제하도록 구성
	REST.GET.BUCKETREPLICATION	버킷에 설정된 복제 설정 조회
	REST.RESTORE.OBJECT	Glacier/Archive에 보관된 오브젝트를 임시 복원해 접근 설정
	REST.GET.OBJECT.TORRENT	(구식) S3 객체를 비트토렌트 방식으로 내려받기 위한 요청
Impact (영향)	REST.DELETE.OBJECT	특정 객체 삭제
	REST.DELETE.BUCKET	버킷 자체 삭제

5.3. AWS DFIR 로그 분석 도구 개발

AWS 클라우드 환경에서의 사고 탐지 및 조사를 지원하기 위해 AWS DFIR 로그 분석 도구(bitParser for AWS Log)를 개발했다. 본 도구는 CloudTrail Log, VPC Flow Logs, S3 Access Log를 수집 및 파싱해 실제 공격에서 빈번하게 관찰되는 로그 패턴을 자동으로 탐지하고, 분석가가 우선적으로 검토해야 할 핵심 분석 포인트를 제시하는 것을 목표로 한다.

또한, CloudTrail Log와 S3 Access Log에 대해서는 앞서 제시한 DFIR CheatSheet에 정리된 주요 이벤트의 탐지 여부를 병행 분석함으로써 전술 기반의 연계·상관 관점에서 더 높은 가시성을 제공한다.



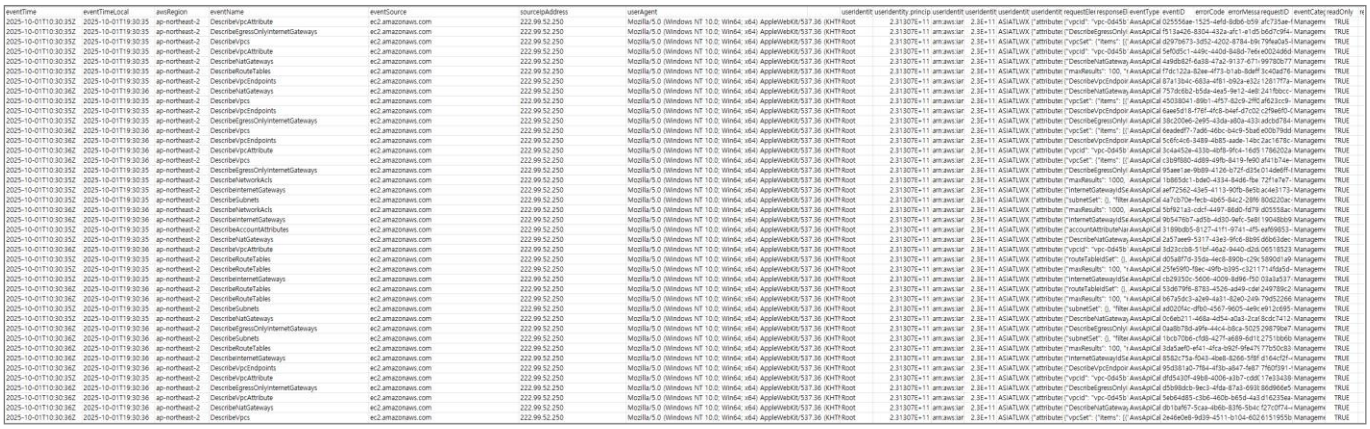
[그림 8] bitParser for AWS Log 실행 화면 예시

도구의 주요 기능은 다음과 같다.

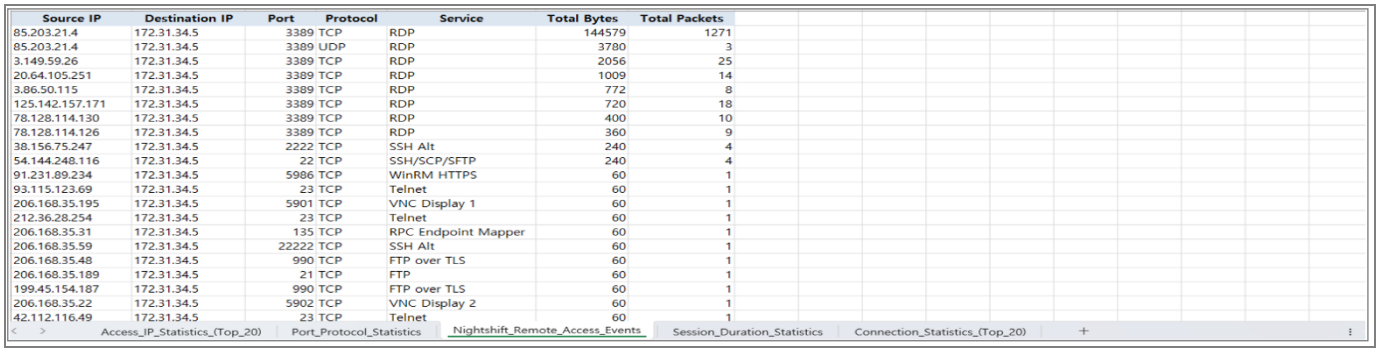
[표 147] bitParser for AWS Log 주요 기능

기능 구분	설명
통합 로그 파싱	• CloudTrail Log: AWS API 호출 이벤트 파싱 (JSON 평탄화) • VPC Flow Logs: 네트워크 트래픽 로그 파싱 • S3 Access Log: S3 버킷 접근 로그 파싱
CloudTrail Log 분석	• 이벤트 호출 상위 20개 IP의 접근 이력 통계 (최초/마지막 접근일시, 접근 횟수) • 철야 시간 (22:00 – 06:00)에 이벤트를 호출한 IP 통계 • 철야 시간 (22:00 – 06:00)에 발생한 이벤트 빈도 분석 • 전체 이벤트 기준 통계 및 빈도 분석 • User-Agent 상세 분류 및 통계 (AWS CLI, SDK, 브라우저 등) • 계정 생성 이력 분석 • AWS 관리 콘솔 로그인 이력 분석 • 실패한 인증/권한 통계 • AWS 리전 통계 및 빈도 분석 • MITRE ATT&CK 전술별 이벤트 통계 (DFIR Cheat Sheet에 기재된 175개 이벤트) • MITRE ATT&CK 전술별 이벤트에 매핑된 모든 CloudTrail Log 상세 로그
VPC Flow Logs 분석	• 전송량 기준 상위 20개 IP (srcIP, dstIP) • 상위 20개 포트의 네트워크 트래픽 통계 • 철야 시간 (22:00 – 06:00)에 발생한 원격 접근 이벤트 (RDP, SSH 등) • 상위 20개 세션 지속 시간 통계 • 상위 20개 총 바이트 기준 네트워크 트래픽 통계
S3 Access Log 분석	• 상위 20개 요청자 ARN 및 IP 통계 • 오퍼레이션 발생 빈도 및 통계 (오퍼레이션, S3 버킷, Prefix, 발생 횟수) • User-Agent 상세 분류 및 통계 (AWS CLI, SDK, 브라우저 등) • MITRE ATT&CK 전술별 오퍼레이션 통계 (DFIR Cheat Sheet에 기재된 47개 오퍼레이션) • MITRE ATT&CK 전술별 오퍼레이션에 매핑된 모든 S3 Access Log 상세 로그
다중 출력 형식	• 파싱된 원본 로그 (출력 폴더의 Parse_Logs 폴더 내 csv 파일로 저장) • 로그 분석 결과 시트 (출력 폴더의 Analysis_Log 폴더 내 xlsx 파일로 저장) • 로그 분석 결과 기반 요약 보고서 (출력 폴더의 Report 폴더 내 html 파일로 저장)
로그 시간 변환	• CloudTrail 로그에 기입된 AWS 리전을 기준으로 변환된 로그 시간 추가 제공

도구 결과 파일은 다음과 같은 형식으로 구성되어 있다.



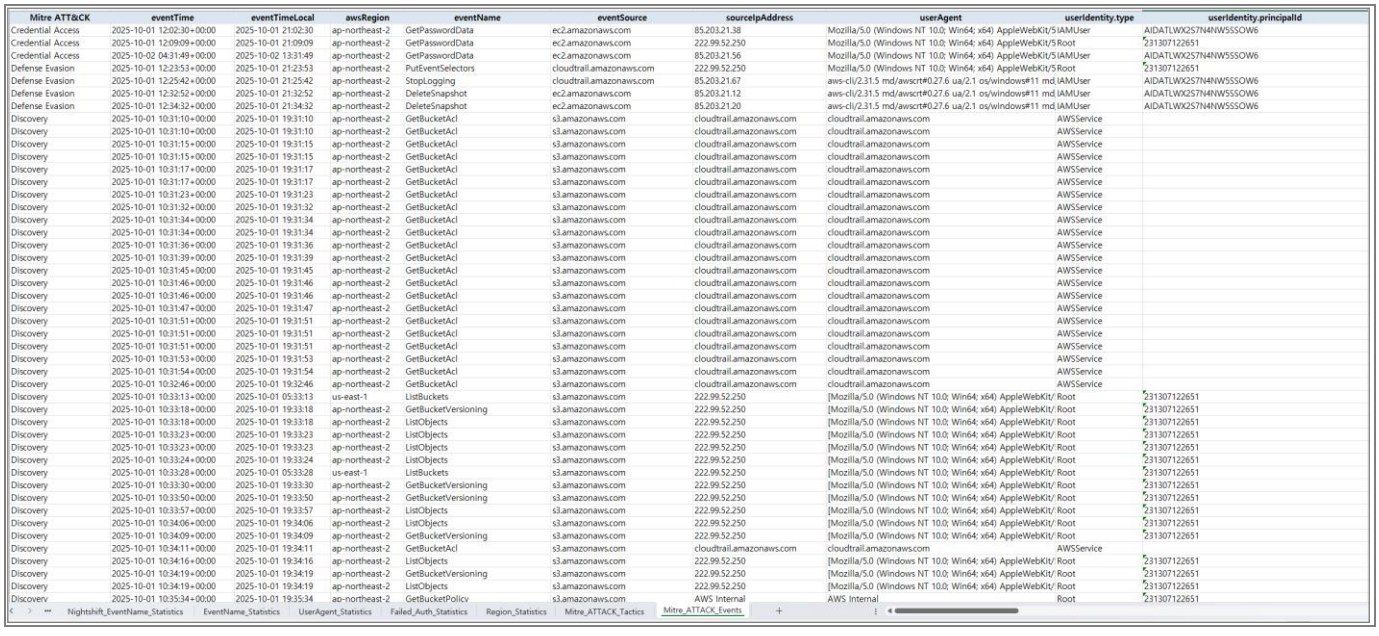
[그림 9] 원본 로그 파싱 결과 예시 화면 (CloudTrail Log)



Source IP	Destination IP	Port	Protocol	Service	Total Bytes	Total Packets
85.203.21.4	172.31.34.5	3389	TCP	RDP	144579	1271
85.203.21.4	172.31.34.5	3389	UDP	RDP	3780	3
3.149.59.26	172.31.34.5	3389	TCP	RDP	2056	25
20.64.105.251	172.31.34.5	3389	TCP	RDP	1009	14
3.86.50.115	172.31.34.5	3389	TCP	RDP	772	8
125.142.157.171	172.31.34.5	3389	TCP	RDP	720	18
78.128.114.130	172.31.34.5	3389	TCP	RDP	400	10
78.128.114.126	172.31.34.5	3389	TCP	RDP	360	9
38.156.75.247	172.31.34.5	2222	TCP	SSH Alt	240	4
54.144.248.116	172.31.34.5	22	TCP	SSH/SCP/SFTP	240	4
91.231.89.234	172.31.34.5	5986	TCP	WinRM HTTPS	60	1
93.115.123.69	172.31.34.5	23	TCP	Telnet	60	1
206.168.35.195	172.31.34.5	5901	TCP	VNC Display 1	60	1
212.36.28.254	172.31.34.5	23	TCP	Telnet	60	1
206.168.35.31	172.31.34.5	135	TCP	RPC Endpoint Mapper	60	1
206.168.35.59	172.31.34.5	22222	TCP	SSH Alt	60	1
206.168.35.48	172.31.34.5	990	TCP	FTP over TLS	60	1
206.168.35.189	172.31.34.5	21	TCP	FTP	60	1
199.45.154.187	172.31.34.5	990	TCP	FTP over TLS	60	1
206.168.35.22	172.31.34.5	5902	TCP	VNC Display 2	60	1
42.112.116.49	172.31.34.5	23	TCP	Telnet	60	1

Access_IP_Statistics_(Top_20) | Port_Protocol_Statistics | Nightshift_Remote_Access_Events | Session_Duration_Statistics | Connection_Statistics_(Top_20)

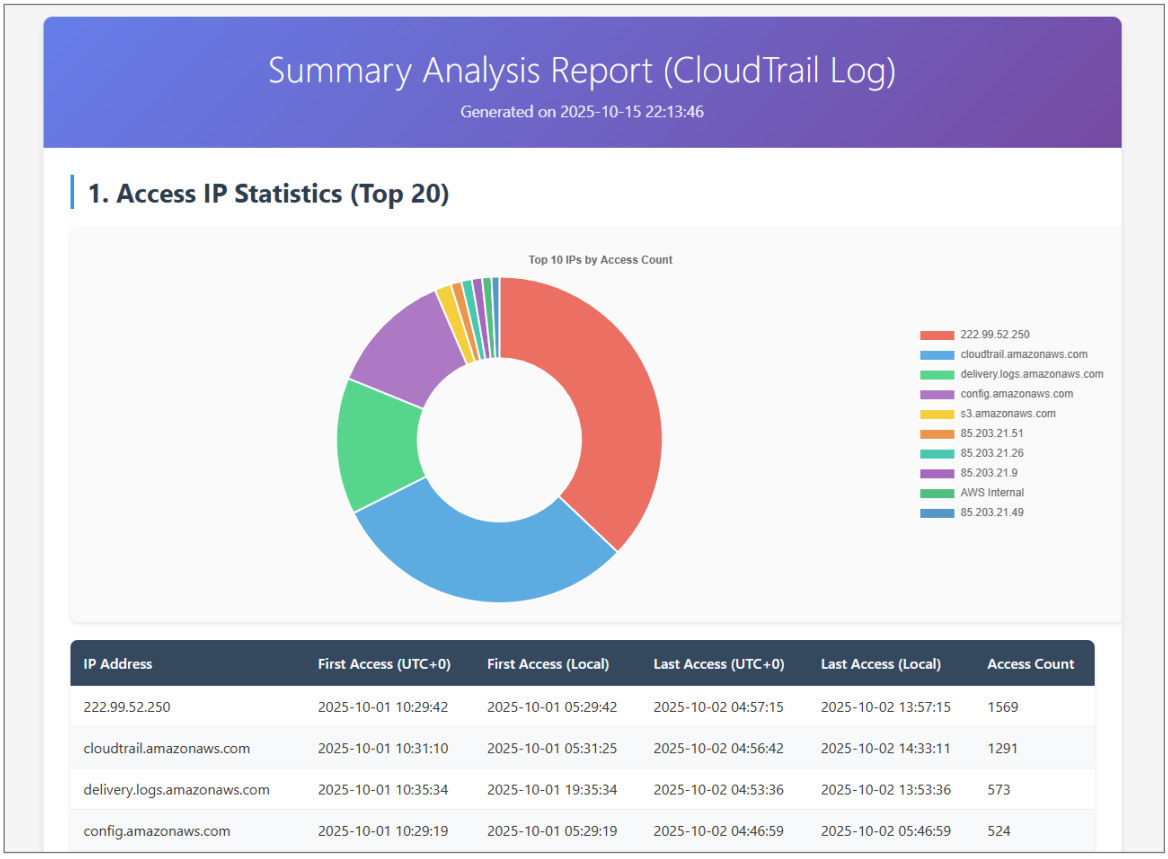
[그림 10] 로그 분석 결과 시트 예시 화면 (VPC Flow Logs – 철야 시간에 발생한 원격 접근 이벤트)



Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.principalId
Credential Access	2025-10-01 12:02:30+00:00	2025-10-01 21:02:30	ap-northeast-2	GetPasswordData	ec2.amazonaws.com	85.203.21.38	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/5	IAMUser	AIDATLWX2S7N4NW5SSOW6
Credential Access	2025-10-01 12:09:09+00:00	2025-10-01 21:09:09	ap-northeast-2	GetPasswordData	ec2.amazonaws.com	222.99.52.250	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/5	Root	'231307122651
Credential Access	2025-10-02 04:31:49+00:00	2025-10-02 13:31:49	ap-northeast-2	GetPasswordData	ec2.amazonaws.com	85.203.21.56	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/5	IAMUser	AIDATLWX2S7N4NW5SSOW6
Defense Evasion	2025-10-01 12:23:53+00:00	2025-10-01 21:23:53	ap-northeast-2	PutEventSelectors	cloudtrail.amazonaws.com	222.99.52.250	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/5	Root	'231307122651
Defense Evasion	2025-10-01 12:25:42+00:00	2025-10-01 21:25:42	ap-northeast-2	StopLogging	cloudtrail.amazonaws.com	85.203.21.67	aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md	IAMUser	AIDATLWX2S7N4NW5SSOW6
Defense Evasion	2025-10-01 12:32:52+00:00	2025-10-01 21:32:52	ap-northeast-2	DeleteSnapshot	ec2.amazonaws.com	85.203.21.12	aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md	IAMUser	AIDATLWX2S7N4NW5SSOW6
Defense Evasion	2025-10-01 12:34:32+00:00	2025-10-01 21:34:32	ap-northeast-2	DeleteSnapshot	ec2.amazonaws.com	85.203.21.20	aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md	IAMUser	AIDATLWX2S7N4NW5SSOW6
Discovery	2025-10-01 10:31:10+00:00	2025-10-01 19:31:10	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:10+00:00	2025-10-01 19:31:10	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:15+00:00	2025-10-01 19:31:15	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:15+00:00	2025-10-01 19:31:15	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:17+00:00	2025-10-01 19:31:17	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:17+00:00	2025-10-01 19:31:17	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:23+00:00	2025-10-01 19:31:23	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:32+00:00	2025-10-01 19:31:32	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:34+00:00	2025-10-01 19:31:34	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:34+00:00	2025-10-01 19:31:34	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:36+00:00	2025-10-01 19:31:36	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:39+00:00	2025-10-01 19:31:39	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:45+00:00	2025-10-01 19:31:45	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:46+00:00	2025-10-01 19:31:46	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:46+00:00	2025-10-01 19:31:46	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:47+00:00	2025-10-01 19:31:47	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:51+00:00	2025-10-01 19:31:51	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:51+00:00	2025-10-01 19:31:51	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:51+00:00	2025-10-01 19:31:51	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:53+00:00	2025-10-01 19:31:53	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:31:54+00:00	2025-10-01 19:31:54	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:32:46+00:00	2025-10-01 19:32:46	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:33:13+00:00	2025-10-01 19:33:13	us-east-1	ListBuckets	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:33:18+00:00	2025-10-01 19:33:18	ap-northeast-2	GetBucketVersioning	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:33:18+00:00	2025-10-01 19:33:18	ap-northeast-2	ListObjects	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:33:23+00:00	2025-10-01 19:33:23	ap-northeast-2	ListObjects	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:33:23+00:00	2025-10-01 19:33:23	ap-northeast-2	ListObjects	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:33:24+00:00	2025-10-01 19:33:24	ap-northeast-2	ListObjects	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:33:28+00:00	2025-10-01 05:33:28	us-east-1	ListBuckets	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:33:30+00:00	2025-10-01 19:33:30	ap-northeast-2	GetBucketVersioning	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:33:50+00:00	2025-10-01 19:33:50	ap-northeast-2	GetBucketVersioning	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:33:57+00:00	2025-10-01 19:33:57	ap-northeast-2	ListObjects	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:34:06+00:00	2025-10-01 19:34:06	ap-northeast-2	ListObjects	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:34:09+00:00	2025-10-01 19:34:09	ap-northeast-2	GetBucketVersioning	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:34:11+00:00	2025-10-01 19:34:11	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.com	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 10:34:16+00:00	2025-10-01 19:34:16	ap-northeast-2	ListObjects	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:34:19+00:00	2025-10-01 19:34:19	ap-northeast-2	GetBucketVersioning	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:34:19+00:00	2025-10-01 19:34:19	ap-northeast-2	ListObjects	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/	Root	'231307122651
Discovery	2025-10-01 10:35:34+00:00	2025-10-01 19:35:34	ap-northeast-2	GetBucketPolicy	s3.amazonaws.com	AWS Internal	AWS Internal	Root	'231307122651

Nightshift_EventName_Statistics | EventName_Statistics | UserAgent_Statistics | Failed_Auth_Statistics | Region_Statistics | Mitre_ATTACK_Tactics | Mitre_ATTACK_Events

[그림 11] 로그 분석 결과 시트 예시 화면 (CloudTrail Log – DFIR Cheat Sheet 기반 이벤트 탐지 로그)



Summary Analysis Report (CloudTrail Log)

Generated on 2025-10-15 22:13:46

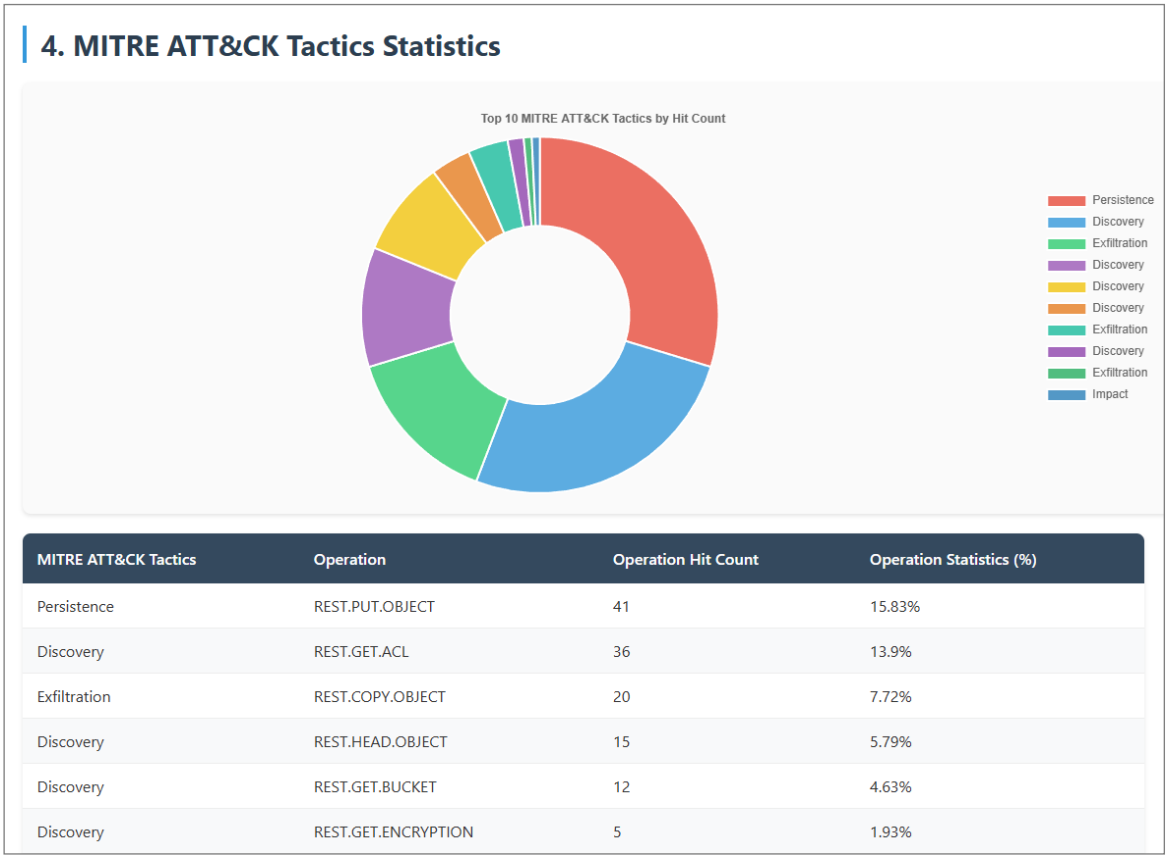
1. Access IP Statistics (Top 20)

Top 10 IPs by Access Count

222.99.52.250
cloudtrail.amazonaws.com
delivery.logs.amazonaws.com
config.amazonaws.com
s3.amazonaws.com
85.203.21.51
85.203.21.26
85.203.21.9
AWS Internal
85.203.21.49

IP Address	First Access (UTC+0)	First Access (Local)	Last Access (UTC+0)	Last Access (Local)	Access Count
222.99.52.250	2025-10-01 10:29:42	2025-10-01 05:29:42	2025-10-02 04:57:15	2025-10-02 13:57:15	1569
cloudtrail.amazonaws.com	2025-10-01 10:31:10	2025-10-01 05:31:25	2025-10-02 04:56:42	2025-10-02 14:33:11	1291
delivery.logs.amazonaws.com	2025-10-01 10:35:34	2025-10-01 19:35:34	2025-10-02 04:53:36	2025-10-02 13:53:36	573
config.amazonaws.com	2025-10-01 10:29:19	2025-10-01 05:29:19	2025-10-02 04:46:59	2025-10-02 05:46:59	524

[그림 12] 분석 결과 요약 보고서 화면 예시 (CloudTrail Log)



4. MITRE ATT&CK Tactics Statistics

Top 10 MITRE ATT&CK Tactics by Hit Count

Persistence
Discovery
Exfiltration
Discovery
Discovery
Discovery
Exfiltration
Discovery
Exfiltration
Impact

MITRE ATT&CK Tactics	Operation	Operation Hit Count	Operation Statistics (%)
Persistence	REST.PUT.OBJECT	41	15.83%
Discovery	REST.GET.ACL	36	13.9%
Exfiltration	REST.COPY.OBJECT	20	7.72%
Discovery	REST.HEAD.OBJECT	15	5.79%
Discovery	REST.GET.BUCKET	12	4.63%
Discovery	REST.GET.ENCRYPTION	5	1.93%

[그림 13] 분석 결과 요약 보고서 화면 예시 (S3 Access Log)

6. 시나리오 기반 실증 분석

선행 연구에서 분석한 공격 전술 등을 참고해 AWS 클라우드 환경에서 발생할 수 있는 랜섬웨어 사고 시나리오를 제작했으며, AWS 클라우드 환경에서 랜섬웨어가 발생했을 때 주요 로그에서 어떻게 분석할 수 있는지 살펴본다. 또한, 연구를 통해 개발한 AWS DFIR 로그 분석 도구(bitParser for AWS Log, 이하 bitParser)를 활용하면 어떤 정보를 얻을 수 있는지 분석을 수행했으며, 그 결과는 다음과 같다.

6장에서 다루는 내용은 다음과 같다.

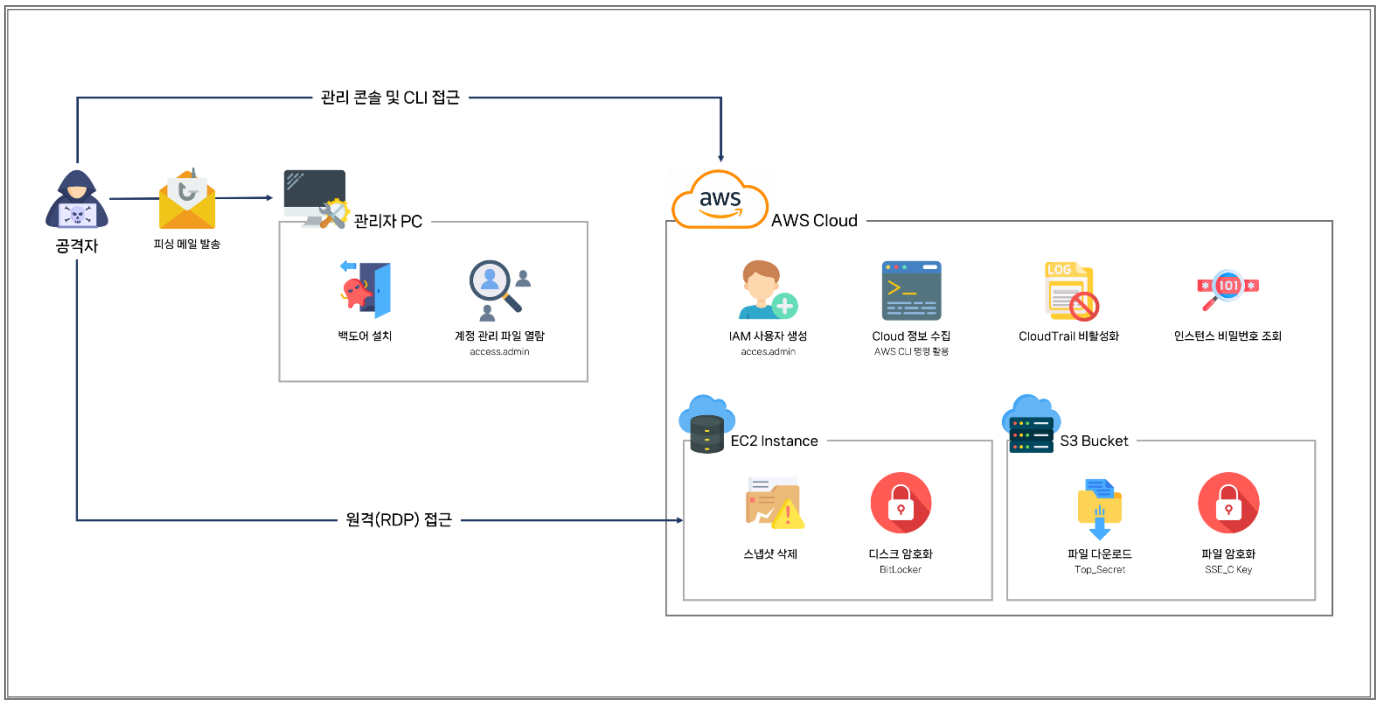
[표 148] 주요 연구 내용 – 시나리오 기반 실증 분석

번호	소제목	주요 내용
1	공격 시나리오 개요	선행 연구의 전술 분석을 기반으로 구성한 AWS 랜섬웨어 시나리오 개요 및 공격 전술 설명
2	시나리오 분석 결과	랜섬웨어 시나리오르 기반으로 주요 로그 분석 및 bitParser 도구를 활용한 검증 결과 제시

6.1. 공격 시나리오 개요

공격자는 피싱 메일을 통해 백도어에 감염된 관리자 PC로 침투해 계정 관리 파일 내 존재하는 AWS IAM 계정 크리덴셜과 키페어 파일(PEM)을 획득했다. 획득한 AWS IAM 계정 크리덴셜로 AWS Cloud 콘솔에 접근해 IAM 사용자를 생성했으며, 생성한 IAM 사용자를 통해 AWS CLI 명령으로 Cloud 정보를 수집했다. 또한, 사전에 획득한 키페어 파일(PEM)을 통해 EC2 Instance 비밀번호를 획득했다.

이후, 공격자는 EC2 Instance 스냅샷을 삭제한 뒤 원격(RDP) 접근해 디스크를 암호화했으며, AWS CLI 명령을 통해 S3 Bucket의 파일을 다운로드 받고 내부 파일들을 공격자 SSE_C Key로 암호화했다.



[그림 14] 공격 시나리오 개요도

공격자가 수행한 공격 행위를 전술로 구분하면 다음과 같다.

[표 149] MITRE ATT&CK 전술별 공격행위 구분

전술	공격 기법 설명
Initial Access (최초 침투)	• 피싱 메일을 통한 백도어 설치 후 침투
Discovery & Collection (정보 수집)	• 계정 관리 파일 열람 • 키페어 파일(PEM) 탈취 • AWS 클라우드 콘솔 접속 후 EC2, S3 등 확인 • AWS CLI를 통한 정보 수집
Persistence (지속)	• 공격 전용 IAM 사용자 생성 (access.admin) • AWS CLI 계정 연동
Lateral Movement (내부 이동)	• AWS EC2 인스턴스 원격 접근 (RDP)
Defense Evasion (방어 회피)	• AWS CloudTrail 비활성화
Impact (영향)	• AWS EC2 인스턴스 스냅샷 데이터 삭제 • AWS EC2 인스턴스 디스크 암호화 • AWS S3 버킷 데이터 다운로드 • AWS S3 버킷 데이터 암호화

6.2. 시나리오 분석 결과

AWS 클라우드 환경에서 랜섬웨어가 발생했을 때 어떻게 분석할 수 있는지 살펴보기 위해 주요 로그(CloudTrail, VPC Flow Logs, S3 Access Log)를 수집해 분석했으며, 다음과 같은 이벤트들을 통해 위협을 식별할 수 있었다.

1) 관리자 PC에서 획득한 크리덴셜을 활용한 AWS Console 로그인

CloudTrail에서 관리자의 IAM 사용자(access.admin)가 MFA 없이 85.203.21.5(싱가포르) IP에서 Chrome 브라우저를 통해 AWS Console에 성공적으로 로그인한 것을 확인할 수 있었다.


```
2025-10-01T11:32:37.275Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7NSJAGFFTJW","arn":"arn:aws:iam::2313
07122651:user/access.admin","accountId":"231307122651","userName":"access.admin"},"eventTime":"2025-10-01T11:30:00Z","
eventSource":"signin.amazonaws.com","eventName":"ConsoleLogin","awsRegion":"ap-southeast-2","sourceIPAddress":"85.203.
21.5","userAgent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0
Safari/537.36","requestParameters":null,"responseElements":{"ConsoleLogin":"Success"},"additionalEventData":{"LoginTo"
:"https://console.aws.amazon.com/console/home?hashArgs=%23&isauthcode=true&nc2=h_si&src=header-signin&state=hashArgsFr
omTB_ap-southeast-2_56595ecf92c30140","MobileVersion":"No","MFAUsed":"No"},"eventID":"77d3119a-5db1-44d4-bacc-d0bfb77c
d3c4","readOnly":false,"eventType":"AwsConsoleSignIn","managementEvent":true,"recipientAccountId":"231307122651","even
tCategory":"Management","tlsDetails":{"tlsVersion":"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHos
tHeader":"ap-southeast-2.signin.aws.amazon.com"}}
```


[그림 15] CloudTrail에서 확인되는 AWS Console 로그인 이벤트

[표 150] AWS Console 로그인 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS Console 로그인	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/access.admin - userName: access.admin • eventTime: 2025-10-01T11:30:00Z • eventSource: signin.amazonaws.com • eventName: ConsoleLogin • awsRegion: ap-southeast-2 • sourceIPAddress: 85.203.21.49 • userAgent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36 • responseElements - ConsoleLogin: Success • additionalEventData - MobileVersion: No - MFAUsed: No

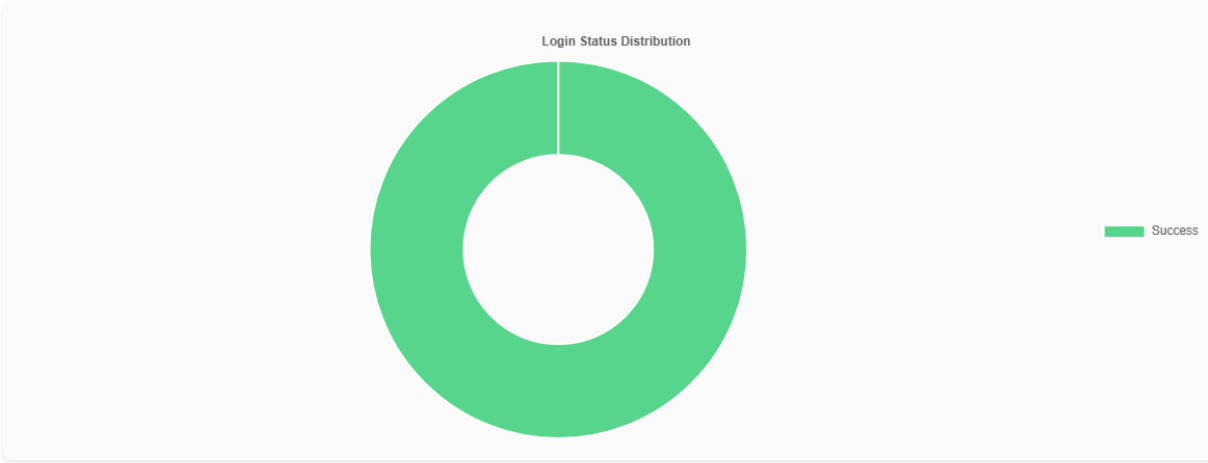
bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Initial Access	2025-10-01 11:24:06+00:00	2025-10-01 06:24:06	us-east-1	ConsoleLogin	signin.amazonaws.com	222.99.52.250	Mozilla/5.0 (Windows NT 10.0; Win64;	Root	
Initial Access	2025-10-01 11:25:30+00:00	2025-10-01 12:25:30	eu-north-1	ConsoleLogin	signin.amazonaws.com	222.99.52.250	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	access.admin
Initial Access	2025-10-01 11:30:00+00:00	2025-10-01 21:30:00	ap-southeast-2	ConsoleLogin	signin.amazonaws.com	85.203.21.5	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	access.admin
Initial Access	2025-10-01 11:40:31+00:00	2025-10-01 21:40:31	ap-southeast-2	ConsoleLogin	signin.amazonaws.com	85.203.21.67	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	acces.admin
Initial Access	2025-10-01 12:06:22+00:00	2025-10-01 07:06:22	us-east-1	ConsoleLogin	signin.amazonaws.com	222.99.52.250	Mozilla/5.0 (Windows NT 10.0; Win64;	Root	
Initial Access	2025-10-01 12:22:42+00:00	2025-10-01 07:22:42	us-east-1	ConsoleLogin	signin.amazonaws.com	222.99.52.250	Mozilla/5.0 (Windows NT 10.0; Win64;	Root	
Initial Access	2025-10-02 04:26:26+00:00	2025-10-01 23:26:26	us-east-1	ConsoleLogin	signin.amazonaws.com	222.99.52.250	Mozilla/5.0 (Windows NT 10.0; Win64;	Root	
Initial Access	2025-10-02 04:26:36+00:00	2025-10-01 23:26:36	us-east-1	ConsoleLogin	signin.amazonaws.com	222.99.52.250	Mozilla/5.0 (Windows NT 10.0; Win64;	Root	
Initial Access	2025-10-02 04:31:00+00:00	2025-10-02 14:31:00	ap-southeast-2	ConsoleLogin	signin.amazonaws.com	85.203.21.53	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	acces.admin

[그림 16] bitParser 분석 결과 파일에서 확인한 AWS Console 로그인 이벤트

또한, bitParser 분석 결과 요약 보고서 파일의 'ConsoleLogin Statistics' 화면에서 기존에 접근하지 않았던 IP나 외부 IP가 존재하는지 확인해 식별할 수 있다. 분석 시트에서는 콘솔 로그인 전체 이력 분석이 가능하나, 보고서 상에서는 최근 30일에 해당하는 이력만 확인 가능하다.

8. ConsoleLogin Statistics (Last 30 Days)



User	Event Time (UTC)	Event Time (Local)	Source IP	Login Status	User Agent
acces.admin	2025-10-02 04:31:00	2025-10-02 14:31:00	85.203.21.53	Success	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36
acces.admin	2025-10-01 11:40:31	2025-10-01 21:40:31	85.203.21.67	Success	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36
access.admin	2025-10-01 11:30:00	2025-10-01 21:30:00	85.203.21.5	Success	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36
access.admin	2025-10-01 11:25:30	2025-10-01 12:25:30	222.99.52.250	Success	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36

[그림 17] bitParser 분석 결과 요약 보고서 파일에서 확인한 'ConsoleLogin Statistics' 화면

2) 공격전용 AWS IAM 사용자 생성

CloudTrail에서 IAM 사용자(access.admin)가 85.203.21.49(싱가포르) IP에서 Chrome 브라우저를 통해 새로운 IAM 사용자(acces.admin)를 생성하는 것을 확인할 수 있었다.


```
2025-10-01T11:36:11.121Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7NSJAGFFTJW","arn":"arn:aws:iam::231307
122651:user/access.admin","accountId":"231307122651","accessKeyId":"ASIATLWX2S7N3TMEGKOW","userName":"access.admin","ses
sionContext":{"attributes":{"creationDate":"2025-10-01T11:30:00Z","mfaAuthenticated":"false"}}},"eventTime":"2025-10-01T
11:34:02Z","eventSource":"iam.amazonaws.com","eventName":"CreateUser","awsRegion":"us-east-1","sourceIPAddress":"85.203.
21.49","userAgent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0
Safari/537.36","requestParameters":{"userName":"acces.admin"},"responseElements":{"user":{"path":"/","userName":"acces.a
dmin","userId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307122651:user/acces.admin","createDate":"Oct 1, 2025,
11:34:02 AM"}},"requestID":"80f8ebe6-6fe0-4d57-b629-db2740af547d","eventID":"4f293c91-9768-47f8-8ba8-d21041eb1cd9","readO
nly":false,"eventType":"AwsApiCall","managementEvent":true,"recipientAccountId":"231307122651","eventCategory":"Manageme
nt","tlsDetails":{"tlsVersion":"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHostHeader":"iam.amazonaw
s.com"},"sessionCredentialFromConsole":"true"}
```


[그림 18] CloudTrail에서 확인되는 AWS IAM 사용자 생성 이벤트

[표 151] AWS IAM 사용자 생성 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS IAM 사용자 생성	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/access.admin - userName: access.admin • eventTime: 2025-10-01T11:34:02Z • eventSource: iam.amazonaws.com • eventName: CreateUser • awsRegion: us-east-1 • sourceIPAddress: 85.203.21.49 • userAgent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36 • responseElements - user:userName: acces.admin - user:userId: AIDATLWX2S7N4NW5SSOW6 - user:createDate: Oct 1, 2025, 11:34:02 AM

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Persistence	2025-10-01 11:34:02+00:00	2025-10-01 06:34:02	us-east-1	CreateUser	iam.amazonaws.com	85.203.21.49	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	access.admin

[그림 19] bitParser 분석 결과 파일에서 확인한 AWS IAM 사용자 생성 이벤트

3) 공격전용 AWS IAM 사용자 Console 액세스 활성화

CloudTrail에서 IAM 사용자(access.admin)가 85.203.21.26(싱가포르) IP에서 Chrome 브라우저를 통해 IAM 사용자(acces.admin)의 Console 액세스를 활성화하기 위해 LoginProfile을 생성하는 것을 확인할 수 있었다.


```
2025-10-01T11:36:11.122Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7NSJAGFFTJW","arn":"arn:aws:iam::231307
122651:user/access.admin","accountId":"231307122651","accessKeyId":"ASIATLWX2S7N3TMEGKOW","userName":"access.admin","ses
sionContext":{"attributes":{"creationDate":"2025-10-01T11:30:00Z","mfaAuthenticated":"false"}}},"eventTime":"2025-10-01T
11:35:31Z","eventSource":"iam.amazonaws.com","eventName":"CreateLoginProfile","awsRegion":"us-east-1","sourceIPAddress":
"85.203.21.26","userAgent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/140.0.0.0
Safari/537.36","requestParameters":{"userName":"acces.admin","passwordResetRequired":false},"responseElements":{"loginPr
ofile":{"userName":"acces.admin","createDate":"Oct 1, 2025,
11:35:31 AM","passwordResetRequired":false}},"requestID":"1811ea73-1b88-4fd8-af58-ec0778bf78dd","eventID":"01ed1d85-404e-
48cd-be40-81bc8093ba7f","readOnly":false,"eventType":"AwsApiCall","managementEvent":true,"recipientAccountId":"231307122
651","eventCategory":"Management","tlsDetails":{"tlsVersion":"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientPro
videdHostHeader":"iam.amazonaws.com"},"sessionCredentialFromConsole":"true"}
```


[그림 20] CloudTrail에서 확인되는 AWS IAM 사용자 Console 액세스 활성화 이벤트

[표 152] AWS IAM 사용자 Console 액세스 활성화 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS IAM 사용자 Console 액세스 활성화	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/access.admin - userName: access.admin • sessionContext - creationDate: 2025-10-01T11:30:00Z - mfaAuthenticated: false • eventTime: 2025-10-01T11:35:31Z • eventSource: iam.amazonaws.com • eventName: CreateLoginProfile • awsRegion: us-east-1 • sourceIPAddress: 85.203.21.26 • userAgent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36 • responseElements - loginprofile:userName: acces.admin - loginprofile:createDate: Oct 1, 2025, 11:35:31 AM - loginprofile:passwordResetRequired: false

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	useridentity.type	useridentity.userName
Persistence	2025-10-01 11:35:31+00:00	2025-10-01 06:35:31	us-east-1	CreateLoginProfile	iam.amazonaws.com	85.203.21.26	Mozilla/5.0 (Windows NT 10.0; Win64; IAMUser		access.admin

[그림 21] bitParser 분석 결과 파일에서 확인한 AWS IAM 사용자 Console 액세스 활성화 이벤트

4) 공격전용 AWS IAM 사용자 Access Key 생성

CloudTrail에서 IAM 사용자(access.admin)가 85.203.21.26(싱가포르) IP에서 Chrome 브라우저를 통해 IAM 사용자(acces.admin)의 Access Key를 생성하는 것을 확인할 수 있었다.


```
2025-10-01T11:38:31.369Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7NSJAGFFTJW","arn":"arn:aws:iam::231307
122651:user/access.admin","accountId":"231307122651","accessKeyId":"ASIATLWX2S7N3TMEGKOW","userName":"access.admin","ses
sionContext":{"attributes":{"creationDate":"2025-10-01T11:30:00Z","mfaAuthenticated":"false"}}},"eventTime":"2025-10-01T
11:36:30Z","eventSource":"iam.amazonaws.com","eventName":"CreateAccessKey","awsRegion":"us-east-1","sourceIPAddress":"85
.203.21.26","userAgent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/140.0.0.0
Safari/537.36","requestParameters":{"userName":"acces.admin"},"responseElements":{"accessKey":{"userName":"acces.admin",
"accessKeyId":"AKIATLWX2S7NSYHHZ2BL","status":"Active","createDate":"Oct 1, 2025,
11:36:30 AM"}},"requestID":"66c2abe9-d3f8-43cf-958f-8b91dd5467be","eventID":"7a361c64-f972-4a02-ab8c-a9047a180b11","readO
nly":false,"eventType":"AwsApiCall","managementEvent":true,"recipientAccountId":"231307122651","eventCategory":"Manageme
nt","tlsDetails":{"tlsVersion":"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHostHeader":"iam.amazonaw
s.com"},"sessionCredentialFromConsole":"true"}
```


[그림 22] CloudTrail에서 확인되는 AWS IAM 사용자 Access Key 생성 이벤트

[표 153] AWS IAM 사용자 Access Key 생성 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS IAM 사용자 Access Key 생성	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/access.admin - userName: access.admin • sessionContext - creationDate: 2025-10-01T11:30:00Z - mfaAuthenticated: false • eventTime: 2025-10-01T11:36:30Z • eventSource: iam.amazonaws.com • eventName: CreateAccessKey • awsRegion: us-east-1 • sourceIPAddress: 85.203.21.26 • userAgent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36 • responseElements - userName: acces.admin - accessKeyId: AKIATLWX2S7NSYHHZ2BL - status: Active - createDate: Oct 1, 2025, 11:36:30 AM

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	useridentity.type	useridentity.userName
Persistence	2025-10-01 11:36:30+00:00	2025-10-01 06:36:30	us-east-1	CreateAccessKey	iam.amazonaws.com	85.203.21.26	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	access.admin

[그림 23] bitParser 분석 결과 파일에서 확인한 AWS IAM 사용자 Access Key 생성 이벤트

5) AWS Console을 통해 EC2 페이지 접근

CloudTrail에서 IAM 사용자(acces.admin)가 85.203.21.24(싱가포르) IP에서 Chrome 브라우저를 통해 EC2 인스턴스 정보를 출력하는 것을 확인할 수 있었다.


```
2025-10-01T11:42:48.617Z
{"eventVersion":"1.10","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"ASIATLWX2S7NSPQL5JUB","userName":"acces.admin","sessi
onContext":{"attributes":{"creationDate":"2025-10-01T11:40:32Z","mfaAuthenticated":"false"}}},"eventTime":"2025-10-01T11
:41:37Z","eventSource":"ec2.amazonaws.com","eventName":"DescribeInstances","awsRegion":"ap-southeast-2","sourceIPAddress
":"85.203.21.24","userAgent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/140.0.0.0
Safari/537.36","requestParameters":{"maxResults":100,"instancesSet":{},"filterSet":{}},"responseElements":null,"requestI
D":"5bf4ca1a-bcee-4b4a-96d2-ed759294b7aa","eventID":"5a83ab6f-910a-46b8-bd7f-e0e7af1f7079","readOnly":true,"eventType":"
AwsApiCall","managementEvent":true,"recipientAccountId":"231307122651","eventCategory":"Management","tlsDetails":{"tlsVe
rsion":"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHostHeader":"ec2.ap-southeast-2.amazonaws.com"},"
sessionCredentialFromConsole":"true"}
```


[그림 24] CloudTrail에서 확인되는 AWS Console을 통한 EC2 페이지 접근 이벤트

[표 154] AWS Console을 통한 EC2 페이지 접근 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS Console을 통한 EC2 페이지 접근	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • sessionContext - creationDate: 2025-10-01T11:40:32Z - mfaAuthenticated: false • eventTime: 2025-10-01T11:41:37Z • eventSource: ec2.amazonaws.com • eventName: DescribeInstances • awsRegion: ap-southeast-2 • sourceIPAddress: 85.203.21.24 • userAgent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Discovery	2025-10-01 11:26:00+00:00	2025-10-01 20:26:00	ap-northeast-2	DescribeInstances	ec2.amazonaws.com	222.99.52.250	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	access.admin
Discovery	2025-10-01 11:26:03+00:00	2025-10-01 20:26:03	ap-northeast-2	DescribeInstances	ec2.amazonaws.com	222.99.52.250	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	access.admin
Discovery	2025-10-01 11:41:37+00:00	2025-10-01 21:41:37	ap-southeast-2	DescribeInstances	ec2.amazonaws.com	85.203.21.24	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	acces.admin
Discovery	2025-10-01 11:42:37+00:00	2025-10-01 20:42:37	ap-northeast-2	DescribeInstances	ec2.amazonaws.com	85.203.21.8	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	acces.admin

[그림 25] bitParser 분석 결과에서 확인한 AWS Console을 통한 EC2 페이지 접근 이벤트

6) AWS Console을 통해 S3 페이지 접근

CloudTrail과 S3 Access Log에서 IAM 사용자(acces.admin)가 85.203.21.53(싱가포르) IP에서 Chrome 브라우저를 통해 S3 버킷 목록을 출력하는 것을 확인할 수 있었다.


```
2025-10-01T11:45:01.907Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"ASIATLWX2S7NVAUPWEIJ","userName":"acces.admin","sessi
onContext":{"attributes":{"creationDate":"2025-10-01T11:40:32Z","mfaAuthenticated":"false"}}},"eventTime":"2025-10-01T11
:43:36Z","eventSource":"s3.amazonaws.com","eventName":"ListBuckets","awsRegion":"us-east-1","sourceIPAddress":"85.203.21
.48","userAgent":"[Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0
Safari/537.36]","requestParameters":{"Host":"s3.us-east-1.amazonaws.com"},"responseElements":null,"additionalEventData":
{"SignatureVersion":"SigV4","CipherSuite":"TLS_AES_128_GCM_SHA256","bytesTransferredIn":0,"AuthenticationMethod":"AuthHe
ader","x-amz-id-2":"MRAUFnaL2XY/KAbAH46Ki/v1Yjb6XLnzaPiH6Y9hVVctiCmeEXoGE2Ujiv6JZ9BT91JC/nqkzMg=","bytesTransferredOut":
461},"requestID":"1WGSGAB2AKD1RHGA","eventID":"22d89195-333f-4671-9126-b20eca006c03","readOnly":true,"eventType":"AwsApi
Call","managementEvent":true,"recipientAccountId":"231307122651","eventCategory":"Management","tlsDetails":{"tlsVersion"
:"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHostHeader":"s3.us-east-1.amazonaws.com"}}
```


[그림 26] CloudTrail에서 확인되는 AWS Console을 통한 S3 페이지 접근 이벤트


```
5043e0a2f5c9e33bf501c24bcde8204bf2bf4f8e4be8f9b60878bcce1c01d406 plainbit-s3 [01/Oct/2025:11:43:37 +0000] 85.203.21.53
- EZ8Z6TM31Q4DW84F REST.OPTIONS.PREFLIGHT - "OPTIONS /plainbit-s3 HTTP/1.1" 200 - - - 3 -
"https://ap-northeast-2.console.aws.amazon.com/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/140.0.0.0 Safari/537.36" -
UahR/WnaqAxuGrKODA6zrf0Lf/Rql494Ro8CnNRXTZv9gKszcx/Im+KLcjsg/+zYSiEd4VKdN0Q= - TLS_AES_128_GCM_SHA256 -
s3.ap-northeast-2.amazonaws.com TLSv1.3 - -
5043e0a2f5c9e33bf501c24bcde8204bf2bf4f8e4be8f9b60878bcce1c01d406 plainbit-s3 [01/Oct/2025:11:43:37 +0000] 85.203.21.53
arn:aws:iam::231307122651:user/acces.admin EZ8MXQ252W3QQVFW REST.HEAD.BUCKET - "HEAD /plainbit-s3 HTTP/1.1" 200 - - -
22 21 "https://ap-northeast-2.console.aws.amazon.com/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36" -
WdMzcy6NqftnF7t8Ce9rUlDRZ80gRSyssK4qsLxNeM9Zz293tN1ARmTjFXV1xWjTowg6QWeiaYk= SigV4 TLS_AES_128_GCM_SHA256 AuthHeader
s3.ap-northeast-2.amazonaws.com TLSv1.3 - -
```

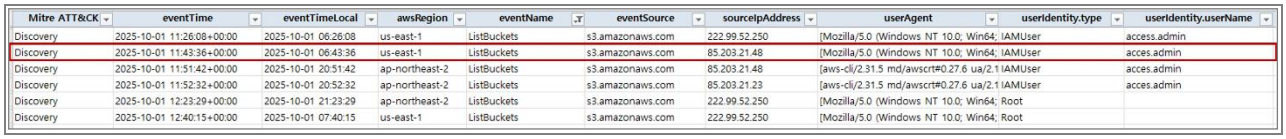

[그림 27] S3 Access Log에서 확인되는 AWS Console을 통한 S3 페이지 접근 이벤트

[표 155] AWS Console을 통한 S3 페이지 접근 이벤트의 주요 필드 내용

구분	주요 필드 내용
(CloudTrail) AWS Console을 통한 S3 페이지 접근	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • sessionContext - creationDate: 2025-10-01T11:40:32Z - mfaAuthenticated: false • eventTime: 2025-10-01T11:43:36Z • eventSource: s3.amazonaws.com • eventName: ListBuckets • awsRegion: us-east-1 • sourceIPAddress: 85.203.21.48 • userAgent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36

구분	주요 필드 내용
(S3 Access Log) AWS Console을 통한 S3 페이지 접근	• eventTime: [01/Oct/2025:11:43:37 +0000] • sourceIPAddress: 85.203.21.53 • task: REST.OPTIONS.PREFLIGHT or REST.HEAD.BUCKET • StatusCode: 200 • User-Agent: https://ap-northeast-2.console.aws.amazon.com/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.



Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Discovery	2025-10-01 11:26:08+00:00	2025-10-01 06:26:08	us-east-1	ListBuckets	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	access.admin
Discovery	2025-10-01 11:43:36+00:00	2025-10-01 06:43:36	us-east-1	ListBuckets	s3.amazonaws.com	85.203.21.48	[Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	acces.admin
Discovery	2025-10-01 11:51:42+00:00	2025-10-01 20:51:42	ap-northeast-2	ListBuckets	s3.amazonaws.com	85.203.21.48	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin
Discovery	2025-10-01 11:52:32+00:00	2025-10-01 20:52:32	ap-northeast-2	ListBuckets	s3.amazonaws.com	85.203.21.23	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin
Discovery	2025-10-01 12:23:29+00:00	2025-10-01 21:23:29	ap-northeast-2	ListBuckets	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64;	Root	
Discovery	2025-10-01 12:40:15+00:00	2025-10-01 07:40:15	us-east-1	ListBuckets	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64;	Root	

[그림 28] bitParser 분석 결과 파일에서 확인한 AWS Console을 통한 S3 페이지 접근 이벤트

7) AWS CLI 명령을 통해 IAM 사용자 목록 수집

CloudTrail에서 IAM 사용자(acces.admin)가 85.203.21.67(싱가포르) IP에서 AWS CLI로 'iam list-users' 명령을 실행해 IAM 사용자 목록을 수집하는 것을 확인할 수 있었다.


```
2025-10-01T11:48:12.075Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-01T11:46:03Z","eventSource":"iam.amazonaws.com","eventName":"ListUsers","awsRegion":"us-east-1","sourceI
PAddress":"85.203.21.67","userAgent":"aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64
lang/python#3.13.7 md/pyimpl#CPython m/C,E,Z,b,n cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#iam.list-users","requestParameters":null,"responseElements":null,"requestID":"e3277e78-4b30-4196-ad1b-96cd208
bb090","eventID":"fbc72f15-6980-4f46-8776-2dc44709d9b3","readOnly":true,"eventType":"AwsApiCall","managementEvent":true,
"recipientAccountId":"231307122651","eventCategory":"Management","tlsDetails":{"tlsVersion":"TLSv1.3","cipherSuite":"TLS
_AES_128_GCM_SHA256","clientProvidedHostHeader":"iam.amazonaws.com"}}
```


[그림 29] CloudTrail에서 확인되는 AWS CLI 명령을 통한 IAM 사용자 목록 수집 이벤트

[표 156] AWS CLI 명령을 통한 IAM 사용자 목록 수집 이벤트의 주요 필드 내용

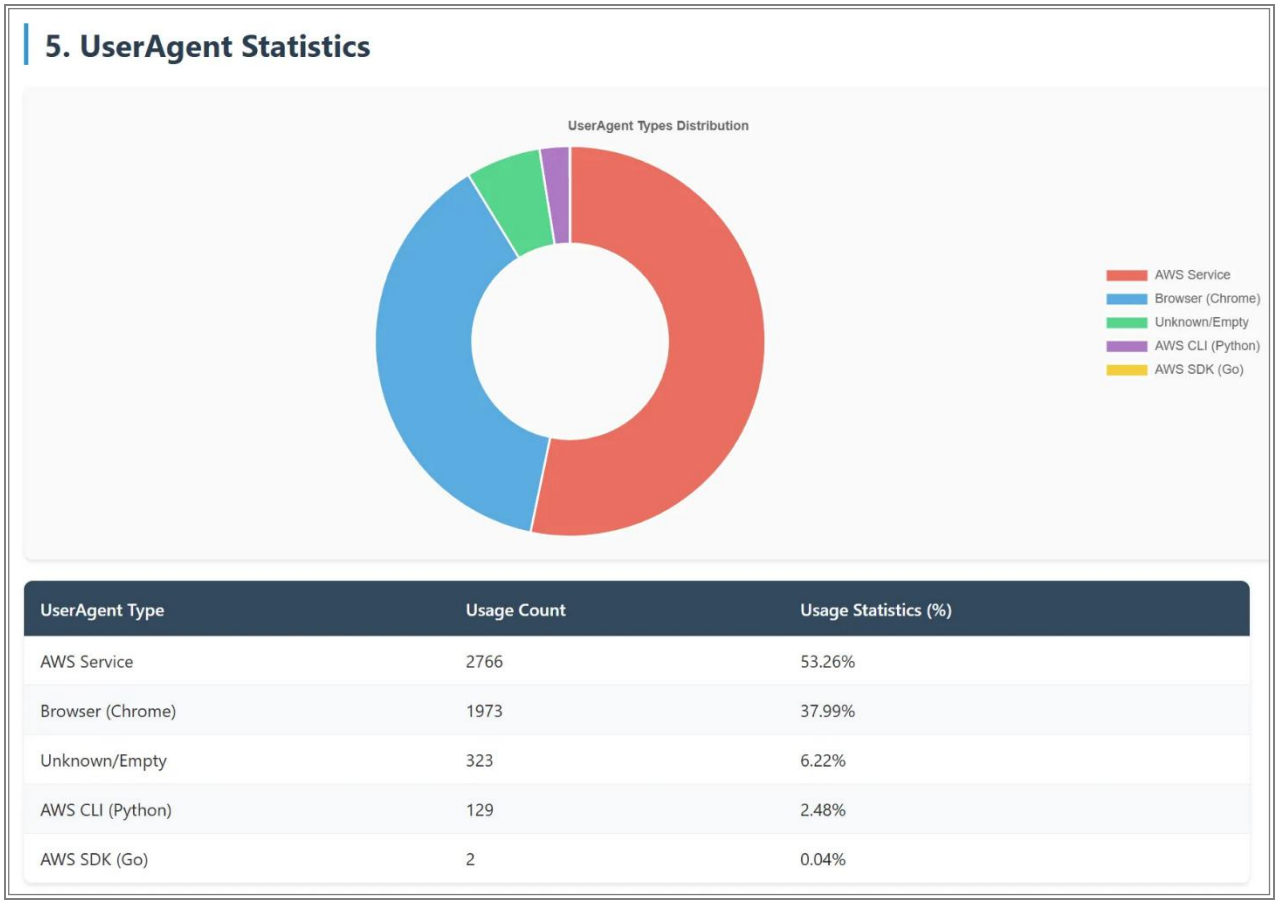
구분	주요 필드 내용
AWS CLI 명령을 통한 IAM 사용자 목록 수집	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T11:46:03Z • eventSource: iam.amazonaws.com • eventName: ListUsers • awsRegion: us-east-1 • sourceIPAddress: 85.203.21.67 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#Cpython m/C,E,Z,b,n cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#iam.list-users

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Discovery	2025-10-01 11:30:48+00:00	2025-10-01 06:30:48	us-east-1	ListUsers	iam.amazonaws.com	85.203.21.51	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	access.admin
Discovery	2025-10-01 11:34:05+00:00	2025-10-01 06:34:05	us-east-1	ListUsers	iam.amazonaws.com	85.203.21.38	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	access.admin
Discovery	2025-10-01 11:46:03+00:00	2025-10-01 06:46:03	us-east-1	ListUsers	iam.amazonaws.com	85.203.21.67	aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin

[그림 30] bitParser 분석 결과 파일에서 확인한 IAM 사용자 목록 수집 이벤트

또한, bitParser 분석 결과 요약 보고서 파일의 'UserAgent Statistics' 화면에서 기존에 사용하지 않던 UserAgent가 존재하는지 확인해 식별할 수 있다.



UserAgent Type	Usage Count	Usage Statistics (%)
AWS Service	2766	53.26%
Browser (Chrome)	1973	37.99%
Unknown/Empty	323	6.22%
AWS CLI (Python)	129	2.48%
AWS SDK (Go)	2	0.04%

[그림 31] bitParser 분석 결과 요약 보고서 파일에서 확인한 'UserAgent Statistics' 화면

8) AWS CLI 명령을 통해 IAM 역할 목록 수집

CloudTrail에서 IAM 사용자(acces.admin)가 85.203.21.49(싱가포르) IP에서 AWS CLI로 'iam list-roles' 명령을 실행해 IAM 역할 목록을 수집하는 것을 확인할 수 있었다.


```
2025-10-01T11:48:12.075Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-01T11:47:22Z","eventSource":"iam.amazonaws.com","eventName":"ListRoles","awsRegion":"us-east-1","sourceI
PAddress":"85.203.21.49","userAgent":"aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64
lang/python#3.13.7 md/pyimpl#CPython m/Z,C,E,b,n cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#iam.list-roles","requestParameters":null,"responseElements":null,"requestID":"969c0128-1a91-41eb-950d-9ebb6a5
5a8e3","eventID":"b6f4ad5c-2539-42ce-b378-74a5de0931ab","readOnly":true,"eventType":"AwsApiCall","managementEvent":true,
"recipientAccountId":"231307122651","eventCategory":"Management","tlsDetails":{"tlsVersion":"TLSv1.3","cipherSuite":"TLS
_AES_128_GCM_SHA256","clientProvidedHostHeader":"iam.amazonaws.com"}}
```


[그림 32] CloudTrail에서 확인되는 AWS CLI 명령을 통한 IAM 역할 목록 수집 이벤트

[표 157] AWS CLI 명령을 통한 IAM 역할 목록 수집 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS CLI 명령을 통한 IAM 역할 목록 수집	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T11:47:22Z • eventSource: iam.amazonaws.com • eventName: ListRoles • awsRegion: us-east-1 • sourceIPAddress: 85.203.21.49 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/Z,C,E,b,n cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#iam.list-roles

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Discovery	2025-10-01 11:47:22+00:00	2025-10-01 06:47:22	us-east-1	ListRoles	iam.amazonaws.com	85.203.21.49	aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin

[그림 33] bitParser 분석 결과 파일에서 확인한 IAM 역할 목록 수집 이벤트

9) AWS CLI 명령을 통해 IAM 정보 수집

CloudTrail에서 IAM 사용자(acces.admin)가 85.203.21.25(싱가포르) IP에서 AWS CLI로 'iam get-account-authorization-details' 명령을 실행해 IAM 정보를 수집하는 것을 확인할 수 있었다.


```
2025-10-01T11:50:32.385Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-01T11:48:24Z","eventSource":"iam.amazonaws.com","eventName":"GetAccountAuthorizationDetails","awsRegion"
:"us-east-1","sourceIPAddress":"85.203.21.25","userAgent":"aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11
md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/C,Z,b,n,E cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#iam.get-account-authorization-details","requestParameters":null,"responseElements":null,"requestID":"0a6c0fa2
-3432-41c6-b9ac-95f8fdc3e6a3","eventID":"2a90bc9e-a776-440d-ae38-9c4331c6e982","readOnly":true,"eventType":"AwsApiCall",
"managementEvent":true,"recipientAccountId":"231307122651","eventCategory":"Management","tlsDetails":{"tlsVersion":"TLSv
1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHostHeader":"iam.amazonaws.com"}}
```


[그림 34] CloudTrail에서 확인되는 AWS CLI 명령을 통한 IAM 정보 수집 이벤트

[표 158] AWS CLI 명령을 통한 IAM 정보 수집 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS CLI 명령을 통한 IAM 정보 수집	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T11:48:24Z • eventSource: iam.amazonaws.com • eventName: GetAccountAuthorizationDetails • awsRegion: us-east-1 • sourceIPAddress: 85.203.21.25 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/C,Z,b,n,E cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#iam.get-account-authorization-details

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Discovery	2025-10-01 11:48:24+00:00	2025-10-01 06:48:24	us-east-1	GetAccountAuthorizationDetails	iam.amazonaws.com	85.203.21.25	aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin
Discovery	2025-10-01 11:48:26+00:00	2025-10-01 06:48:26	us-east-1	GetAccountAuthorizationDetails	iam.amazonaws.com	85.203.21.25	aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin
Discovery	2025-10-01 11:48:28+00:00	2025-10-01 06:48:28	us-east-1	GetAccountAuthorizationDetails	iam.amazonaws.com	85.203.21.25	aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin

[그림 35] bitParser 분석 결과 파일에서 확인한 IAM 정보 수집

10) AWS CLI 명령을 통해 EC2 인스턴스 목록 수집

CloudTrail에서 IAM 사용자(acces.admin)가 85.203.21.38(싱가포르) IP에서 AWS CLI로 'ec2 describe-instances' 명령을 사용해 EC2 인스턴스 목록을 수집하는 것을 확인할 수 있었다.


```
2025-10-01T11:51:42.663Z
{"eventVersion":"1.10","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-01T11:49:32Z","eventSource":"ec2.amazonaws.com","eventName":"DescribeInstances","awsRegion":"ap-northeas
t-2","sourceIPAddress":"85.203.21.38","userAgent":"aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64
lang/python#3.13.7 md/pyimpl#CPython m/b,E,C,Z,n cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#ec2.describe-instances","requestParameters":{"instancesSet":{},"filterSet":{}},"responseElements":null,"reque
stID":"0c7242e9-58ce-456a-a551-a7f1d92b9e30","eventID":"ed51f182-e6f9-4607-b879-ae7c2c3de6c5","readOnly":true,"eventType
":"AwsApiCall","managementEvent":true,"recipientAccountId":"231307122651","eventCategory":"Management","tlsDetails":{"tl
sVersion":"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHostHeader":"ec2.ap-northeast-2.amazonaws.com"
}}
```


[그림 36] CloudTrail에서 확인되는 AWS CLI 명령을 통한 EC2 인스턴스 목록 수집 이벤트

[표 159] AWS CLI 명령을 통한 EC2 인스턴스 목록 수집 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS CLI 명령을 통한 EC2 인스턴스 목록 수집	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T11:49:32Z • eventSource: ec2.amazonaws.com • eventName: DescribeInstances • awsRegion: ap-northeast-2 • sourceIPAddress: 85.203.21.38 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#Cpython m/b,E,C,Z,n cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#ec2.describe-instances

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Discovery	2025-10-01 11:42:37+00:00	2025-10-01 20:42:37	ap-northeast-2	DescribeInstances	ec2.amazonaws.com	85.203.21.8	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	acces.admin
Discovery	2025-10-01 11:49:32+00:00	2025-10-01 20:49:32	ap-northeast-2	DescribeInstances	ec2.amazonaws.com	85.203.21.38	aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin
Discovery	2025-10-01 11:58:54+00:00	2025-10-01 20:58:54	ap-northeast-2	DescribeInstances	ec2.amazonaws.com	85.203.21.49	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	acces.admin
Discovery	2025-10-01 11:59:09+00:00	2025-10-01 20:59:09	ap-northeast-2	DescribeInstances	ec2.amazonaws.com	85.203.21.42	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	acces.admin
Discovery	2025-10-01 12:00:12+00:00	2025-10-01 21:00:12	ap-northeast-2	DescribeInstances	ec2.amazonaws.com	85.203.21.38	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	acces.admin

[그림 37] bitParser 분석 결과 파일에서 확인한 EC2 인스턴스 목록 수집

11) AWS CLI 명령을 통해 S3 버킷 목록 수집

CloudTrail에서 IAM 사용자(acces.admin)가 85.203.21.48(싱가포르) IP에서 AWS CLI로 's3 ls' 명령을 사용해 S3 버킷 목록을 수집하는 것을 확인할 수 있었다.


```
2025-10-01T11:53:53.120Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-01T11:51:42Z","eventSource":"s3.amazonaws.com","eventName":"ListBuckets","awsRegion":"ap-northeast-2","s
ourceIPAddress":"85.203.21.48","userAgent":"[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64
lang/python#3.13.7 md/pyimpl#CPython m/E,C,Z,n,b cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#s3.ls]","requestParameters":{"Host":"s3.ap-northeast-2.amazonaws.com"},"responseElements":null,"additionalEve
ntData":{"SignatureVersion":"SigV4","CipherSuite":"TLS_AES_128_GCM_SHA256","bytesTransferredIn":0,"AuthenticationMethod"
:"AuthHeader","x-amz-id-2":"ATMitRJyphp9RlmeZX+TgffGanAHHOOWBTBluibZKR9xEcFe8yDFX7pUTno9GO1KLBrWnGOKo3E=","bytesTransfer
redOut":461},"requestID":"NNZVK0J7R4A05S2J","eventID":"b845a6c1-fe2a-4e44-bbe7-e11ce1df97de","readOnly":true,"eventType"
:"AwsApiCall","managementEvent":true,"recipientAccountId":"231307122651","eventCategory":"Management","tlsDetails":{"tls
Version":"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHostHeader":"s3.ap-northeast-2.amazonaws.com"}}
```


[그림 38] CloudTrail에서 확인되는 S3 버킷 목록 수집 이벤트

[표 160] AWS CLI 명령을 통한 S3 버킷 목록 수집 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS CLI 명령을 통한 S3 버킷 목록 수집	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T11:51:42Z • eventSource: s3.amazonaws.com • eventName: ListBuckets • awsRegion: ap-northeast-2 • sourceIPAddress: 85.203.21.48 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/E,C,Z,n,b cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#s3.ls

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Discovery	2025-10-01 11:26:08+00:00	2025-10-01 06:26:08	us-east-1	ListBuckets	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	access.admin
Discovery	2025-10-01 11:43:36+00:00	2025-10-01 06:43:36	us-east-1	ListBuckets	s3.amazonaws.com	85.203.21.48	[Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	acces.admin
Discovery	2025-10-01 11:51:42+00:00	2025-10-01 20:51:42	ap-northeast-2	ListBuckets	s3.amazonaws.com	85.203.21.48	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin
Discovery	2025-10-01 11:52:32+00:00	2025-10-01 20:52:32	ap-northeast-2	ListBuckets	s3.amazonaws.com	85.203.21.23	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin
Discovery	2025-10-01 12:23:29+00:00	2025-10-01 21:23:29	ap-northeast-2	ListBuckets	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64;	Root	
Discovery	2025-10-01 12:40:15+00:00	2025-10-01 07:40:15	us-east-1	ListBuckets	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64;	Root	

[그림 39] bitParser 분석 결과 파일에서 확인한 S3 버킷 목록 수집 이벤트

12) AWS CLI 명령을 통해 SecretsManager 정보 수집

CloudTrail에서 IAM 사용자(acces.admin)가 85.203.21.56(싱가포르) IP에서 AWS CLI로 명령을 사용해 SecretsManager 정보를 수집하는 것을 확인할 수 있었다.


```
2025-10-01T11:53:53.121Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-01T11:53:32Z","eventSource":"secretsmanager.amazonaws.com","eventName":"ListSecrets","awsRegion":"ap-nor
theast-2","sourceIPAddress":"85.203.21.56","userAgent":"aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11
md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/b,E,n,C,Z cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#secretsmanager.list-secrets","requestParameters":null,"responseElements":null,"requestID":"ab028708-1fdc-41f8
-9599-2310d0ad2bad","eventID":"c8e79443-beb4-49a5-b585-b5b7d872114a","readOnly":true,"eventType":"AwsApiCall","managemen
tEvent":true,"recipientAccountId":"231307122651","eventCategory":"Management","tlsDetails":{"tlsVersion":"TLSv1.3","ciph
erSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHostHeader":"secretsmanager.ap-northeast-2.amazonaws.com"}}
```


[그림 40] CloudTrail에서 확인되는 SecretsManager 정보 수집 이벤트

[표 161] AWS CLI 명령을 통한 SecretsManager 정보 수집 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS CLI 명령을 통한 SecretsManager 정보 수집	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T11:53:32Z • eventSource: secretsmanager.amazonaws.com • eventName: ListSecrets • awsRegion: ap-northeast-2 • sourceIPAddress: 85.203.21.56 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/b,E,n,C,Z cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#secretsmanager.list-secrets

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	awsRegion	eventName	eventSource	eventTimeLocal	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userNam
Credential Access	2025-10-01 11:53:32	ap-northeast-2	ListSecrets	secretsmanager.amazonaws.com	2025-10-01 20:53:32	85.203.21.56	aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin

[그림 41] bitParser 분석 결과 파일에서 확인한 SecretsManager 정보 수집 이벤트

13) AWS CLI 명령을 통해 S3 버킷 정책 정보 수집 시도

CloudTrail과 S3 Server Access Log에서 IAM 사용자(acces.admin)가 85.203.21.30(싱가포르) IP에서 AWS CLI로 ‘s3api get-bucket-policy’ 명령을 사용해 S3 버킷(plainbit-s3) 정책 정보를 수집 시도한 것을 확인할 수 있었다.


```
2025-10-01T11:56:43.781Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-01T11:54:32Z","eventSource":"s3.amazonaws.com","eventName":"GetBucketPolicy","awsRegion":"ap-northeast-2
","sourceIPAddress":"85.203.21.30","userAgent":"[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64
lang/python#3.13.7 md/pyimpl#CPython m/b,n,E,Z cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#s3api.get-bucket-policy]","errorCode":"NoSuchBucketPolicy","errorMessage":"The bucket policy does not
exist","requestParameters":{"bucketName":"plainbit-s3","Host":"plainbit-s3.s3.ap-northeast-2.amazonaws.com","policy":""}
,"responseElements":null,"additionalEventData":{"SignatureVersion":"SigV4","CipherSuite":"TLS_AES_128_GCM_SHA256","bytes
TransferredIn":0,"AuthenticationMethod":"AuthHeader","x-amz-id-2":"sEYPjrACtxlK035Xy1vG8SOGg5gF4BBBO26TNWa8/BffNryK8h/XE
t4oEQgYmFSEc3y3hAdqK2dg0jlpOhFSv/xie8E/Jtph","bytesTransferredOut":324},"requestID":"VT6VY0ZM8E2Y9V4P","eventID":"a3daab
5b-97f1-4c0f-82cf-2743d46b17f7","readOnly":true,"resources":[{"accountId":"231307122651","type":"AWS::S3::Bucket","ARN":
"arn:aws:s3:::plainbit-s3"}],"eventType":"AwsApiCall","managementEvent":true,"recipientAccountId":"231307122651","eventC
ategory":"Management","tlsDetails":{"tlsVersion":"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHostHea
der":"plainbit-s3.s3.ap-northeast-2.amazonaws.com"}}
```


[그림 42] CloudTrail에서 확인되는 AWS CLI 명령을 통한 S3 버킷 정책 정보 수집 시도 이벤트


```
5043e0a2f5c9e33bf501c24bcde8204bf2bf4f8e4be8f9b60878bcce1c01d406 plainbit-s3 [01/Oct/2025:11:54:32 +0000] 85.203.21.30
arn:aws:iam::231307122651:user/acces.admin VT6VY0ZM8E2Y9V4P REST.GET.BUCKETPOLICY - "GET /?policy HTTP/1.1" 404
NoSuchBucketPolicy 324 - 23 - "-" "aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64
lang/python#3.13.7 md/pyimpl#CPython m/b,n,E,Z cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#s3api.get-bucket-policy" -
sEYPjrACtxlK035Xy1vG8SOGg5gF4BBBO26TNWa8/BffNryK8h/XEt4oEQgYmFSEc3y3hAdqK2dg0jlpOhFSv/xie8E/Jtph SigV4
TLS_AES_128_GCM_SHA256 AuthHeader plainbit-s3.s3.ap-northeast-2.amazonaws.com TLSv1.3 - -
```


[그림 43] S3 Access Log에서 확인되는 AWS CLI 명령을 통한 S3 버킷 정책 정보 수집 시도 이벤트

[표 162] AWS CLI 명령을 통한 S3 버킷 정책 정보 수집 이벤트의 주요 필드 내용

구분	주요 필드 내용
(CloudTrail) AWS CLI 명령을 통한 S3 버킷 정책 정보 수집	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T11:54:32Z • eventSource: s3.amazonaws.com • eventName: GetBucketPolicy • awsRegion: ap-northeast-2 • sourceIPAddress: 85.203.21.30 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#Cpython m/b,n,E,Z cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#s3api.get-bucket-policy • errorCode: NoSuchBucketPolicy • requestParameters - bucketName: plainbit-s3

구분	주요 필드 내용
(S3 Server Access Log) AWS CLI 명령을 통한 S3 버킷 정책 정보 수집	• bucketName: plainbit-s3 • eventTime: [01/Oct/2025:11:54:32 +0000] • sourceIPAddress: 85.203.21.30 • arn: arn:aws:iam::231307122651:user/acces.admin • task: REST.GET.BUCKETPOLICY • request: GET /?policy HTTP/1.1 • statusCode: 404 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#Cpython m/b,n,E,Z cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#s3api.get-bucket-policy

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Discovery	2025-10-01 11:54:32+00:00	2025-10-01 20:54:32	ap-northeast-2	GetBucketPolicy	s3.amazonaws.com	85.203.21.30	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin
Discovery	2025-10-01 12:23:52+00:00	2025-10-01 21:23:52	ap-northeast-2	GetBucketPolicy	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64;	Root	
Discovery	2025-10-02 04:56:19+00:00	2025-10-02 13:56:19	ap-northeast-2	GetBucketPolicy	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64;	Root	

[그림 44] bitParser 분석 결과 파일에서 확인한 S3 버킷 정책 정보 수집 시도 이벤트(CloudTrail)

MITRE ATT&CK	bucket_owner	Bucket	original_timestamp	Timestamp (UTC)	Source IP	Requester	request_id	Operation	Key	Request URI	http_status_code
Discovery	5043e0a2f5c9e33bf501c24bcde	plainbit-s3	[01/Oct/2025:11:54:32 +0000]	2025-10-01 11:54:32	85.203.21.30	arn:aws:iam::231307122651:user/acces.a	VT6VY0ZM8E2Y9V4P	REST.GET.BUCKETPOLICY	-	GET /?policy HTTP/1.1	404
Discovery	5043e0a2f5c9e33bf501c24bcde	plainbit-s3	[01/Oct/2025:12:23:52 +0000]	2025-10-01 12:23:52	222.99.52.250	5043e0a2f5c9e33bf501c24bcde8204bf2JFZKKZVWYA24VAC9		REST.GET.BUCKETPOLICY	-	GET /?policy HTTP/1.1	404

[그림 45] bitPaser 분석 결과 파일에서 확인한 S3 버킷 정책 정보 수집 시도 이벤트(S3 Access Log)

14) AWS CLI 명령을 통해 S3 버킷 ACL 정보 수집

CloudTrail과 S3 Access Log에서 IAM 사용자(acces.admin)가 85.203.21.53(싱가포르) IP에서 AWS CLI로 ‘s3api get-bucket-acl’ 명령을 사용해 S3 버킷(plainbit-s3) ACL 정보를 수집하는 것을 확인할 수 있었다.


```
2025-10-01T11:56:43.781Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-01T11:55:11Z","eventSource":"s3.amazonaws.com","eventName":"GetBucketAcl","awsRegion":"ap-northeast-2","
sourceIPAddress":"85.203.21.53","userAgent":"[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64
lang/python#3.13.7 md/pyimpl#CPython m/b,E,n,Z cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#s3api.get-bucket-acl]","requestParameters":{"bucketName":"plainbit-s3","Host":"plainbit-s3.s3.ap-northeast-2.
amazonaws.com","acl":""},"responseElements":null,"additionalEventData":{"SignatureVersion":"SigV4","CipherSuite":"TLS_AE
S_128_GCM_SHA256","bytesTransferredIn":0,"AuthenticationMethod":"AuthHeader","x-amz-id-2":"NaZp3awqotiggkG1R1tyT71IHjJOC
+68vKubomyoNwNEdDuwTXlfL+G6BgUF9y5/LPJR8zwuiViW4TJJtuuoq/GA8vNq91p6","bytesTransferredOut":480},"requestID":"09VD362A1ZY
J5PF0","eventID":"52ff0880-4da5-426a-8608-fc129bdab783","readOnly":true,"resources":[{"accountId":"231307122651","type":
"AWS::S3::Bucket","ARN":"arn:aws:s3:::plainbit-s3"}],"eventType":"AwsApiCall","managementEvent":true,"recipientAccountId
":"231307122651","eventCategory":"Management","tlsDetails":{"tlsVersion":"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256
","clientProvidedHostHeader":"plainbit-s3.s3.ap-northeast-2.amazonaws.com"}}
```


[그림 46] CloudTrail에서 확인되는 AWS CLI 명령을 통한 S3 버킷 ACL 정보 수집 이벤트


```
5043e0a2f5c9e33bf501c24bcde8204bf2bf4f8e4be8f9b60878bcce1c01d406 plainbit-s3 [01/Oct/2025:11:55:11 +0000] 85.203.21.53
arn:aws:iam::231307122651:user/acces.admin 09VD362A1ZYJ5PF0 REST.GET.ACL - "GET /?acl HTTP/1.1" 200 - 480 - 21 - "-"
"aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/b,E,n,Z
cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#s3api.get-bucket-acl" -
NaZp3awqotiggkG1R1tyT71IHjJOC+68vKubomyoNwNEdDuwTXlfL+G6BgUF9y5/LPJR8zwuiViW4TJJtuuoq/GA8vNq91p6 SigV4
TLS_AES_128_GCM_SHA256 AuthHeader plainbit-s3.s3.ap-northeast-2.amazonaws.com TLSv1.3 - -
```

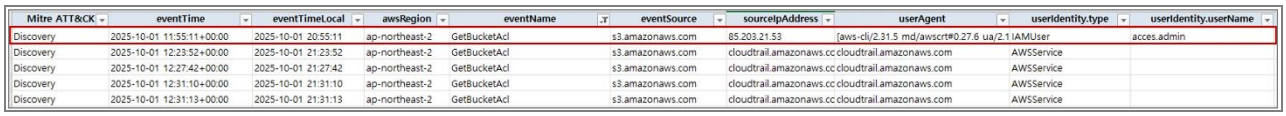

[그림 47] S3 Access Log에서 확인되는 AWS CLI 명령을 통한 S3 버킷 ACL 정보 수집 이벤트

[표 163] AWS CLI 명령을 통한 S3 버킷 ACL 정보 수집 이벤트의 주요 필드 내용

구분	주요 필드 내용
(CloudTrail) AWS CLI 명령을 통한 S3 버킷 ACL 정보 수집	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T11:55:11Z • eventSource: s3.amazonaws.com • eventName: GetBucketAcl • awsRegion: ap-northeast-2 • sourceIPAddress: 85.203.21.53 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#Cpython m/b,E,n,Z cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#s3api.get-bucket-acl • requestParameters - bucketName: plainbit-s3

구분	주요 필드 내용
(S3 Server Access Log) AWS CLI 명령을 통한 S3 버킷 ACL 정보 수집	• bucketName: plainbit-s3 • eventTime: [01/Oct/2025:11:55:11 +0000] • sourceIPAddress: 85.203.21.53 • arn: arn:aws:iam::231307122651:user/acces.admin • task: REST.GET.ACL • request: GET /?acl HTTP/1.1 • statusCode: 200 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#Cpython m/b,E,n,Z cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#s3api.get-bucket-acl

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.



Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Discovery	2025-10-01 11:55:11+00:00	2025-10-01 20:55:11	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	85.203.21.53	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin
Discovery	2025-10-01 12:23:52+00:00	2025-10-01 21:23:52	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.cc	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 12:27:42+00:00	2025-10-01 21:27:42	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.cc	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 12:31:10+00:00	2025-10-01 21:31:10	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.cc	cloudtrail.amazonaws.com	AWSService	
Discovery	2025-10-01 12:31:13+00:00	2025-10-01 21:31:13	ap-northeast-2	GetBucketAcl	s3.amazonaws.com	cloudtrail.amazonaws.cc	cloudtrail.amazonaws.com	AWSService	

[그림 48] bitParser 분석 결과 파일에서 확인한 S3 버킷 ACL 정보 수집 이벤트(CloudTrail)

MITRE ATT&CK	bucket_owner	Bucket	original_timestamp	Timestamp (UTC)	Source IP	Requester	request_id	Operation	Key	Request URI	http_status_code
Discovery	5043e0a2f5c9e33bf501c24bcde	plainbit-s3	[01/Oct/2025:11:55:11 +0000]	2025-10-01 11:55:11	85.203.21.53	arn:aws:iam::231307122651:user/acces.admin	09VD362A1ZYJ5PF0	REST.GET.ACL	-	GET /?acl HTTP/1.1	200
Discovery	5043e0a2f5c9e33bf501c24bcde	plainbit-s3	[01/Oct/2025:13:49:30 +0000]	2025-10-01 13:49:30	-	svc.cloudtrail.amazonaws.com	QMWF1NSWAZAR6XZ0	REST.GET.ACL	-	GET /?acl HTTP/1.1	200

[그림 49] bitParser 분석 결과 파일에서 확인한 S3 버킷 ACL 정보 수집 이벤트(S3 Access Log)

15) EC2 인스턴스에 Public IP 주소 할당

CloudTrail에서 IAM 사용자(acces.admin)가 85.203.21.49(싱가포르) IP에서 Chrome 브라우저를 통해 네트워크 인터페이스(eni-01af8a2eb6f8e8394)의 Public IP Address를 활성화하는 것을 확인할 수 있었다.


```
2025-10-01T12:00:54.729Z
{"eventVersion":"1.10","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"ASIATLWX2S7NQLYV27QL","userName":"acces.admin","sessi
onContext":{"attributes":{"creationDate":"2025-10-01T11:40:32Z","mfaAuthenticated":"false"}}},"eventTime":"2025-10-01T12
:00:11Z","eventSource":"ec2.amazonaws.com","eventName":"ModifyNetworkInterfaceAttribute","awsRegion":"ap-northeast-2","s
ourceIPAddress":"85.203.21.49","userAgent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/140.0.0.0
Safari/537.36","requestParameters":{"networkInterfaceId":"eni-01af8a2eb6f8e8394","associatePublicIpAddress":true},"respo
nseElements":{"requestId":"8b1bfd0f-0ad0-40ae-ab5a-067de7309c06","_return":true},"requestID":"8b1bfd0f-0ad0-40ae-ab5a-06
7de7309c06","eventID":"79ec1a05-d61d-4746-ac8c-d46a259f40c7","readOnly":false,"eventType":"AwsApiCall","managementEvent"
:true,"recipientAccountId":"231307122651","eventCategory":"Management","tlsDetails":{"tlsVersion":"TLSv1.3","cipherSuite
":"TLS_AES_128_GCM_SHA256","clientProvidedHostHeader":"ec2.ap-northeast-2.amazonaws.com"},"sessionCredentialFromConsole"
:"true"}
```


[그림 50] CloudTrail에서 확인되는 EC2 인스턴스 Public IP 주소 활성화 이벤트

[표 164] EC2 인스턴스 Public IP 주소 활성화 이벤트의 주요 필드 내용

구분	주요 필드 내용
EC2 인스턴스 Public IP 주소 활성화	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T12:00:11Z • eventSource: ec2.amazonaws.com • eventName: ModifyNetworkInterfaceAttribute • awsRegion: ap-northeast-2 • sourceIPAddress: 85.203.21.49 • userAgent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36 • requestParameters - networkInterfaceId: eni-01af8a2eb6f8e8394 - associatePublicIpAddress: true

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	awsRegion	eventName	eventSource	eventTimeLocal	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userNam
Defense Evasion	2025-10-01 12:00:11	ap-northeast-2	ModifyNetworkInterfaceAttribute	ec2.amazonaws.com	2025-10-01 21:00:11	85.203.21.49	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	acces.admin

[그림 51] bitParser 분석 결과 파일에서 확인한 EC2 인스턴스 Public IP 주소 활성화 이벤트

16) EC2 인스턴스 비밀번호 획득

CloudTrail에서 IAM 사용자(acces.admin)가 85.203.21.56(싱가포르) IP에서 Chrome 브라우저를 통해 EC2 인스턴스의 비밀번호를 획득하는 것을 확인할 수 있었다.


```
2025-10-02T04:33:31.587Z
{"eventVersion":"1.10","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"ASIATLWX2S7NUIT44SNP","userName":"acces.admin","sessi
onContext":{"attributes":{"creationDate":"2025-10-02T04:31:00Z","mfaAuthenticated":"false"}}},"eventTime":"2025-10-02T04
:31:49Z","eventSource":"ec2.amazonaws.com","eventName":"GetPasswordData","awsRegion":"ap-northeast-2","sourceIPAddress":
"85.203.21.56","userAgent":"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/140.0.0.0
Safari/537.36","requestParameters":{"instanceId":"i-01d86d270432b7980"},"responseElements":null,"requestID":"09ca83e7-a2
fd-47bf-86bc-4584c32953bb","eventID":"e1826151-d94c-4b70-908e-9fa6df7e97e0","readOnly":true,"eventType":"AwsApiCall","ma
nagementEvent":true,"recipientAccountId":"231307122651","eventCategory":"Management","tlsDetails":{"tlsVersion":"TLSv1.3
","cipherSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHostHeader":"ec2.ap-northeast-2.amazonaws.com"},"sessionCredenti
alFromConsole":"true"}
```


[그림 52] CloudTrail에서 확인되는 EC2 인스턴스 비밀번호 획득 이벤트

[표 165] EC2 인스턴스 비밀번호 획득 이벤트의 주요 필드 내용

구분	주요 필드 내용
EC2 인스턴스 비밀번호 획득	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-02T04:31:49Z • eventSource: ec2.amazonaws.com • eventName: GetPasswordData • awsRegion: ap-northeast-2 • sourceIPAddress: 85.203.21.56 • userAgent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Safari/537.36 • requestParameters - instanceId: i-01d86d270432b7980

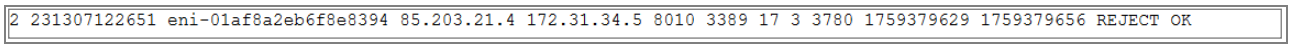
bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Credential Access	2025-10-01 12:02:30+00:00	2025-10-01 21:02:30	ap-northeast-2	GetPasswordData	ec2.amazonaws.com	85.203.21.38	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	acces.admin
Credential Access	2025-10-01 12:09:09+00:00	2025-10-01 21:09:09	ap-northeast-2	GetPasswordData	ec2.amazonaws.com	222.99.52.250	Mozilla/5.0 (Windows NT 10.0; Win64;	Root	
Credential Access	2025-10-02 04:31:49+00:00	2025-10-02 13:31:49	ap-northeast-2	GetPasswordData	ec2.amazonaws.com	85.203.21.56	Mozilla/5.0 (Windows NT 10.0; Win64;	IAMUser	acces.admin

[그림 53] bitParser 분석 결과 파일에서 확인한 EC2 인스턴스 비밀번호 획득 이벤트

17) EC2 인스턴스 원격 접근(RDP)

VPC Flow Logs에서 85.203.21.4(싱가포르) IP가 172.31.34.5(Public IP)로 RDP(3389) 접근 성공(ACCEPT)한 것을 확인할 수 있었다.



[그림 54] VPC Flow Logs에서 확인되는 EC2 인스턴스 원격 접근(RDP) 이벤트

[표 166] 원격 접근(RDP) 이벤트의 주요 필드 내용

구분	주요 필드 내용
원격 접근(RDP)	• account-id: 231307122651 • interface-id: eni-01af8a2eb6f8e8394 • srcaddr: 85.203.21.4 • dstaddr: 172.31.34.5 • srcport: 20813 • dstport: 3389 • protocol: 6 • packets: 1167 • bytes: 137739 • start: 1759379663 • end: 1759379673 • action: ACCEPT

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Source IP	Destination IP	Port	Protocol	Service	Total Bytes	Total Packets
85.203.21.4	172.31.34.5	3389	TCP	RDP	144579	1271
85.203.21.4	172.31.34.5	3389	UDP	RDP	3780	3
3.149.59.26	172.31.34.5	3389	TCP	RDP	2056	25
20.64.105.251	172.31.34.5	3389	TCP	RDP	1009	14
3.86.50.115	172.31.34.5	3389	TCP	RDP	772	8

[그림 55] bitParser 분석 결과 파일에서 확인한 원격 접근(RDP) 이벤트

18) AWS CLI 명령을 통해 CloudTrail 목록 수집

CloudTrail에서 IAM 사용자(acces.admin)가 85.203.21.7(싱가포르) IP에서 AWS CLI로 'cloudtrail describe-trails' 명령을 실행해 CloudTrail 목록을 수집하는 것을 확인할 수 있었다.


```
2025-10-01T12:27:01.240Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-01T12:25:03Z","eventSource":"cloudtrail.amazonaws.com","eventName":"DescribeTrails","awsRegion":"ap-nort
heast-2","sourceIPAddress":"85.203.21.7","userAgent":"aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11
md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/n,Z,b,E cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#cloudtrail.describe-trails","requestParameters":null,"responseElements":null,"requestID":"1fefc6fd-4b32-4726-
bd02-83cdeee9a5c3","eventID":"a8b7b504-6e44-45be-998a-7e5672b4379e","readOnly":true,"eventType":"AwsApiCall","management
Event":true,"recipientAccountId":"231307122651","eventCategory":"Management","tlsDetails":{"tlsVersion":"TLSv1.3","ciphe
rSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHostHeader":"cloudtrail.ap-northeast-2.amazonaws.com"}}
```


[그림 56] CloudTrail에서 확인되는 AWS CLI 명령을 통한 CloudTrail 목록 수집 이벤트

[표 167] AWS CLI 명령을 통한 CloudTrail 목록 수집 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS CLI 명령을 통한 CloudTrail 목록 수집	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T12:25:03Z • eventSource: cloudtrail.amazonaws.com • eventName: DescribeTrails • awsRegion: ap-northeast-2 • sourceIPAddress: 85.203.21.7 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/n,Z,b,E cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#cloudtrail.describe-trails

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	awsRegion	eventName	eventSource	eventTimeLocal	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userNam
Discovery	2025-10-01 12:25:03	ap-northeast-2	DescribeTrails	cloudtrail.amazonaws.com	2025-10-01 21:25:03	85.203.21.7	aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin

[그림 57] bitParser 분석 결과 파일에서 확인한 CloudTrail 목록 수집 이벤트

19) AWS CLI 명령을 통해 CloudTrail 비활성화

CloudTrail에서 IAM 사용자(acces.admin)가 85.203.21.67(싱가포르) IP에서 AWS CLI로 'cloudtrail stop-logging' 명령을 사용해 CloudTrail(PLAINBIT-TRAIL)의 동작 상태를 비활성화하는 것을 확인할 수 있었다.


```
2025-10-01T12:27:01.240Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-01T12:25:42Z","eventSource":"cloudtrail.amazonaws.com","eventName":"StopLogging","awsRegion":"ap-northea
st-2","sourceIPAddress":"85.203.21.67","userAgent":"aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64
lang/python#3.13.7 md/pyimpl#CPython m/b,n,Z,E cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#cloudtrail.stop-logging","requestParameters":{"name":"PLAINBIT-TRAIL"},"responseElements":null,"requestID":"b
5c2e56d-50e5-4f14-a1c5-a1d713482580","eventID":"73ce5819-e90c-4b12-ae32-51799eed5ccf","readOnly":false,"eventType":"AwsA
piCall","managementEvent":true,"recipientAccountId":"231307122651","eventCategory":"Management","tlsDetails":{"tlsVersio
n":"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHostHeader":"cloudtrail.ap-northeast-2.amazonaws.com"
}}
```


[그림 58] CloudTrail에서 확인되는 AWS CLI 명령을 통한 CloudTrail 비활성화 이벤트

[표 168] AWS CLI 명령을 통한 CloudTrail 비활성화 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS CLI 명령을 통한 CloudTrail 비활성화	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T12:25:42Z • eventSource: cloudtrail.amazonaws.com • eventName: StopLogging • awsRegion: ap-northeast-2 • sourceIPAddress: 85.203.21.67 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/b,n,Z,E cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#cloudtrail.stop-logging • requestParameters - name: PLAINBIT-TRAIL

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Defense Evasion	2025-10-01 12:25:42+00:00	2025-10-01 21:25:42	ap-northeast-2	StopLogging	cloudtrail.amazonaws.com	85.203.21.67	aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin

[그림 59] bitParser 분석 결과 파일에서 확인한 CloudTrail 비활성화 이벤트

20) AWS CLI 명령을 통해 EC2 스냅샷 목록 수집

CloudTrail에서 IAM 사용자(acces.admin)가 85.203.21.9(싱가포르) IP에서 AWS CLI로 'ec2 describe-snapshots' 명령을 사용해 EC2 스냅샷 목록을 수집하는 것을 확인할 수 있었다.


```
2025-10-01T12:29:42.820Z
{"eventVersion":"1.10","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-01T12:27:33Z","eventSource":"ec2.amazonaws.com","eventName":"DescribeSnapshots","awsRegion":"ap-northeas
t-2","sourceIPAddress":"85.203.21.9","userAgent":"aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64
lang/python#3.13.7 md/pyimpl#CPython m/b,n,E,Z,C cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#ec2.describe-snapshots","requestParameters":{"maxResults":1000,"snapshotSet":{},"ownersSet":{},"sharedUsersSe
t":{},"filterSet":{}},"responseElements":null,"requestID":"9ca9f668-8972-40f2-8ca0-ef420573b8fc","eventID":"e052834d-ac5
f-4e9f-b572-f0279c1a1a2b","readOnly":true,"eventType":"AwsApiCall","managementEvent":true,"recipientAccountId":"23130712
2651","eventCategory":"Management","tlsDetails":{"tlsVersion":"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientPr
ovidedHostHeader":"ec2.ap-northeast-2.amazonaws.com"}}
```


[그림 60] CloudTrail에서 확인되는 AWS CLI 명령을 통한 EC2 스냅샷 목록 수집 이벤트

[표 169] AWS CLI 명령을 통한 EC2 스냅샷 목록 수집 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS CLI 명령을 통한 EC2 스냅샷 목록 수집	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T12:27:33Z ~ 2025-10-01T12:27:59Z • eventSource: ec2.amazonaws.com • eventName: DescribeSnapshots • awsRegion: ap-northeast-2 • sourceIPAddress: 85.203.21.9 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/b,n,E,Z,C cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#ec2.describe-snapshots

21) AWS CLI 명령을 통해 EC2 스냅샷 삭제

CloudTrail에서 IAM 사용자(acces.admin)가 85.203.21.20(싱가포르) IP에서 AWS CLI로 'ec2 delete-snapshot' 명령을 사용해 EC2 스냅샷을 삭제한 것을 확인할 수 있었다.


```
2025-10-01T12:36:35.367Z
{"eventVersion":"1.10","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-01T12:34:32Z","eventSource":"ec2.amazonaws.com","eventName":"DeleteSnapshot","awsRegion":"ap-northeast-2
","sourceIPAddress":"85.203.21.20","userAgent":"aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64
lang/python#3.13.7 md/pyimpl#CPython m/E,b,Z,n cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#ec2.delete-snapshot","requestParameters":{"snapshotId":"snap-002c6b72b2e789904","force":false},"responseEleme
nts":{"requestId":"e1ba66e7-8803-48de-a490-69a547b86471","_return":true},"requestID":"e1ba66e7-8803-48de-a490-69a547b864
71","eventID":"de0c5bb5-16f7-4e0f-81b1-d35c1da0c249","readOnly":false,"eventType":"AwsApiCall","managementEvent":true,"r
ecipientAccountId":"231307122651","eventCategory":"Management","tlsDetails":{"tlsVersion":"TLSv1.3","cipherSuite":"TLS_A
ES_128_GCM_SHA256","clientProvidedHostHeader":"ec2.ap-northeast-2.amazonaws.com"}}
```


[그림 61] CloudTrail에서 확인되는 AWS CLI 명령을 통한 EC2 스냅샷 삭제 이벤트

[표 170] AWS CLI 명령을 통한 EC2 스냅샷 삭제 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS CLI 명령을 통한 EC2 스냅샷 삭제	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T12:34:32Z • eventSource: ec2.amazonaws.com • eventName: DeleteSnapshot • awsRegion: ap-northeast-2 • sourceIPAddress: 85.203.21.20 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/E,b,Z,n cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#ec2.delete-snapshot • requestParameters - snapshotId: snap-002c6b72b2e789904

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Defense Evasion	2025-10-01 12:32:52+00:00	2025-10-01 21:32:52	ap-northeast-2	DeleteSnapshot	ec2.amazonaws.com	85.203.21.12	aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin
Defense Evasion	2025-10-01 12:34:32+00:00	2025-10-01 21:34:32	ap-northeast-2	DeleteSnapshot	ec2.amazonaws.com	85.203.21.20	aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin

[그림 62] bitParser 분석 결과 파일에서 확인한 EC2 스냅샷 삭제 이벤트

22) AWS CLI 명령을 통해 S3 버킷 내 객체 목록 수집

CloudTrail과 S3 Access Log에서 IAM 사용자(acces.admin)가 85.203.21.16(싱가포르) IP에서 AWS CLI로 ‘s3api list-objects’ 명령을 사용해 S3 버킷(plainbit-s3)의 객체 목록을 수집하는 것을 확인할 수 있었다.


```
2025-10-01T12:38:49.649Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-01T12:35:32Z","eventSource":"s3.amazonaws.com","eventName":"ListObjects","awsRegion":"ap-northeast-2","s
ourceIPAddress":"85.203.21.16","userAgent":"[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64
lang/python#3.13.7 md/pyimpl#CPython m/Z,b,C,n,E cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#s3api.list-objects]","requestParameters":{"bucketName":"plainbit-s3","Host":"plainbit-s3.s3.ap-northeast-2.am
azonaws.com","encoding-type":"url"},"responseElements":null,"additionalEventData":{"SignatureVersion":"SigV4","CipherSui
te":"TLS_AES_128_GCM_SHA256","bytesTransferredIn":0,"AuthenticationMethod":"AuthHeader","x-amz-id-2":"XLm71FYzj6/WmmRDJP
2OGE2Ch3UoZpRkGjFxsdDOzGlVmPrJdsCdYGqs1NYFqBNuEqhJhlGzi92mZ6snOYRKirNXcQqppwEKMUmRDZPgobE=","bytesTransferredOut":2349},
"requestID":"D9CRTKWH2WAMTJMG","eventID":"ee80bb19-c5e1-46da-89f0-179dfc6580ff","readOnly":true,"resources":[{"accountId
":"231307122651","type":"AWS::S3::Bucket","ARN":"arn:aws:s3:::plainbit-s3"},{"type":"AWS::S3::Object","ARNPrefix":"arn:a
ws:s3:::plainbit-s3/"}],"eventType":"AwsApiCall","managementEvent":false,"recipientAccountId":"231307122651","eventCateg
ory":"Data","tlsDetails":{"tlsVersion":"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHostHeader":"plai
nbit-s3.s3.ap-northeast-2.amazonaws.com"}}
```


[그림 63] CloudTrail에서 확인되는 AWS CLI 명령을 통한 S3 버킷 내 객체 목록 수집 이벤트


```
5043e0a2f5c9e33bf501c24bcde8204bf2bf4f8e4be8f9b60878bcce1c01d406 plainbit-s3 [01/Oct/2025:12:35:32 +0000] 85.203.21.16
arn:aws:iam::231307122651:user/acces.admin D9CRTKWH2WAMTJMG REST.GET.BUCKET - "GET /?encoding-type=url HTTP/1.1" 200 -
2349 - 38 37 "-" "aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7
md/pyimpl#CPython m/Z,b,C,n,E cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#s3api.list-objects" -
XLm71FYzj6/WmmRDJP2OGE2Ch3UoZpRkGjFxsdDOzGlVmPrJdsCdYGqs1NYFqBNuEqhJhlGzi92mZ6snOYRKirNXcQqppwEKMUmRDZPgobE= SigV4
TLS_AES_128_GCM_SHA256 AuthHeader plainbit-s3.s3.ap-northeast-2.amazonaws.com TLSv1.3 - -
```


[그림 64] S3 Access Log에서 확인되는 AWS CLI 명령을 통한 S3 버킷 내 객체 목록 수집 이벤트

[표 171] AWS CLI 명령을 통한 S3 버킷 내 객체 목록 수집 이벤트의 주요 필드 내용

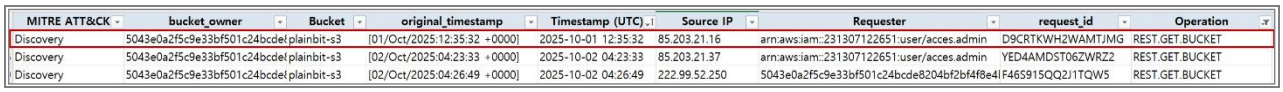
구분	주요 필드 내용
(CloudTrail) AWS CLI 명령을 통한 S3 버킷 내 객체 목록 수집	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T12:35:32Z • eventSource: s3.amazonaws.com • eventName: ListObjects • awsRegion: ap-northeast-2 • sourceIPAddress: 85.203.21.16 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/Z,b,C,n,E cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#s3api.list-objects • requestParameters - bucketName: plainbit-s3 - encoding-type: url

구분	주요 필드 내용
(S3 Server Access Log) AWS CLI 명령을 통한 S3 버킷 내 객체 목록 수집	• bucketName: plainbit-s3 • eventTime: [01/Oct/2025:12:35:32 +0000] • sourceIPAddress: 85.203.21.16 • arn: arn:aws:iam::231307122651:user/acces.admin • task: REST.GET.BUCKET • request: GET /?encoding-type=url HTTP/1.1 • statusCode: 200 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/Z,b,C,n,E cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#s3api.list-objects

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Discovery	2025-10-01 12:35:32+00:00	2025-10-01 21:35:32	ap-northeast-2	ListObjects	s3.amazonaws.com	85.203.21.16	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin

[그림 65] bitParser 분석 결과 파일에서 확인한 S3 버킷 내 객체 목록 수집 이벤트(CloudTrail)



MITRE ATT&CK	bucket_owner	Bucket	original_timestamp	Timestamp (UTC)	Source IP	Requester	request_id	Operation
Discovery	5043e0a2f5c9e33bf501c24bcde	plainbit-s3	[01/Oct/2025:12:35:32 +0000]	2025-10-01 12:35:32	85.203.21.16	arn:aws:iam::231307122651:user/acces.admin	D9CRTKWH2WAMTJMG	REST.GET.BUCKET
Discovery	5043e0a2f5c9e33bf501c24bcde	plainbit-s3	[02/Oct/2025:04:23:33 +0000]	2025-10-02 04:23:33	85.203.21.37	arn:aws:iam::231307122651:user/acces.admin	YED4AMDST06ZWRZ2	REST.GET.BUCKET
Discovery	5043e0a2f5c9e33bf501c24bcde	plainbit-s3	[02/Oct/2025:04:26:49 +0000]	2025-10-02 04:26:49	222.99.52.250	5043e0a2f5c9e33bf501c24bcde8204bf2bf4f8e4	F465915QQ2J1TQW5	REST.GET.BUCKET

[그림 66] bitParser 분석 결과 파일에서 확인한 S3 버킷 내 객체 목록 수집 이벤트(S3 Access Log)

23) AWS CLI 명령을 통해 S3 버킷 내 객체 다운로드

CloudTrail과 S3 Access Log에서 IAM 사용자(acces.admin)가 85.203.21.21(싱가포르) IP에서 AWS CLI로 's3api get-objects' 명령을 실행해 S3 버킷(plainbit-s3) 내 객체(Top_Secret)를 다운로드하는 것을 확인할 수 있었다.


```
2025-10-01T12:45:29.665Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-01T12:41:02Z","eventSource":"s3.amazonaws.com","eventName":"GetObject","awsRegion":"ap-northeast-2","sou
rceIPAddress":"85.203.21.21","userAgent":"[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64
lang/python#3.13.7 md/pyimpl#CPython m/E,n,b,Z cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#s3api.get-object]","requestParameters":{"bucketName":"plainbit-s3","Host":"plainbit-s3.s3.ap-northeast-2.amaz
onaws.com","key":"Secret/Top_Secret"},"responseElements":null,"additionalEventData":{"SignatureVersion":"SigV4","CipherS
uite":"TLS_AES_128_GCM_SHA256","bytesTransferredIn":0,"AuthenticationMethod":"AuthHeader","x-amz-id-2":"Z4w4W5wqIDEZTJg4
jhdxYBjLAydgpMIuh5bNgkENtllRTvSjMbRScCxg4fdGqEXQ6kVs2oUsmfRyQ+br6rdDFlcsRNXV4qx84zUpwSfQIpU=","bytesTransferredOut":15},
"requestID":"35X1GPG4ANQWJP15","eventID":"50971eb9-a7c0-458a-bfcb-db189a6a1798","readOnly":true,"resources":[{"accountId
":"231307122651","type":"AWS::S3::Bucket","ARN":"arn:aws:s3:::plainbit-s3"},{"type":"AWS::S3::Object","ARN":"arn:aws:s3:
::plainbit-s3/Secret/Top_Secret"}],"eventType":"AwsApiCall","managementEvent":false,"recipientAccountId":"231307122651",
"eventCategory":"Data","tlsDetails":{"tlsVersion":"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientProvidedHostHe
ader":"plainbit-s3.s3.ap-northeast-2.amazonaws.com"}}
```


[그림 67] CloudTrail에서 확인되는 AWS CLI 명령을 통한 S3 버킷 내 객체 다운로드 이벤트


```
5043e0a2f5c9e33bf501c24bcde8204bf2bf4f8e4be8f9b60878bcce1c01d406 plainbit-s3 [01/Oct/2025:12:41:02 +0000] 85.203.21.21
arn:aws:iam::231307122651:user/acces.admin 35X1GPG4ANQWJP15 REST.GET.OBJECT Secret/Top_Secret "GET /Secret/Top_Secret
HTTP/1.1" 200 - 15 15 33 32 "-" "aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7
md/pyimpl#CPython m/E,n,b,Z cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#s3api.get-object" -
Z4w4W5wqIDEZTJg4jhdxYBjLAydgpMIuh5bNgkENtllRTvSjMbRScCxg4fdGqEXQ6kVs2oUsmfRyQ+br6rdDFlcsRNXV4qx84zUpwSfQIpU= SigV4
TLS_AES_128_GCM_SHA256 AuthHeader plainbit-s3.s3.ap-northeast-2.amazonaws.com TLSv1.3 - -
```


[그림 68] S3 Access Log에서 확인되는 AWS CLI 명령을 통한 S3 버킷 내 객체 다운로드 이벤트

[표 172] AWS CLI 명령을 통한 S3 버킷 내 객체 다운로드 이벤트의 주요 필드 내용

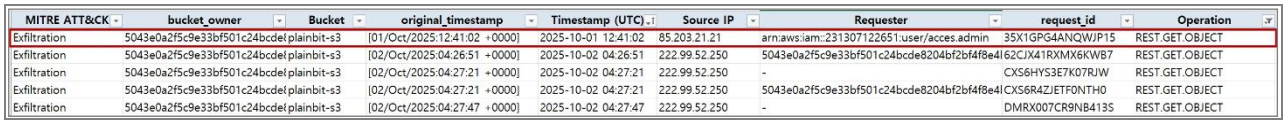
구분	주요 필드 내용
(CloudTrail) AWS CLI 명령을 통한 S3 버킷 내 객체 다운로드	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-01T12:41:02Z • eventSource: s3.amazonaws.com • eventName: GetObject • awsRegion: ap-northeast-2 • sourceIPAddress: 85.203.21.21 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/E,n,b,Z cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#s3api.get-object • requestParameters - bucketName: plainbit-s3 - key: Secret/Top_Secret

구분	주요 필드 내용
(S3 Server Access Log) AWS CLI 명령을 통한 S3 버킷 내 객체 다운로드	• bucketName: plainbit-s3 • eventTime: [01/Oct/2025:12:41:02 +0000] • sourceIPAddress: 85.203.21.21 • arn: arn:aws:iam::231307122651:user/acces.admin • task: REST.GET.OBJECT • targetObject(Key): Secret/Top_Secret • request: GET /Secret/Top_Secret HTTP/1.1 • statusCode: 200 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/E,n,b,Z cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#s3api.get-object

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Exfiltration	2025-10-01 12:41:02+00:00	2025-10-01 21:41:02	ap-northeast-2	GetObject	s3.amazonaws.com	85.203.21.21	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1	IAMUser	acces.admin

[그림 69] bitParser 분석 결과 파일에서 확인한 S3 버킷 내 객체 다운로드 이벤트(CloudTrail)



MITRE ATT&CK	bucket_owner	Bucket	original_timestamp	Timestamp (UTC)	Source IP	Requester	request_id	Operation
Exfiltration	5043e0a2f5c9e33bf501c24bcde	plainbit-s3	[01/Oct/2025:12:41:02 +0000]	2025-10-01 12:41:02	85.203.21.21	arn:aws:iam::231307122651:user/acces.admin	35X1GPG4ANQWJP15	REST.GET.OBJECT
Exfiltration	5043e0a2f5c9e33bf501c24bcde	plainbit-s3	[02/Oct/2025:04:26:51 +0000]	2025-10-02 04:26:51	222.99.52.250	5043e0a2f5c9e33bf501c24bcde8204bf2bf4f8e4	62CJX41RXMX6KWB7	REST.GET.OBJECT
Exfiltration	5043e0a2f5c9e33bf501c24bcde	plainbit-s3	[02/Oct/2025:04:27:21 +0000]	2025-10-02 04:27:21	222.99.52.250	-	CXS6HYS3E7K07RJW	REST.GET.OBJECT
Exfiltration	5043e0a2f5c9e33bf501c24bcde	plainbit-s3	[02/Oct/2025:04:27:21 +0000]	2025-10-02 04:27:21	222.99.52.250	5043e0a2f5c9e33bf501c24bcde8204bf2bf4f8e4	CXS6R4ZJETF0NTH0	REST.GET.OBJECT
Exfiltration	5043e0a2f5c9e33bf501c24bcde	plainbit-s3	[02/Oct/2025:04:27:47 +0000]	2025-10-02 04:27:47	222.99.52.250	-	DMRX007CR9NB413S	REST.GET.OBJECT

[그림 70] bitParser 분석 결과 파일에서 확인한 S3 버킷 내 객체 다운로드 이벤트(S3 Access Log)

24) AWS CLI 명령을 통해 S3 버킷 내 객체 암호화

CloudTrail에서 IAM 사용자(acces.admin)가 85.203.21.7(싱가포르) IP에서 AWS CLI로 ‘s3 cp’ 명령을 사용해 S3 버킷(plainbit-s3)의 객체를 암호화(SSE_C) 후 복사하는 것을 확인할 수 있었다.


```
2025-10-02T04:26:11.433Z
{"eventVersion":"1.11","userIdentity":{"type":"IAMUser","principalId":"AIDATLWX2S7N4NW5SSOW6","arn":"arn:aws:iam::231307
122651:user/acces.admin","accountId":"231307122651","accessKeyId":"AKIATLWX2S7NSYHHZ2BL","userName":"acces.admin"},"even
tTime":"2025-10-02T04:23:36Z","eventSource":"s3.amazonaws.com","eventName":"CopyObject","awsRegion":"ap-northeast-2","so
urceIPAddress":"85.203.21.7","userAgent":"[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64
lang/python#3.13.7 md/pyimpl#CPython m/G,Z,b,E cfg/retry-mode#standard md/installer#exe md/prompt#off
md/command#s3.cp]","requestParameters":{"bucketName":"plainbit-s3","Host":"plainbit-s3.s3.ap-northeast-2.amazonaws.com",
"x-amz-server-side-encryption-customer-algorithm":"AES256","x-amz-copy-source":"plainbit-s3/AWSLogs/231307122651/CloudTr
ail-Digest/ap-northeast-2/2025/10/01/231307122651_CloudTrail-Digest_ap-northeast-2_PLAINBIT-TRAIL_ap-northeast-2_2025100
1T122353Z.json.gz","key":"AWSLogs/231307122651/CloudTrail-Digest/ap-northeast-2/2025/10/01/231307122651_CloudTrail-Diges
t_ap-northeast-2_PLAINBIT-TRAIL_ap-northeast-2_20251001T122353Z.json.gz"},"responseElements":{"x-amz-copy-source-version
-id":"dkU1h.djxGpfrPVYHmDVwwr_LrMcCWxe","x-amz-server-side-encryption-customer-algorithm":"AES256","x-amz-version-id":"_
j4.x8d.rhqCODPTGmF1yvnfuc.KrdEi"},"additionalEventData":{"SignatureVersion":"SigV4","CipherSuite":"TLS_AES_128_GCM_SHA25
6","bytesTransferredIn":0,"SSEApplied":"SSE_C","AuthenticationMethod":"AuthHeader","x-amz-id-2":"pJL2Oma92jqen+OILefaKtN
KlHeH3svXX8N+Quivy9eFiu3JVsqwfL553sSLvmMmmlLzq6jnSDw9CCJFgzN4il8QzbuhIzNm","bytesTransferredOut":275},"requestID":"Q66M6
NSVJBC30TA6","eventID":"1232bb4c-e8f5-44bb-8469-24f5ff7618dd","readOnly":false,"resources":[{"accountId":"231307122651",
"type":"AWS::S3::Bucket","ARN":"arn:aws:s3:::plainbit-s3"},{"type":"AWS::S3::Object","ARN":"arn:aws:s3:::plainbit-s3/AWS
Logs/231307122651/CloudTrail-Digest/ap-northeast-2/2025/10/01/231307122651_CloudTrail-Digest_ap-northeast-2_PLAINBIT-TRA
IL_ap-northeast-2_20251001T122353Z.json.gz"}],"eventType":"AwsApiCall","managementEvent":false,"recipientAccountId":"231
307122651","eventCategory":"Data","tlsDetails":{"tlsVersion":"TLSv1.3","cipherSuite":"TLS_AES_128_GCM_SHA256","clientPro
videdHostHeader":"plainbit-s3.s3.ap-northeast-2.amazonaws.com"}}
```


[그림 71] CloudTrail에서 확인되는 AWS CLI 명령을 통한 S3 버킷 내 객체 암호화 이벤트

[표 173] AWS CLI 명령을 통한 S3 버킷 내 객체 암호화 이벤트의 주요 필드 내용

구분	주요 필드 내용
AWS CLI 명령을 통한 S3 버킷 내 객체 암호화	• userIdentity - type: IAMUser - arn: arn:aws:iam::231307122651:user/acces.admin - userName: acces.admin • eventTime: 2025-10-02T04:23:36Z • eventSource: s3.amazonaws.com • eventName: CopyObject • awsRegion: ap-northeast-2 • sourceIPAddress: 85.203.21.7 • userAgent: aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.1 os/windows#11 md/arch#amd64 lang/python#3.13.7 md/pyimpl#CPython m/G,Z,b,E cfg/retry-mode#standard md/installer#exe md/prompt#off md/command#s3.cp • requestParameters - bucketName: plainbit-s3 - x-amz-server-side-encryption-customer-algorithm: AES256 • additionalEventData - SSEApplied: SSE_C

bitParser 분석 결과 파일에서도 해당 행위를 다음과 같이 확인할 수 있다.

Mitre ATT&CK	eventTime	eventTimeLocal	awsRegion	eventName	eventSource	sourceIpAddress	userAgent	userIdentity.type	userIdentity.userName
Exfiltration	2025-10-01 12:40:36+00:00	2025-10-01 21:40:36	ap-northeast-2	CopyObject	s3.amazonaws.com	222.99.52.250	[Mozilla/5.0 (Windows NT 10.0; Win64;	Root	
Exfiltration	2025-10-02 04:23:36+00:00	2025-10-02 13:23:36	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.7	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:36+00:00	2025-10-02 13:23:36	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.44	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:36+00:00	2025-10-02 13:23:36	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.36	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:36+00:00	2025-10-02 13:23:36	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.11	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:36+00:00	2025-10-02 13:23:36	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.16	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:36+00:00	2025-10-02 13:23:36	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.5	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:36+00:00	2025-10-02 13:23:36	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.12	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:36+00:00	2025-10-02 13:23:36	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.59	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:36+00:00	2025-10-02 13:23:36	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.67	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:36+00:00	2025-10-02 13:23:36	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.53	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.44	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.12	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.36	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.36	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.59	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.16	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.11	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.5	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.16	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.11	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.16	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.11	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.12	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.44	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.53	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.7	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.7	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.59	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin
Exfiltration	2025-10-02 04:23:37+00:00	2025-10-02 13:23:37	ap-northeast-2	CopyObject	s3.amazonaws.com	85.203.21.59	[aws-cli/2.31.5 md/awscrt#0.27.6 ua/2.'	IAMUser	acces.admin

[그림 72] bitParser 분석 결과 파일에서 확인한 S3 버킷 내 객체 암호화 이벤트

또한, GuardDuty를 사용하는 경우 S3 버킷의 객체가 암호화되는 것을 이벤트로 탐지하는 것을 확인할 수 있다.


```
[
  {
    "AccountId": "231307122651",
    "Arn":
    "arn:aws:guardduty:ap-northeast-2:231307122651:detector/c8ccd1c2255a3473ef16f77caf52d1d8/finding/9cccd1d36931de5e355
    e95ac13d9de64",
    "AssociatedAttackSequenceArn":
    "arn:aws:guardduty:ap-northeast-2:231307122651:detector/c8ccd1c2255a3473ef16f77caf52d1d8/finding/9cccd1d36931de5e355
    e95ac13d9de64",
    "CreatedAt": "2025-10-02T06:40:43.107Z",
    "Description": "A sequence of actions involving 1 signals indicating a potential data compromise of one or more S3
    bucket(s) was observed for IAMUser/acces.admin with principalId AIDATLWX2S7N2NUZWF7WC in account 231307122651
    between 2025-10-02T06:32:04Z and 2025-10-02T06:32:04Z.\nEvidence:\n- 2 MITRE ATT&CK tactics observed: Exfiltration,
    Impact\n- 2 MITRE ATT&CK techniques observed:\n  - T1567 - Exfiltration Over Web Service\n  - T1486 - Data
    Encrypted for Impact\n- Connected with sensitive networks:\n  - Internet Utilities Europe and Asia Limited:
    ALLOWS_CRYPTO, ALLOWS_TORRENTS, CATEGORY_COMMERCIAL_VPN, CLIENT_BEHAVIOR_FILE_SHARING, IS_ANONYMOUS,
    KNOWN_THREAT_OPERATOR, OPERATOR_EXPRESS_VPN, RISK_CALLBACK_PROXY, TUNNEL_VPN\n- Connected from sensitive IP
    addresses:\n  - 85.203.21.48: ALLOWS_CRYPTO, ALLOWS_TORRENTS, CATEGORY_COMMERCIAL_VPN,
    CLIENT_BEHAVIOR_FILE_SHARING, IS_ANONYMOUS, KNOWN_THREAT_OPERATOR, OPERATOR_EXPRESS_VPN, RISK_CALLBACK_PROXY,
    TUNNEL_VPN\n- 1 sensitive APIs called: s3:CopyObject\n",
    "Id": "9cccd1d36931de5e355e95ac13d9de64",
```


[그림 73] GuardDuty를 통해 탐지된 S3 버킷 내 객체 암호화 이벤트 내용 중 일부

7. 연구 결과

본 연구는 AWS 클라우드 환경에서 사고 발생 시 효율적인 대응과 분석을 지원하기 위한 DFIR 데이터 수집 및 분석 체계를 제안하고 실증했다. 클라우드 구조의 특성상 물리적 접근이 제한되고 데이터가 서비스별로 분산되어 있으므로, 본 연구는 재현 가능한 증거 확보와 타임라인 기반의 행위 규명을 핵심 목표로 설정하고 연구를 수행했다.

연구를 통해 도출된 수집 절차·분석 기법·도구는 AWS 클라우드 환경에서 DFIR 수행의 표준화와 실무 적용성을 크게 향상시킬 수 있다. 명령 기반·로그 기반·포렌식 이미지의 통합적 수집 구조, 전술 기반의 이벤트 매핑, 자동화된 로그 파싱·가시화 도구의 결합은 사고 대응가가 사건 재현성과 분석 신뢰성을 확보하는 데 실질적 도움을 줄 수 있다. 주요 성과는 다음과 같다.

1) DFIR 데이터 수집 체계 정립

AWS 환경의 구조적 제약을 고려해 수집 유형을 다음과 같이 3가지로 구분하고, 각 범주별 수집 항목 및 방안을 체계화했다.

- **명령 기반 수집 (Command-based)**

AWS 환경 내 인스턴스 및 서버 리소스에 직접 명령을 전달해 시스템 정보, 구성 설정, 로그 파일 등을 확보하는 방식이다. 이는 사고 발생 시 신속하게 시스템의 현황과 보안 구성을 파악하기 위한 절차로, 본 연구에서는 AWS CLI, AWS Systems Manager(SSM), Prowler를 활용한 명령 기반 수집 체계를 정립했다.

- **로그 기반 수집 (Log-based)**

AWS 서비스가 생성하는 운영 로그를 수집하는 방식으로, 공격 단계별 징후를 식별하는 데 핵심 근거로 활용된다. 본 연구에서는 CloudTrail, VPC Flow Logs, S3 Server Access Log, CloudWatch Logs, GuardDuty Findings, WAF Log 등을 대상으로 각 로그의 수집 경로와 주요 로그 필드, 분석 포인트를 정의했다.

- **포렌식 이미지 수집 (Forensic Image)**

침해가 발생한 인스턴스의 스냅샷을 확보하는 방식으로, EC2/EKS Worker Node에 대해 EBS 스냅샷 생성 방안 정립했다.

2) DFIR 분석 방안 도출 및 도구 개발

클라우드 DFIR의 분석 단계에서는 수집된 데이터를 바탕으로 공격 행위를 규명하고 타임라인 형태로 재구성하는 것을 목표로 하기 위해 다음과 같이 수행했다.

● 전술 기반 분석 프레임워크 및 CheatSheet 개발

MITRE ATT&CK의 전술을 참조해 CloudTrail, VPC Flow, S3 Access Log 등에서 빈번하게 관찰되는 이벤트를 전술별로 매핑한 AWS DFIR Cheat Sheet를 개발하였다. Cheat Sheet에는 각 이벤트의 의미, 공격에서의 악용 유형, 사고 시 연계 가능성이 높은 이벤트 및 핵심 로그 컬럼 등을 정리해 분석가의 표준화된 해석 지침으로 활용할 수 있도록 했다.

● 분석 도구(bitParser for AWS Log) 개발

CloudTrail, VPC Flow, S3 Access Log를 입력으로 삼아 로그를 정규화(flatten)하고 전술별 이벤트를 자동 식별·시각화하는 도구를 구현했다. 해당 도구는 주요 공격 징후를 탐지해 분석가가 우선 검토해야 할 이벤트를 제시함으로써 초동 분석의 효율성을 향상시키고, 로그 해석 과정에서의 편차를 줄이도록 설계되었다.

3) 시나리오 기반 실효성 검증

랜섬웨어 공격을 가정한 AWS 침해 시나리오를 구성해 제안한 수집·분석 체계를 검증했다. 사고의 전반적인 공격 행위는 주로 CloudTrail 로그에서 분석할 수 있었으며, 내부 이동과 같은 네트워크 행위는 VPC Flow Logs를 통해 파악할 수 있었다. 또한, 본 연구에서 개발한 bitParser 도구를 적용한 결과, 전술(Tactics) 기반의 위협 이벤트를 자동으로 식별·우선순위화하고 타임라인을 구성하는 과정에서 수동 분석 대비 명확한 효율성 향상을 확인했다.

8. 결론 및 향후 연구

본 연구는 AWS 클라우드 환경에서의 사고 대응을 위한 DFIR 데이터 수집 및 분석 체계를 제안하고, 랜섬웨어 사고 시나리오를 통해 그 실효성을 검증했다. 이를 통해 AWS 클라우드 환경에서 재현 가능한 데이터 확보 체계와 자동화된 분석 기반을 마련했다는 점에서 연구의 의의가 있다.

기존 온프레미스 중심의 DFIR 연구가 개별 로그나 도구에 한정되었던 것과 달리, 본 연구는 클라우드 서비스의 구조적 제약을 고려해 '수집–분석–도구화'의 통합 프레임워크를 제시하였다는 점에서 차별성을 가진다. 명령 기반·로그 기반·포렌식 이미지 기반의 수집 절차를 체계화하고 전술(Tactic) 기반의 로그 분석을 자동화한 점은 클라우드 사고 대응의 표준화 가능성과 실무 적용성을 입증했다.

또한, 연구는 사고 대응가가 AWS 환경에서 로그와 시스템 데이터를 일관된 절차로 확보하고 공격 행위를 신속히 재구성할 수 있는 기반을 마련했다. 이를 통해 기존 온프레미스 DFIR 중심의 연구를 확장하고, 클라우드 환경에 특화된 DFIR 프레임워크의 이론적 토대를 제시했다.

본 연구는 AWS 단일 환경을 중심으로 수행되어, 클라우드 전반의 다양한 운영 형태를 모두 반영하지는 못했다. 따라서, 향후 연구에서는 다음과 같은 보완과 확장이 필요하다.

첫째, 멀티클라우드 환경으로의 확장 필요

본 연구는 AWS 환경에 한정되어 수행되었으나, Azure와 GCP 등 이기종 클라우드 간에는 로그 포맷과 보안 구조가 상이하다. 따라서, 각 플랫폼의 로그 구조를 통합 분석할 수 있는 표준화된 DFIR 프레임워크 연구가 필요하다.

둘째, AI 기반 이상행위 분석의 고도화

현재 제안된 체계는 전술(Tactic) 기반 이벤트 매핑 수준에서 공격 행위를 식별하도록 구성되어 있으나, 향후에는 머신러닝과 인공지능 기법을 적용해 공격 행위를 자동으로 탐지·분류하는 지능형 분석 모델로 발전시킬 필요가 있다.

셋째, 실시간 대응 체계로의 확장

AWS EventBridge, Step Functions, Lambda 등 네이티브 서비스를 활용해 탐지–격리–증거 보존을 자동화하는 실시간 대응 오케스트레이션 구조를 연구함으로써 사고 대응의 신속성과 일관성을 강화할 수 있을 것이다.

참고 문헌

번호	참고 문헌
1	Google Cloud, “M-Trends 2025 Report”, https://cloud.google.com/security/resources/m-trends?hl=ko, 2025.05.27.
2	MITRE ATT&CK, “Enterprise - Cloud Matrix”, attack.mitre.org/matrices/enterprise/cloud/, 2025.04.25.
3	AWS, “Shared responsibility in the cloud”, learn.microsoft.com/en-us/azure/security/fundamentals/shared-responsibility, 2024.09.29.
4	AWS, “The AWS Security Reference Architecture”, https://docs.aws.amazon.com/en_us/prescriptive-guidance/latest/security-reference-architecture/architecture.html, 2025.10.13.
5	Chris Champa, “What Is Cloud Incident Response?”, https://www.wiz.io/academy/cloud-incident-response, 2025.07.14.
6	CSA, “Top Threats to Cloud Computing 2024”, https://cloudsecurityalliance.org/artifacts/top-threats-to-cloud-computing-2024, 2024.08.05.
7	Google Cloud, “M-Trends 2023 Report”, https://services.google.com/fh/files/misc/m_trends_2023_report.pdf, 2023.04.18.
8	Daniel Leussink and Kantaro Komiya, “More than 2 million Toyota users face risk of vehicle data leak in Japan”, https://www.reuters.com/business/autos-transportation/toyota-flags-possible-leak-more-than-2-mln-users-vehicle-data-japan-2023-05-12/?ref=thestack.technology, 2023.05.12.
9	jumpcloud, “[Security Update] June 20 Incident Details and Remediation”, https://jumpcloud.com/blog/security-update-june-20-incident-details-and-remediation, 2023.09.07.
10	Pierluigi Paganini, “DARKBEAM LEAKS BILLIONS OF EMAIL AND PASSWORD COMBINATIONS”, https://securityaffairs.com/151566/security/darkbeam-data-leak.html, 2023.09.27.
11	Ionut Arghire, “Mercedes Source Code Exposed by Leaked GitHub Token”, https://www.securityweek.com/leaked-github-token-exposed-mercedes-source-code/, 2024.01.31.
12	AWS, “AWS Security Incident Response Guide”, https://docs.aws.amazon.com/pdfs/whitepapers/latest/aws-security-incident-response-guide/aws-security-incident-response-guide.pdf, 2025.08.15.
13	AWS, “What is Amazon GuardDuty?”, https://docs.aws.amazon.com/en_us/guardduty/latest/ug/what-is-guardduty.html, 2025.10.14.
14	AWS, “What is Amazon CloudWatch Logs?”, https://docs.aws.amazon.com/en_us/AmazonCloudWatch/latest/logs/WhatIsCloudWatchLogs.html, 2025.10.14.
15	AWS, “What is Amazon Detective?”, https://docs.aws.amazon.com/en_us/detective/latest/userguide/what-is-detective.html, 2025.10.14.
16	AWS, “What is Amazon Athena?”, https://docs.aws.amazon.com/en_us/athena/latest/ug/what-is.html, 2025.10.10.
17	AWS, “Introduction to AWS Security Hub”, https://docs.aws.amazon.com/en_us/securityhub/latest/userguide/what-is-securityhub-v2.html, 2025.10.14.
18	AWS, “What is AWS Systems Manager?”, https://docs.aws.amazon.com/en_us/systems-manager/latest/userguide/what-is-systems-manager.html, 2025.10.14.
19	AWS, “What is Amazon Macie?”, https://docs.aws.amazon.com/en_us/macie/latest/user/what-is-macie.html, 2025.10.14.
20	AWS, “What Is AWS Config?”, https://docs.aws.amazon.com/en_us/config/latest/developerguide/WhatIsConfig.html, 2025.10.14.

번호	참고 문헌
21	AWS, “What is Amazon Inspector?”, https://docs.aws.amazon.com/en_us/inspector/latest/user/what-is-inspector.html, 2025.10.14.
22	AWS, “What is AWS CloudFormation?”, https://docs.aws.amazon.com/en_us/AWSCloudFormation/latest/UserGuide/Welcome.html, 2025.10.14.
23	prowler-cloud, “prowler”, https://github.com/prowler-cloud/prowler, 2025.09.30.
24	AWS, “AWS Shield”, https://docs.aws.amazon.com/en_us/waf/latest/developerguide/shield-chapter.html, 2025.10.14.
25	aws-samples, “aws-incident-response-playbooks”, https://github.com/aws-samples/aws-incident-response-playbooks, 2025.07.08.
26	aws-samples, “aws-customer-playbook-framework”, https://github.com/aws-samples/aws-customer-playbook-framework/tree/main/docs, 2025.08.05.
27	aws-samples, “aws-incident-response-playbooks-workshop”, https://github.com/aws-samples/aws-incident-response-playbooks-workshop, 2024.02.20.
28	AWS, “What Is AWS CloudTrail?”, https://docs.aws.amazon.com/en_us/awscloudtrail/latest/userguide/cloudtrail-user-guide.html, 2025.10.14.
29	AWS, “Logging IP traffic using VPC Flow Logs”, https://docs.aws.amazon.com/en_us/vpc/latest/userguide/flow-logs.html, 2025.10.14.
30	AWS, “Enabling Amazon S3 server access logging”, https://docs.aws.amazon.com/en_us/AmazonS3/latest/userguide/enable-server-access-logging.html, 2025.10.14.
31	AWS, “Monitoring Amazon RDS log files”, https://docs.aws.amazon.com/en_us/AmazonRDS/latest/UserGuide/USER_LogAccess.html, 2025.10.14.
32	AWS, “Logging AWS WAF protection pack (web ACL) traffic”, https://docs.aws.amazon.com/en_us/waf/latest/developerguide/logging.html, 2025.10.14.